

Shreekumar T.
Dinesha L.
Sreeja Rajesh (Eds.)

Communications in Computer and Information Science

2232

Intelligent Systems in Computing and Communication

First International Conference, ISCComm 2023
Moodabidri, India, December 8–9, 2023
Proceedings, Part II

Part 2

Communications in Computer and Information Science

2232

Series Editors

Gang Li , *School of Information Technology, Deakin University, Burwood, VIC, Australia*

Joaquim Filipe , *Polytechnic Institute of Setúbal, Setúbal, Portugal*

Ashish Ghosh , *Indian Statistical Institute, Kolkata, West Bengal, India*

Zhiwei Xu, *Chinese Academy of Sciences, Beijing, China*

Rationale

The CCIS series is devoted to the publication of proceedings of computer science conferences. Its aim is to efficiently disseminate original research results in informatics in printed and electronic form. While the focus is on publication of peer-reviewed full papers presenting mature work, inclusion of reviewed short papers reporting on work in progress is welcome, too. Besides globally relevant meetings with internationally representative program committees guaranteeing a strict peer-reviewing and paper selection process, conferences run by societies or of high regional or national relevance are also considered for publication.

Topics

The topical scope of CCIS spans the entire spectrum of informatics ranging from foundational topics in the theory of computing to information and communications science and technology and a broad variety of interdisciplinary application fields.

Information for Volume Editors and Authors

Publication in CCIS is free of charge. No royalties are paid, however, we offer registered conference participants temporary free access to the online version of the conference proceedings on SpringerLink (<http://link.springer.com>) by means of an http referrer from the conference website and/or a number of complimentary printed copies, as specified in the official acceptance email of the event.

CCIS proceedings can be published in time for distribution at conferences or as post-proceedings, and delivered in the form of printed books and/or electronically as USBs and/or e-content licenses for accessing proceedings at SpringerLink. Furthermore, CCIS proceedings are included in the CCIS electronic book series hosted in the SpringerLink digital library at <http://link.springer.com/bookseries/7899>. Conferences publishing in CCIS are allowed to use Online Conference Service (OCS) for managing the whole proceedings lifecycle (from submission and reviewing to preparing for publication) free of charge.

Publication process

The language of publication is exclusively English. Authors publishing in CCIS have to sign the Springer CCIS copyright transfer form, however, they are free to use their material published in CCIS for substantially changed, more elaborate subsequent publications elsewhere. For the preparation of the camera-ready papers/files, authors have to strictly adhere to the Springer CCIS Authors' Instructions and are strongly encouraged to use the CCIS LaTeX style files or templates.

Abstracting/Indexing

CCIS is abstracted/indexed in DBLP, Google Scholar, EI-Compendex, Mathematical Reviews, SCImago, Scopus. CCIS volumes are also submitted for the inclusion in ISI Proceedings.

How to start

To start the evaluation of your proposal for inclusion in the CCIS series, please send an e-mail to ccis@springer.com.

Shreekumar T. · Dinesha L. · Sreeja Rajesh
Editors

Intelligent Systems in Computing and Communication

First International Conference, ISCComm 2023
Moodabidri, India, December 8–9, 2023
Proceedings, Part II

Editors

Shreekumar T. 
MITE
Moodabidri, Karnataka, India

Dinesha L. 
MITE
Moodabidri, Karnataka, India

Sreeja Rajesh 
MITE
Moodabidri, Karnataka, India

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-3-031-75607-8 ISBN 978-3-031-75608-5 (eBook)
<https://doi.org/10.1007/978-3-031-75608-5>

© The Editor(s) (if applicable) and The Author(s), under exclusive license
to Springer Nature Switzerland AG 2025, corrected publication 2025

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

If disposing of this product, please recycle the paper.

MITE



Invent Solutions

Preface

We are delighted and honored to announce the success of the First International Conference on Intelligent Systems in Computing and Communications (ISCCComm 2023), held on December 8–9, 2023, at the Mangalore Institute of Technology & Engineering in Moodabidri. This conference brought together leading researchers, academicians, and industry professionals from various countries worldwide. The conference featured prominent figures from academia and industry as our conference chair, co-chair, program chairs, and track chairs, including experts from top-ranked Indian Institutes such as NITs and MITs, and prestigious foreign universities. The dedication of the Advisory, Program, and Branding & Outreach committees played a crucial role in ensuring the quality of submissions and the overall success of the conference.

We received a total of 245 technical papers across four tracks: AI & ML, IoT and Blockchain Technology, Communication Technology and Data Processing, and Image/Video Analysis and Processing. From these submissions, 53 papers (<22%) were selected for publication in Springer CCIS, comprising 36 long papers (≥ 12 pages) and 17 short papers (<12 pages), each undergoing three technical reviews. We extend our gratitude to the staff of Springer CCIS for their continuous support and guidance.

The technical papers presented covered a wide range of application domains, including healthcare, agriculture, automobile, civil and mechanical engineering, pharmacy, cybercrime, and sports. We commend all the enthusiastic authors for submitting their innovative research works.

The conference featured keynote addresses by renowned experts such as Raghavendra Ramachandra from the Information Security & Communication department at NTNU, Norway, Shanthi Sekhar from the Training Vision Institute, Singapore, and Mahadeva Prasanna from IIT Dharwad, India.

We extend our sincere thanks to our esteemed sponsors, including Canara Bank and Mangalore Refinery & Petrochemicals Limited, for their generous support.

We believe this collection of papers will be highly beneficial for researchers and practitioners in AI & ML and their related fields.

December 2023

Shreekumar T.
Dinesha L.
Sreeja Rajesh

Organization

Chief Patron

Rajesh Chouta	MITE, India
---------------	-------------

Patron

Prashanth C. M.	MITE, India
-----------------	-------------

Conference Chair

Ashok Rao	IISc, India
-----------	-------------

Conference Co-chair

Raghavendra Ramachandra	NTNU, Norway
-------------------------	--------------

Program Committee Chairs

Shreekumar T.	MITE, India
Dinesha L.	MITE, India
Sreeja Rajesh	MITE, India

Steering Committee

Pradeep B. S.	MITE, India
Nagesh Prabhu	NITTE University, India

Advisory Committee

Thejas G. S.	Tarleton State University, USA
Mahesha Narayana	University of the West Indies at Mona, Jamaica
Zhang Jingang T. C.	Linyi Top Network Co. Ltd., China

Vinod P.	CUSAT, India
Balakrishna R.	Raja Rajeswari College of Engineering, India
Nagaratna Hegde	Vasavi College of Engineering, India
Mohit Tahilian	NITK Suratkal, India
Sushil Kumar Pandey	NITK Suratkal, India
Suleiman Yerima	De Montfort University, UK
P. S. Hiremath	KLE Technological University, India
Brahmananda S. H.	GITAM School of Technology, India
P. Vijaya	Modern College of Business and Science, Oman
Priyamvada Singh	DigiPen Inst. of Technology, Redmond, USA
Periyasamy	NIT Tiruchirappalli, India
Venugopala P. S.	NMAMIT, India
Prakash Kumar Udupi	Middle East College, Oman

Publicity Committee

Srikrishna Shastri	MITE, India
Ravinarayana B.	MITE, India
Rashmi Samanth	MITE, India
Sunitha	MITE, India
Sunil Kumar S.	MITE, India
Uday J.	MITE, India
Jyothi S.	MITE, India
Narendra U. P.	MITE, India
Swaroop	MITE, India
Ranjith H. D.	MITE, India
Rashmi Praveen	MITE, India
Deepthi Shetty	MITE, India
Ashwin Kumar	MITE, India
Sonia Peal K. P.	MITE, India
Vasudha	MITE, India
Shivaprasad T. K.	MITE, India
Bindu Madhavi J.	MITE, India

Main Technical Committee

Nawali Nousath	University of Technology and Applied Sciences, Sohar, Oman
Raghavendra Ramachandra	NTNU, Norway
Sreeja Rajesh	MITE, India

Nagesh Prabhu
Roshan Fernandes

NITTE University, India
NITTE University, India

Technical Program Committee Chairs

Sunil Jacob	SOTI Aerospace, Canada
Muhammed Siddique	Majan University College, Oman
Jacintha Menezes	Majan University College, Oman
Saira Jacob	SOTI, Canada
Ramalingam	EC-Council, Oman
Manju Jose	Majan University College, Oman
Sannidhan M. S.	NMAMIT, India
Syed Ziaur Rahman	Majan University College, Oman
Shiva Prakash M.	Dr. AIT, India
Srinath Acharya	SJEC, India
Radhakrishna Doddamane	NMAMIT, India
V. Sowmya	Amrita Vishwa Vidyapeetham, Coimbatore, India
Sudharshan K. M.	REVA University, India
Geetha	NITK Suratkal, India
Karthik	KSSEM, India
Narendra V. G.	Manipal Institute of Technology, India
S. G. Totad	KLE Technological University, India
Ravi	GAT, India
Guruprasad M.	Graphic Era University, Dehradun, India
J. D. Pujari	SDM College of Engineering, India
Shreeknath Acharya	St. Joseph Engineering College, India
Ashwini	NMAMIT, India
Varun G. Menon	SCMS School of Engineering & Technology, India
Varghese Paul	Rajagiri School of Engineering and Technology, India
Mahesh T. R.	Jain University, India
Krishna Prakash	Manipal Institute of Technology, India
Ramakrishna	Manipal Institute of Technology, India
Regitha M. R.	Sacred Heart College, Thevara, India
Srikrishna Shastry	MITE, India
Ramalingam	MITE, India
Kiran Kumar	MITE, India
Babu Rao	MITE, India
Maryjo M. George	MITE, India

Additional Reviewers

Ahamed Shafeeq B. M.
Ajanthaa Lakkshmanan
Akash Gandhar
Anitha N.
Anoop B. K.
Arunadevi
Arunakumari B. N.
Ashwini
B. Perumal
Babitha Ganesh Kulal
Basappa Bharamappa Kodada
Bhargavi K.
Biswajit R. Bhowmik
Chandan Kumar Behera
Dhanesha R.
Divya Mohan
Gaurav Prasad
Geeta R. Bharamagoudar
Geetha
Geetha Mary A.
Guruprasad M.
Haraprasad Mondal
Harshalatha Y.
Hemalatha N.
J. D. Pujari
Jacintha Menezes
Jose Alex Mathew
Jyoti Metan
K. A. Varunkumar
K. Annapoorneshwari Shetty
Kallinatha Hd
Kanakam Siva Rama Prasad
Kanmani
Karthik
Krisha M.
Krishna Prakash
Krishna Prakasha K.
Kshama S. B.
Kuldeep BabanVayadande
Latha H. N.
Laxmi Gulappagol
Madhusudhan G.
Mahesh T. R.

Malathi S. Y.
Manju Jose
Manjula
Manjula Gururaj Rao H.
Mohammed Siddique
Mohan Naik R.
Muhammed Siddique
Nandana Anantha Prabhu
Nandini Prasad K. S.
Narendra V. G.
Neelima K.
Nguyen Quang Vinh
Nimisha Patel
P. Vijaya
Padmanayana
Pavan Kunchur
Poovammal
Prabha Niranjana
Pradeep Kanchan
Pradeep V.
Prajna M. R.
Prakash S. Hiremath
Pramod H. B.
Prashant Narayankar
Praveen Gujjar J.
Praveen J.
Premjith B.
Priyadarshini Patil
Puneet Mittal
Pushpa R.
RadhakrishnaDoddamane
RafidhaRehiman K. A.
Rajan Patel
Rajashekhar B. Somasagar
Rajitha M. R.
Raju K.
Ramakrishna
Ramalingam
Ramesh G.
Ramesh N. Koppar
Ramesh S. Wadawadagi
Rashmi Laxmikant Malghan
Rashmi Naveen

Rashmi P. Karchi
Ratnababu Pilli
Ravi
Ravishankara Kulamarva
Roshan Fernandes
Rudresh Deepak Shirwaikar
S. Gopal Krishna Patro
S. Sethu Selvi
S. G. Totad
S. Kannadhasan
Sai Venkatramana Prasada G. S.
Sainithin Artham
Saira Jacob
Sanjay Thakur
Sanjeevakumar M. Hatture
Sanket S. Kulkarni
Sannidhan M. S.
Sarojadevi H.
Savitha C. K.
Savitha G.
Shashank M. Gowda
Shashank Shetty
Shekhappa Ankaliki
Shiva Prakash M.
Shiva Tyagi
Shreeknath Acharya
Sindhu C.

Sowmya V.
Srikanta Dash
Srinath Acharya
Srinivasa K.
Srinivasa Rao Burri
Sudharshan K. M.
Sudheendra P.
Sudheer Shetty
Suja C. Nair
Sumangala Biradar
Sumathi
Sunil Jacob
Sunil Kumar Nahak
Suresha D.
Syed Ziaur Rahman
T. Ganesh Kumar
Thangam S.
Urmila T. S.
Varghese Paul
Varun G. Menon
Vasudeva
Veena Devi Shastrimath V.
Vidya A.
Vidya Rao
Vidyalakshmi K.
Vimuktha E. Salis
Viswavardhan Reddy Karna

Contents – Part II

IoT&Blockchain Technology

Enhancing Resource Scalability in Cloud Computing: A Comparative Investigation of Neural Network Predictive Techniques	3
<i>Yaswanth Kumar Damarla and Bhagya Rekha Deenadayal</i>	
Revolutionizing Dental Treatment Through IoT Integrated Force Sensors: Design and Calibration	21
<i>Chaithra Salian, Prakyath Shetty, Charishma Shetty, Durga Prasad, G. R. Pradyumna, K. B. Bommegowda, M. S. Ravi, and P. S. Murali</i>	
Streamlining Software Migration with Model-Driven Techniques	32
<i>Yaswanth Kumar Damarla and Bhagya Rekha Deenadayal</i>	
A Survey - IOT Edge Based Garbage Sorter Bin with Bio Gas Generator	45
<i>A. R. Sathyabama, Jeevaa Katiravan, M. Muthudhivakar, and S. Sabarinath</i>	
Architecture of Biometric System Using Multiphase	57
<i>M. P. Nayana and S. Soumya</i>	
Reinforcement Learning with Enhanced Bubble Sort for Wireless Mesh Networks Node Selection	72
<i>G. Revathy, J. Senthilkumar, E. Gurumoorthi, and M. Shyamalagowri</i>	
Development of an Soil Health Monitoring IoT Integrated System	81
<i>M. C. Jayaprakash, Vinayambika S. Bhat, and M. P. Shama</i>	
One Nation One Health Card and Patient Health Monitoring System Using IOT and Cloud Technologies	89
<i>K. M. Deepika, Akshata B. Anjutagi, M. N. Yashodha, K. Lavanya, and Renuka Karadagi</i>	
Amalgamation of Semantic Web with IoT: Semantic Web of Things (SWoT) ...	100
<i>Gaurav Jaglan, Aman Jolly, Indrasen Singh, Vikas Pandey, and Shashikant</i>	

Developing an Autonomous Fruit Picking and Sorting Robot for Vertical Farming	111
<i>S. Santhosh, Pavitra kumari, J. Bindu Madhavi, and B. Mohan Kumar Naik</i>	

An Efficient System for Water Quality Monitoring in Sustainable Ground Water Supply Using IoT Technology	121
<i>C. Sri Krishna Shastri and M. C. Jayaprakash</i>	

Stable Control of Underwater Target Search Robot Support Rescue and Rescue Work	130
<i>Nguyen Duc Anh, Ravishankara Kulamarva, D. Suresha, and Nguyen Quang Vinh</i>	

Communication Technology and Data Processing

Comparative Analysis of Network on Chip Topologies	139
<i>Mondal Churantal, C. S. Chandana, Sangavi Siddarth, V. Sanju, Ramakrishnan Nithin, and Farhanaaz</i>	

Adaptive Handover Decision and Load Balancing Optimization in LTE Networks: A Fuzzy Logic System Perspective	150
<i>Divya Mohan, A. Geetha Mary, K. Priyadharshini, and Rashmi Naveen</i>	

A Comprehensive Review of Optimization Techniques for Automatic Voltage Regulator Controllers	168
<i>Rumana Ali and Vinayambika S. Bhat</i>	

Implementation of UWB Receiver Front-End in FPGA	181
<i>Aravinda Koithyar, S. Arun, and K. Bhaskar</i>	

Design and Development of a Virtual Laboratory Using the Python Web Framework Django to Carry Out Experiments in the Stream of ML, DSP, and Digital Communication	195
<i>Shivaprasad and A. S. Dinesha</i>	

Image/Video Analysis and Processing

Swift Detection of Human Fall Events in Compressed Videos	217
<i>S. M. Praveenkumar, Prakashgoud Patil, and P. S. Hiremath</i>	

Automatic Music Control Using Image Processing and MediaPipe	235
<i>Sudheer Shetty, R. Rakshitha, S. Arundhathi Bhat, Lathesh, Ravish Acharya, Suresh Kumar Peddoju, and Hemanth Kumar Nichenametla</i>	
Study of Lung Nodules Using CT Scan Image	248
<i>P. V. Raja Suganya, T. Devi, and V. Joshita</i>	
Mathematical Formulation and Genetic Algorithm for Permutation Flow Shop Scheduling with Release Date to Minimize Makespan	262
<i>Sachin Karadgi and P. S. Hiremath</i>	
Estimate Initial and Final Data Bounding for Big Data Progressive Sampling in Parallel Computing Environment	275
<i>B. C. Yathish Aradhya, H. A. Dinesh, and M. R. Lokesh</i>	
Digital Glamour: Exploring the Intersection of Facial Makeup and Augmented Reality	291
<i>Vinayambika S. Bhat, Deepthi Shetty, and Deepthi Kotian</i>	
Local Stereo Matching Technique Based on Collaborative Cost Aggregation and Improved Disparity Refinement	303
<i>Deepa, K. Jyothi, and Abhishek A. Udupa</i>	
Self-Healing Networks Leverage Intelligent Protocols Enable Autonomous Fault Detection and Recovery	317
<i>V. Lakshmi and S. M. A. K. Azad</i>	
A Secure System for Decentralized Preservation of Digital Library Collections Using Private Blockchain and Interplanetary File System (IPFS)	331
<i>Patrick Kipkorir Laboso, A. Martin, and P. Thiagarajan</i>	
Learning Systems in Forgery Image Detection	342
<i>Sudheer Shetty, Devadiga Likhit kumar Ganesh, Rahul R. Poojary, Raviraj, R. Tejas, Nagaratna P. Hegde, and Sireesha Vikkurty</i>	
Correction to: A Comprehensive Review of Optimization Techniques for Automatic Voltage Regulator Controllers	C1
<i>Rumana Ali and Vinayambika S. Bhat</i>	
Author Index	355

Contents – Part I

Artificial Intelligence and Machine Learning

A Novel Approach for Skin Infections Classification Using Transfer Learning	3
<i>Dhruv Singhal, Deepanshu Verma, Ankita Nainwal, B. Dhananjaya, and Garima Sharma</i>	
A Unified Image Repository for Cucurbits Diseases and Pests	14
<i>Rithika Adapa, Raksha, N. Hemalatha, Prathamesh Pai, K. M. Sreekumar, P. K. Sajeesh, and Sainamole Kurian</i>	
Towards Hands-Free Computing: AI Virtual Mouse Interface Powered by Gestures	26
<i>A. Menaka Gowda, M. Krisha, Shrividya Shashidhara, and B. N. Lakshmi</i>	
SmartAgro: Precision Yield Prediction, Crop Insights, and Real-Time Dashboard	50
<i>Parth Mody, Ankur Singh, Param Sejpal, Anurag Singh, and Archana Nanade</i>	
Artificial Intelligence with MRI-Guided Radiation Therapy for Cancer Treatment	72
<i>K. Priyadharshini, Divya Mohan, and Amulya S. Bhat</i>	
Software Requirements to UML Class Diagrams Using Machine Learning and Rule-Based Approach	86
<i>Feisal Alaswad, E. Poovammal, Batoul Aljaddouh, and B. Supriya</i>	
CFALEA_LSTM: Adaptive Lotus Effect Algorithm Enabled Long Short-Term Memory for Rainfall Prediction Using Time Series Data	102
<i>P. Vijaya, Satish Chander, Roshan Fernandes, Anisha P. Rodrigues, and B. Supriya</i>	
A Deep Learning Survey on Diseases Prediction and Detection in Health Care	119
<i>R. Priyanka and M. R. Pooja</i>	

Fault Diagnosis in Belts Using Signal Processing Techniques and Machine Learning	128
<i>M. V. Kirankumar, Rajeshwari, Nishmitha, Pavitra Kumari, B. K. Raghavendra, and Kulkarni Sourabh Vyankatesh</i>	
Machine Learning Techniques Based Chronic Kidney Disease Detection with Performance Analysis of Fuzzy Rough Set and Correlation Attribute Selection	145
<i>Ch. Ravindra Babu, E. V. N. Jyothi, Kunchala Teja, Annem Tharun Kumar Reddy, Mutyala Venkata Gopi Chand, Peram Rajesh, and R. J. Shwetha</i>	
Segmentation and Classification of Unharvested Arecanut Bunches Using Deep Learning	158
<i>R. Dhanesha, D. K. Umesha, Gurudeva Shastri Hiremath, G. N. Girish, and C. L. Shrinivasa Naika</i>	
Enhanced Satellite Image Fusion Using Deep Learning and Feature Extraction Techniques: A Survey	171
<i>Swathi Nallagachu and R. Sandanalakshmi</i>	
Intelligent Aircraft Antiskid Braking Systems – A Review	188
<i>P. Sudheendra and Vinayambika S. Bhat</i>	
Artificial Neural Networks Applied in the Detection of Breast Cancer	207
<i>Deepthi Kotian, Vinayambika S. Bhat, Dhanush Poojary, and V. Kavya</i>	
Supervised and Unsupervised Learning Techniques for Malware Classification Based on Opcode Frequency Features	221
<i>K. A. Asmitha, P. Vinod, K. A. Rafidha Rehimani, Renil M. Cherian, Rohith Balachandran, Bins Titus, R. Rohith Raj, K. N. Fathima Afrah, and A. R. Archana</i>	
Predictive Analytics for Diagnosing Alzheimer's Disease Using Artificial Intelligence and Machine Learning Algorithms	232
<i>Hemanth Kumar Nichenametla, Suresh Kumar Peddoju, and Sudheer Shetty</i>	
A Comprehensive Study on Artificial Intelligence (AI) Driven Internet of Healthcare Things (IOHT)	254
<i>Astha Sah, Bharat Bhushan, and Chinmai Shetty</i>	

Predictive Models for the Early Diagnosis and Prognosis of Knee Osteoarthritis Using Deep Learning Techniques	270
<i>S. Y. Malathi, Geeta Bharamagoudar, Sharada K. Shiragudikar, and Shashikumar G. Totad</i>	
Predicting Salinity Resistance of Rice at the Seedling Stage: An Evaluation of Transfer Learning Methods	285
<i>Sharada K. Shiragudikar, Geeta Bharamagoudar, Manohara K. K., Malathi S. Y., and Shashikumar G. Totad</i>	
Multi-camera HD Pedestrian Dataset for Person Detection and Re-identification	309
<i>Prashant Narayankar, Vishwanath P. Baligar, Prajwalkumar A. Gouli, Naresh Illur, Pooja Doddannavar, and Pooja M. Malagund</i>	
Fire Detection System Using Deep CNN	318
<i>Sireesha Vikkurty, P. Nagaratna Hegde, Vennela Preethi Chinthakrinda, G. P. Hegde, and Sudheer Shetty</i>	
Deep Learning Classification Model for Analysing and Predicting Road Accidents	326
<i>Prakash Kumar Udupi and Manju Jose</i>	
Web Scraping System for Recommendation of Mobiles from Multiple E-commerce Websites	334
<i>Shivananda V. Seeri, K. P. Sonia Peal, Sandeep Seetaram Naik, and A. Aneesh Kumar</i>	
A Machine Learning Approaches with Special Emphasis for Identification of Plant Leaf Diseases in Crop Yield Prediction	342
<i>Ch Ravindra Babu, E. V. N. Jyothi, K. Naga Prassana, T. Siva Jayanth, Sk. Sandani, S. Veera Venkata Gopi Chand, and R. J. Shwetha</i>	
State of the Art Recurrent Neural Network with Bidirectional Long Short-Term Memory for Cursive Handwriting Recognition	354
<i>Manju Jose and Prakash Kumar Udupi</i>	
BERT-Based Question-Answering for Narrative Text on Document Corpus	364
<i>Ramesh Wadawadagi, Shrikant Tiwari, and Sanjay Hanji</i>	
Author Index	375

IoT&Blockchain Technology



Enhancing Resource Scalability in Cloud Computing: A Comparative Investigation of Neural Network Predictive Techniques

Yaswanth Kumar Damarla^{1,2}✉ and Bhagya Rekha Deenadayal²

¹ Jawaharlal Nehru Technological University Kakinada - University College of Engineering
Vizianagaram, Vizianagaram, India
yaswanthkumardamarla23@gmail.com

² Vasireddy Venkatadri Institute of Technology, Guntur, India

Abstract. The field of cloud computing is changing in field of computer science. Software companies are using cloud services to deploy their applications. As a result, a lot of interest has been shown by researchers in this field. An important challenge faced in this field is the question of how and when to scale the use of hardware and virtual machine resources. As cloud services are costly, making efficient use of them is necessary to prevent wastage of resources and which will help companies in preserving there business. In this paper, we present a parameter value predictor using LSTMs and RNNs, both of then different predictive mechanisms using neural networks and finding out which is more effective. Previously, approaches like different processing, mathematical models were used to scale microservices. Those approaches were not optimal as they are not good at predicting loads and hard to manage. So, we use Reinforcement Learning, namely Double Deep Q-Networks Algorithm, as this approach automatically monitors the parameters which are essential for scaling and triggers the effect (action) onsite immediately. It also balances the acts of exploiting the environment and exploring the environment, using the computing resources and computing cost-effectively.

Keywords: autoscaling · reinforcement learning · double DQN · Average Memory Utilization SLA

1 Introduction

The traditional approach to application deployment has been to deploy applications in a monolithic architecture. These applications typically consist of a single directory. As the number of clients increases, multiple instances of the application may be deployed, resulting in an increase in load that must be balanced. To address the numerous constraints, such as overloaded containers, IDEs, and the inability to dynamically scale, a new architecture, known as microservice architecture, has been developed. This architecture divides applications into atomic modules, and scales these modules rather than the entire application. This model allows for the creation of new instances of these modules when scaling, providing improved fault tolerance and maintainability, among other advantages.

In microservice architectures, cloud service providers face the challenge of ensuring that Service Level Agreement violations do not take place. Service Level Agreements usually specify the performance expected from the computing infrastructure and failure to adhere to it will lead to monetary penalties. At the same time, cloud service providers also have physical and financial constraints to scale up or down. Thus, ensuring that the hardware resources, power, etc. offered by the Cloud providers are used at an optimal level is a tough question to answer.

Predicting when to scale in a model is most important, they are different prediction available among those we chosen LSTM, an artificial RNN model used in deep learning. In such network, as opposing a typical feed forward neural network, there are feedback connections. Along with processing data with individual points, images for example, it can also process data sequences, like speech and video. A standard LSTM entity has the following components: a memory component called cell, and three gates for overseeing the information flow, called input, output, and forget.

An RNN formed using LSTM entities can undergo supervised training on a pre-determined training data, employing an optimization algorithm such as gradient descent, along with back propagation to calculate the loss gradients. Values of errors are sent back from output layer, and it is saved in the cell of the LSTM entity. Such a mechanism enables continuous learning from the errors.

Reinforcement-Learning does not require labeled datasets or any supervision. The model has zero knowledge about the environment and it makes its own decisions based on trial and error approach and develops policies that will yield our rewards and punishments based on the action done for a particular state. This is done for n episodes till we get an accurate model. Generally, training in reinforcement learning is a very lengthy process as it takes a long time for values to converge to a point and it consumes a lot of memory for storing action-state pairs. This can be solved by Deep Reinforcement Learning where we use Q-value function and convolutional neural networks to predict the future rewards and actions to be taken at a particular state so that we get maximum rewards. This method minimizes memory wastage when we deal with a ample state action pairs. This study has described the working of an LSTM based value predictor which can be given as input for a DDQN to take autoscaling decisions based on the Average memory Utilization. We try to design and develop an autoscaling policy for microservices using Deep Reinforcement Learning.

2 Autoscaling Algorithms

Scaling is the ability of cloud computing resources to handle increasing or decreasing demands in a flexible and graceful manner. Scaling is classified into horizontal and vertical scaling based on how computing resources are distributed in the infrastructure. Within the framework of scalability, horizontal enlargement encompasses the integration of supplementary machines into a prevailing machine cluster, while vertical scaling encompasses the augmentation of potency or resources for an extant machine. In our project, we will be using the Horizontal Pod Autoscaler along with a Double DQN service. Based on memory being used at that moment, Horizontal Pod Autoscaler modifies replicas. In scenarios where memory consumption rises, HPA will initiate the creation of

new instances, potentially leading to a situation where the cluster lacks sufficient room. When resources are inadequate, the Cluster Autoscaler will strive to deploy additional nodes, allowing the newly spawned pods from HPA to find a suitable environment. In instances of decreased workload, HPA will deactivate some replicas. Consequently, certain nodes could experience reduced utilization or become entirely vacant, prompting the Cluster Autoscaler to terminate these surplus nodes.

Based on when scaling decisions are taken, scaling is divided into reactive scaling and proactive scaling and hybrid scaling. In reactive scaling, resources are scaled based on loads that have already appeared. In proactive scaling, resources are scaled based on loads that are expected to appear in the future. In the hybrid type, a combination of proactive and reactive scaling is used. For example, containers or virtual machines may be scaled using a reactive approach but individual pods inside the containers are scaled using a proactive approach. In our project, we use the hybrid scaling approach. We use an LSTM model to predict metric values, which will be sent to the Double DQN module to take scaling decisions.

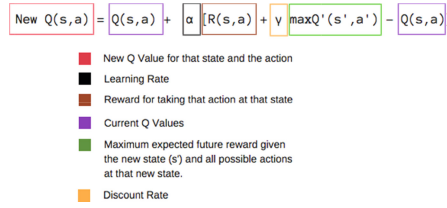
Generally, parameters like CPU usage, memory usage, requests served per unit time are the parameters used by most researchers to make scaling decisions as these are the parameters which are directly obtained from the machine and so, give us a better picture about the state of the machines which we are trying to scale. Metrics like packet rate, bandwidth, average end to end delay are obtained from the machines as well, but they tell us more about the state of the network's state than that of machine.

3 Reinforcement-Learning

It is a learning methodology that does not require labeled datasets or any supervision. Here, an agent learns to solve a problem using the trial and error method. For every action the agent performs in the environment, there is either a reward or a penalty. The agent's goal will be to take the rewards value to maximal as it learns through each and every iteration. Here, it is the programmer who sets the rules i.e., the rewards and penalties for a given problem. Another important feature of reinforcement learning is that the agent does not spend all its time on maximizing the rewards. There are times when it will settle for a smaller reward in the present so that it may get a bigger reward in the future. This act of settling for a lower reward at the present at the possibility of getting a bigger reward in the future is called exploration. On the other hand, preferring to stay with the option of getting a bigger score is called exploitation. Reinforcement learning systems need to balance both these tasks. In general, a bulk of the time, about 80%–90% is spent on exploitation and the remaining time is spent on exploration. It is the epsilon greedy strategy.

Commonly, issues resolved through Q-Learning can be framed within the context of a Markov Decision Process challenge. Markov Decision Processes encompass the following elements: a finite assortment of states, a selection of actions accessible for each state, transitions connecting states, rewards linked with each transition, and a discount factor, ranging from 0 to 1, that gauges the distinction in significance between immediate and future rewards.

Q-learning constitutes an approach that determines the optimal course of action based on a self-evaluative variable signifying the value attributed to being in a specific state or undertaking a particular action in that state. An intrinsic Q feature takes a state and an action as inputs, predicting the resultant reward, including all subsequent actions, for that action. Until an exploration phase occurs, Q maintains a constant, predetermined value. But then, when we further analyze the setting, Q gives us a good estimation of the worth of a specific action at a given state. The Q function is updated as we run more and more iterations. This is a formula that demonstrates how to solve a problem using Q-Learning (Fig. 1).

$$\text{New } Q(s, a) = Q(s, a) + \alpha [R(s, a) + \gamma \max_{a'} Q'(s', a') - Q(s, a)]$$


The diagram shows the Q-learning update equation with color-coded components:

- New Q Value for that state and the action**: Represented by a red box around the left-hand side of the equation, $\text{New } Q(s, a)$.
- Learning Rate**: Represented by a black box around the coefficient α .
- Reward for taking that action at that state**: Represented by a brown box around the term $R(s, a)$.
- Current Q Values**: Represented by a purple box around the term $Q(s, a)$ on the right-hand side.
- Maximum expected future reward given the new state (s') and all possible actions at that new state**: Represented by a green box around the term $\max_{a'} Q'(s', a')$.
- Discount Rate**: Represented by an orange box around the coefficient γ .

Fig. 1. Equation 1

These are the variables which are used in the given equation:

Alpha: This is the learning rate, which specifies how aggressive our algorithm should be when it wishes to update score values.

Reward: It is the particular reward obtained for taking up a particular action at a particular state.

Gamma: “The factor that characterizes the discount parameter is tasked with measuring the distinction in importance between the immediate rewards derived from exploitation and the time allocated for conducting exploratory actions. It is the epsilon greedy method.

However, if we only use Q-learning, it has its disadvantages concerning memory and time to make decisions, so Deep Reinforcement learning was introduced to overcome this disadvantage. Using this, we can generalize the Q-function values and rewards for the future, unlike normal Q-learning, where we need to store each every Q-function value and search for the highest Q-value from the table, which is time taking and long. So in deep RL, we have a network layer introduced where we create the network to predict the value of the upcoming Q-function and give the reward according to the q value predicted, which reduces our work. Even if we use deep Q-networks to predict the best Q-value from the Q-function, it overestimates the value as it always tries to make greedy decisions. It creates what is called maximization bias in the learning part with the overestimation causing many problems such as creating as giving importance to unnecessary decisions in policy, which will reduce the efficiency of our network. So in order to overcome this problem, a different version of DQN was introduced, which was called Double DQN. It was able to solve the problem as it used two networks, namely a Deep Q network and Target Q network, which help us to check the decisions made by the network using the target network and add that decision to the policy. This will help us to reduce the overestimations and get better Q-values. As a result, we are using this

approach for our implementation, where we take a set of actions and states for scaling based on the Q-value generated by the double DQN to scale out the application or not in the cloud platform, which efficient.

4 Related Work

Yenel et al. proposal utilizes proactive scheduling, the scaling is done by iterating through a state's set to be achieved among a given states' set, an idea which is used in our project. Podolskiy et al. propose a predictive scaling model is used, but requests per second are used as a parameter for making scaling decisions [13]. The graph-based model of the microservice application is used to balance the application capacity statically. The schedule of future scaling actions is generated according to one of the following policies: Naive, best resource pair, only-delta-load policy. However, the downside of this paper is that it is limited to microservice web applications with a homogeneous workload. Models employed are limited by the capacity of the application expressed in terms of RPS.

Kwan et al. propose a reactive scaling model can be used with CPU utilization the metric for scaling [14]. In scenarios where resource allocation surpasses usage and triggers a signal to the algorithm, it indicates the availability of untapped CPU resources for the specific microservice. For each microservice falling under this category, an endeavor is initiated to engage in downward vertical scaling, entailing the reclamation of resources from individual replicas to align the instance with the desired utilization benchmark. Instances that undergo downward vertical scaling and subsequently witness their assigned resources falling beneath a predefined minimum threshold are comprehensively removed. Conversely, the reverse procedure is followed for upward scaling. It is crucial to acknowledge that this algorithm lacks the capacity to effectively manage workloads characterized by memory constraints and crashes. Gias et al. propose a model where the length of message queues is used as a measure for scaling. Scaling decisions are taken by solving multiple LQN models. To generate these models, queueing theory is used. Baeur et al. propose a model where predictive scaling is used coupled with a reactive fallback. To attain accurate predictions using a seasonal pattern model, it is imperative to have access to a minimum of two days' historical data. This recommended methodology necessitates the integration of an external monitoring component that systematically gathers the essential values. An important drawback associated with this approach pertains to the uniformity of resources allocated for each instance of the service [16]. In the research conducted by Song et al., they advocate the substitution of a concealed RNN layer with an LSTM block. This strategic replacement enhances the ability to effectively capture and model long-term dependencies, which is critical for accurate metric value predictions aimed at facilitating scalable operations [17]. Kumar et al. propose a model where one can predict the metric values using a three-layered neural network. It uses self-adaptive differential evolution for training neurons in the neural network. However, this is a costly approach compared to the one mentioned in [17], so we have decided to go with LSTM for our project.

Li et al. propose a paper where reactive scaling is used, and the average end to end delay is used as the metric for scaling. If the average end to end latency is more than the maximum agreed upon in the SLA, scaling up is performed. The problem with this approach is that end to end delay is affected by external factors relating to networking conditions, including interference, channel errors and congestion [19].

Li et al. propose a mechanism which tries to use multiple metrics. Proactive scaling is done using packet rate, link bandwidth and the number of cores as the metric. Utilizing a linear programming paradigm alongside randomized rounding techniques, this strategy systematically and efficiently achieves an outcome that is nearly optimal in nature while also exhibiting a proactive approach to address potential challenges [20]. Hyun Woo Kim et al. propose a mechanism which uses reactive scaling; The assessment of user storage utilization involves an investigation into the monthly time aspect, allowing for the anticipation of the envisaged fixed storage volume. This evaluation of performance aspects is rooted in the analysis of clustered desktop resources [21]. The examination of performance metrics is conducted distinctly for dynamic performance parameters (such as the performance of idle CPUs, remaining memory, and storage capacities) and static performance variables (comprising the performance of the underlying desktop - installed CPU, along with the upper bounds of memory and storage capacities). Another paper proposes using a complicated context switching method, along with a hybrid mechanism and a combination of horizontal-vertical scaling [22].

Gu et al. studied application needs and their performance requirement (virtual queueing system) is used as a metric for scaling [23]. A shadow routing algorithm maintains virtual queues and a greedy Primal-Dual (GPD) applied to the virtual system. In [24], hybrid scale-up and horizontal scale-down are used. Utilizing a framework rooted in the queueing model, this strategy involves the prediction of arrival times for individual customers, subsequently leading to the determination of the minimal resource allocation necessary to meet their distinct requirements. In contrast to conventional practices involving the fixed deployment of resources according to peak loads, this approach is adept at substantially enhancing resource utilization, all the while ensuring the steadfast maintenance of service quality.

Rossi et al. propose RL based Solutions for Container Deployment and adaptation in the real-time environments of cloud services [5]. Generally, scalings are of two kinds: Horizontal and Vertical for the deployment of containers; most of them use Horizontal scaling. Fabiana Rossi, the author, combines both of these scaling methods and uses RL to apply the scaling or not under varying workload conditions. RL is model-free, and because of that, it takes a longer time to build a good model as it does not have a priori knowledge about the environment. So author Uses a Model-based approach to overcome this disadvantage so that our agent can adapt to the environment faster and better resource usage whenever there is a change in the environment. Validation of his work was done by implementing his approach in Docker Swarm, which adapts to the environment accordingly and it was also done on AWS cloud services which showed the comparisons of using A model-based solution and a model-free solution.

Yu et al. propose the Microscaler tool, which helps in identifying the microservices to be scaled and scaling them automatically [6]. The design of Microscaler is unique as it takes SLA violations as criteria to determine whether to add or remove resources

instead of resource usage, which is generally done. As we know, microservices have so many instances; we do not know which microservices are going to trigger the other. So they have used service mesh layer over the microservices to detect SLA violations and to get QoS metrics from the micro-services instances based on those metrics if there is any trigger in SLA violations by any of the services then they have used Service power to determine whether that app is underprovisioning or overprovisioning and act accordingly. As QoS gives us the detailed conditions with numbers and they have used Kubernetes to deploy microscaler on Hipster shop E-commerce services and showed that the convergence of the microscaler better than the generally used ones.

Young et al. propose an approach to deal with massive development and increasing complexity concerning development deployment and management of software [7]. They have proposed a solution that combines powerful functionality of microservices and emergent systems software to overcome this problem. Microservices are independent and running on its own, which has a container or pods which communicate with surrounding instances. Microservices are very flexible in terms of deployment; scaling and deleting was added to the emergent system software, which has three components: assembly, perception, learning. In assembly, small tightly packed building blocks of the local system are put together, and the model is developed. Perception module checks the performance of the developed model and learning uses these models to get understanding about the environment and adapt accordingly. Using this process, they have demonstrated using smart city microservice architecture that adapts to the parameters generated randomly by the system. Combining these two things can yield better results and help reduce the complexity of development.

Nanda et al. proposed a solution for container scheduling in microservice architecture using machine learning algorithms. Microservices architecture advancement over monolithic architecture and container as add-on to it making it easy to use compared to VMs in terms of deployment time or adjustments. Scheduling is done based on the load pressure of services with the help of ML using four eigenvectors of services in the current time window. These four factors help us in predicting how to schedule our container next time window. CSML proposed in the paper helps in balancing load pressure and also system performance which involves Random forest to differentiate and get a value from the input retrieved from four factors, giving an efficiency over container expansion by reduced 50% and accuracy of 10–38% higher compared to feedback CSML.

Magableh et al. present an architectural model for microservices conforming to the MAPE-K model for the purpose of self-adaptation. The proposed adaptation agent uses a Markov decision process (MDP). It uses a deep Q-learning Network (DQN) to pick the action that gives the most reward. Deep Q-learning is one type of Q-learning. It makes the function approximation with a deep neural network of many layers. Per the results, the paper claims a reasonable rate of success using its approach in performing scaling in reaction to different environmental alterations [1].

Magableh et al. provide a distributed model of a microservices architecture, in compliance with the MAPE-K (Monitor-Analyze-Plan-Execute over a Knowledgebase) model to attain self-adaptability. A central controller is used, which is backed by multi adaptation agents (MDP). These agents continuously check on the environment and take

actions that are chosen according to the results of a deep recurrent Q-network (DRQN) algorithm [2]. This paper combines DRQN and Gated Recurrent Unit (GRU) cells.

Nouri et al. propose a controller that can react to varying patterns of inflowing jobs, based on reinforcement learning and a distributed architectural base [3]. The architecture detailed is decentralized, and it can get and manage resources to meet the Service Level Agreement (SLA) goals of web-based applications, and it is also aiming to maintain the infrastructure cost as less as possible. The design and solution of the Autonomic Decentralized Elasticity Controller (ADEC) to handle elastic scaling is detailed.

Rossi et al. explore the adaptation of container-applications that are deployed in the environment with varied resources and have Quality of Service (QoS) requirements that must be met [4]. It proposes reinforcement learning (RL) based decentralized policies to adjust the application deployment, where the changes are in container migrations and elasticity. The paper first focuses on container - based applications' elasticity, concerning itself with vertical as well as horizontal scaling. The adaptation is structured per Monitor, Analyze, Plan, and Execute (MAPE) design. The control strategy considered is hierarchical in nature. Next, the paper combines the placement and elasticity of containers and elasticity of Virtual Machines (VMs).

Nanda et al. present a problem of the bin packing for multiple-resources, and proposes a deep learning-based procedure to tackle the problem of container - placement on physical machines [8]. This paper tackles the problem when containers are having varying needs for resources, and physical machines have varying capacities. The problem is converted into a problem of deep learning using a deep reinforcement learning formulation. It focuses primarily on two goals: learning every job's resource needs adaptively and tracking the resources already present.

Tozer et al. propose to improve system security by reducing the surface of attack. The paper describes a method for building systems so that it is possible to transform them into Markov Decision Processes for multi-objectives. It shows how policies are learned, using a multi-objective reinforcement learning algorithm, that aids in minimizing the attack surface of the system and executes the policies to achieve diversity while the system is running. Microservices architecture is used to implement a dynamic system that allows diversity in the configuration. Reinforcement learning algorithms for multi-objective are evaluated. The three evaluated employ Pareto dominance to assess possible actions that can be taken in any state. The first is the Pareto Q-learning algorithm. The second is a variation of SARSA for multiple objectives. The paper also develops the TD(0) temporal-difference algorithm for multiple objectives [9].

Ianucci et al. propose an approach to intrusion response by exploiting deep reinforcement learning [10]. A system based on microservices is modeled, and Deep Q-Learning and Q-Learning are evaluated and compared. A Markov Decision Process is used to model the system. Experimental evaluation of the proposed approach shows that it takes less time for calculating the policies while giving almost-optimal rewards. For future work, the paper proposes that General Purpose Graphical Processing Units (GPGPUs) can use deep reinforcement learning. It also proposes extending the approach presented in this paper to situations having many attackers and many defenders.

Belhaj et al. describe a broad framework for creating component-based applications that can adapt themselves to changing situations [11]. It also provides a way to integrate

self-adaptive behavior into an existing application so that it will learn the best policy when it is running. The framework is designed as an autonomic container. The process of taking decisions is modeled as a finite MDP (Markov Decision Process). Experimental results show a minimal extra cost, and the application adjusts well to ever-changing environmental conditions, as well as the reduction in Service Level Agreement (SLA) faults while learning, and improved behavior of the application once it converges.

5 Methodology

The typical LSTM entity has a cell, and three gates which are input, output, and forget. The cell is for holding values in memory over time, and the gates oversee information flow to and from the cell. Different types of the LSTM entity may not have some of the gates, or they may also have different gates.

Theoretically, traditional RNNs observe dependencies in the input sequences, which are for long periods of time. Practically, when using back-propagation to train a traditional RNN, the back-propagated gradients may lean towards either of two extremities, zero or infinity, due to the nature of the calculations done. LSTM entities can partly resolve this issue of the vanishing gradient, because they can also let the gradients to pass unvaried. LSTM networks may however face the problem of the exploding gradient.

The approach of Double Deep Q-Network (DQN) revolves around the utilization of a pair of distinct Q-value estimators, with each estimator alternately updating the other. This unique configuration aims to provide unbiased Q-value assessments for actions chosen based on the alternative estimator. By doing so, the issue of overestimation, often encountered in traditional DQN implementations, is effectively mitigated. In standard DQN, an agent might consistently select suboptimal actions due to the highest Q-value criterion. This challenge is addressed through the application of the subsequent equations. Q-function is used in selecting the best action a with max Q-value among the next state.

$$a = \max_a Q(s_{t+1}, a) \quad (1)$$

Q' is used for calculating expected Q-value by using the action a .

$$q_{estimated} = Q'(s_t + 1, a) \quad (2)$$

Updating Q function by using the expected Q-value of Q' function

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(R_{t+1} + \gamma Q'(s_{t+1}, a) - Q(s_t, a_t)) \quad (3)$$

Average memory utilization is a parameter which can be fetched from the kubernetes at any time span of the process. so this parameter helps us to know the current running status of the containers. Memory utilization of a particular pod in the Node which is very useful for scaling of microservices if any containers is under utilized or over utilized the system can identify those containers and scale the node accordingly. Basing on these we always try to keep container utilization around 30–70% of its total memory limit so that containers are utilized more and less resources consumption is done.

The flow of the Double DQN module starts with Kubernetes, which has different metrics to monitor the performance of the environment that is essential for maintaining the cost-effectiveness of services. We monitor average memory used by each pod in their respective containers after a particular amount of time, and we transfer the collected data to the LSTM model which is used for predicting values for the given input for the future.

With the help of LSTM, we get the next memory utilization values. This will trigger the deep Reinforcement Learning model which uses Double Deep Q-Networks which will help us get Q-values without overestimation unlike in DQN. Table 1 shows the states-actions-rewards behavior for the proposed reinforcement learning solution. The set of states is, $S = S1, S2, S3$, where each state is representative of the memory utilization of a container. Then $S = S1$: $\text{MemUse} < 30\%$, $S2$: $30\% \leq \text{MemUse} \leq 70\%$, $S3$: $\text{MemUse} > 70\%$. The set of actions is, $A = ++$: scale up, $-$: scale down. The set of rewards is, $R = -1, 0, +1, +3$, where a higher positive value indicates a positive favorable reward. The table shows the possible actions that may be taken at each state, the possible states that the actions may lead to, and the associated reward for each. $S2$ is considered to be the most favorable state, assuming optimal memory utilization, and actions that lead to it have the highest reward. Conversely, $S1$ has poor memory utilization, and has poor reward. $S3$ may lead to overburdening of the memory, so even that has a poor reward.

Table 1. Transition states in the problem

Current State	Action	Net State	Reward
S1	-	S1	0
S1	-	S2	+3
S1	-	S3	+1
S1	++	S1	0
S2	-	S2	+3
S2	-	S3	-1
S2	++	S1	-3
S2	++	S2	+3
S3	-	S3	-1
S3	++	S1	0
S3	++	S2	+3
S3	++	S3	-1

6 Experimentation and Results

We have used a Long short-term memory (LSTM)-network for prediction of the next arrival time of requests. The dataset contained 172000 samples. Of these, it was divided into a standard training/testing split in the ratio 80:20. The single feature of timestamp was considered for performing prediction. The data was standardized by the use of Z-score normalization. 500 previous samples were used as a reference for predicting the next immediate value of timestamp. An LSTM neural network model was built to solve this regression problem. The LSTM built had a sequential input layer, an LSTM layer, and a dense layer at the output. The training and testing of the model were done on Google Colab. Keras and Tensorflow were used to handle the neural network.

In analyzing the early stages of training, we saw the model's prediction was way off from expected and intended values. When we looked at the behavior of training and validation loss for the LSTM-based model, we noticed a sharp drop within 4 epochs. After that, while the validation loss didn't reduce to the level of the training loss, it did seem to be decreasing very slowly. Further better observations can be found with tuning the model parameters, taking approaches to mitigate overfitting such as getting more samples, reducing network size.

Commenting on the variation in training and validation loss with a simple RNN model, the behavior is more varying than that with the LSTM. The validation loss takes more time than the LSTM to near the training loss, but it doesn't seem to be converging. More training, or slight modifications in the network itself might provide even better results.

Table 2. Network architecture of the Q-Network and the Target Q Network

Layer (Type)	Output Shape	Number of parameters
dense ₁ (<i>Dense</i>)	(None, 32)	192
dense ₂ (<i>Dense</i>)	(None, 16)	528
dense ₃ (<i>Dense</i>)	(None, 8)	136
dense ₄ (<i>Dense</i>)	(None, 2)	18

To test our Double DQN module, we have used a Random number generator for each episode to obtain average memory utilization used by each container. We ran 7000 episodes in total for the experiment and we noted down the reward obtained in each episode and the discounted reward in each episode. We used Random number generate because currently we don't have data from kubernetes attached to the model so To balance exploration and exploitation which help in better learning of the agent, The hyperparameters we used in our model are following we have used the epsilon greedy strategy with the minimum and maximum values of epsilon being 0.01 and 1.00 respectively and the epsilon decay rate being 0.99. We have used a discounting factor (the factor used to discount future rewards to current step) of 0.9 and a learning rate of 0.001. Each episode runs 999 steps as we wanted each episode to explore the action space as much as possible.

We have used the MLP DNN architecture in both the Q and the Q target networks and both of them have the same structure. The input layer has as many nodes as the number of states, i.e. 3. The output layer has as many layers as action space cardinality follows i.e. 2 layers. The ReLU activation function is implemented across input and hidden layers, while linear activation is employed for output layers. The neural network architecture encompasses three intermediate layers in total. The initial hidden layer generates an overall output of 32 units, succeeded by a subsequent hidden layer with 16 units, and ultimately culminating in a third hidden layer yielding a combined total of 8 units. We have used a total of 874 weights in each of these networks, with all of them being trainable. We have used the ADAM optimizer and Mean Square Error method to calculate losses from Table 2 (Figs. 2, 3, 4 and 5).

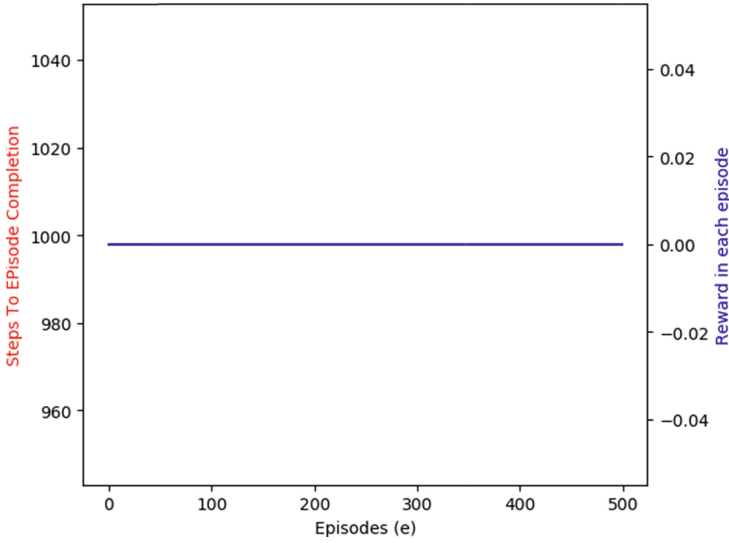


Fig. 2. Graph showing total rewards when the agent stays in S1 throughout all episodes

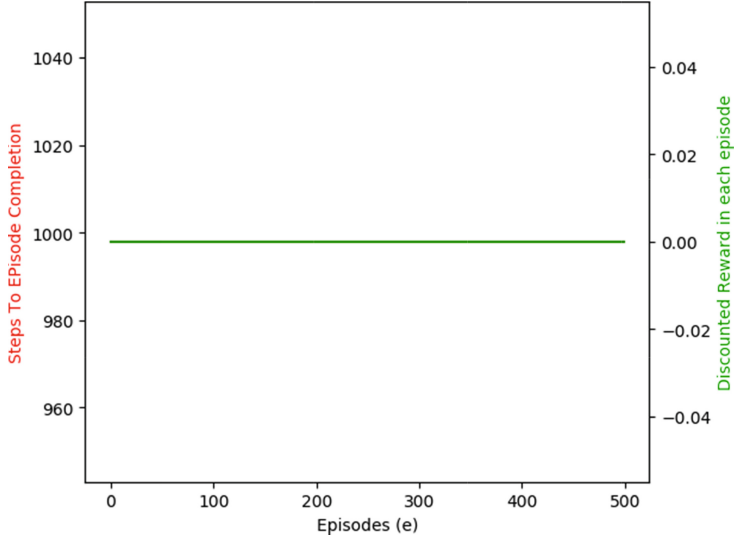


Fig. 3. Graph showing discounted rewards when the agent stays in S1 throughout all episodes

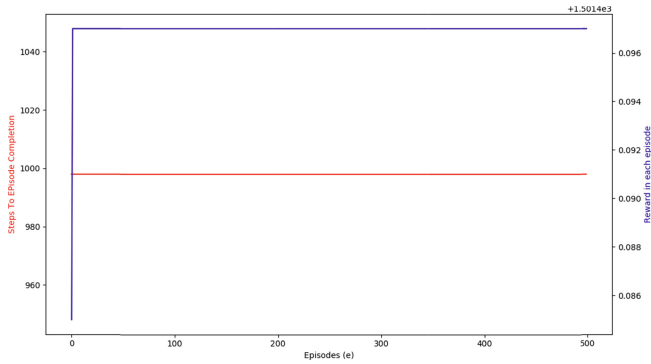


Fig. 4. Graph showing total rewards when the agent stays in S2 throughout all episodes

Figure 6 shows the total reward obtained when the agent stays in S1 throughout all episodes obtained for each episode (in blue) and the number of iterations (steps) in each of those episodes (in red). In this graph, the blue line overlaps the red line so it is not to be seen. From the graph, we can see that in all the episodes, the reward is same. This is because, according to Table 1, a reward of 0 is given when the agent moves from S1 to itself, irrespective of whether upscaling or downscaling is performed.

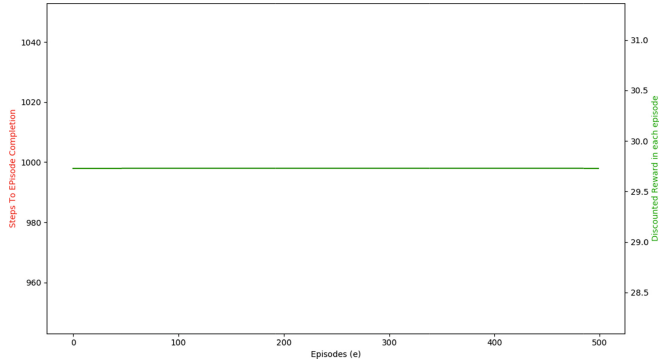


Fig. 5. Graph showing discounted rewards when the agent stays in S1 throughout all episodes

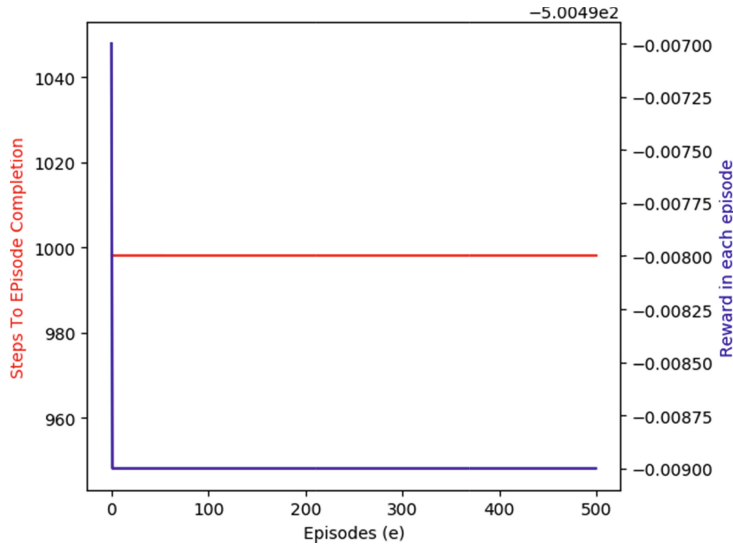


Fig. 6. Graph showing total rewards when the agent stays in S3 throughout all episodes (Color figure online)

Figure 7 shows the discounted reward obtained when the agent stays in S1 throughout all episodes obtained for each episode (in green) and the number of iterations (steps) in each of those episodes (in red). In this graph, the blue line overlaps the red line so it is not to be seen. From the graph, we can see that in all the episodes, the discounted reward is the same. This is because, according to Table 1, a reward of 0 is given when the agent moves from S1 to itself and by multiplying the discount factor with 0, the result will not change.

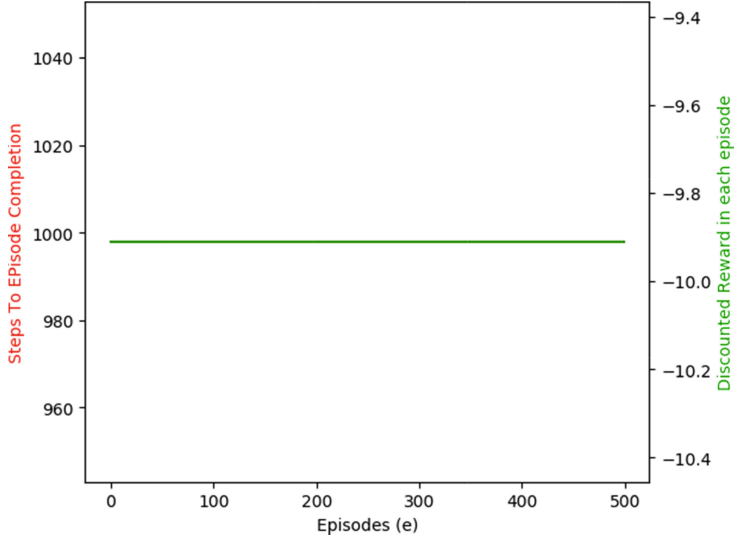


Fig. 7. Graph showing discounted rewards when the agent stays in S1 throughout all episodes (Color figure online)

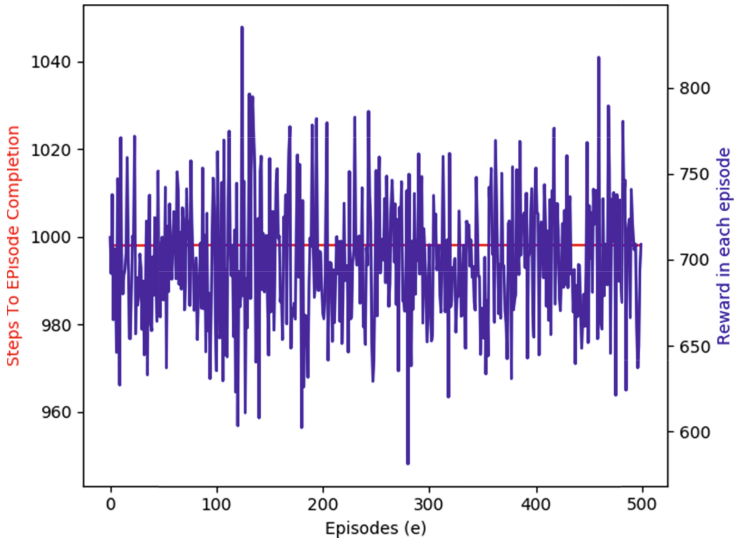


Fig. 8. Graph showing total rewards when the agent goes from one state to another throughout all episodes but spends most time in S2 (Color figure online)

Figure 8 shows the total reward obtained when the agent stays in S2 throughout all episodes obtained for each episode (in blue) and the number of iterations (steps) in each of those episodes (in red). From the graph, we can see that in all the episodes, the reward is the same i.e., 1500. This is because, according to Table 1, a reward of +3 is given when

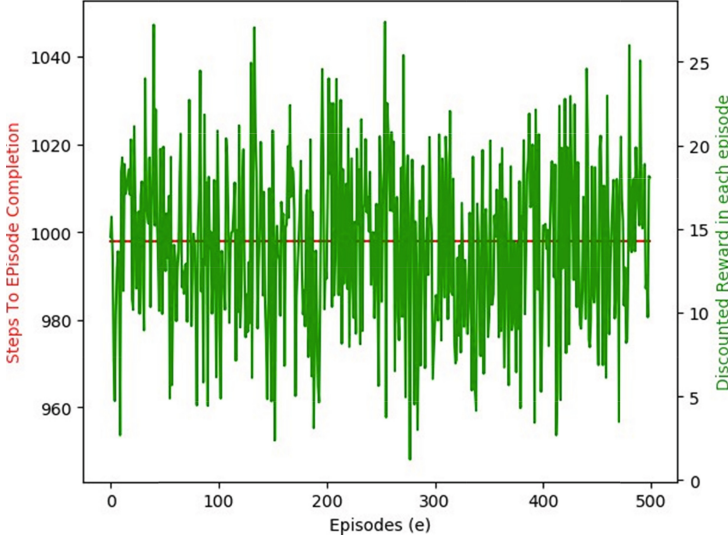


Fig. 9. Graph showing discounted rewards when the agent goes from one state to another throughout all episodes but spends most time in S2 (Color figure online)

the agent moves from S2 to itself, irrespective of whether upscaling or downscaling is performed.

Figure 9 shows the discounted reward obtained when the agent stays in S2 throughout all episodes obtained for each episode (in green) and the number of iterations (steps) in each of those episodes (in red). From the graph, we can see that in all the episodes, the reward is the same. This is because, according to Table 1, a reward of +3 is given when the agent moves from S2 to itself, irrespective of whether upscaling or downscaling is performed. When we multiply the total reward with the discounting factor, we get a constant value.

Figure 10 shows the total reward obtained when the agent stays in S3 throughout all episodes obtained for each episode (in blue) and the number of iterations (steps) in each of those episodes (in red). From the graph, we can see that in all the episodes, the reward is the same i.e., -500. This is because, according to Table 1, a reward of -1 is given when the agent moves from S3 to itself, irrespective of whether upscaling or downscaling is performed.

Figure 11 shows the total reward obtained when the agent stays in S3 throughout all episodes obtained for each episode (in green) and the number of iterations (steps) in each of those episodes (in red). From the graph, we can see that in all the episodes, the reward is the same i.e., -500. This is because, according to Table 1, a reward of -1 is given when the agent moves from S3 to itself, irrespective of whether upscaling or downscaling is performed. When we multiply the total reward with the discounting factor, we get a constant value.

Figure 12 shows the total reward obtained when the agent goes to different episodes throughout all episodes, with a favouring to S2 though. S2 was favoured to show that the graph is not entirely random and is dense in the middle, which is close to real

life situations. The total reward obtained for each episode (in blue) and the number of iterations (steps) in each of those episodes (in red) is presented in the graph. From the graph, we can see that in all the episodes, the reward is different, indicating that the rewards mechanism did work even when the agent moved from one state to another.

Figure 13 shows the discounted reward obtained when the agent goes to different episodes throughout all episodes, with a favouring to S2 though. As explained above, S2 was favoured to show that the graph is not entirely random and is dense in the middle, which is close to real life situations. The total reward obtained for each episode (in green) and the number of iterations (steps) in each of those episodes (in red) is presented in the graph. From the graph, we can see that in all the episodes, the discounted reward is different as well, indicating that the discounting rewards mechanism did work even when the agent moved from one state to another.

From these graphs, it is clear that the agent was correctly rewarded as per the proposed rewarding method in Table 1.

7 Conclusion and Future Work

In this paper, we have worked on an LSTM predictor for predicting parameter values. From our experiments with the LSTM predictor and the RNN predictor, we can conclude that the LSTM predictor is a better choice as the RNN predictor is taking more time to converge to a value.

We have also worked on an autoscaling algorithm using Double DQN. It was able to solve the problem as it used two networks, namely a Deep network and Target network, which help us to check the decisions made by the network using the target network and add that decision to the policy. This will help us to reduce the overestimations and get better Q-values. In the future, we will simulate the algorithm in the Kubernetes autoscaler which will help us decide the optimal values for constants like learning rate, discounting factor, epsilon decay rate other hyperparameters and better reward functions.

References

1. Magableh, B., Almiani, M.: Deep Q learning for self-adaptive distributed microservices architecture. *IEEE Access* (2019, in Press)
2. Magableh, B.: A deep recurrent Q network towards self-adapting distributed microservices architecture. *CoRR*, cs.SE, [arXiv:1901.04011](https://arxiv.org/abs/1901.04011). <https://doi.org/10.21427/c4v9-an45>. Accepted Jan 2019
3. Nouri, S.M.R., Li, H., Venugopal, S., Guo, W., He, M., Tian, W.: Autonomic decentralized elasticity based on a reinforcement learning controller for cloud applications. *Future Gener. Comput. Syst.* **94**, 765–780 (2019)
4. Rossi, F.: Self-management of containers deployment in decentralized environments. In: 2019 IEEE World Congress on Services (SERVICES), Milan, Italy, pp. 315–318 (2019)
5. Rossi, F., Nardelli, M., Cardellini, V.: Horizontal and vertical scaling of container-based applications using reinforcement learning. In: 2019 IEEE 12th International Conference on Cloud Computing (CLOUD), Milan, Italy, pp. 329–338 (2019). <https://doi.org/10.1109/CLOUD.2019.00061>

6. Yu, G., Chen, P., Zheng, Z.: Microscaler: automatic scaling for microservices with an online learning approach. In: 2019 IEEE International Conference on Web Services (ICWS), Milan, Italy, pp. 68–75 (2019). <https://doi.org/10.1109/ICWS.2019.00023>
7. Young, M.: The Technical Writer's Handbook. University Science, Mill Valley (1989)
8. Nanda, S., Hacker, T.J.: RACC: resource-aware container consolidation using a deep learning approach. In: Proceedings of the First Workshop on Machine Learning Computing System—MLCS18, pp. 1–5 (2019)
9. Tozer, B., Mazzuchi, T., Sarkani, S.: Optimizing attack surface and configuration diversity using multi-objective reinforcement learning. In: 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Miami, FL, pp. 144–149 (2015)
10. Iannucci, S., Barba, O.D., Cardellini, V., Banicescu, I.: A performance evaluation of deep reinforcement learning for model-based intrusion response. In: 2019 IEEE 4th International Workshops on Foundations and Applications of Self* Systems (FAS*W), Umea, Sweden, pp. 158–163 (2019)
11. Belhaj, N., Belaïd, D., Mukhtar, H.: Framework for building self-adaptive component applications based on reinforcement learning. In: 2018 IEEE International Conference on Services Computing (SCC), San Francisco, CA, pp. 17–24 (2018)
12. Yenel, A., Podolskiy, V., Gerndt, M.: Predictive auto scaling scheduling application (2018)
13. Podolskiy, V., Jindal, A., Ramirez, Y., Yenel, A., Ahmed, S., Gerndt, M.: SCALENDAR: predictive autoscaling engine for microservice applications in cloud (2019). <https://doi.org/10.13140/RG.2.2.18399.51361>
14. Kwan, A., Wong, J., Jacobsen, H., Muthusamy, V.: HyScale: hybrid and network scaling of dockerized microservices in cloud data centres. In: 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), Dallas, TX, USA, 2019, pp. 80–90 (2019)
15. Gias, A.U., Casale, G., Woodside, M.: ATOM: model-driven autoscaling for microservices. In: 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), Dallas, TX, USA, pp. 1994–2004 (2019)
16. Bauer, A., Lesch, V., Versluis, L., Ilyushkin, A., Herbst, N., Kounev, S.: Chamulteon: coordinated auto-scaling of microservices (2019). <https://doi.org/10.1109/ICDCS.2019.00199>
17. Song, B., Yu, Y., Zhou, Y., et al.: J. Supercomput. **74**, 6554 (2018). <https://doi.org/10.1007/s11227-017-2044-4>
18. Kumar, J., Singh, A.K.: Future Gener. Comput. Syst. **81**, 41–52 (2018)
19. Toosi, A.N., Son, J., Chi, Q., Buyya, R.: ElasticSFC: auto-scaling techniques for elastic service function chaining in network functions virtualization-based clouds (2019). <https://doi.org/10.1016/j.jss.2019.02.052>
20. Li, Y., Phan, L.T.X., Loo, B.T.: Network functions virtualization with soft real-time guarantees. In: IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications, San Francisco, CA, pp. 1–9 (2016)
21. Kim, H.-W., Jeong, Y.-S.: Efficient auto-scaling scheme for rapid storage service using many-core of desktop storage virtualization based on IoT, **209**, 67–74 (2016)
22. Hadley, J., Roedig, U., Elkhatib, Y.: Using context switches for VM scaling. In: 2016 IEEE 35th International Performance Computing and Communications Conference (IPCCC), Las Vegas, NV, pp. 1–2 (2016)
23. Guo, Y., Stolyar, A., Walid, A.: Online VM auto-scaling algorithms for application hosting in a cloud. IEEE Trans. Cloud Comput. (2018)
24. Huang, G., et al.: Auto scaling virtual machines for web applications with queueing theory. In: 2016 3rd International Conference on Systems and Informatics (ICSAI), Shanghai, pp. 433–438 (2016)



Revolutionizing Dental Treatment Through IoT Integrated Force Sensors: Design and Calibration

Chaithra Salian¹, Prakyath Shetty¹, Charishma Shetty¹, Durga Prasad¹(✉),
G. R. Pradyumna¹, K. B. Bommegowda¹, M. S. Ravi², and P. S. Murali²

¹ Department of Electronics and Communication Engineering, NMAM Institute of Technology,
NITTE (Deemed to be University), Nitte 574110, Karnataka, India

{shettycharishma, durgaprasad, pradyumna, bgowda_kbl}@nitte.edu.in

² A B Shetty Memorial Institute of Dental Sciences, NITTE (Deemed to be University),
Mangaluru, Karnataka 575018, India

{drmsravi, drmuralips}@nitte.edu.in

Abstract. The measurement of bite force stands a crucial parameter in assessing oral health and functional capabilities. This research expands on this significance by delving into the calibration of the Flexiforce A301 sensor, specialized for bite force measurement, and its interfacing with the Raspberry Pi Pico W microcontroller. Through calibration, the study uncovers a clear linear relationship between applied force and resulting resistance changes. This calibration process is optimized by utilizing low drive voltage, which in turn informs the optimal configuration for an amplifier circuit with a 100 k Ω feedback resistor. Notably, with a 20 kg force application, this setup yields a voltage output of 0.66 V, effectively enabling the sensor to accurately measure forces up to 100 kg. The study highlights precise calibration's importance for accurate measurements and the practical integration of calibrated sensors with microcontrollers to enhance dental and medical applications, enabling real-time monitoring and analysis in bite force measurement.

Keywords: Bite force · force – sensing resistor · IoT · data acquisition

1 Introduction

In the world of dentistry, a technological revolution is underway. Computer-based technologies have transformed dental care, enhancing accuracy, efficiency, and patient comfort [1]. These innovations, including the Internet of Medical Things (IoMT), link dental devices to gather and transmit patient data, improving diagnosis and treatment planning. Intraoral cameras, digital radiography sensors, and 3D printers are part of IoMT in dentistry, elevating patient care and practice management [2].

By seamlessly integrating wireless devices and sensors, healthcare professionals can now monitor patients' well-being in real-time, a critical advancement in early disease identification and preventive care [3, 4]. The concept of teledentistry takes this progress

further, making it possible to deliver dental services remotely. This groundbreaking method employs tools like video conferencing, digital imaging, and remote monitoring to offer dental consultations and treatments to individuals who may not have access to in-person dental care [5]. Teledentistry, an emerging and evolving field, has the potential to revolutionize dental care by widening its scope and making it accessible to a more diverse range of patients [6].

Bite force measurement is a pivotal advancement in dentistry, offering crucial insights into oral health and masticatory function. It holds clinical significance, aiding in dental evaluations, prosthetic adjustments, and diagnosing disorders like temporomandibular issues. Accurate measurements empower healthcare professionals to make informed decisions, leading to improved treatment strategies and patient wellbeing. This innovation underscores bite force's broader applicability in medical and dental contexts, enhancing diagnostics and treatment outcomes [7–9].

The proposed bite system uses the versatile Flexiforce A301 sensor as an affordable, adaptable, and user-friendly force and pressure sensing solution when compared to load cell [10], strain gauge sensor [11] and capacitive type pressure mapping sensor [12]. Unlike bulkier load cells, the Flexiforce A301's compact and flexible design simplifies integration, even in tight spaces. In comparison to traditional strain gauge sensors [13, 14], it offers a simplified electronics interface for easy measurement with basic tools, reducing the need for complex components.

Our system innovatively incorporates a low drive voltage concept for calibrating the Flexiforce A301 sensor, enabling accurate force measurements of up to 100 kg. The system's versatility and increased measurement range make it suitable for more than just its primary intended use, opening up possibilities for its utilization in numerous other areas ranging from medical applications, sports equipment testing, to industrial machinery, among others.

IoT and force sensors in dentistry mean real-time procedure adjustments for better treatment. Sensor data with digital systems streamlines workflows. Success depends on specificity, environment, and technical expertise, with potential to revolutionize dental care. Steps involve sensor selection, hardware design, wireless integration, and software, creating intelligent dental tools for improved patient care.

2 Design and Methodology

The Flexiforce sensor is a slim, adaptable force-sensing resistor used in robotics, medicine, and automobiles for precise force and pressure measurement. With a sleek design (0.203 mm thick) and a 9.53 mm sensing area, it ensures accuracy. It can measure forces from 4 N to 4448 N, operates in temperatures from -40°C to 60°C , and responds in less than 5 μsec . This structure comprises two polyester substrates, with a layer of conductive silver applied to each, and a subsequent layer of pressure-sensitive ink [15]. Its reliability and versatility make it suitable for various industries. Additionally, a custom 3D printed puck illustrated in Fig. 1 is employed to support weight on the A301 Flexiforce Sensor's sensing area, with the puck's upper part (9 mm diameter, 12.7 mm height) securely attached to the sensor and the lower part (21 mm diameter, 2 mm height) designed for pressure application.

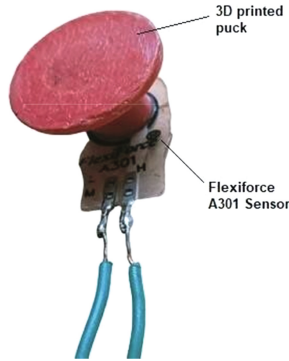


Fig. 1. 3D printed puck

Figure 2 shows the connection between the Flexiforce A301 sensor and the LM358P. This setup enhances signal strength for easier analysis. The LM358P op-amp is vital for accurate amplification and conditioning of the signal, ensuring reliable observation. This configuration enables efficient signal amplification and monitoring, facilitating precise analysis for diverse applications.

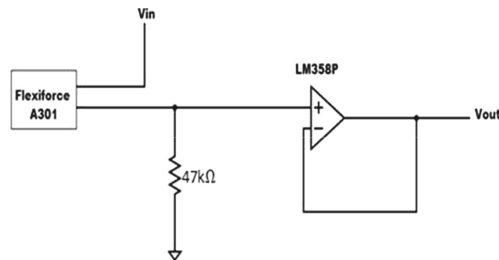


Fig. 2. Flexiforce A301 Sensor Connection with LM358P Op-Amp

2.1 Obtaining Low Drive Voltage

The Flexiforce A301 sensor measures forces up to 4,448 N (1,000 lb). Adjustments can extend its range. Using a low drive voltage enhances its piezoresistive sensing, accurately detecting force changes. This maintains linearity and captures both small and large forces. A low drive voltage optimizes accuracy, reduces power consumption, and avoids self-heating. For obtaining higher forces, lower drive voltages like -0.5 V or -0.25 V and low feedback resistor resistance is used, but not below 1 k Ω for proper functionality [16]. Negative voltage converter using ICL7662 depicted in Fig. 3 generates required lower drive voltage of -5 V.

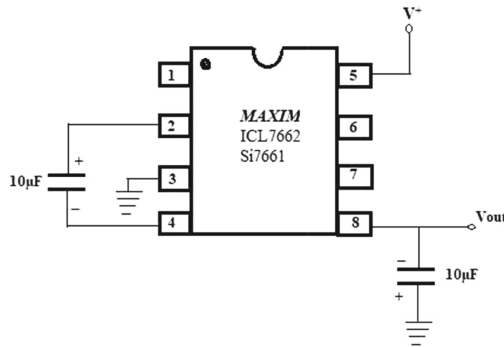


Fig. 3. ICL7662 Negative Voltage Converter for Flexiforce A301 Sensor Setup

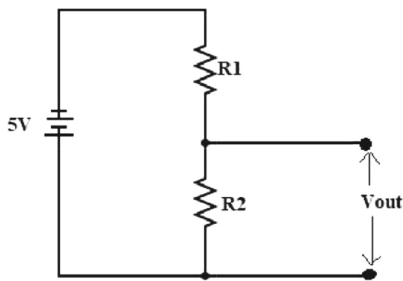


Fig. 4. Voltage divider circuit

The generated -5 V is fed to the voltage divider circuit shown in Fig. 4. Table 1 shows reference voltages for two setups. With $R_1 = 1\text{ k}\Omega$ and $R_2 = 122\text{ k}\Omega$, the output is about -0.49 V . For $R_1 = 2\text{ k}\Omega$ and $R_2 = 110\text{ k}\Omega$, the output is approximately -0.22 V . These adjustments establish the desired low drive voltage for the sensor, ensuring precise measurements within the specified range.

Table 1. V_{out} values for combination of R_1 and R_2

$V_{\text{in}}\text{ (V)}$	$R_1\text{ (}\Omega\text{)}$	$R_2\text{ (}\Omega\text{)}$	$V_{\text{out}}\text{ (V)}$
-5	1k	122	-0.49
-5	2k	110	-0.22

To enhance the circuit’s performance, a range of feedback resistors is introduced in conjunction with the LM358P op-amp as shown in Fig. 5. The objective is to determine the optimal resistor for achieving the highest precision in force measurement while aligning with the fundamental graph. This calibration spans from 0 to 20 kg. The connection between the Flexiforce sensor and the LM358P IC, employing a feedback resistor R_f ,

is illustrated in Fig. 5. The sensor's resistance is denoted as R_s . The circuit's assessment involves the use of two reference voltages: -0.49 V and -0.22 V, each paired with varying R_f values. The behavior of this circuit is mathematically expressed by an Eq. (1).

$$V_{out} = -V_{ref} \frac{R_f}{R_s} \quad (1)$$

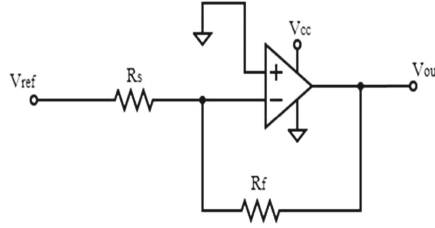


Fig. 5. LM358P Op-Amp Circuit Calibration for Precision Force Measurement with Flexiforce Sensor (R_s).

An optimal interfacing of the sensor with the Raspberry Pi Pico W involves selecting the most appropriate feedback resistor and utilizing a compatible low drive voltage. This configuration facilitates the automatic storage of force values, which can subsequently be harnessed for in-depth analysis and seamless wireless transmission.

2.2 Interfacing of Flexiforce A301 Sensor with Raspberry Pi Pico W Along with LM358P

The Flexiforce A301 sensor is integrated with the Raspberry Pi Pico W using an LM358P amplifier circuit as shown in Fig. 6. This circuit choice with a $100\text{ k}\Omega$ feedback resistor and a -0.49 V reference voltage offers excellent linearity. The amplifier is powered by 5 V from V_{cc} and connects to GPIO 26 on the Raspberry Pi. A system-started Raspberry Pi is supported by a switch between GPIO 20 and ground.

Using a suitable Micropython code an effective connection between the Flexiforce sensor and the Raspberry Pi Pico W is done, achieving accurate voltage level conversion for precise measurements with the help of the code and circuit setup.

At first the code enables the acquisition of data from the Flexiforce A301 sensor, with the collected values automatically stored in a CSV file for future wireless data transmission. The code allows us to derive voltage readings corresponding to different applied forces. By plotting a graph of Force versus Voltage, we can observe a clear relationship between the two variables. From this graph, we derive an equation that serves as a reliable method to calculate force values based on voltage measurements.

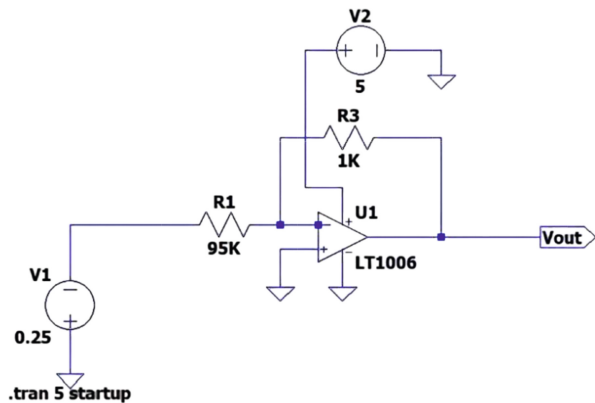


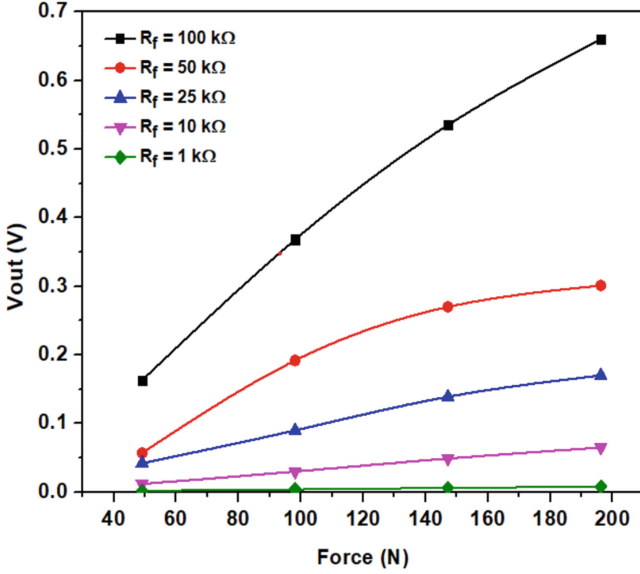
Fig. 7. Schematic diagram with feedback resistor and negative voltage reference

Table 2. Experimental output voltage for different R_f and V_{ref}

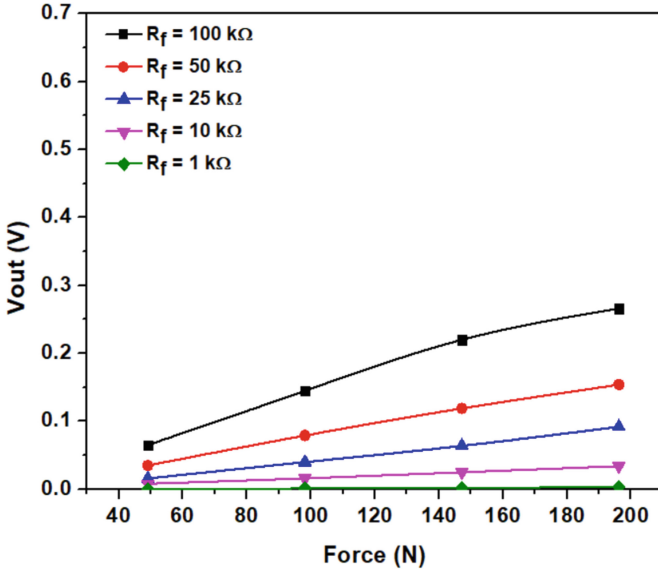
Force (kg)	Force (N)	R _s (kΩ)	V _{out} (V) for different values of R _f				
			100 kΩ	50 kΩ	25 kΩ	10 kΩ	1 kΩ
For V _{ref} = −0.49 V							
5	49.05	345	0.163	0.057	0.042	0.012	0.002
10	98.1	160	0.368	0.192	0.09	0.03	0.004
15	147.15	110	0.535	0.27	0.139	0.049	0.006
20	196.2	95	0.66	0.301	0.17	0.065	0.008
For V _{ref} = −0.22 V							
5	49.05	345	0.065	0.035	0.016	0.008	0
10	98.1	160	0.145	0.079	0.04	0.016	0.001
15	147.15	110	0.22	0.119	0.064	0.025	0.002
20	196.2	95	0.266	0.154	0.092	0.034	0.003

However, the plot achieved with $R_f = 100$ k Ω and $V_{ref} = -0.49$ V demonstrates that at a 20 kg load, the reading registers as 0.46 V. This observation implies that the circuit’s capacity to capture output voltage can extend to a range of up to 100 kg. Consequently, for further analysis, the configuration employing $R_f = 100$ k Ω and $V_{ref} = -0.49$ V is selected as the optimal choice.

The output voltage obtained by interfacing sensor to Raspberry Pi Pico W using $-V_{ref}$ is represented in Table 3 and the linear regression plot of Force (N) versus V_{out} (V) is depicted in Fig. 9. The graph illustrates a linear increase in values, indicating a proportional relationship between the applied force and the measured output voltage.



(a) Force vs V_{out} for $V_{ref} = -0.49$ V



(b) Force vs V_{out} for $V_{ref} = -0.22$ V

Fig. 8. Experimental results of Force Vs V_{out} (a) for $V_{ref} = -0.49$ V, (b) for $V_{ref} = -0.22$ V

Table 3. Output voltage obtained by interfacing sensor to Raspberry Pi Pico W

Force (kg)	Force (N)	V _{out} (V)
1	9.81	0.0192
2	19.62	0.0248
3	29.43	0.0282
4	39.24	0.0451
5	49.05	0.0666
6	58.86	0.096
7	68.67	0.11
8	78.48	0.1669
9	88.29	0.1926
10	98.1	0.2354
11	107.91	0.2499
12	117.72	0.2617
13	127.53	0.2752
14	137.34	0.2936
15	147.15	0.3151
16	156.96	0.3538
17	166.77	0.3882
18	176.58	0.417
19	186.39	0.4415
20	196.2	0.4952

The relationship between force and voltage is established through careful analysis of the graph. By analyzing the data points, an equation is derived that accurately describes this relationship as follows: $\text{Voltage} = (0.0026 * \text{Force}) - 0.0415$. With this equation, we can conveniently convert voltage readings back into force values. This conversion process allows us to obtain precise force measurements when pressure is applied to the sensor and is stored in the CSV file, as demonstrated in the corresponding Fig. 10. This approach enables us to utilize the force-voltage equation to reliably record force values from various pressure applications on the sensor. These valuable force readings can then be used in dental procedures or other relevant applications, offering crucial insights to dental professionals, and enhancing overall treatment outcomes.

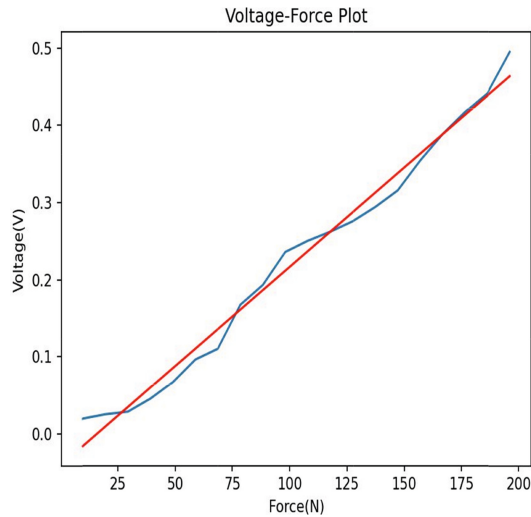


Fig. 9. Linear regression plot for Force vs Vout

Force(N)
33.624172
42.300558
52.216420
58.413835
59.343448
84.442987

Fig. 10. Voltage-to-Force Conversion in Sensor Data

4 Conclusion

This study focuses on calibrating the Flexiforce A301 sensor for precise measurements and integrating it seamlessly with the Raspberry Pi Pico W microcontroller. Calibration and optimization of the sensor through low drive voltage and a 100 kΩ feedback resistor, establishes a strong correlation between applied force and resistance changes. The achievement of accurate measurements, notably a voltage output of 0.46 V for a 20 kg force, demonstrates the sensor’s capability to gauge forces up to 100 kg. Calibration techniques ensured accurate measurements across a broad force spectrum, highlighting improved treatment planning, data-driven decisions, and dental research. Successful implementation requires careful consideration of application needs and technical capabilities.

Acknowledgment. The authors are greatly indebted for the financial assistance provided by the Indian Council of Medical Research (ICMR), GoI, under Ad-hoc Scheme, Grant No. 202110219 dated 16-11-2022 and also acknowledge with thanks for the support and the encouragement of

the management of NITTE (Deemed to be university), NMAM Institute of Technology, Nitte and A. B. Shetty Memorial Institute of Dental Sciences, Mangalore for carrying out this work.

Conflicts of Interest. Authors do not have a conflict of interest.

References

1. Raja'a, M., Al-Mussawi, A., Farid, F.: Computer-based technologies in dentistry: types and applications. *J. Dent. (Tehran)* **13**(3), 215–222 (2016)
2. Salagare, S., Prasad, R.: An overview of internet of dental things: new frontier in advanced dentistry. *Wirel. Pers. Commun.* **110**, 1–27 (2019)
3. Ganesh, B., Sugumar, K.: Internet of Things—a novel innovation in dentistry. *J. Adv. Oral Res.* **12**(1), 42–48 (2021)
4. Lanzara, R., Kumar, D., Viswambaran, M.: Internet of Things: Alexa application in dentistry: a futuristic review. *Indian J. Appl. Res.* **11**, 1–3 (2021)
5. Jha, K., Mandal, Y., Jnaneswar, A., Kumar, G., Suresan, V., Singh, A.: Teledentistry and advancements in traditional dental care. *J. Pharm. Res. Int.* **33**(64B), 515–24 (2021)
6. Salagare, S., Prasad, R.: Internet of Dental Things (IoDT). Intraoral wireless sensors, and teledentistry: a novel model for prevention of dental caries. *Wirel. Pers. Commun.* **123**, 3263–3274 (2021)
7. Koc, D., Dogan, A., Bek, B.: Bite force and influential factors on bite force measurements: a literature review. *Eur. J. Dent.* **4**(2), 223–232 (2010)
8. Lepley, C.R., Throckmorton, G.S., Ceen, R.F., Buschang, P.H.: Relative contributions of occlusion, maximum bite force, and chewing cycle kinematics to masticatory performance. *Am. J. Orthod. Dentofac. Orthop.* **139**, 609–6013 (2011)
9. Verma, T.P., Kumathall, K.R., Jain, V., Kumar, R.: Bite force recording devices – a review. *J. Clin. Diagn. Res.* **11**(9), 1–5 (2017)
10. Suppelt, S., Chadda, R., Schafer, N., Sader, R., Kupnik, M.: Sensor for bilateral human bite force measurements. In: 2021 IEEE Sensors, Sydney, Australia, November 31–October 3, pp. 1–4. IEEE (2021)
11. Shimada, A., Yamabe, Y., Torisu, T., Baad-Hansen, L., Murata, H., Svensson, P.: Measurement of dynamic bite force during mastication. *J. Oral Rehab.*, 1–8 (2012)
12. Iwasaki, M., Maeda, I., Kokubo, Y., Tanaka, Y., Ueno, T., Takahashi, W., et al.: Capacitive-type pressure-mapping sensor for measuring bite force. *Int. J. Environ. Res. Public Health* **24**(19), 1–9 (2021)
13. Steffen, C., Duda, K., Wulsten, D., Voss, J.O., Koerdt, S., Nahles, S., et al.: Clinical and technical validation of novel bite force measuring device for functional analysis after mandibular reconstruction. *Diagnostics* **05**(13), 1–11 (2023)
14. De Pasquale, P., et al.: A low-cost wireless bite force measurement device. *Materials* **15**(1), 1–22 (2022)
15. Parmar, S., Khodasevych, I., Troynikov, O.: Evaluation of flexible force sensors for pressure monitoring in treatment of chronic venous disorders. *Sensors* **17**(8) (2017). <https://doi.org/10.3390/s17081923>
16. Tekscan Inc.: FlexiForce Standard Model A301 Datasheet (2021). <https://www.tekscan.com/sites/default/files/FLX-Datasheet-A301-RevI.pdf>



Streamlining Software Migration with Model-Driven Techniques

Yaswanth Kumar Damarla^{1,2(✉)} and Bhagya Rekha Deenadayal²

¹ Jawaharlal Nehru Technological University Kakinada - University College of Engineering
Vizianagaram, Vizianagaram, India
yaswanthkumardamarla23@gmail.com

² Vasireddy Venkatadri Institute of Technology, Guntur, India

Abstract. Software migration is a common problem for developers working in industry or even working on personal projects. It is usually time-consuming and difficult to test if it has been done correctly. Nowadays, many web application frameworks have different versions, which makes it difficult to migrate, given how frequently they change. One key example is the Ionic Framework, which is used to help build desktop as well as mobile applications, with a simple “write once, deploy anywhere” type of strategy. The Ionic Framework has a manual migration guide that can help developers migrate from different versions. We believe that the tedious work involved can be mitigated partially through automation. Therefore, we propose a model-driven approach to aid the migration process. By focusing on a model-driven approach, we can develop a model that is a representation of the system that will generate code for a specific version. Our approach was applied to a case study in order to validate whether this approach was successful or not.

1 Introduction

At some point in the software lifecycle, the current system must be migrated into a new system or version. Software migration is the intricate process of moving an application from one environment or version to another. Software migration in general, usually consumes lots of resources, time, and effort.

According to the online article (IBM, 2014) [1], it described several common challenges in migrating software applications. In terms of business aspects, these include determining when to migrate, the resistance of users’ culture, and the migration itself not being completed on time. The technical aspects include minimizing disruption of mission-critical applications and ensuring that the application functions correctly after migration.

These migration challenges are common for new technologies, especially for web and mobile-based applications, as they are constantly changing [5]. Recently, hybrid apps have become popular as an alternative to native mobile applications due to their overall benefits [2]. Hybrid applications host a web application inside a native web view while utilizing bridges that allow access to native app functions such as camera access, geolocation, etc. This means that developers are able to carry over their expertise with web technologies to mobile app development without needing to learn native tech stacks.

A key example of this is the Ionic Framework, which is a very popular hybrid application development framework that employs a ‘write once and run anywhere’ [3] approach. The latest version of Ionic (version 4) was a complete rewrite that makes it incompatible with projects written in Ionic 3, which makes it a suitable candidate for the research project.

There are several differences between Ionic 3 and 4, which makes it worth the port, including compatibility with any front-end framework as opposed to Ionic 3’s tight coupling with Angular. There are quite a few issues that can deter development teams from doing a manual port to Ionic 4, which includes breaking changes to the library versions used in both framework versions. Despite there being a guide on how to manually migrate from an Ionic 3 project to an Ionic 4 project, there are still a vast number of forums dedicated to helping solve the manual migration process.

One potential solution to solve these problems is to create models to help generate code to support the migration process. This might be achieved with a software engineering technique called model-driven development [4]. Model-driven approaches are used mainly in software design to generally simplify a complex process of the system into a higher-level abstraction. By abstracting the migration process into a code generation tool, it could possibly speed up the migration process with high accuracy and consistency as intended. In addition, it would benefit developers as it is a more efficient way to deal with the migration problems.

Finally, applying a model-driven approach would not just be applicable to the Ionic framework but potentially other web frameworks as well. Therefore, model-driven approaches might solve software migration problems toward hybrid app development in regard to technical and business aspects that the industry is currently facing. Using a model-driven approach as well as focusing on Ionic versions 3 and 4, the research undertaken will determine whether the issue of model-driven approaches for migration can be applied.

This report will go into detail about our research questions in Sect. 2 and then move into how we approached the problem in Sect. 3. In Sect. 4 we will evaluate our research, looking specifically at our case study (Meetup For Pets) and other Ionic version 3 projects. Furthermore, the model will be discussed in detail, as well as the code generated using this model. Subsequently, we will discuss our experiences applying a model-driven approach in Sect. 5 to solve the migration issue and tie in some other related literature to our own work in Sect. 6. Eventually, we will formulate a conclusion in Sect. 7 and mention possible future work that our research may have.

2 Related Work

In regards to the software migration that utilizes model-driven engineering methods, there are several studies that have been conducted extensively before. Those are; Migrating C/C++ Software to Mobile Platforms in the ADM Context [9], FASMM: Fast and Accessible Software Migration Method [10], Reverse Engineering Strategies for Software Migration [11], Model-Driven Engineering for Software Migration in a Large Industrial Context [12], and Towards a Model-Driven Approach for Planning a Standard-Based Migration of Enterprise Applications to SOA [13]

According to Martinez et al. [9] study is related to our work in terms of processing different kinds of programming languages and paradigms to generate mobile application code. It is also the main feasible example of a model-driven approach in the software migration of native-mobile applications, which in our case is the migration of the hybrid mobile application. This study took an example of C/C++ language to produce Android, iOS, and Windows mobile applications. In our study, the extension type of the generated code is still maintained, but the main code-generation process in this study is technically similar.

Furthermore, this study gives insight into the feasibility of a model-driven development approach that is advanced to ADM (Architecture-Driven Modernization) within a similar domain, mobile application migration. The validation tool that is used in this study is the same as what we are using in our approach, which is the Eclipse Modelling Framework [2]. Mobile applications are formed using different kinds of programming languages, which is technically how hybrid application systems work [6]. The ADM approach that is described in this study could potentially be part of our future works.

In terms of methodologies, our study is closely related to the FASMM approach as described by Forite et al. [3]. This study provides extensive guides and methods to conduct software migration. The migration process that it is aiming is at the short and non-optimized migration phase. It is quickly done and not necessarily in an optimized way. For example, inaccurate functionality implementation loosely added new architecture, and non-capitalized gained knowledge during the migration. These conditions are relatable to the environment of our current study. Therefore, our study followed the described methodologies that enable the validation of a hybrid application migration process with model-driven engineering techniques. It also allows capitalizing knowledge as transformation rules to make it highly reusable in future migration projects.

In [12] and [13], studies are proving the feasibility of model-driven engineering in different domains and contexts from our studies. Those are the migration to a completely new framework, migration to a new architecture (legacy architecture to service-oriented architecture), and migration of different languages to a specific mobile application framework. They also provided insights into how model-driven engineering processes are economically profitable and cost-effective.

In this paper [12], the author describes the process of migrating a large-scale software application from Mainframe to J2EE using a Model-Driven Engineering process. The migration process in this study starts by describing the general processes of the current legacy system. First, a parser is created to make an abstract syntax tree from the legacy code. Then, it is processed by a transformation to build a model that conforms to the meta-model of the legacy language. During the second stage, all symbols are resolved and bound to the appropriate model elements. Afterward, from the model representing the code, a reverse-engineering process is conducted to produce a platform-independent model. Finally, this model will be mapped into a platform-specific model that can be used to generate code for the final product of the migrated application.

Lastly, Aboulsamh describes [13] the feasibility of model-driven approaches to be used in the software migration context in general. Our proposed method that utilizes

model transformation to generate the migrated code conforms with this research argument. The migration system that it utilizes is a sequential execution of model transformation [13] that is quite similar to our approach. However, this research uses much more complex software migration modeling methodologies that are slightly irrelevant to our current study objectives. These include the usage of the service model [7] and migration path model to define common meta-data for semantic integration [8].

3 Research Question

Looking into the problem a bit more, we have devised a main research question below.

Can Model-Driven Development Provide a Valid Migration Solution?

Branching away from this primary question, we have additional research questions that target the concerns about the migration tool we are proposing:

1. RQ: Can it account for major changes like different libraries/versions?
Looking at the Ionic Framework, there are specific libraries or versions used in a project that may differ from one version to another. These may involve small changes, like simply changing the name to match the new version, or larger changes, such as changing the way the application handles routing to different pages. Our research will involve taking a look at some of these changes.
2. RQ: Is the generated code of acceptable quality?
With the model being used to help convert one Ionic 3 project file to a compatible Ionic 4 version, there can be a chance that the migration may not be perfect. There might be flaws with the output of the conversion, which breaks the integrity of the original file. Our research will also observe the quality of the code and take note of any changes.
3. RQ: Can the tool be used for other case studies?
Our modeling tool will be primarily used for our case study (Meetup For Pets); however, every Ionic project is expected to be different. Our research will observe to see if the tool will be applicable not just for our case study but for other Ionic 3 projects as well.

4 Approach

In order to migrate Ionic Framework applications from one version to another, the migration tool, called the *IonicMigrator* (<https://github.com/rqmok/IonicMigrator>), was modeled. This section describes how the Eclipse Modeling Framework and Acceleo have been used for this purpose.

4.1 Eclipse Modeling Framework

The Eclipse Modeling Framework (EMF), is a standard for data models with full support for meta-modeling and modeling [14].

During this research, EMF has been used to create a meta-model from which a model for the *IonicMigrator* has been derived. EMF is pre-configured with the Ecore meta-model (or super-metamodel) for constructing the domain-specific meta-model.

4.2 Overview of Operation

After creating the metamodel and, hence, the model, Acceleo has been utilized to help invoke the appropriate operations in the model to read the source files, modify their contents, and then write back to the files. Figure 1 shows a diagram of this workflow.

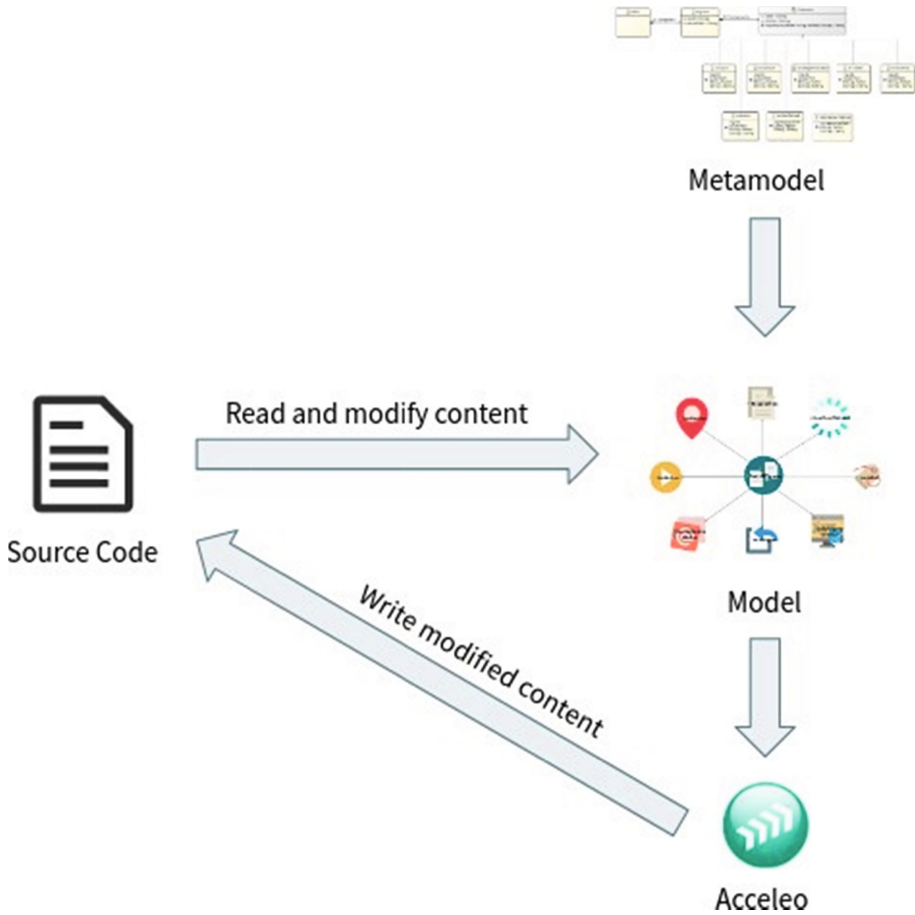


Fig. 1. IonicMigrator Operation Overview

4.3 The Metamodel

The solution to the *IonicMigrator* metamodel is shown in Fig. 2. The meta-model contains an entity called *mBase*, which is used as a container for all other entities for creating the model.

The *Migrator* entity stores the path to the project that needs to be migrated. It contains a set of *Components*. Each *Component* entity is a part of the source code for the application that needs to be migrated.

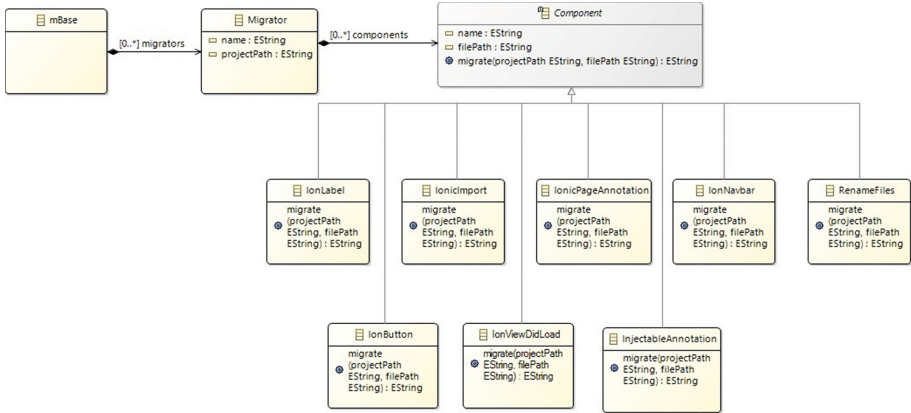


Fig. 2. IonicMigrator Metamodel

Each *Component* contains a path to the file, relative to the project path, that contains the source code for that component, and also a *migrate* operation. The *migrate* operation is given the project path and the file path relative to the project path. The operation combines these paths to produce an absolute path and read the file to modify its contents.

These components are separated in the metamodel rather than the model because each one has a separate way of migrating the source code that it is responsible for. Furthermore, each of the *migrate* operations for the components are manually written in Java and inserted into the Java source files generated by EMF.

4.4 The Model

Once the metamodel Java source code is generated and the *migrate* operation for each *Component* is manually inserted into this code, the *IonicMigrator* can finally be modeled.

Figure 5 shows the *IonicMigrator* model, which simply contains instances of the entities that were created in the metamodel. Through this model, the project path is provided to the *Migrator* entity instance, and the individual file paths relative to the project path are provided to each *Component* entity instance.

4.5 Acceleo

Acceleo is a template-based source code generation technology to facilitate model-to-text transformations (MTT). It is used as an add-on for EMF to be used with the model [16].

The model in Fig. 5 contains a project path, file path for each *Component*, and a *migrate* operation in each component that takes both paths as its parameters. In order to utilize these operations, Acceleo is used as shown in Fig. 3.

The Acceleo script in Fig. 3 goes through each migrator's components in the model, retrieves the project path and file path from a component, and calls that component's *migrate* operation, giving it both paths as parameters. The *migrate* operation then returns

```

[comment encoding = UTF-8 /]
[module generate('http://www.fit4003.org/migrator'')]

[template public migrateComponents(amBase : mBase)]
  [comment @main/]

  [comment]Go through each migrator in the model.[/comment]
  [for (migrator : Migrator | amBase.migrators)]
    [comment]Get the project path and store as variable.[/comment]
    [let projectPath : String = migrator.projectPath]
    [comment]Go through each component in the migrator.[/comment]
    [for (component : Component | migrator.components)]
      [comment]Get the file path and store as a variable.[/comment]
      [let filePath : String = component.filePath]
      [comment]Get migration result.[/comment]
      [let result : String = component.migrate(projectPath, filePath)]
      [comment]Check that migration is not empty.[/comment]
      [if (result.size() > 0)]
        [comment]Get the file at the stored file path.[/comment]
        [file (filePath, false, 'UTF-8')]
      [comment]Write result to the file[/comment]
      [result/]
    [/for]
  [/for]
[/template]
[/file]
[/if]
[/let]
[/let]
[/for]
[/for]
[/template]

```

Fig. 3. IonicMigrator Acceleco Script

a string that contains the contents of the entire file, including the migration modifications. Acceleco will then write this string back to the file, essentially overwriting the entire file. However, the script ignores writing to a file when the *migrate* operation returns an empty string. This can be useful for operations that do not require changing a file's contents. Once the model is created and the project path and file path are given, the Acceleco script is run by creating the appropriate run configuration in EMF.

5 Evaluation

5.1 Case Study

As a proof-of-concept, the application that will be developed for our case study will be called **Meetups for Pets** (<https://github.com/rbnguyen231/Meetup-For-Pets>).

It is a hybrid application written in Ionic version 3 to connect pet enthusiasts to promote face-to-face interactions with pets and their owners. The users will have the ability to organize a meetup with each pet's owner, given a selection of pets. Furthermore, a user can add their own pets and upload or take a photo using the camera of their pet. Finally, if a meetup is organized, the event will be added to their calendar on their device. The main purpose of this case study is to test the model-driven migration tool; this can be shown in Fig. 4

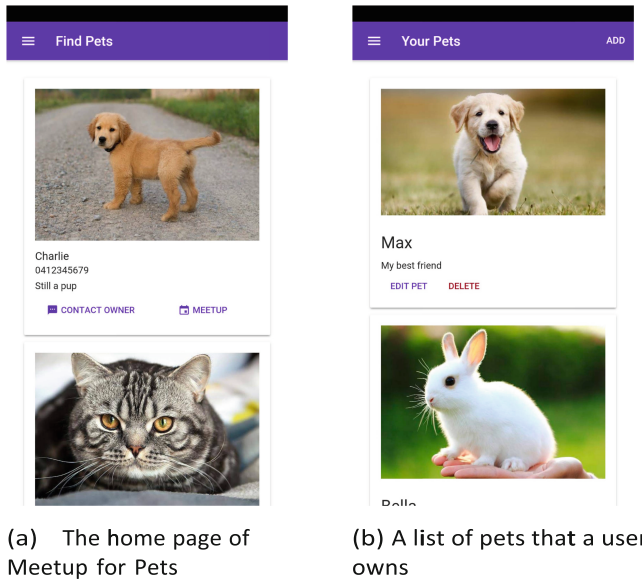


Fig. 4. Meetup For Pets Ionic 3 Application

In a real-world scenario, an application would normally take advantage of the hardware capabilities of the device it is running on. Therefore, there are some Ionic native framework features that were included, such as camera, contacts, calendar, and messaging.

Using the case study above, we attempted to perform some manual migration to look deeper into the issues that may arise when migrating. In our research, we also attempted to use another open-source Ionic 3 application from GitHub (<https://github.com/growu/drip-ionic3>) in order to test if our model had generated the correct Ionic 4 code.

5.2 The Model

Sirius is an Eclipse project that allows for creating a graphical modeling work-bench in EMF [15]. It has been used to visualize the model shown in Fig. 5.

The model in Fig. 5 gives flexibility to the user by allowing a *Component* to be removed if it shouldn't be included in the migration process and then added back in if it is. However, it should be remembered that the file path will always need to be given where the component resides in the source code, and each file will need to have a separate *Component* created for it.



Fig. 5. IonicMigrator Model

5.3 Code Generation

The results of the *IonicMigrator* we developed were quite favorable. The case study involved migrating three categories of changes:

1. HTML markup
2. Typescript code
3. Architectural changes (file and folder structure)

HTML Markup. The *IonicMigrator* tackles the problem by separating the full text into components (i.e., buttons, labels, etc.). This is the most general category, as the migration process for different components has a lot of common procedures. The migration of HTML markup reveals the tool's adeptness at component-wise migration. By dissecting the full text into components, such as buttons and labels, the *IonicMigrator* demonstrates a systematic approach. The presented Figs. (6a and 6b) vividly illustrate the transformation from Ionic 3 to Ionic 4, showcasing a harmonious migration of the button component.

Typescript Code. Migration of Typescript code involves looking through code files to find specific tokens and change them to fit the Ionic 4 specification. Unlike HTML markup, every different migration process in this category is almost completely different and can only be generalized in broad terms, such as find-and-replace functionality.

In contrast to HTML markup, the migration of Typescript code unfolds as a more intricate process. The necessity to examine code files for specific tokens and adapt them to Ionic 4 specifications introduces diversity in migration procedures.


```
<button ion-button type="button" clear (click)="addPhoto()">Add Image</button>
```

(a) Ionic 3

```
<ion-button type="button" fill="clear" (click)="addPhoto()">  
  Add Image  
</ion-button>
```

(b) Ionic 4

Fig. 6. Ionic 3–4 Button Migration (Markup)

Architectural Changes. The structure of an Ionic 4 project is drastically different from an Ionic 3 project, making its migration a very specific process. The *IonicMigrator* focuses on a small subset, which involves reorganizing files associated with one specific feature of an app to the equivalent Ionic 4 folder structure, as well as renaming them to fit the new convention. The inherent complexity of architectural changes highlights the tool's capability to navigate project-wide transformations, ensuring that the migrated project adheres to the evolved Ionic framework's structural norms (Figs. 7 and 8).

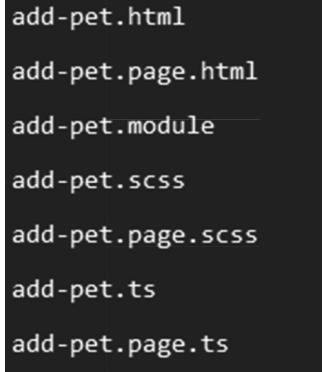
```
ngOnInit() {  
  console.log('ngOnInit AddPetPage');  
}
```

(a) Ionic 3

```
ionViewDidLoad() {  
  console.log('ionViewDidLoad AddPetPage');  
}
```

(b) Ionic 4

Fig. 7. Ionic 3–4 Lifecycle Methods Migration (Typescript)



```
add-pet.html  
add-pet.page.html  
add-pet.module  
add-pet.scss  
add-pet.page.scss  
add-pet.ts  
add-pet.page.ts
```

Fig. 8. Ionic 3–4 File Rename-and-Relocate Migration (Architectural)

6 Discussion/Experiences

The overall consensus is that model-driven development is a feasible approach for software migration, especially regarding this case study in particular.

For HTML markup, in particular, automating component-wise migration is remarkably simple, and it is well within the scope of possibility to create a general function that will migrate any Ionic 3 component to its Ionic 4 equivalent, provided a library of changes specific to each component is collated for use by this function.

Typescript code can be a little more complicated to deal with. During the development of the *IonicMigrator*, we encountered some difficulty with porting over two components in particular - **Toasts** and **Alerts**. The difficulty stems from the asynchronous nature of the two in Ionic 4 as opposed to their counterparts in Ionic 3 - handler function callbacks and deeply nested toasts and alerts have to be handled differently, which makes migrating these a more involved process than just pure automation.

Architectural changes vary in difficulty. A project-wide relocation and renaming of files is well within the scope of possibility, but changes to the routing mechanisms adopted in the different versions are drastic and require significant changes to the code-base. Ionic 4 uses the routing functionality provided by the frontend framework specified when creating a project using its command-line interface, meaning that a separate migration process would need to be adopted for, say, React vs. Angular, for example. Ionic 3 used its own proprietary router, on the other hand, which means at least that part of the code is easier to deal with. While Ionic 4 adopted this change in order to be future-proof, this does make a generalized migration tool for this specific feature a lot harder to implement.

7 Conclusion

In conclusion, our study contributes to the following in the domain of software migration and model-driven approaches:

- We have introduced a case study Ionic 3 hybrid application (MeetUp For Pets) built primarily to test the viability of a proposed migration tool
- An Ionic migration tool was built which performs code generation to convert from Ionic 3 to Ionic 4
- We have evaluated the feasibility of using a model-driven approach for migrating using the Ionic Framework as the primary focus

At this stage, our approach only proves a small fraction of the hybrid application migration because this study only focused on one framework (Ionic) and only based on the selected versions (version 3 to version 4).

At the moment, within various constraints, the validation tool that we had developed only handles basic migration functionalities. More specifically, it is only using string parsing, transformations, and code generation. These processes might be insufficient to handle more complex functionalities, as described in our limitations.

This leads to many other challenges that need to be tackled in the proposed methods to fully migrate the Ionic version 3 project to version 4. The main goal of this study is to prove the feasibility of migration hybrid applications with model-driven engineering methods, and it is proven to be true. Therefore, building a fully functional software migration tool for a hybrid application is intentionally left for future work.

Here, we used the Eclipse Modeling Framework as the validator and the *IonicMigrator* to handle the migration process of the Ionic framework from version 3 to version 4. Within this tool, the abstraction level was intentionally increased for the cohesiveness and extensibility of migration functions. By following our approaches and methodologies, future researchers could add and upgrade the tool to handle more complex functionalities according to the limitations described above.

At the moment, our approaches are only applicable to solving basic software migration problems in hybrid applications. It requires more extensive research to solve more complex functionalities, such as dealing with architectural changes to arrange files in the correct directories. ADM (Architecture-Driven Modernization) and KDM (Knowledge Discovery Metamodel) methods [9, 13] could potentially be candidates to solve the current limitations in this study.

There will be a time in the future for the *IonicMigrator* to be mature enough so that the migration of Ionic Framework applications from version 3 to version 4 can be done thoroughly without introducing any migration bugs. This tool is also envisioned to handle the migration of the Ionic framework from any version to the latest. In the future, by following our approaches and adding further research, it could potentially migrate other hybrid application frameworks.

References

1. IBM Knowledge Center: Ibm.com (2019). www.ibm.com/support/knowledgecenter/en/SSEP7J_10.2.1/com.ibm.swg.ba.cognos.ug_mfdm.10.2.1.doc/c_mfdm_chal_mig.html. Accessed 07 Sept 2019
2. Dunka, B., Emmanuel, E., Oyeyinka, D.: Hybrid mobile application based on ionic framework technologies. In: Int. J. Recent Adv. Multidiscip. Res. **04**(12), 3121–3130 (2017). Accessed 27 Aug 2019

3. Huynh, M., Ghimire, P., Truong, D.: Hybrid app approach: could it mark the end of native app domination? *Issues Informing Sci. Inf. Technol. Educ.* **14**, 49–65 (2017). Accessed 07 Sept 2019
4. García Díaz, V., et al.: A brief introduction to model-driven engineering. *Tecnura* **18**, 127–142 (2014). Accessed 27 Aug 2019
5. Kwon, J., Moon, S.: Web application migration with closure reconstruction. In: *International World Wide Web Conference Committee (IW3C2)*, Seoul (2014). Accessed 23 Aug 2019
6. Thomas, C.G., Devi, J.: A study and overview of the mobile app development industry. *Int. J. Appl. Eng. Manag. Lett. (IIAEM)* **5**(1), 115–130 (2021)
7. Bragagnolo, S., et al.: Appendix to software migration: a theoretical framework a grounded theory approach on systematic literature review (2021)
8. Heil, S.: Web migration revisited: addressing effort and risk concerns. Diss. dissertation, Chemnitz, Technische Universität Chemnitz (2020, 2021)
9. Martinez, L., Pereira, C., Favre, L.: Migrating C/C++ software to mobile platforms in the ADM context. *Int. J. Interact. Multimedia Artif. Intell.* **4**(3), 34 (2017)
10. Forite, L., Hug, C.: FASMM: fast and accessible software migration method. In: *2014 IEEE Eights International Conference on Research Challenges in Information Science (RCIS)*. IEEE (2014)
11. Muller, H.A.: Reverse engineering strategies for software migration. In: *Proceedings of the 19th International Conference on Software Engineering*, p. 659. IEEE, May 1997
12. Fleurey, F., Breton, E., Baudry, B., Nicolas, A., Jézéquel, J.-M.: Model-driven engineering for software migration in a large industrial context. In: Engels, G., Opdyke, B., Schmidt, D.C., Weil, F. (eds.) *MODELS 2007*. LNCS, vol. 4735, pp. 482–497. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-75209-7_33
13. Aboulsamh, M.A.: Towards a model-driven approach for planning a standard-based migration of enterprise applications to SOA. In: *2009 Congress on Services – I*. IEEE (2009)
14. Gronback, R.: Eclipse Modeling Project — The Eclipse Foundation. Eclipse.org (2019). <https://www.eclipse.org/modeling/emf/>. Accessed 04 Nov 2019
15. Sirius – Overview. Eclipse.org (2019). <https://www.eclipse.org/sirius/overview.html>. Accessed 04 Nov 2019
16. Accelele — Home. Eclipse.org (2019). <https://www.eclipse.org/accelele/>. Accessed 04 Nov 2019



A Survey - IOT Edge Based Garbage Sorter Bin with Bio Gas Generator

A. R. Sathyabama^(✉), Jeevaa Katiravan, M. Muthudhivakar, and S. Sabarinath

Information Technology, Velammal Engineering College, Chennai, India
sathya.it1@gmail.com

Abstract. Waste management is a pressing contemporary environmental challenge, stemming from the daily generation of industrial and household waste by humans. This waste poses threats to human health, emits foul odours, and even contributes to air pollution. Inadequate waste handling practices can result in severe environmental consequences. People often hesitate to open and close trash cans due to concerns about bacterial contamination on their hands. To address these environmental issues, a smart trash bin equipped with waste sorting technology is suggested. Garbage presents environmental problems, including health risks, unpleasant odors, and potential air pollution. The proposed solution involves utilizing an ultrasonic sensor capable of detecting approaching individuals within a 50 cm range, automatically opening the bin's lid for a 7-s duration. A stepper motor is employed to rotate the waste sorter to different positions based on the type of waste detected. For instance, if the soil moisture sensor identifies organic waste, a servo motor rotates the sorter to direct it into the organic waste bin. Likewise, if the inductive proximity sensor detects metal, the servo motor positions the sorter for metal recycling. A camera module captures images of the waste, employing a CNN algorithm for waste categorization. Once classified, the servo motor guides the sorter to the appropriate bin. This system streamlines waste sorting, enhancing accuracy. The separated waste is stored in dedicated compartments within the bin, reducing landfill contributions. Organic waste can be processed to produce biogas, serving kitchen needs. Additionally, the system can automatically alert cleaners when the bin is full, mitigating overflow issues and odours. This proposed smart trash bin holds promise as a valuable tool for waste management, applicable in both public and private settings.

Keywords: Inductive Proximity Sensor · Soil moisture sensor · Camera module · CNN algorithm · Stepper motor · Biogas digester

1 Introduction

Solid waste management is a major challenge facing the world today. The amount of solid waste generated is increasing rapidly, and many cities and municipalities are struggling to keep up. In recent years, there has been a growing interest in developing new and innovative solutions to the solid waste problem. A number of recent surveys have been conducted to assess the current state of solid waste management and to identify areas for improvement.

One recent survey by the World Bank found that more than half of the world's solid waste is not collected or disposed of properly. The survey also found that the amount of solid waste generated is expected to increase by 70% by 2050. Another recent survey by the Global Alliance for Incinerator Alternatives (GAIA) found that more than 70% of cities in developing countries are struggling to manage their solid waste. The survey also found that open burning of waste is a major problem in many developing countries. These surveys highlight the need for urgent action to address the solid waste problem. There is a need to invest in new and innovative waste management technologies, and to educate the public about the importance of waste reduction and recycling.

Our research's main goal is to classify waste and make it useful for recycling, thereby avoiding pollution. We will sort the waste based on its type and recycle organic waste into biogas. There are a variety of different ways to classify waste. One common method is to classify waste into organic and inorganic waste. Organic waste is derived from living organisms, such as food scraps, yard waste, and animal manure. Inorganic waste is derived from non-living organisms, such as metal, plastic, and glass. A renewable energy source, biogas can be utilized to provide electricity, heat, and fuel for vehicles. It is created through the anaerobic digestion of organic material, including manure, yard trash, and food scraps. We must first separate organic garbage from other types of waste in order to turn it into biogas. Numerous techniques, including optical sorting, manual sorting, and sensor-based sorting, can be used to accomplish this.

After the segregation of organic waste, it can be introduced into an anaerobic digester. Within a sealed container called an anaerobic digester, microorganisms break down organic material in the absence of oxygen. This process results in the production of biogas along with a nutrient-rich liquid, often used as fertilizer. The biogas generated from the anaerobic digester can serve multiple purposes, including generating electricity, producing heat, or acting as fuel for vehicles. Furthermore, it can undergo enhancements and purification processes to yield bio methane, a renewable natural gas compatible with existing natural gas infrastructure.

Our research will focus on developing new and innovative methods for waste develop new markets for biogas products. We believe that our research has the potential to make a significant contribution to the reduction of waste and pollution, and to the promotion of sustainable waste management practices.

2 Related Work

This project aims to design and implement an Internet of Things (IoT) powered smart waste bin capable of autonomously recognizing and sorting different types of waste [1]. The core objective of this initiative is to establish an IoT-driven waste management system capable of monitoring the waste bin fill levels and sending alerts to the waste collection service when bins require emptying [2]. The overarching goal of this proposal is to develop a smart waste management system that leverages IoT sensors to monitor the fill levels of trash bins and trigger notifications to the relevant authorities when it's time for emptying. Additionally, this proposal introduces an IoT-based intelligent trash can that can automatically identify the waste type and segregate it into distinct compartments [4]. It also elaborates on the components of such systems, the challenges faced during

implementation, and the benefits of harnessing big data analytics to optimize waste management [5].

The system uses sensors to monitor the fullness of waste bins and sends alerts to the waste collection company when bins need to be emptied. This helps to improve the efficiency and effectiveness of waste collection [6]. The framework employs IoT devices to continuously monitor the capacity status of waste bins and promptly alert the waste collection company when bins necessitate emptying. Furthermore, it harnesses machine learning to forecast patterns in garbage generation, enhancing the efficiency of waste collection routes [7]. When garbage bins reach their capacity, the system utilizes sensors to assess the level of fullness and communicates this information to the waste collection provider. Additionally, it incorporates a web application to provide real-time updates to residents regarding the status of trash cans [8]. The proposed smart waste management system featured in this project utilizes ultrasonic sensors for waste detection, triggering the automatic opening of the garbage bin's lid [9].

This project introduces a smart waste and garbage collection bin that operates on the Internet of Things (IoT) framework. The bin incorporates a camera module for capturing images of the waste and utilizes a CNN algorithm to classify the waste into specific categories [10]. The system outlined in this project relies on ultrasonic sensors to gauge the filling level of trash bins and notify waste collectors when they reach capacity. It's an IoT-driven intelligent garbage monitoring system [11]. The proposal advocates for an IoT-based intelligent waste management system that employs a combination of sensors, including ultrasonic sensors, soil moisture sensors, and inductive proximity sensors, to identify the presence, type, and quantity of waste within garbage bins [12].

The project puts forward an IoT-driven smart garbage bin monitoring system that utilizes sensors to keep tabs on the trash levels within the bins and promptly notifies the relevant authorities when they reach capacity [13]. This smart waste management system, proposed in the project, employs sensors for monitoring the trash levels in various bins and triggering alerts to the appropriate authorities when a bin is full. The system operates on an IoT infrastructure. Additionally, residents have the capability to track the status of bins in their neighborhoods and even request waste pickups through the system's mobile application [14].

Furthermore, the project delves into the exploration of anaerobic digestion as a method to convert slaughterhouse by-products into biogas. Anaerobic digestion is a biological process that decomposes organic material in an oxygen-deprived environment. The resulting biogas can serve as a sustainable energy source or be harnessed for electricity generation [15]. The study revealed that optimizing the pH, temperature, and nutrient levels within the microenvironment can significantly enhance the efficiency of biogas production [16]. In the essay, the concept of smart cities is explored, along with how the Internet of Things (IoT) can be effectively applied to implement this concept. It discusses how IoT technology can be leveraged to improve multiple facets of smart cities, such as waste management, electricity systems, and transportation networks [17].

The various frameworks for smart waste management systems are reviewed in this research. It covers the various parts of a smart waste management system as well as the opportunities and limitations of using IoT for trash management [18]. This essay examines recent developments in wastewater-derived biogas production. It goes over

the various setups and this essay examines recent developments in wastewater-derived biogas production. It goes over the various setups and operational variables that can be employed to maximize biogas production [19]. This study describes a smart bin that autonomously sorts rubbish using IoT and converts organic waste into biogas. A camera module, soil moisture sensor, inductive proximity sensor, and ultrasonic sensor are just a few of the sensors the bin is equipped with [20].

This project presents a smart trash can that opens its lid when a person approaches and uses ultrasonic sensors to identify their presence [21]. The smart garbage collection system described in this project employs ultrasonic sensors to determine when bins are full and automatically sends a signal to the collection truck. The system is IoT-based. In order to track the location of the bins and improve the collection routes, the system also includes a GPS tracker [22]. This project describes a smart garbage collection system that uses ultrasonic sensors to detect when bins are full and automatically sends a signal to the collection truck [23]. The trash management system described in this project is IoT-based, employs ultrasonic sensors to determine when bins are full, and automatically alerts the pickup truck. Additionally, the device features a camera that can recognize the type of waste and route it to the proper bin [24]. This project suggests an IoT-based smart waste collection system. The technology delivers data from sensors that measure the amount of trash in bins to a central server [25]. Using LoRaWAN, this project suggests an Internet of Things-based smart trash management system. The system uses sensors to gather information about the amount of trash in the bins, their temperature, and their humidity. After that, LoRaWAN is used to transmit this data to a central server [26].

The factors affecting the production of biogas from food waste are examined in this project. The review discusses a wide variety of subjects, including the kind of food waste, its pH, temperature, and retention time [27]. The technology delivers data from sensors that measure the amount of trash in bins to a central server. The server then employs this data to notify users when their trash bins are full and to optimize the collection routes for garbage trucks [28]. For smart cities, this proposal suggests a smart garbage collection system. The technology feeds data from sensors that measure the amount of trash in the bins to a central server. The garbage collection trucks' collection routes are subsequently optimized by the server, leading to more effective waste collection [29].

This project introduces a smart trash can enhanced by Internet of Things (IoT) capabilities, featuring a waste level indicator. The bin communicates data regarding its waste content to a cloud server through an ultrasonic sensor, which gauges the rubbish level. Subsequently, the cloud server calculates the percentage of waste inside the bin and displays this information on an LCD screen [30]. A smart bin IoT-based trash management system is proposed in this research. The technology feeds data from sensors that measure the amount of trash in the bins to a central server [31]. In this project, a smart waste management system that combines cloud computing and IoT is proposed. The system uses sensors to gather information on the temperature, humidity, and amount of trash in the bins. After that, a cloud server receives this data for analysis [32].

This project advocates for the implementation of a smart waste management system utilizing the Internet of Things. The system employs sensors to detect garbage levels, types, and presence within the bins, enhancing trash management efficiency [33]. To improve the effectiveness and efficiency of waste management, the project proposes a

smart system that combines IoT and edge computing. Through sensors, the system collects data on waste generation and disposal patterns, enabling informed decision-making [34]. Additionally, the project introduces a streamlined intrusion detection system for IoT devices utilizing deep learning. By analyzing network traffic data, a lightweight and efficient deep learning model distinguishes between normal and malicious traffic, catering to IoT devices with limited resources [35].

3 Proposed Work

3.1 Overview

The goal of this project is to develop a smart garbage bin that can automatically sort waste into different categories, including organic waste, recyclable materials, and non-recyclable materials. The organic waste will then be fed into a biogas digester to produce biogas, which can be used for cooking, heating, or generating electricity. The biogas digester is a sealed chamber where the organic waste is broken down by anaerobic bacteria. The bacteria produce methane, carbon dioxide, and other gases, which are collectively known as biogas. This system will contribute to diminishing the volume of waste directed to landfills, preserving natural resources, and lowering greenhouse gas emissions.

3.2 Methodology

3.2.1 Ultrasonic Sensor

An ultrasonic sensor is a tool that utilizes sound waves to identify the presence and measure the distance of objects. In the context of a smart trash bin, this sensor can be employed to recognize the proximity of an individual. When the sensor registers a person within a specified range (such as 50 cm), it can transmit a signal to the trash bin, prompting the automatic opening of its cover. This feature is convenient for users, as it eliminates the need to manually open the lid of the garbage bin. It can also be helpful for people with disabilities or mobility issues. Additionally, opening the cover automatically can help to reduce the spread of germs and bacteria.

Distance calculation formula using Ultrasonic Sensor:

$$\text{Distance } L = 1/2 \times T \times C .$$

The amount of time that the cover remains open can be programmed to suit the needs of the user. For example, the cover may remain open for 7 s, which is enough time for most people to dispose of their trash. However, the timer can be adjusted to be longer or shorter, depending on the user's preference.

3.2.2 Soil Moisture Sensor, Inductive Proximity Sensor and Camera Module:

The soil moisture sensor and inductive proximity sensor are two types of sensors that can be used to automatically sort waste into different categories.

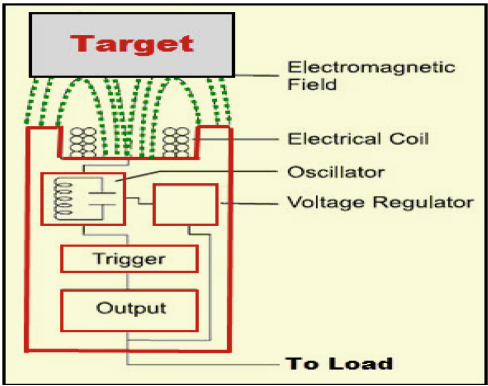
The soil moisture sensor detects the amount of moisture in the waste. Organic waste, such as food scraps and yard waste, typically has a higher moisture content than inorganic waste, such as metal and plastic. Therefore, the soil moisture sensor can be used to distinguish between organic and inorganic waste.

$$\text{Analog Read output} = (\text{Input Voltage} / \text{Maximum Voltage}) * 1024 .$$

Moisture Level Indication:

Level	Analog Value Range
1	0–175
2	175–350
3	350–525
4	525–700
5	700–876

The inductive proximity sensor detects the presence of metal. Metal is a conductor of electricity, so the inductive proximity sensor can detect it by generating an electromagnetic field and sensing any changes in the field.



A camera module can be added to the soil moisture sensor and inductive proximity sensor to increase the accuracy of garbage sorting. The camera module can be used to take pictures of the garbage, which can then be categorized using machine learning techniques. Different kinds of plastic, like PET, HDPE, and PVC, might be recognized using the camera module. The waste sorter might then be sent to the proper recycling bin using this information. The use of soil moisture sensor, inductive proximity sensor, and camera module in a smart garbage bin can help to improve the accuracy and efficiency of waste sorting. This can help to reduce the amount of waste that goes to landfills and promote recycling.

3.2.3 CNN Algorithm

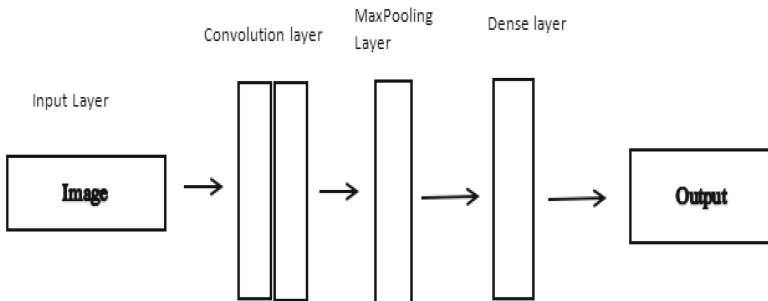
CNN algorithm for waste sorting based on the soil moisture sensor, inductive proximity sensor, and camera module:

Preprocess the images: The images captured by the camera module should be pre-processed to ensure that they are in a consistent format and to remove any noise. This may involve resizing the images, cropping them, and normalizing the pixel values.

Extract features: The CNN algorithm will extract features from the preprocessed images. These features can be based on the color, texture, and shape of the waste.

Classify the waste: The CNN algorithm will then use the extracted features to classify the waste into different categories. This may involve using a deep learning model that has been trained on a large dataset of labeled images of waste.

Control the waste sorter: Once the waste has been classified, the CNN algorithm can send a signal to the waste sorter to direct the waste to the appropriate bin.



3.2.4 Stepper Motor

The stepper motor is a type of electric motor that can be controlled to rotate in precise increments. In a smart garbage bin, the stepper motor can be used to rotate the waste sorter to different positions depending on the type of waste that is being detected. If the waste sorter has three bins for different types of waste (plastic, metal, and glass), the stepper motor can be used to rotate the sorter so that the correct bin is positioned under the opening of the garbage bin. This allows the waste to be automatically sorted into the correct bin, without the need for human intervention. A stepper motor can increase waste sorting effectiveness and decrease the quantity of rubbish that ends up in landfills in a smart garbage can. The spinning of the trash sorter in an intelligent garbage can is easily and effectively controlled by a stepper motor. It might aid in increasing the efficacy and accuracy of garbage sorting, which might be advantageous for the environment.

3.2.5 Working Procedure

The ultrasonic sensor is typically mounted on the front of the garbage bin. When the sensor detects a person within a certain distance, it sends a signal to the garbage bin to open the lid. The garbage bin lid can be opened using a variety of methods, such as a motor, solenoid, or pneumatic actuator. The choice of method will depend on the specific design of the garbage bin.

The person disposes of their trash in the garbage bin. Once the person has disposed of their trash, the garbage bin lid closes automatically. This can be done using the same method that was used to open the lid.

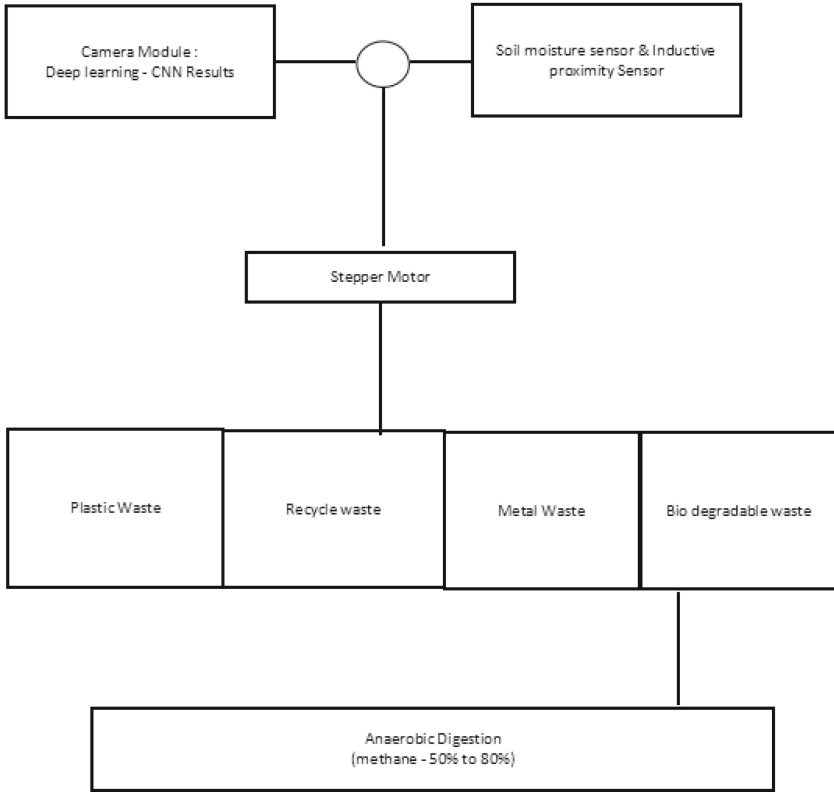
Usually fitted inside the trash can are the inductive proximity sensor and the soil moisture sensor. The inductive proximity sensor identifies the presence of metal, while the soil moisture sensor determines how much moisture is present in the trash.

On top of the trash can is normally where the camera module is positioned. The camera module takes a picture of the waste when the trash can lid is open.

A common machine learning approach for image classification problems is the CNN algorithm. The CNN algorithm will take elements out of the garbage's image and utilize those features to categorize the waste into various groups.

The CNN algorithm will send a signal to the waste sorter to direct the waste to the appropriate bin. The waste sorter is typically a mechanical device that moves the waste to different bins based on its type.

In order to direct the waste to the proper bin, the CNN algorithm will send a signal to the waste sorter. Typically, a trash sorter is a mechanical device that divides waste into several bins according to kind.



4 Results and Discussion

4.1 Reduced Quantity of Waste Ending up in Landfills

Smart trash cans can contribute to a reduction in the amount of waste ending up in landfills by automating the waste sorting process and increasing the accuracy of waste sorting. This is significant because a significant portion of methane, a greenhouse gas 25 times more potent than carbon dioxide, comes from landfills.

4.2 Increased Recycling Rates

By making recycling easier for individuals, smart garbage cans can also contribute to higher recycling rates. Smart trash cans minimize the need for manual waste sorting by automatically classifying rubbish into many categories. Due to its convenience and ease of use, recycling may become increasingly popular.

4.3 Improved Public Health

By halting the transmission of germs and pathogens, intelligent garbage cans can also contribute to better public health. This is due to the possibility of including technologies like garbage compactors and automatic lid openers in smart trash cans, which can lessen the amount of touch that people have with rubbish.

4.4 Reduced Environmental Pollution

Smart trash bins can also help to reduce environmental pollution by preventing waste from overflowing and littering the environment. This is because smart trash bins can be equipped with features such as full bin alerts and automatic waste compaction.

5 Conclusion

Waste management could be revolutionized by the use of intelligent garbage cans with waste sorting technologies. These systems can aid in lowering the quantity of waste that ends up in landfills and promoting recycling by automating the waste sorting process and increasing waste sorting accuracy. Smart trash cans can also help to lower environmental pollution and promote public health. In the context of developing countries, smart trash bins can play a particularly important role in improving waste management. In these countries, waste collection and disposal services are often limited or nonexistent. Smart trash bins can help to address these challenges by providing a more efficient and environmentally friendly way to dispose of waste. Smart trash bins with waste sorting technology have the potential to make a significant contribution to improving waste management and protecting the environment if the proposed system is executed successfully.

References

1. Abu-Mahfouz, A.M., et al.: A smart IoT-based waste bin for efficient solid waste management. *IEEE Access* **7**, 127418–127429 (2019)
2. Ahmed, S.: IoT-based smart waste management system for efficient waste collection. In: *Proceedings of the International Conference on Smart Systems and Data Science*, pp. 357–367 (2019)
3. Bhatia, S., et al.: Smart garbage system: an IoT based approach for waste management. *Procedia Comput. Sci.* **165**, 705–712 (2019)
4. Acharya, A.N., Anjankar, R.D.: IoT-based smart garbage bin for efficient waste management. In: *Proceedings of the Second International Conference on Inventive Communication and Computational Technologies (ICICCT 2019)*, pp. 421–425 (2020)
5. Sasidharan, R., et al.: A review on IoT-based waste management system using big data analytics. *Procedia Comput. Sci.* **132**, 1035–1041 (2018)
6. Nguyen, N.T., et al.: Development of smart waste collection system in IoT platform for sustainable smart cities. In: *2019 International Conference on Advanced Technologies for Communications (ATC)*, pp. 33–37 (2019)
7. Iftikhar, U., et al.: Smart bin: an IoT-based framework for waste collection management. In: *2019 IEEE 2nd International Conference on Sustainable and Renewable Energy Engineering (ICSREE)*, pp. 60–65 (2019)

8. Yadav, A.K., et al.: IoT-based smart waste management system for smart cities. In: 2017 International Conference on Next Generation Intelligent Systems (ICNGIS), pp. 226–230 (2017)
9. Solanki, S., et al.: Smart waste management system using IoT. In: 2017 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT) (2017)
10. Memon, M.H., Soomro, T.R.: IoT-based smart garbage and waste collection bin. In: 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), pp. 1–6 (2018)
11. Sundararajan, V., et al.: Smart garbage monitoring system for efficient waste management using IoT. In: Proceedings of the Second International Conference on Inventive Communication and Computational Technologies (ICICCT 2019), pp. 261–265 (2020)
12. Kumar, A., et al.: Smart waste management system using IoT. In: 2018 IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), pp. 1265–1270 (2018)
13. Verma, R., et al.: IoT-based smart garbage bin monitoring system. In: 2017 International Conference on Intelligent Computing and Control Systems (ICCCS), pp. 782–787 (2017)
14. Jain, A., et al.: IoT-based smart waste management system for efficient and clean cities. In: 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), pp. 1–6 (2019)
15. Angelidaki, I., et al.: Anaerobic digestion of slaughterhouse by-products. *Biotechnol. Bioeng.* **85**(6), 606–612 (2018)
16. Li, J., et al.: Improvement of biogas production from food waste by control of the microbial microenvironment. *Biores. Technol.* **267**, 473–481 (2018)
17. Hossain, M.S., et al.: A review of smart cities based on the Internet of Things concept. *J. King Saud Univ. Comput. Inf. Sci.* (2015)
18. Kalita, R., et al.: A review on smart city framework for smart waste management system. In: 2017 International Conference on Computing Methodologies and Communication (ICCMC), pp. 964–969 (2017)
19. Baroutian, S., et al.: Recent advances in biogas production from wastewater: an overview on configuration and operating parameters. *J. Environ. Chem. Eng.* **7**(1), 102758 (2019)
20. Mishra, A., et al.: Smart bin. In: 2018 International Conference on Control, Power, Communication and Computational Technologies for Sustainable Growth (ICPCCTSG), pp. 1723–1726 (2018)
21. Pandey, V., et al.: Smart waste collection bin. In: 2017 International Conference on Big Data, IoT and Data Science (BIGD), pp. 21–26 (2017)
22. Das, S.K., et al.: IoT-based smart garbage collection system. In: 2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), pp. 1–5 (2019)
23. Sagar, S., et al.: Smart garbage collection system using IoT. In: 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), pp. 1–6 (2019)
24. Ramakrishna, R., et al.: An IoT-based waste management system. In: 2019 International Conference on Emerging Trends in Smart Technologies (ICETST), pp. 1–6 (2019)
25. Gautam, A., et al.: Smart waste collection system based on IoT. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 3945–3951 (2019)
26. Tariq, M.F., et al.: IoT-based smart waste management system using LoRaWAN. In: 2020 International Conference on Emerging Trends in Smart Technologies (ICETST), pp. 1–6 (2020)
27. Phan, H.D., et al.: Biogas production from food waste: a review on process variables. *Biores. Technol.* **337**, 125414 (2021)

28. Uddin, M.Z., et al.: IoT-based smart garbage system for efficient food waste management. In: 2017 IEEE Region 10 Symposium (TENSYPMP), pp. 1–4 (2017)
29. Padya, S.K., et al.: Smart waste collection bins for smart cities. In: 2017 International Conference on Circuit, Power and Computing Technologies (ICCPCT), pp. 1–5 (2017)
30. Bera, S., et al.: Smart dustbin with waste level indicator using IoT. In: 2020 International Conference on Electronics, Communication, and Aerospace Technology (CONECCT), pp. 1–4 (2020)
31. Alam, A., et al.: Smart bin: IoT-based waste management system. In: 2020 IEEE Region 10 Symposium (TENSYPMP), pp. 1257–1260 (2020)
32. Chen, X., et al.: The integration of IoT and cloud computing for smart waste management. In: 2019 International Conference on Artificial Intelligence and Big Data (ICAIBD), pp. 154–157 (2019)
33. Saini, A., et al.: Design of smart dustbin system using IoT. In: 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), pp. 862–866 (2020)
34. Perwitasari, D., et al.: A smart waste management system using IoT and edge computing. In: 2019 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 1173–1180 (2019)
35. Cao, Y., Zhou, Z., Zhang, Y.: A lightweight intrusion detection system for IoT based on deep learning. *IEEE Access* **7**, 33 (2019)



Architecture of Biometric System Using Multiphase

M. P. Nayana¹(✉) and S. Soumya²

¹ MMK and SDM MMV, Mysore, India
mpnayanasm@gmail.com

² Vidyavardhaka First Grade College, Mysore, India

Abstract. The Multiphase biometric System combines multiple biometric modalities or phases to enhance identification or authentication accuracy. It includes physical characteristics of an individual such as fingerprint and patterns of behavior, such as signature. Feature level fusion with normalization techniques aims to create a more robust, discriminative, and consistent multiphase biometric system by standardizing and integrating information from diverse sources effectively. Both LPQ (Local Phase Quantization) and LBP (Local Binary Patterns) are most effective techniques for feature extraction from fingerprint images. The important features extracted for signature analysis are pen pressure, azimuth, and elevation. Multiphase biometric systems leverage the strengths of multiple biometric modalities to enhance recognition accuracy, performance, security, and user acceptance compared to systems based on a single modality. Prior to the fusion of feature sets from two biometric modalities, we normalize the scores using z-score normalization method as the two feature sets resulted with heterogeneous data. Finally, the proposed work on multiphase biometric by the fusion of fingerprint and signature provide the good results.

Keywords: KNN-classifier · SVM · Gait · NIST Finger print image soft ware (NFIS2) · MCYT-100

1 Introduction

Biometrics represents a fundamental shift towards identity verification based on intrinsic biological or behavioral traits. By relying on “who you are” rather than “what you possess,” biometric systems enhance security, convenience, and reliability across various domains, contributing to a more secure and efficient authentication process. Biometric systems utilize a variety of anatomical and behavioral body traits to establish and verify the identity of a person. These biometric traits are difficult to forge or replicate, making them highly reliable for security and identification purposes in various applications. Face, fingerprint, iris, palm print, hand geometry and ear shape are commonly used biometric systems.

Behavioral attributes such as gait, signature dynamics, and keystroke dynamics offer unique patterns that can be leveraged for person authentication in various applications. Voice can indeed be considered in both anatomical and behavioral terms.

2 Problem Statement

Smart biometrics leverages automated methods to identify individuals by analyzing their distinctive biological or behavioral traits. A biometric feature fingerprints, facial features, iris patterns, voice, and others are captured by specialized sensors or devices.

After capturing biometric data using sensors or devices, the next step in smart biometrics is to process this data to extract the unique features that distinguish one individual from another. The extracted template is then compared against stored templates or records in a database. Fingerprint recognition is considered as one of the earliest and most widely recognized forms of biometrics. During the matching process, the system compares the extracted minutiae points from the captured fingerprint against stored templates in a database. The matching algorithm calculates a similarity score or degree of similarity between the captured fingerprint and the stored templates.

Biometrics encompasses the measurement and statistical analysis of individuals' unique physical and behavioral characteristics. The physiological characteristics such as finger print, face, palm prints, iris, retina, vein and hand geometry and ear shape are commonly used. Gait, voice, signature, and typing rhythm commonly used behavioral characteristics. Each person has a unique set of these characteristics, making biometrics a powerful tool for authentication and identification purposes.

For a characteristic to be considered as a biometric measure, it must generally satisfy the following requirements:

- Universality - The biometric characteristic should be present in every individual
- Distinctiveness - No two persons' characteristic should be same
- Permanence - should not vary reliably over time period.
- Collectability - biometric characteristic should be measurable and collectable from the subject, practical biometric system consists of some more issues like:
- Performance – the recognition speed and accuracy are critical performance metric
- Acceptability –acceptance of a particular characteristics
- Circumvention – By using fraudulent methods, the susceptibility of a biometric system to being fooled.

Biometrics encompasses a wide range of characteristics, both physiological and behavioral, that are used in various applications for identification, authentication, and security purposes. Each and every Biometric has got its own advantages and the disadvantages and hence one biometric cannot be used in all applications. A biometric must be chosen considering the requirement of the specific application.

3 Related Work

A smart glove has been developed to assist individuals with visual impairments in selecting items of their choice. Biometric technologies continue to evolve, offering a range of options from newer, innovative approaches:

Gait: Gait recognition, which analyzes the unique way individuals walk as a form of biometric identification, is indeed considered a newer and evolving technology. While it offers unique capabilities as a spatio-temporal biometric, its application is currently more

suitable for low-security environments due to its lower distinctiveness and susceptibility to changes over time. Gait will not remain the same for a long period due to change in body weight or serious brain damage.

Keystroke: Keystroke dynamics, also known as keystroke biometrics or typing rhythm analysis, is indeed a behavioral biometric that identifies individuals based on their unique typing patterns on a keyboard. While it provides continuous authentication benefits and is adaptable to user variations, privacy and ethical considerations must be carefully addressed to ensure responsible use of this technology.

Odor: While chemical sensors can be powerful tools for distinguishing objects based on their odor profiles, the effectiveness can be influenced by external factors such as personal care products and environmental conditions.

Deodorants and perfumes are considered as lower distinctiveness.

Ear: Ear shape recognition, also known as ear biometrics, analyzes the unique shape and structure of the ear's cartilaginous tissue and features for identification purposes. The shape and structure of the ear, including the contours, folds, and position of salient points (such as the earlobe or specific bends in the ear), are considered distinctive. One method used in ear biometrics is measuring the distances and relationships between salient points on the ear relative to a landmark location.

Hand Geometry: Hand geometry biometrics involves the measurement and comparison of various physical characteristics of the hand, focusing on its overall shape, dimensions of fingers, the palm, and the relative positions of joints. As a means of identification or verification. Unlike some other biometric methods, such as fingerprints or iris scans, hand geometry is generally used in a verification mode rather than for identification across a large population. One advantage of hand geometry is that factors like dry skin or environmental conditions such as dry weather do not significantly impact its accuracy. However, it's noted that hand geometry is not highly distinctive, meaning that it may not be unique enough to identify an individual. There are some challenges associated with hand geometry biometrics as well. For instance, the information may not remain invariant during periods of growth, particularly in children.

Fingerprint: fingerprints are highly distinctive patterns found on the tips of fingers and thumbs. Fingerprints are formed by raised ridges (known as friction ridges) and furrows on the skin's surface. The unique characteristics of fingerprints have made them a highly reliable method of personal identification for many centuries [3]. The process begins with the live scanning of a person's finger using a fingerprint reader device. The finger is placed on a sensor surface where the reader captures an image of the fingerprint. The preprocessed fingerprint image is then analyzed to extract distinctive features known as minutiae points. Minutiae points are specific points where ridges end, bifurcate, or intersect within the fingerprint pattern. These minutiae points are critical as they form a unique and stable representation of the fingerprint.

Signature: A signature represents a unique expression of an individual's hand geometry and is considered a behavioral biometric. It encompasses not only the general shape of the signed name but also subtle variations in the way it is executed.. Collecting samples

for signature biometrics requires the cooperation of the subject and typically involves the use of a writing instrument. This cooperative nature distinguishes it from other biometrics that may be captured passively. Signatures are behavioral biometrics because they change over time. Factors such as physical condition (e.g., fatigue, injury) and emotional state (e.g., stress, urgency) can influence how a signature is executed, leading to variations in its appearance.

Face: Facial recognition is one of the most widely adopted biometric methods for the identity of a person due to the non-intrusive nature and versatility in various applications. An overview of facial recognition technology and its applications: Facial recognition is indeed one of the most commonly used biometric characteristics for personal identification due to its non-intrusive nature and suitability for various applications. Facial recognition does not require physical contact or special actions from the individual being identified. It captures a person's facial features from a distance, making it suitable for covert recognition applications where users may not be aware they are being identified. A two-dimensional image of the user's face is captured either by a camera or a specialized facial recognition device. The captured image is processed to extract a set of features that represent key facial attributes such as the shape of the eyes, eyebrows, nose, lips, and chin, as well as their spatial relationships.

Retina: The vascular pattern of the retina is indeed unique to each individual and each eye, making it a highly secure biometric for identification purposes. It is difficult to replicate or forge. Unlike some other biometric methods (like fingerprints or facial recognition), retinal scans require the cooperation of the subject. The person must look through a lens at an alignment point for the occurrence of image acquisition. This can sometimes be perceived as intrusive or uncomfortable. While retinal recognition is considered very secure, advances in other biometric technologies (such as facial recognition) that are less invasive and more user-friendly have gained popularity in various applications.

Iris: Development of the iris, which begins in the third month of gestation, results in its unique and intricate patterns being largely established by the eighth month such as arching ligaments, furrows, ridges, crypts, rings, corona, freckles, and a zigzag collaret, are largely established by the eighth month. This early development ensures that each individual's iris pattern is unique. Unlike retinal scanning, iris scanning is non-intrusive and can be performed from a distance of several meters. Capturing the iris using simple optical equipment and utilizing its response to changes in light are indeed key aspects of iris recognition technology. This feature enhances the security and reliability of iris recognition systems. Even identical twins have different iris patterns. Modern iris recognition systems is more user-friendly and cost-effective over time. They can be integrated into other applications which access control, border security, and identity verification. Extracting a feature vector from an iris image requires careful management of light, focus, resolution, and contrast to ensure accurate and consistent identification.

Palm Print: Similar to fingerprints, the features of palm prints such as pattern of ridges, valleys, and other distinctive features. The surface area of the palm is larger which means there are more unique characteristics that can be captured and analyzed, potentially enhancing the reliability of identification. The larger size of the palm and the greater complexity of its patterns make palm prints potentially more reliable than fingerprints for

biometric identification purposes. This increased reliability stems from the uniqueness and variability of palm print features. A comprehensive biometric system can combine various aspects of palm prints, including hand geometry (size and shape of the hand), principal lines (major ridges), wrinkles, and other features.

Voice: The physical characteristics that contribute to creating an individual's voice. The size and shape of the vocal tracts vary among individuals and greatly influence the quality and pitch of their voice. These anatomical features influence the unique sound patterns produced by a person's speech. These physical characteristics are generally invariant for an individual over time, providing a stable basis for voice recognition. The behavioral aspects of speech include the way in which an individual speaks, such as their accent, pronunciation, intonation, and speaking style. These aspects can vary over in terms of age, health conditions, emotional state, or deliberate changes in speech patterns. The two main categories of voice recognition techniques are Automatic Speaker Verification (ASV) and Automatic Speaker Identification (ASI).

DNA: DNA (Deoxyribonucleic acid) is indeed considered one of the most reliable forms of biometric identification due to its unique and highly individualized nature. DNA samples taken from individuals can reveal sensitive information about genetic predispositions to diseases or other medical conditions. This raises significant privacy concerns regarding the storage, use, and potential misuse of genetic information obtained through DNA analysis. Identical twins share identical DNA profiles, which means DNA alone cannot distinguish between them. This limits the reliability of DNA as a biometric identifier in cases involving identical twins. Despite these limitations, DNA remains invaluable in forensic science for identifying suspects, confirming paternity, and linking individuals to crime scenes. Its reliability and uniqueness make it a powerful tool when used appropriately in controlled and regulated environments.

4 Working Methodology

Combining multiple models into unified ensemble is common in developing systems, applications, or devices, ensuring the efficient and effective operation of available equipment. The proposed system incorporates various essential components, which include the following: In both identification and verification modes fingerprint recognition system can be used. The system compares enrolled fingerprint of a user with an input fingerprint in verification mode, so that they are from same finger (1:1) as in Fig. 1, the system compares with the fingerprint of all the enrolled users with an input fingerprint in Identification mode, to obtain the identity from the known set of identities in the database (1:N).

The three categories of Fingerprint representation are as follows:

- **Correlation -based matching:** the process involves systematically computing the correlation between corresponding pixels of two fingerprint images across various displacements and rotations to determine the best alignment that maximizes the correlation coefficient, thereby identifying a match between the two fingerprints.

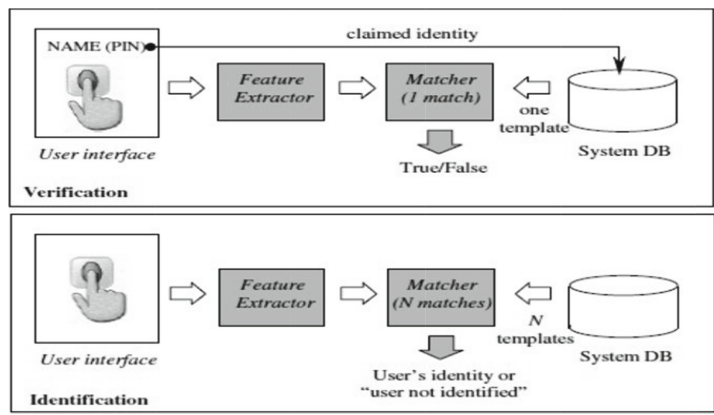


Fig. 1. Verification system for fingerprint recognition

- Minutiae -based matching: This is widely used technique. Minutiae are considered as specific points of interest in a fingerprint such as ridge endings and bifurcations. These are extracted from the fingerprint images and represented as feature sets.
- Texture based matching: Both pattern-based and texture-based algorithms provide alternative approaches to fingerprint recognition when minutiae extraction may be challenging or insufficient. These methods leverage different aspects of fingerprint images—overall patterns and detailed textures—to achieve accurate identification and verification in various applications ranging from security to forensic investigations. Each approach has got its own strengths and based on the specific requirements and conditions of the application environment.

As described above, Each type of fingerprint representation—correlation-based, texture-based, and minutiae-based—has its own strengths and challenges. The choice of representation often depends on the specific requirements of the application, the quality of fingerprint images, and the desired balance between computational efficiency and matching accuracy.

Signature Recognition

A handwritten signature is a person’s unique way of writing their name or marking a document, often used as proof of identity or intent. A handwritten signature is a person’s unique way of writing their name or marking a document, often used as proof of identity or intent. t serves as a legal mark indicating consent, authorization, or verification on documents such as contracts, letters, or forms.

Handwritten signature verification involves assessing the similarity between a query signature and a stored reference signature to determine authenticity. While traditionally done through visual inspection, technological advancements are enabling more objective and automated methods. The reliability of signature verification is crucial for ensuring the integrity and legality of documents and transactions in various sectors.

Online (dynamic) signatures are captured using digital devices like pen pressure-sensitive tablets or stylus pens that record not only the shape but also dynamic properties of the signature as it is written. While both online and offline signatures serve the purpose of authentication, online signatures offer superior security features due to their dynamic properties, making them more resistant to forgery and more suitable for digital transactions and legal authentication processes in Fig. 2.

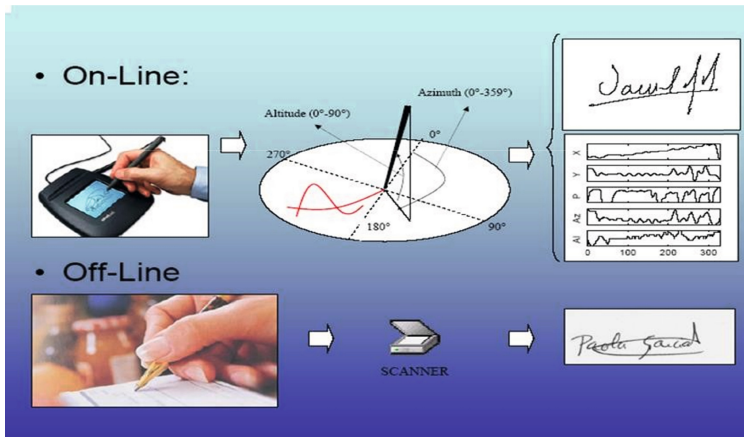


Fig. 2. Online and Offline Signature capturing and different types of signature verification techniques

5 Technical Specification, Configuration and Process

After conducting the experiment, the following outcomes shown in Fig. 3 have been achieved:

Voice Module – A fingerprint is characterized by the dermatoglyphic patterns found on the surface of the finger. It consists of spaces in between the ridges called furrows on the surface of the finger, and these patterns do not change over time except for mechanical alterations like scars and cuts.



Fig. 3. Fingerprint showing with ridges and furrows

- Minutia is a particular shape of ridges and valleys pattern at the local level. Among different types of minutia: ridge bifurcation and ridge ending are considered in Fig. 4 and Fig. 5.

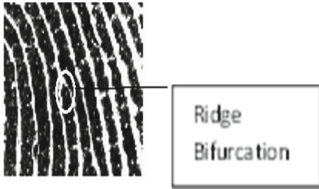


Fig. 4. Ridge Bifurcation

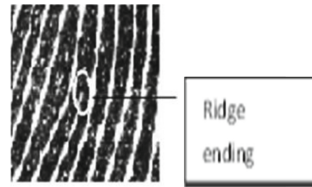


Fig. 5. Ridge ending

- Singularities is a particular shape of fingerprint pattern at a global level. It is classified into three types such as loop, delta and whorl as shown in Fig. 6

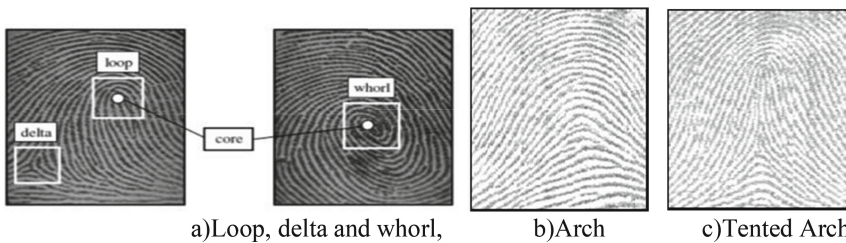


Fig. 6. a) Loop, delta and whorl, b) Arch c) Tented Arch

Loop: is a type of fingerprint pattern where one or more ridges on one side of the impression recurve, and touch or pass near the delta (a point on the fingerprint pattern where ridges diverge) before terminating or tending on the same side of the impression to be terminated from which they entered.

Whorl: Is one of the fingerprint pattern where at least one ridge makes a complete circuit around a central point or points. Whorls are characterized by the presence of two deltas (triangular-shaped ridge patterns) and at least one ridge makes a complete circuit. The whorl class is bit complex and it is classified into 2 types: twin loop and plain whorl.

Arch: (a plain arch): In fingerprint classification, an arch is characterized by ridges that enters one side of the impression, which makes a rise or wave in the center, and flow or to the other side as shown in Fig. 6b.

Tented Arch: Similar to an arch pattern, ridges in a tented arch enter one side of the impression, then rise towards the center, and flow or tend to flow out on the other side. Unlike a plain arch, the feature of a tented arch is the presence of at least one loop

formation and one delta. The loop is a ridge pattern that recurves back on itself, and the delta is a triangular-shaped ridge pattern located near one of the sides of the tented arch shown in Figure 6c.

A high resolution fingerprint imaging device came into existence by Biometrics Research Centre (UGC/CRC) of the Hong Kong Polytechnic University, in which it is used to construct large-scale high resolution fingerprint databases (HRF). It is also designed for effective and efficient algorithms in which it extracts and matches fingerprint additional features.

There are two High resolution databases of fingerprint images(denoted as DBI and DBII).

In order to test and train algorithms in biometric system , finger print recognition databases were created. Each fingerprint image is named in the format “ID_S_X”.

- **ID:** is the identity of the person. Each person has a unique identifier.
- **S:** Represents the session when the fingerprint image was captured. There are two sessions for each person.
- **X:** Represents the image number within each session. This indicates multiple images may have been captured for each session.



Fig. 7. Sample images

In Fig. 7, LPQ is a texture descriptor that captures local phase information in an image. It's computed locally for each pixel and is particularly useful for texture classification tasks. The LPQ is computed for a specific pixel within the image, which is described as 'colored gray'. In LPQ computation, Short-Time Fourier Transform coefficients are used. These coefficients are computed by an M -by- M neighborhood around the pixel for the lowest horizontal, vertical, and diagonal frequencies $(1, 0)$; $(0, 1)$; $(1, 1)$; and $(1, -1)$ (in Figure 12 $M = 5$). After quantization based on sign, you obtain four binary digits (bits) for each complex coefficient (two bits for the real part and two bits for the imaginary part). This results in an 8-bit binary number which is then interpreted as a decimal number, ranging from 0 to 255. Optimizing the LPQ descriptor involves decorrelating frequency coefficients before quantization to maximize information preservation. This is achieved by assuming isotropic image blur and using efficient 1-D convolution techniques for computing frequency coefficients. These optimizations enhance the descriptor's performance in texture analysis tasks, such as classification and recognition, within image processing and computer vision domains.

LPQ (Local Phase Quantization)

The rotation-invariant extension to the Blur Insensitive LPQ descriptor enhances its utility in texture classification tasks, particularly in fingerprint recognition shown in Fig. 8.

The Local Phase Quantization (LPQ) descriptor is derived from the Discrete Fourier Transform (DFT) computed over local image windows, often referred to as the Short-Term Fourier Transform (STFT).

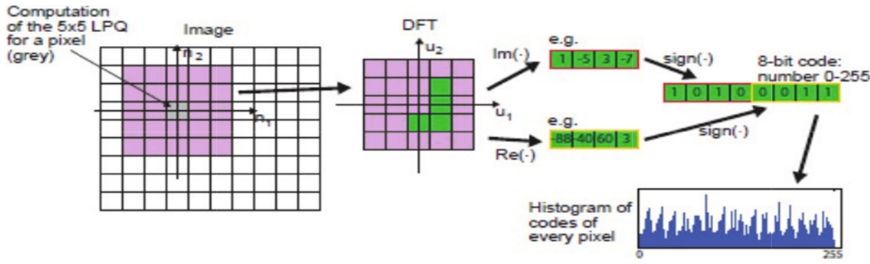


Fig. 8. Texture analysis

(STFT) - LPQ is computed based on the quantized phase information obtained from the Short-Term Fourier Transform (STFT) coefficients within a neighborhood. To describe the entire image or a local region, a histogram is constructed where each bin corresponds to a possible combination of the 8-bit LPQ codes (ranging from 0 to 255). These histograms serve as effective features for texture classification tasks, leveraging distance metrics such as Euclidean distance to classify textures based on their similarity in local texture patterns..

The comparison is done between the Local Phase Quantization (LPQ) method and the Local Binary Pattern (LBP) method and also Gabor filter bank-based method, reveals interesting insights into their performance in texture analysis:. LPQ exhibits higher tolerance to image blurring compared to LBP. This is due to LPQ’s focus on the phase information of the frequency components rather than amplitude, which can be more sensitive to blur. Gabor Filter are designed to capture both spatial frequency and orientation information, making them versatile for texture analysis.

A Signature as a Signal

“The human motor system influences the dynamics of a signature, which is typically defined as a sequence of strokes. These strokes can be categorized and classified based on their assembly.”

The confusion becomes apparent, when looking at the various definitions of a stroke:

- Cognitive feature: A cognitive feature can be understood as a component within a signature that holds perceptual significance. This definition becomes clearer when we consider its frequent occurrence across various signatures, or when the writer consistently draws attention to that specific part of their signature.”

- Time segment: A time segment in signature analysis refers to the interval between two consecutive changes in velocity. The assumption underlying this definition is that the reasons behind these velocity changes may be linked to perceptual relevance.”
- The segment between two consecutive changes in velocity in signature analysis implies that these changes occur for perceptually relevant reasons.”
- the part between two consecutive velocity sign changes. The assumption is here that there must be a reason why the velocity changes and this may have something to do with perceptual relevance.
- Connected component: in signature analysis refers to the segment between the moments when the pen lifts up and touches down again. This phenomenon is often interpreted as indicating moments of perceptual relevance.

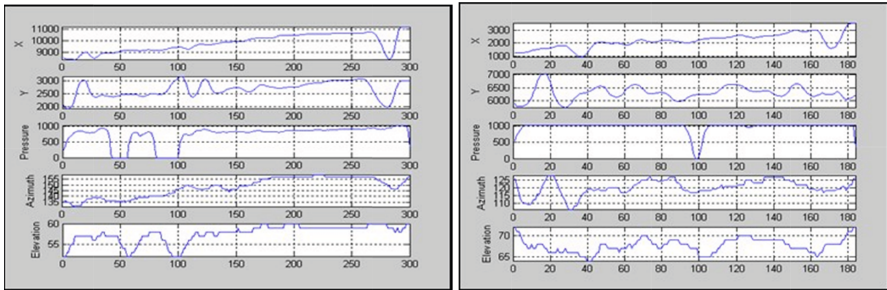


Fig. 9. Signature and related feature as signals

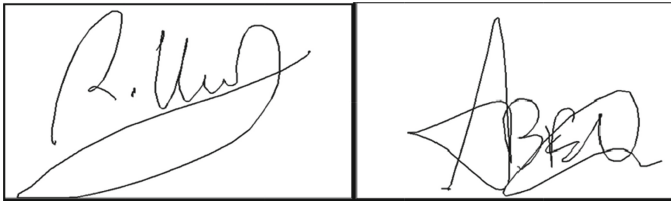


Fig. 10. Samples of signature

Figure 9 shows related feature as different signals. In literature, different features such as X and Y coordinates sequence, Z as Pressure coordinates sequence, AZ as Pen Azimuth coordinates sequence, In as Pen Elevation coordinates sequence, have been used for signature verification in different systems. The reliability of these systems is yet far from human expectation. Therefore, in this paper two different biometric modalities (fingerprint and online signature), the system adds an extra layer of security. Both fingerprint and online signature are unique to individuals and difficult to forge or replicate, shown in Fig. 10.

System Design

Figure 11 modules of our signature verification system is shown. A user's signature is

given as input during the time of enrollment of a new user. Then the data is preprocessed and then the feature extraction takes place. Along with unique identifier, the data is then saved in a database then it is used to retrieve the signatures during matching.



Fig. 11. Modules of a signature verification system

After extracting features from the training signatures, a threshold value is derived by considering matching score. This verification process ensures that the signature verification system maintains a balance between security (through the threshold) and usability (through efficient matching algorithms). It leverages the stored signature templates and derived thresholds to reliably authenticate users based on their signature characteristics. The preprocessing and feature extraction steps are applied. Then signature is compared to each and every samples of training set which is retrieved from the database.

6 Results

The experimental results are shown below tables for different combinations of training and testing using classifier K-NN with Euclidean distance measure (Tables 1, 2 and 3).

Table 1. Result of Recognition, Recall, Precision, Fmeasure of Signature by using different features.

No. of Samples = 100		No. of train = 3		No. of test = 2
Features	PenPressure + Azimuth + Elevation	Pen Pressure (Z)	Azimuth (AZ)	Elevation (In)
Recognition	81	76.5	22.5	19
Recall	81	76.5	22.5	19
Precision	83.2	78.4	20.2	20.5
Fmeasure	82.10	77.47	21.3	19.73

Unimodal biometrics are experimented with different samples of training and testing for online signature, the best accuracy was 83.2% with 3 training samples and 2 testing samples (Table 4).

Table 2. Result of Recognition, Recall, precision, Fmeasure of Signature by using different number of samples for training and testing.

No. of Samples = 100		No. of train = 2		No. of test = 3
Features	PenPressure + Azimuth + Elevation	Pen Pressure (Z)	Azimuth (AZ)	Elevation (In)
Recognition	60	56.33	18.66	14.33
Recall	60	56.33	18.66	14.33
Precision	61.23	57.08	16.19	13.32
Fmeasure	60.61	56.70	17.34	13.81

Table 3. Result of Recognition, Recall, precision, Fmeasure of Signature by varying samples of training and testing.

No. of Samples = 100		No. of train = 4		No. of test = 1
Features	Pen Pressure + Azimuth + Elevation	Pen Pressure (Z)	Azimuth (AZ)	Elevation (In)
Recognition	75	72	8	15
Recall	75	72	8	15
Precision	66.75	62.75	4.79	8.86
Fmeasure	70.63	67.05	5.99	11.14

Table 4. Fusion Result of Fingerprint and Signature

No. of Samples = 100	No. of train = 3	No. of test = 2
Fingerprint	Signature	Recognition
Lpq_basic	Pen pressure	94
Lpq_basic	Azimuth	92
Lpq_basic	Elevation	91.5
Lpq_basic	Pen pressure, Azimuth, Elevation	91.5
Lbp	Pen pressure	90
Lbp	Azimuth	82
Lbp	Elevation	84
Lbp	Pen pressure, Azimuth, Elevation	88.5

When unimodal biometrics are experimented with different samples of training and testing for fingerprint the best accuracy was 83.5% with 3 training samples and 2 testing

samples, for online signature the best accuracy was 83.2% with 3 training samples and 2 testing samples, where as in multimodal, by performing different experiments on the combined feature matrix of fingerprint and signature, the recognition rate was 94% and varying training and testing samples it was resulted out with the best recognition rate of 95%.

7 Conclusion

Designing multimodal biometric systems indeed presents several challenges due to the diverse nature of biometric resources and the complexities involved in integrating them effectively. We have selected two different modalities fingerprint and online signature, in this project and performed fusion to increase the system performance for recognition of an individual. First, challenges were approached in unimodal fingerprint matching from an information fusion perspective. The texture based information present in the fingerprint is extracted from local phase quantization (LPQ) method; The LPQ (Local Phase Quantization) method is considered as an advanced technique that differs significantly from LBP (Local Binary Pattern) methods, particularly in how it handles feature extraction and its applicability to solving complex problems in biometrics, particularly with local minutiae features. Then, designing of a unimodal online signature verification system based on spectral analysis by including features such as Pen Pressure, Azimuth, and Pen Elevation. The spectral analysis of an online signature offers several advantages, particularly in the context of compact representation and efficient matching algorithms. Using global features in fingerprint analysis offers several advantages, one of which is the complementary nature of the information they provide when combined through fusion techniques. Prior to combining the feature sets from two biometrics, we normalized the scores using z-score normalization method because the two feature sets resulted with heterogeneous data. Feature level fusion combines feature vectors from different biometric modalities (such as signatures and fingerprints) into a unified representation for each user in the database. Classification was performed using the effective and robust K-NN classifier.

However, a multimodal biometric system not only enhances matching performance but also effectively addresses challenges like non-universality and spoofing that are prevalent in unimodal systems. By leveraging the strengths of multiple biometric modalities, such systems offer robust security solutions suitable for a wide range of applications.

References

1. Mane, V.M., Jadhav, D.V.: Review of multimodal biometrics: applications, challenges and research areas. *Int. J. Biom.* (2009)
2. Delac, K., Grgic, M.: A survey of biometric recognition methods. In: 46th International Symposium Electronics in Marine, ELMAR-2004, Zadar, Croatia, June 16–18 2004 (2004)
3. Kong, A.W.K.: Palm print Identification Based on Generalization of Iris Code, Ontario, Canada, Waterloo (2007)
4. Anand, P.M.R., Bajpai, G., Bhaskar, V.: 3D signature for efficient authentication in multimodal biometric security systems. *IACSIT Int. J. Eng. Technol.* **2**(2) (2010). ISSN: 1793-8236

5. Richiaadi, J., Drygajlo, A.: Quality measures in unimodal and multimodal biometric verification. In: 15th European Signal Processing Conference(EUSIPCO 2007), 3–7 September 2007. Copyright by EURASIP
6. Mhatre, A., Palla, S., Chikkerur, S., Raju, V.G.: Efficient search and retrieval in biometric databases, Issue 7 (2009)
7. Imran, M., Kumar G, H., Jameen, N.S., Alaei, F.: Accurate person recognition on combining signature and fingerprint. *Int. J. Mach. Intell.* **3**(4), 277–281 (2011). ISSN: 0975–2927, E-ISSN: 0975–9166
8. Sabareeswari, T.C., Lenty Stuwart, S.: Identification of a person using multimodal biometric system. *Int. J. Comput. Appl.* **3**(9) (2010). 0975–8887



Reinforcement Learning with Enhanced Bubble Sort for Wireless Mesh Networks Node Selection

G. Revathy¹(✉), J. Senthilkumar², E. Gurumoorthi³, and M. Shyamalagowri⁴

¹ Department of CSE, SRC, SASTRA Deemed University, Kumbakonam, India
revathyjayabaskar@gmail.com

² Department of CSE, Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology, Chennai, India

³ Department of IT, CMR College of Engineering and Technology, Hyderabad, India

⁴ Department of EEE, Erode Sengunthar Engineering College, Erode, India

Abstract. Wireless Mesh Networks (WMN) plays a vital role in modern day communication. It transfers the data to every node through hop by hop mechanism. It can act as a client as well as the server. The data transmitted to every node from source to destination and hence the next node selection plays a challenging task in WMN. The previous state of art models shows that the node selection is done with lot of time waste or a wrong node is selected. Hence, we propose a novel methodology in which enhanced bubble sorting is used to identify the next node and reinforcement learning is applied every time for the verification of exactness of the node. Traditional sorting algorithms are simple to implement but they lack in efficiency for handling a large number of mesh nodes. The enhanced bubble sort reduces the number of unnecessary comparisons based on the reward and loss of the reinforcement learning and hence choose the best node.

Keywords: WMN · Reinforcement learning · Enhanced bubble sort

1 Introduction

With the help of mesh nodes—which can function as both clients and servers—Wireless Mesh Networks (WMNs) produce a communications structure, allowing for the low-cost formation of networks across a large region. Even after the communication situation is bowdlerized due to catastrophe across a large zone, they nevertheless act rapidly and widely to restore the networks. The Internet of Things (IoT) era is projected to use WMNs as the communication infrastructure for edge systems, although they can also be used in other situations. The choice of the following node, however, has a big impact on how the network functions, plus fitting costs, communication distance, and conservation [11–13].

The goal of the research is to increase the recital of wireless communication complete WMNs by selecting mesh nodes more effectively. Several metaheuristics, including Genetic Algorithms (GA), Hill Climbing (HC), Simulated Annealing (SA), Tabu Search (TS), and Particle Swarm Optimization (PSO), can be used to optimize mesh node selection, however they are not always successful.

With their ubiquitous application in a variety of computing operations and systems, sorting algorithms have a fundamental and important place in the field of computer science. These algorithms are essential for allowing computers to analyze and alter data effectively. They can be found at the foundation of database management systems, in search algorithm optimization, in numerical computations, and in data analysis. Given their major impact on overall computing time and resource utilization, these sorting algorithms' effectiveness and efficiency are increasingly more important as data quantity increases.

With their ubiquitous application in a variety of computing operations and systems, sorting algorithms have a fundamental and important place in the field of computer science. These algorithms serve a crucial role in enabling computers to process and alter data effectively, from the heart of database management systems to the optimization of search engines, to numerical computations and data analysis. With their major impact on overall processing time and resource use, these sorting algorithms' effectiveness and efficiency become even more critical as data sizes increase.

Although artificial intelligence (AI) systems are currently outperforming humans in certain target areas like speech and picture recognition, most jobs that we conduct on a daily basis can still be handled considerably more readily by the human brain than by AI. A state and a prize from the situation are provided to an agent during reinforcement learning. Finding the right course of action is the agent's responsibility. The right behaviour in reinforcement learning is one which maximizes the reward, which is often referred to as exploitative behaviour. However given the lack of rewards, it might not be possible to solve issues only through exploitation. The agent has nothing to exploit because reward sparsity makes it unlikely that it will discover a reward immediately. Because of this, solving issues with sparse rewards requires an exploration method.

2 Literature Survey

As a result, authors claim that closing data must frequently be transferred between internal and external memory, external sorting—a core technique of database management systems—suffers from a high I/O burden. Non-volatile memory could be an alternative (NVM), however, cannot be used with the standard external sorting algorithm because it was designed specifically for disk storage. The benefit of LazySort over conventional algorithms is also amplified as the volume of data grows [1].

Authors says that multi-criteria sorting problems are a common model for many real-world problems. Many options are categorized into predetermined types in this kind of situation. Provide a novel hybrid way to classifying choices while taking decision-maker restrictions into account. Easy cluster best-worst and multiple-criteria categorization methods will both be used in this strategy. As a result, the decision-maker may find the results ineffectual. The weighting methodology provided in this study can be regarded as the maximum real cluster best-worst technique ever published because it can be solved without an optimization model or specialized software. It is also suggested to use a way of sorting using several criteria. The results demonstrate the validity and accuracy of the methodologies that were created. The recommended method can be used to solve classification-based real-world challenges and aids managers in considering their restrictions during the decision-making process [2].

The authors review deep reinforcement learning exploration strategies. Exploration strategies are a key factor in problem-solving with little reward. In situations when incentives are rare, the reward is infrequent, therefore the agent won't often find the reward by acting arbitrarily. Reinforcement learning finds it difficult to associate rewards with behaviors in such a situation. Hence, more advanced exploration techniques must be developed [3]. Authors said that data privacy can be efficiently safeguarded with fully homomorphic encryption (FHE). Users can accomplish this on a cloud server deprived of skimpy their identities and trust a third party with handling their data. The Users may envoi data processes to a third party on a cloud server and do so deprived of divulging their identities. The fundamental data management and processing algorithms. According to the experiments, when an element is larger than 32 bits, AdditionSort outperforms the bubble sort by over 50% [4].

Viewpoints of the authors Ad hoc networks, such as MANET, link a large number of mobile devices to form decentralized wireless mesh networks. Because of its intricacy and dynamic nature, MANETs are a type of network that frequently changes its topology. It is feasible to reduce the cost of topology by utilizing the cluster-based structure. The clustering problem is formulated using Elephant Herding Optimization to select the optimal cluster head (CH) in MANET (EHO). The simulation results that were utilized to validate the proposed EHO technique show that the work creates stable clusters where simulation results are compared to existing algorithms. In terms of measures like network durability and clustering overhead, the EHO performs better than older methods [5].

ExSeisDat was created by the authors to manage seismic data on high-speed super-computer clusters utilizing the standard message passing interface (MPI) library. The bulk of the time, these clusters are built to handle complex workloads and large-scale IO well. Several programs trying to access data from parallel networked storage raise concerns about IO performance reduction. Both the lack of innovation in the storage hardware and the constrained set of protocols used by the parallel file system (PFS) that manages the disks are factors in these issues. Users who are mainly concerned with creating parallel applications usually miss this, demanding and resulting in the changing of specific configuration parameters to improve IO performance. Yet, there are specific situations where it is necessary to change the setting values. With a huge seismic data file scaling to petabytes in SEG-Y format, all of this leads to further IO performance reduction. Two of ExSeisDat's most crucial functions are the SEG-Y IO and file sorting processes. This article presents a strategy for leveraging artificial neural networks to enhance current SEG-Y approaches (ANNs). To forecast IO bandwidth and optimize the relevant configuration parameters, machine learning (ML) uses trained ANN models. We also go through statistical analysis of auto-tuning bandwidth findings and the impact of hidden layer node structure variations on ANN prediction accuracy [6].

Authors says that each day, trillions of times are used for simple algorithms like sorting and hashing. The need for these algorithms to be as performant as possible has grown along with the need for computation. Although there has been notable progress in the past, it has been difficult for both human scientists and computational technologies to further increase the effectiveness of these routines. Here, we demonstrate how finding previously undiscovered routines allows artificial intelligence to development beyond the current public of the art. Designed the objective of discovering a improved sorting

procedure as a single-player game to realize this. Subsequently taught AlphaDev, a fresh deep reinforcement learning agent, how to show this game. Little categorization algorithms were created from scratch by AlphaDev and outperformed existing human standards. The LLVM standard C++ sort library³ now incorporates these methods. With this modification to this section of the sort library, an algorithm that was automatically found by reinforcement learning takes the place of a component [7].

Authors claim that Wireless Mesh Networks (WMNs) are capable of developing a communications infrastructure with only routers (referred to as mesh routers). Mesh routers, also known as mesh clients, enable mesh clients to communicate with various nodes. Because it influences how well WMNs communicate, mesh router placement can be changed to increase communication performance. Here, we offer a mesh router placement technique that maximizes coverage (CCM). A combined optimization method that applies both Simulated Annealing (SA) and Delaunay Edges (DE) in CCM is also provided in order to increase the effectiveness of mesh router placement optimization. The suggested method gives WMNs the ability to establish and supply a communication infrastructure in emergency scenarios. By placing mesh clients in Kurashiki City, an evacuation zone for Japan, we take into account a real-world scenario. The simulation results show that using the suggested technique, mesh routers may be placed in the best locations to reach every mesh client in the evacuation zone. The DECCM-based SA strategy, which frequently covers more mesh clients than the CCM-based SA strategy, can also increase WMN network connectivity [8].

For real-time pressure process control and industrial wireless mesh networks (WMNs) optimization, the authors suggest a novel hybrid Harris Hawk-Arithmetic Optimization Algorithm (HHAOA). In order to address position relocation issues, premature convergence, and the subpar accuracy of the existing methodologies, the suggested method incorporates concepts from the Arithmetic Optimization Algorithm and Harris Hawk Optimization. The Arithmetic Optimization Algorithm, Moth Flame Optimization, Sine Cosine Optimization, GreyWolf Optimization, and Harris Hawk Optimization were used to test the HHAOA methodology against a number of benchmark functions as well as in comparison to other optimization methods.

3 Methodology

By hopping from node to node, data is conveyed. The hardest part of mesh network operations is choosing the next node. We recommend a tweaked Bubble sort algorithm to choose the following node. It is necessary to tackle the problem with a methodical and thorough technique that takes into account mesh network practicalities in the effort to optimize Bubble Sort. We may check whether the list has already been sorted by keeping note of whether any swaps were made during each pass. If none were, we can stop the algorithm early and see if the list is already sorted.

Due to this enhancement, the algorithm is now able to attain a best-case time complexity of $O(n)$, which is much better than the average and worst-case outcomes. The biggest element in the list is always placed at the end of the list after each pass. It is a straightforward observation that once an element has been placed correctly, it is unnecessary to include it in further comparisons. The final item on the list can thus be left out of

the subsequent pass. This improvement immediately lowers the amount of comparisons and swaps carried out during each pass, reducing the algorithm's temporal complexity and actual runtime.

It is important to retain in attention that a variability of issues, with the programming language, the hardware used to perform the algorithm, the size and distribution of the dataset, and the specific implementation of the method, might affect how long it takes to complete a task. We made sure that the standard and Optimized Bubble Sort algorithms were implemented in the same setting, on the same hardware, and using the same programming language in order to reduce the effects of these external factors and concentrate on the inherent efficiency of the algorithms. The quantity of passes through the list was a different performance metric we took into account. Fewer passes often result in higher performance because each pass through the list represents a full iteration over all of its components. As the number of passes closely correlates with the number of comparisons and swaps performed by the algorithm, this parameter is particularly important for Bubble Sort and its optimized counterpart.

Reinforcement An agent learns how to function in a particular environment by taking actions and seeing the results of those actions, which is a feedback-based machine learning technique. The agent gets positive feedback for every action that is effective, and they get bad feedback or a penalty for every unsuccessful action. RL algorithms that use models as well as those that don't can both be generically described. The environment in general, or the MDP in particular, is not particularly represented by model-free methods. They are similar to algorithms that pick the best course of action after testing environmental ideas through behavior. The two types of algorithms that lack models are value-based and policy-based.

The promise of value-based algorithms is that the exact prediction of each state's value function leads directly to the formulation of the best possible policy. The agent engages with the environment to sample the state and reward trajectories using the Bellman equation as a recursive relation. The MDP's value function can be roughly estimated with enough trajectory data. Once the value function is understood, all that's left to do is act greedily toward it at each step of the process to determine the best course of action. Q-learning and SARSA are two examples of value-based algorithms. Without having to describe the value function, policy-based algorithms can predict the best course of action.

The learning problem becomes an explicit optimization problem when the policy is directly parametrized using learnable weights. Similar to value-based algorithms, the agent samples the trajectory of states and rewards, but by maximizing the average value function across all states, this knowledge is specifically employed to improve the policy. The Monte Carlo policy gradient (REINFORCE) and the deterministic policy gradient are two well-known policy-based RL techniques (DPG). High variation in policy-based methods causes instability to show up throughout training. Although more trustworthy, value-based modeling methodologies are inadequate for continuous action space modeling. One of the best RL algorithms, actor-critic combines a value-based and a policy-based methodology. The value function (critic) and the policy (actor) in this method are both parametrized to enable the efficient use of training data with continuous convergence.

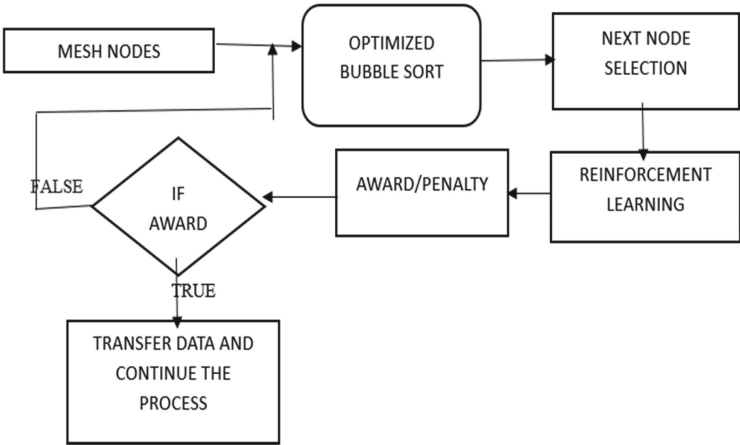


Fig. 1. Architecture Diagram

Figure 1 explains the workflow architecture of the model. Mesh nodes are passed as input to Optimized bubble sort. Using optimized bubble sort very fast next node is selected. The selected node is applied for Reinforcement learning. The RL mechanisms verifies whether the selected node is the next best node. If the node is best node award is generated and if not penalty is generated. If award we proceed continuously with next nodes, if penalty we again apply optimized bubble sort and the process continues till best node selected.

4 Results and Discussions

Using mesh nodes of various sizes, we assessed the performance of the conventional Bubble Sort method and the Improved Bubble Sort algorithm (ranging from 10 to 1000 nodes). For each algorithm, datasets were generated at random and run to measure the average execution time. Table 1 shows that the Improved Bubble Sort consistently showed faster execution times than the conventional Bubble Sort.

Table 1. Comparison Table for Sort

Elements	Avg. Time (Traditional) in ms	Avg. Time (EBS) in ms
10	0.06	0.01
100	0.15	0.8
500	1.6	1.1
1000	2.1	1.6

In Fig. 2, the computation time is computed based on the Table 1.

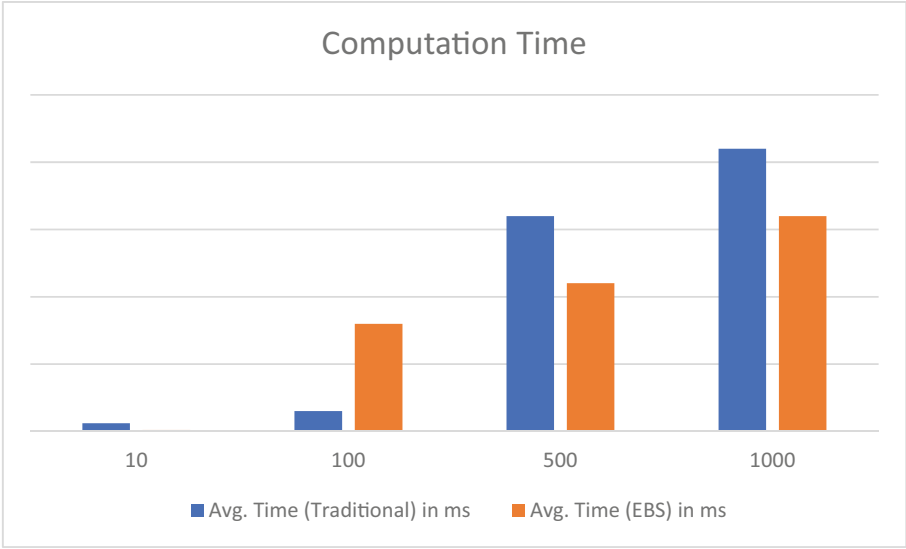


Fig. 2. Computation Time Graph

In Fig. 3 the Enhanced bubble sort with Reinforcement learning is compared with Tabu search with distributed scheduling Optimization [10] in which our EBS+RL gives the best results and outperforms TS+DS

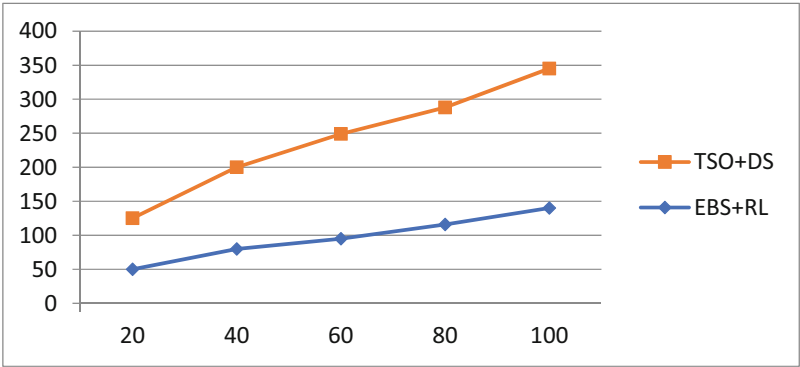


Fig. 3. Throughput Comparison

Figure 4 shows the reduction in error rate graph, When compared with traditional TSO with DS shows high error rate when compared with our proposed EBS +RL approach for wireless mesh networks.

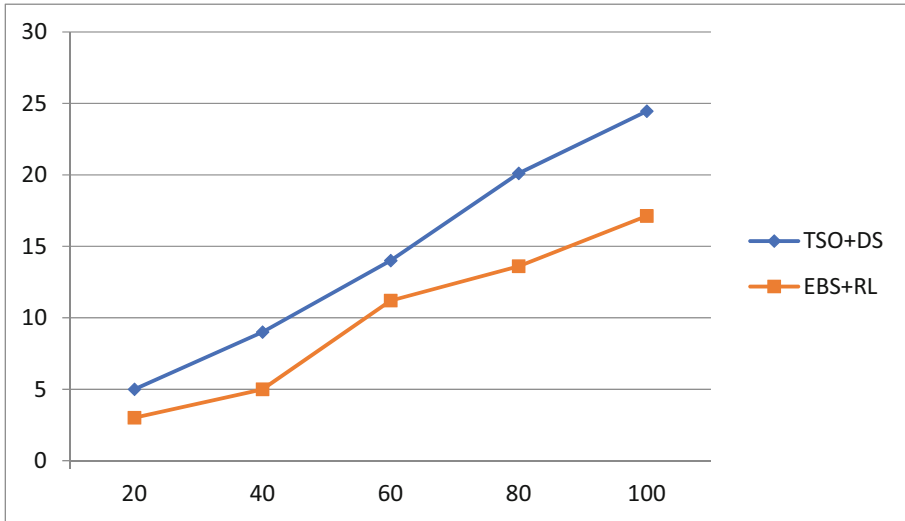


Fig. 4. Error rate reduction

5 Conclusion

In the modern era of communication the most fast and efficient transfer is needed among the nodes in mesh. The proposed model enabled with enhanced bubble and reinforcement learning shows best results when compared with previous state of art model. Hence transmission can be done in a more quick and efficient means in the mesh networks.

References

1. Liu, Y., et al.: LazySort: a customized sorting algorithm for non-volatile memory. *Inf. Sci.* **641**, 119137 (2023)
2. Emamat, M.S.M.M., et al.: A novel hybrid simplified group BWM and multi-criteria sorting approach for stock portfolio selection. *Expert Syst. Appl.* **215**, 119332 (2023)
3. Ladosz, P., et al.: Exploration in deep reinforcement learning: a survey. *Inf. Fus.* **85**, 1–22 (2022)
4. Wang, C., et al.: An efficient fully homomorphic encryption sorting algorithm using addition over TFHE. In: 2022 IEEE 28th International Conference on Parallel and Distributed Systems (ICPADS). IEEE (2023)
5. Devi, S.U., Selvanayagi, D.: An efficient optimal cluster head selection algorithm in MANET using elephant herding optimization. In: 2023 Second International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT). IEEE (2023)
6. Tipu, A.J.S., Conbhui, P.Ó., Howley, E.: Seismic data IO and sorting optimization in HPC through ANNs prediction based auto-tuning for ExSeisDat. *Neural Comput. Appl.* **35**(8), 5855–5888 (2023)
7. Mankowitz, D.J., et al.: Faster sorting algorithms discovered using deep reinforcement learning. *Nature* **618**(7964), 257–263 (2023)

8. Oda, T.: A delaunay edges and simulated annealing-based integrated approach for mesh router placement optimization in wireless mesh networks. *Sensors* **23**(3), 1050 (2023)
9. Devan, P.A.M., et al.: A novel hybrid Harris Hawk-Arithmetic Optimization Algorithm for industrial wireless mesh networks. *Sensors* **23**(13), 6224 (2023)
10. Revathy, G., Selvakumar, K.: Route maintenance using Tabu search and priority scheduling in wireless mesh networks. *J. Adv. Res. Dyn. Control Syst.* **9** (2017). ISSN 1943023X
11. Revathy, G., Selvakumar, K.: Channel assignment using Tabu search in wireless mesh networks. *Wirel. Pers. Commun.*
12. Revathy, G., Selvakumar, K.: Escalating quality of services with channel assignment and traffic scheduling in wireless mesh networks. *Clust. Comput.* (2018)
13. Revathy, G., Selvakumar, K., Sustain route by Tabu and amplified QoS by distributed scheduling in WMN. *Int. J. Recent Trends Eng. Res.*



Development of an Soil Health Monitoring IoT Integrated System

M. C. Jayaprakash¹(✉), Vinayambika S. Bhat¹, and M. P. Shama²

¹ Mangalore Institute of Technology and Engineering, Affiliated to V.T.U., Belagavi, Approved by AICTE, New Delhi, Badga Mijar, Moodabidri, Karnataka 574225, India
jayaprakash@mite.ac.in

² Aarbee Structures Pvt. Ltd., Bengaluru, Karnataka, India

Abstract. In this study, developed integrated soil health monitoring system using the Internet of Things (IoT) for evaluating soil health which is irrigated by the effluent from Sewage Treatment Plant (STP). The system makes use of sensors to gather information on different characteristics like pH levels, temperature, soil moisture and NPK. The microcontroller unit receives the collected data and using pre-programmed threshold values activates the integrated system. A cloud-based platform for data processing and storage is also part of the system. The proposed system may be able to handle issues like the necessity for regular monitoring and the fluctuation of effluent quality that are related to using STP effluent for irrigation. The technology makes irrigation precise and effective, which promotes healthy plant development and healthy soil conservation.

Keywords: IoT · Soil health monitoring · Cloud-based platform · Microcontroller · STP effluent · Soil conservation

1 Introduction

Water adaptation strategies are being studied considering the anticipated effects of global warming in order to guarantee water availability for agricultural production, human consumption, and ecosystem sustainability. In addition, it's important to keep water safe for consumption by humans and for reuse in the environment. Risks from climate change include increased water scarcity, poor water quality, increased soil salinity, biodiversity loss, increased irrigation needs, and the cost of emergency and remedial action. These variables have led to an increase in the number of studies that concentrate on innovative ways to use water for irrigation. The Internet of Things (IoT) is now more than just an idea; it is being used in practical applications. Since then, there have been a lot of technology and application challenges [1].

In this scenario, the incorporation of sensors into IoT devices ensures a secure connection to the real world. Modern IoT devices do indeed have a variety of sensors, including accelerometers, gyroscopes, microphones, light sensors, and others, allowing for more effective and user-friendly applications. IoT devices can improve any ongoing task efficiently by using these sensors to detect any changes in their environment. The IoT sector expanded at a phenomenal rate as a result of the growing acceptance and usefulness of IoT devices in a variety of application domains [2]. It is advantageous for the user to interact with sensors from anywhere in the Arduino, which reduces power consumption by increasing system life and executes on a wide scale for comparatively little investment, thanks to the communication through the website. The system can also be created for a cooling system with temperature sensors. They must stop engaging in other activities because they must wait until the field has been properly watered even after that [3]. The irrigation system will be controlled by the suggested system according to the amount of soil moisture in order to maximize water utilization. This increases agricultural productivity and lessens water waste. Also, the device offers current data on the amount of soil moisture, assisting farmers in making well-informed irrigation decisions [4].

1.1 Methodology

The development of the proposed system includes its hardware and software components, and its design and implementation. A real-world testing is used to evaluate the system, and the findings demonstrate how well the suggested approach works to maximize water use in agriculture. Although irrigation is difficult in dry areas and there is less rainfall there. Because of its low cost, ease of application, and ease of maintenance, the ESP8266 Wi-Fi based communication system was chosen. A soil moisture sensor is used in an Internet of Things based smart irrigation system to optimize water utilization in agriculture. A soil moisture sensor, a microcontroller, a Wi-Fi module, and a web application make up the system. The microcontroller receives the data from the soil moisture sensor, which is used to gauge the soil's moisture content. Using the Wi-Fi module, the microcontroller processes the data before sending it to the cloud server. The online application collects the information from the cloud server and offers real-time data on the irrigation system and soil moisture level (Figs. 1, 2 and 3).

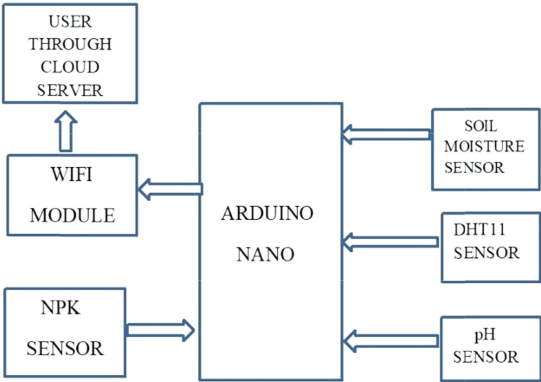


Fig. 1. Block Diagram of Proposed IoT System

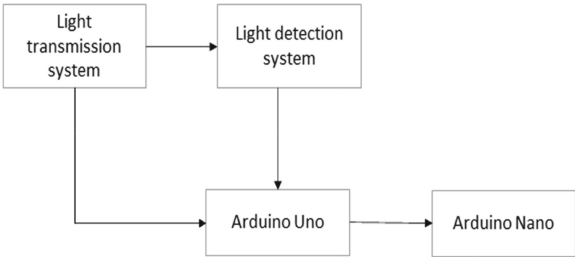


Fig. 2. Block Diagram for NPK Sensor

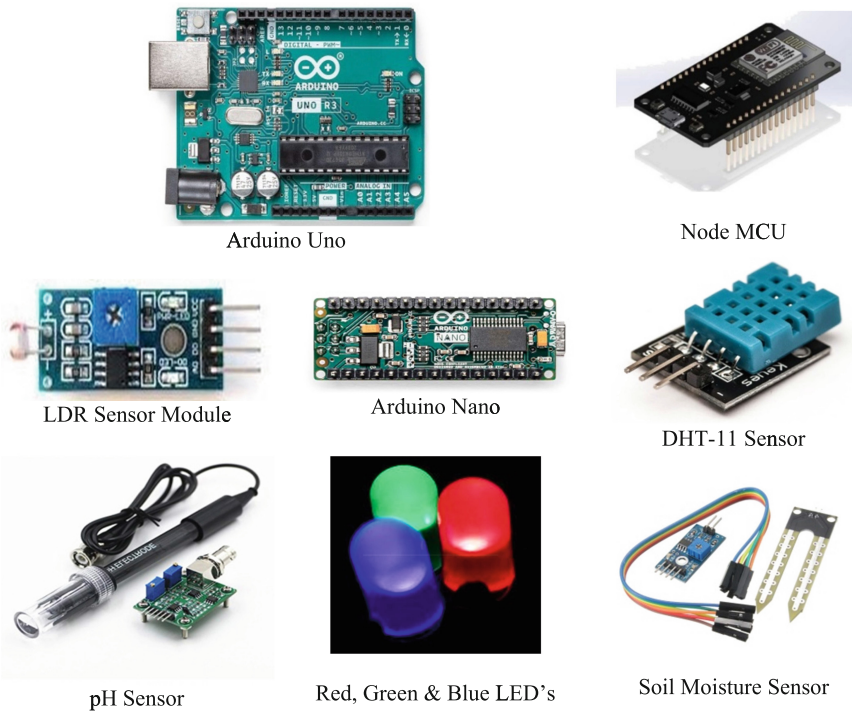


Fig. 3. Sensors Used

2 Results and Discussion

2.1 Representation of the Model

In Fig. 4, pictorial overview of working model is represented. All the respective sensors of the parameters are connected to the micro-controller Arduino UNO. Readings were taken simultaneously (Fig. 5).

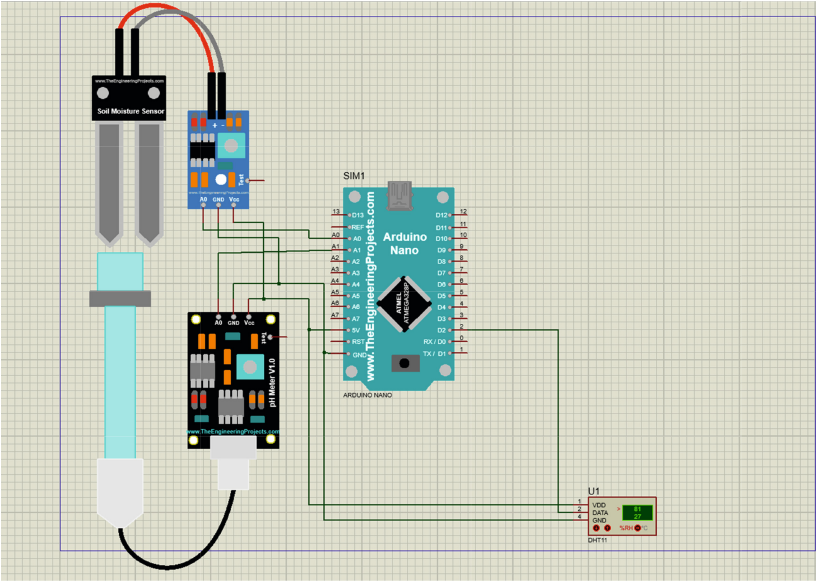


Fig. 4. Pictorial representation of the hardware connections (Arduino Nano Circuit)



Fig. 5. Rapid growing crops grown for the experiment which is irrigated by treated effluent

2.2 Tested Values of Effluent Water Quality in the Laboratory

(See Table 1)

Table 1. Summarized Results obtained by manual test in the Environmental Laboratory.

Sl. No.	Parameters	Units	Results of STP effluent Water	*Desirable-Maximum limit
1	pH	–	8.4	6.5 – 8.5
2	Turbidity	NTU	8	5–10
3	Total Dissolved Solids (TDS)	mg/L	334	1000
4	Biochemical oxygen demand (BOD)	mg/l	18	20–30
5	Dissolved oxygen (DO)	mg/l	4.5	4–5
6	Alkalinity	mg/l as CaCO ₃	48.5	30–100
7	Total hardness	mg/l	208	399–429.97
8	Calcium hardness	mg/l	257	300–400
9	Magnesium hardness	mg/l	11.95	50–100

* Irrigation effluent water standards as per Central Pollution Control Board

2.3 N, P and K Fertility Status of Dakshina Kannada and Udupi Districts in Karnataka as Determined by IISS, Bhopal

(See Tables 2 and 3).

Table 2. N, P and K fertility status of Dakshina Kannada and Udupi districts.

Nutrient	Status	Districts
Nitrogen	High	Dakshina Kannada and Udupi
Phosphorous	Low	Dakshina Kannada and Udupi
Potassium	Low	Dakshina Kannada and Udupi

2.4 NPK Consumption Ratio During the year 2007–08, 2008–09, 2009–10 and 2010–11 in Karnataka.

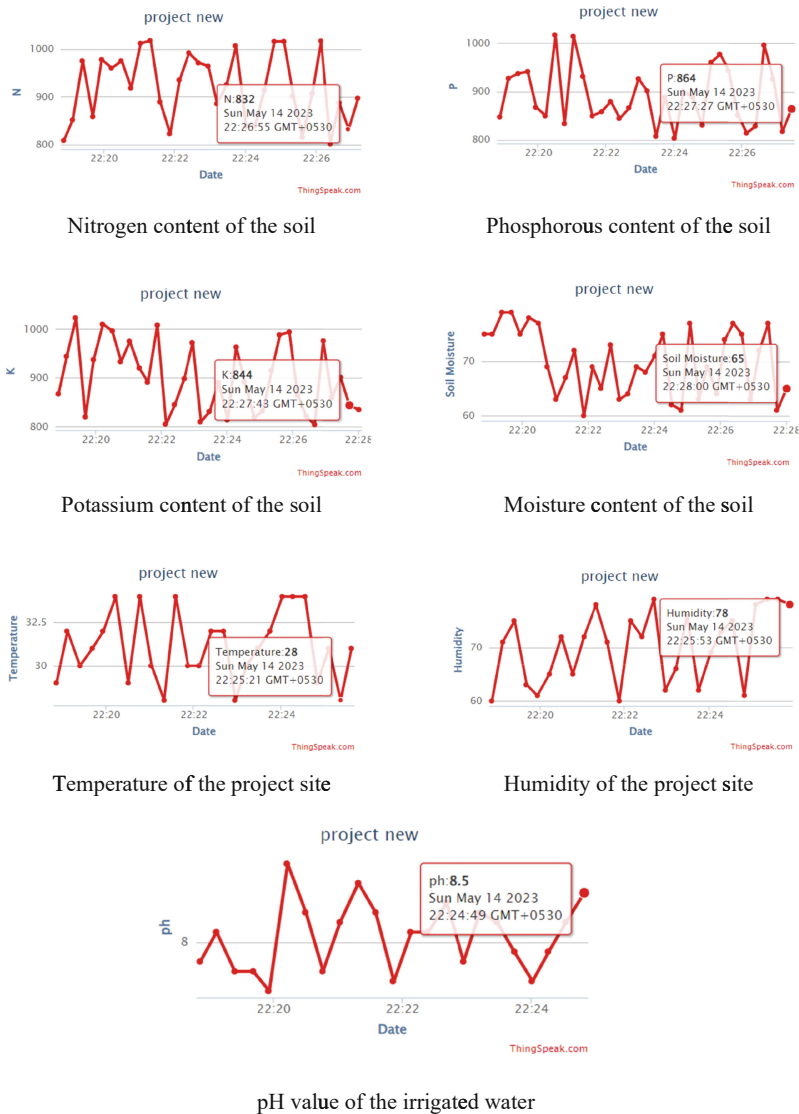
Table 3. N, P and K consumption ratio in Karnataka.

State	2007–08	2008–09	2009–10	2010–11
	N: P: K	N: P: K	N: P: K	N: P: K
Karnataka	2.4: 1.2: 1	2.1: 1.4: 1	2.1: 1.4: 1	2.6: 1.7: 1

2.5 Results Acquired from Sensors as IoT System

To visualize the data received from the sensor like DHT11, pH sensor, soil moisture sensor and the NPK sensors were sending the data by using ESP8266 as it is a wi-fi module and also acts a microcontroller (Table 4).

Table 4. Acquired digital data from sensors as IoT system.



3 Conclusion

The developed IoT integrated soil health monitoring system for the assessment of STP effluent has tremendous potential to address the issues of water quality and pollution in irrigation land, this will helps to STP working efficiency and monitoring. IoT technology makes it possible to monitor and manage effluent water use in real-time quality test, improving effluent water use and boosting crop output. In addition, monitoring sewage treatment plant effluent with IoT sensors offers a practical and sustainable way to reuse effluent water for irrigation.

Monitoring and detection of water quality parameter such as pH, turbidity, TDS, Temperature and NPK was done by sophisticated inbuilt NPK sensors which are embedded in the prototype is done and it will be accessed via node MCU through Thing-Speak and the acquired data is correlated with the data acquired from manual effluent water testing. The main focus of the research work ease of monitoring and detecting for the end user to differentiate the effluent water quality for the irrigation purpose as well as soil health monitoring. The prototype developed is working according to the desired IS standards and purpose and passed all the set of simulation and tests conducted as per IS standards.

Acknowledgement. We would like to acknowledge Mangalore Institute of Technology & Engineering, Moodabidri, Mangaluru, Karnataka, India, for the constant support and encouragement to carry out the research work.

References

1. Shams, A.M., Alim, M.A.: IoT-based smart irrigation system using soil moisture sensor. In: 2020 International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2–2020), pp. 1–4 (2020)
2. Hamilton, A.J., Stagnitti, F., Xiong, X., Kreidl, S.I., Benke, K.K., Maher, P.: Wastewater irrigation: the state of play (2020)
3. Narain-Ford, D.M., Bartholomeus, R.P., Dekker, S.C., Van Wezel, A.P.: Natural purification through soils: risks and opportunities of sewage effluent reuse in sub-surface irrigation (2020)
4. Monteiro-Silva, F., Jorge, P.A.S., Martins, R.C.: Optical sensing of nitrogen, phosphorus and potassium: a spectrophotometrical approach toward smart nutrient deployment (2019)
5. Sharma, H., Pal, R.: IoT-based smart irrigation system using machine learning algorithm. In: International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), pp. 1–7 (2020)
6. Patil, S.S. Shukre, V.A.: Comparative study of different methodologies used for measuring soil parameters: a review (2020)



One Nation One Health Card and Patient Health Monitoring System Using IOT and Cloud Technologies

K. M. Deepika^(✉), Akshata B. Anjutagi, M. N. Yashodha, K. Lavanya,
and Renuka Karadagi

Information Science and Engineering, Nitte Meenakshi Institute of Technology, Bangalore, India
deepika.km@nmit.ac.in

Abstract. The healthcare sector plays a pivotal role in a nation's economic landscape. Despite its significance, India faces limitations in healthcare communication innovation, hindering service quality enhancement. This paper identifies the absence of an electronic healthcare system in India as a major obstacle. In response, we propose an e-Health card system designed to boost the efficiency and accessibility of healthcare services. Authorized parties can securely access patient data, doctors' prescriptions, and medical histories through a dedicated web page. The introduction of digital records ensures improved tracking and standardized documentation of patient interactions, thereby minimizing the risk of errors. Notably, our project addresses the critical need for a comprehensive database accessible to all hospitals, particularly when transferring patients to facilities with advanced capabilities. By providing a centralized platform, our initiative aims to bridge existing gaps in the healthcare system and elevate patient care standards in India.

Keywords: Healthcare system · E-health card · Doctor's prescription · Patient's data

1 Introduction

The healthcare sector, a cornerstone of a nation's well-being, stands as a critical contributor to both public health and a country's economic vitality. In the context of India, however, the full potential of healthcare services remains untapped due to a limited implementation of communication innovation. The absence of a robust electronic healthcare system has posed challenges, hindering the enhancement of service quality and impeding the seamless delivery of medical care.

Globally, many countries have successfully embraced e-health card systems, leveraging the power of communication technology to elevate healthcare services. In contrast, India currently grapples with the absence of an electronic healthcare infrastructure. This lacuna has prompted the need for innovative solutions that can enhance the efficiency and accessibility of healthcare services, ensuring that the burgeoning population receives timely and quality medical care.

Addressing this imperative, our proposed e-Health card system represents a strategic response to the existing gaps in India's healthcare landscape. Grounded in the principles of communication technology, our system seeks to empower authorized parties, including healthcare providers and stakeholders, with seamless access to vital information. This includes patients' data, doctors' prescriptions, and comprehensive medical histories—all accessible through an intuitive web-based platform.

One of the key advantages of our proposed system lies in its ability to introduce digital records, a transformative approach that facilitates meticulous tracking and standardized documentation of patient interactions. By doing so, the system aims to mitigate the likelihood of errors, ensuring that healthcare practitioners are equipped with accurate and up-to-date information for informed decision-making.

Furthermore, our project recognizes the critical need for a comprehensive database accessible to all hospitals. In situations where patients require transfer to facilities with advanced capabilities, the availability of a centralized platform becomes imperative. Through this initiative, we aspire to establish a robust foundation that promotes data interoperability and accessibility across the healthcare spectrum, ultimately elevating the standards of patient care across the nation.

2 Literature Review

The healthcare paradigm centers around the continuous measurement and monitoring of various biological parameters within a patient's body, encompassing vital signs such as blood pressure, oxygen levels, and temperature. This intricate process is facilitated through the integration of a web server and an Android application. By leveraging the Android app, medical professionals can seamlessly monitor a patient's condition in real-time via their smartphones. Moreover, the patient's complete medical history is securely stored on the web server, enabling remote access for physicians and concerned parties, thereby eliminating the necessity for physical presence [1].

An exemplification of this technology is in medical services for assessing patient well-being. The Internet of Things (IoT) enhances the efficiency of clinical devices by enabling continuous monitoring of patient health. Through sensors that capture patient data, the potential for human error is minimized [2].

This paper proposes a health monitoring system utilizing non-invasive biomedical sensors specifically designed for temperature and heartbeat measurement. The acquired results are showcased on any digital monitoring system using an Arduino Mega. Subsequently, the sensor-generated data is transmitted to the ThingSpeak cloud, facilitating the storage and accessibility of patient information by healthcare professionals or concerned individuals for instantaneous follow-ups [3].

The inception of Radio Frequency Identification (RFID) technology traces back to 1948, with its early application being the Identification, Friend or Foe (IFF) system for aircraft. In healthcare, RFID plays a pivotal role in automated object identification. Two noteworthy applications include Cloud-Based Smart Card Healthcare Monitoring and Global Positioning System, utilizing RFID integrated with smart cards, and Global Med Info Smart Assistant with Cloud Storage, combining RFID technology with cloud storage for secure information management [4].

By integrating GSM and IoT technology, effective patient condition monitoring becomes feasible. This innovative approach incorporates smart sensors, including temperature and heartbeat sensors, to gauge the patient's internal temperature and pulse. The system operates on an ARDUINO-UNO board as a microcontroller, implementing cloud computing principles [5].

For patients requiring continuous monitoring in hospital settings, IoT-driven systems offer a solution. Employing sensors to collect comprehensive physiological data, this setup utilizes gateways and cloud infrastructure for data analysis and storage. The processed information is then remotely transmitted for further scrutiny [6].

RFID technology's integration into the healthcare sector is propelled by its enhanced functionality, affordability, reliability, and user-friendly features. The primary objective of this study, as delineated in this paper, is the development of an RFID-based system architecture and data model. This design aims to provide efficient methods for essential information management, particularly in emergency care scenarios, transcending hospital boundaries and national borders [7].

3 Design and Architecture

In times of emergencies, the utilization of pervasive computing technologies, including wireless communication, sensors, and automated data capture, becomes pivotal for facilitating information exchange. The system comprises three main components: a sensor module dedicated to collecting and analyzing data from the human body, a controller module for processing the gathered data, and a cloud-based storage system. The collected data is transmitted to the cloud through Wi-Fi, serving medical support purposes (Figs. 1 and 2).

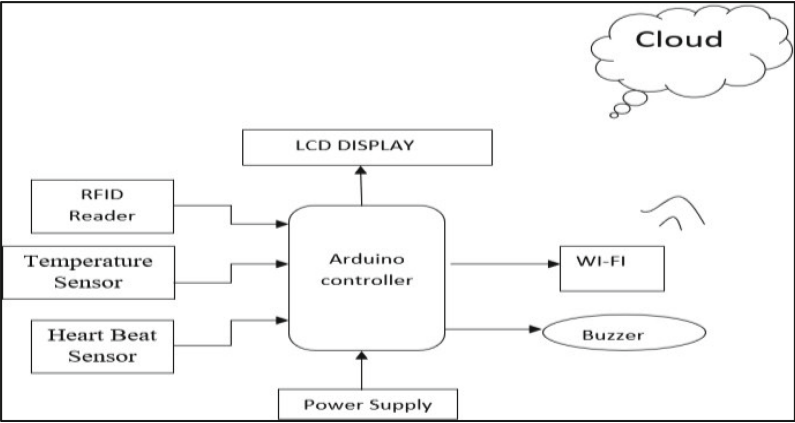


Fig. 1. Architecture Diagram.

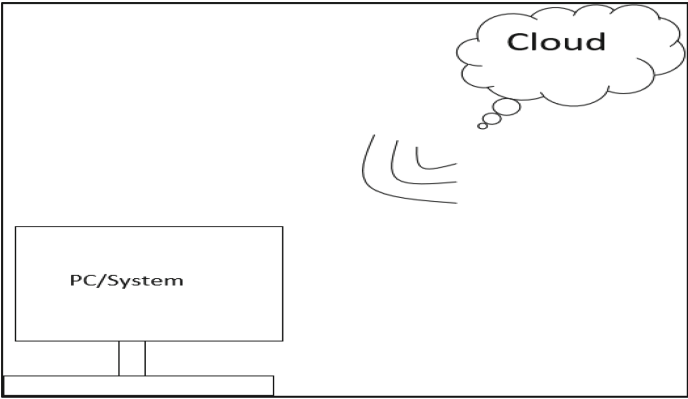


Fig. 2. Display Unit

3.1 Temperature Sensor

The MAX30205 Human Body Temperature Sensor from Maxim Integrated provides accurate temperature measurements and includes an output for over-temperature alarm, interrupt, and shutdown. It utilizes a high-resolution sigma-delta analog-to-digital converter (ADC) to convert temperature readings into a digital format. Thanks to its high accuracy and low-voltage operation, this sensor aids designers in fulfilling requirements related to errors and power consumption. Notably, the sensor’s precision of 0.1°C renders an ADC unnecessary for interpreting data, as it reliably delivers accurate human body temperature readings (Figs. 3 and 4).

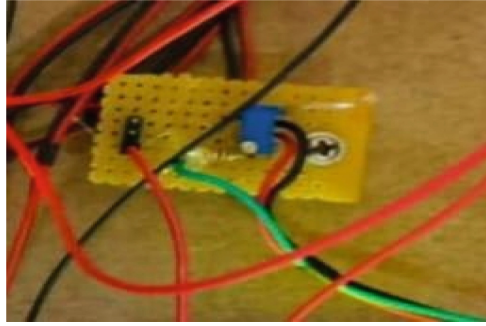


Fig. 3. Temperature Sensor



Fig. 4. Heartbeat Sensor

3.2 Heartbeat Sensor

The heartbeat sensor is employed to measure the digital output of heart beats per minute. It includes two LEDs emitting red and infrared light. Red light is used to gauge the pulse rate, while infrared light is essential as the heart contracts and relaxes, determining the heartbeat rate by the increase or decrease in oxygenated blood.

3.3 Radio Frequency Identification (RFID)

Radio Frequency Identification (RFID) is a wireless technology that utilizes radio waves for data communication. It has garnered significant interest in the market owing to its versatile applications, providing numerous advantages to the healthcare sector. Notably, RFID enhances safety and operational efficiency, offering key benefits such as the ability to function without line-of-sight and providing read/write capabilities.

4 Implementation

The information accessed is securely stored in the cloud through personal computers. When the patient swipes their RFID card during a doctor's visit, the complete medical history is systematically presented to the doctor. This not only eliminates confusion but also optimizes the doctor's time, sparing them from verbally seeking the patient's history.

Every diagnosis, scan, and procedure conducted is meticulously recorded, enabling the doctor to prescribe medications based on the patient's comprehensive medical history. There is no longer a need for individuals to memorize medication names or maintain hard copies of prescriptions provided by doctors, as all data is securely stored in the cloud and can be accessed globally at any given time.

The key features of this system include:

- Cloud-based maintenance and storage of patients' health records.
- Provision of patients' past health details to doctors and hospitals for streamlined diagnosis.
- Monitoring of health parameters, such as body temperature and heartbeat, utilizing sensors.

In case the patient's body temperature rises to 40 degrees, the temperature sensor triggers alert, notifying designated physicians. Similarly, the heartbeat sensors activate an alert when the patient's pulse exceeds 140, sending individualized alarm messages.

4.1 Hardware Implementation

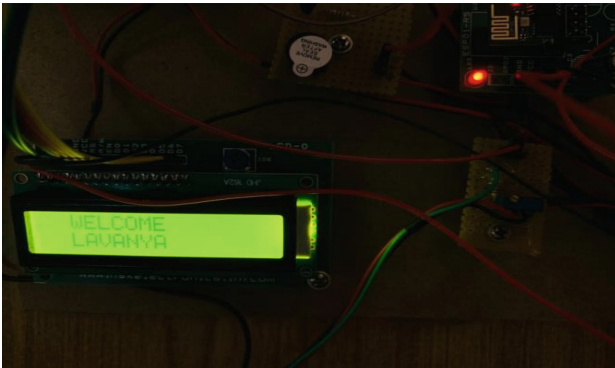


Fig. 5. LED Display

Once the RFID card is being swiped then it fetches all the data and displays on it (Figs. 5 and 6).

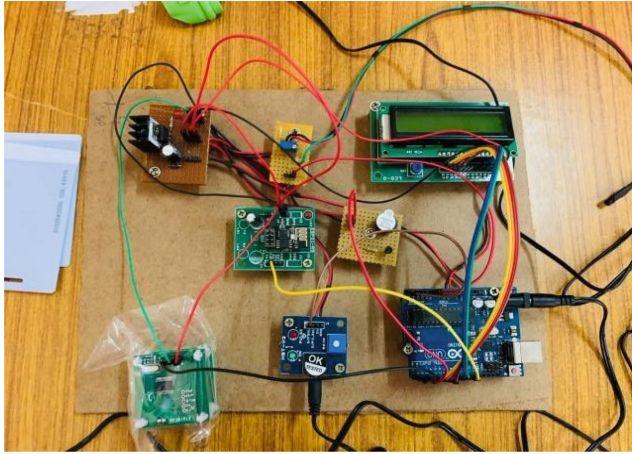


Fig. 6. Connections

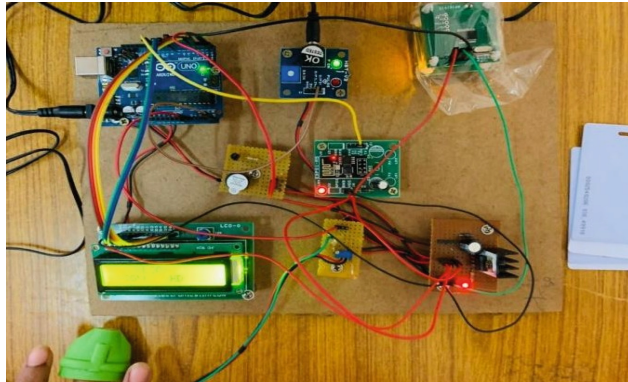


Fig. 7. Temperature and heartbeat sensors

In this setup, temperature and heartbeat sensors are integrated into the patient's bed. If a person's temperature surpasses 40 degrees, the buzzer alerts, and similarly, it activates when the heartbeat reaches 140 (Fig. 7).

5 Results

Upon swiping the RFID card, it retrieves all the information from the cloud and displays it. When the RFID card is swiped, patient data such as patient ID, name, age, disease, and RFID number will be fetched (Figs. 8, 9, 10, 11, 12 and 13).

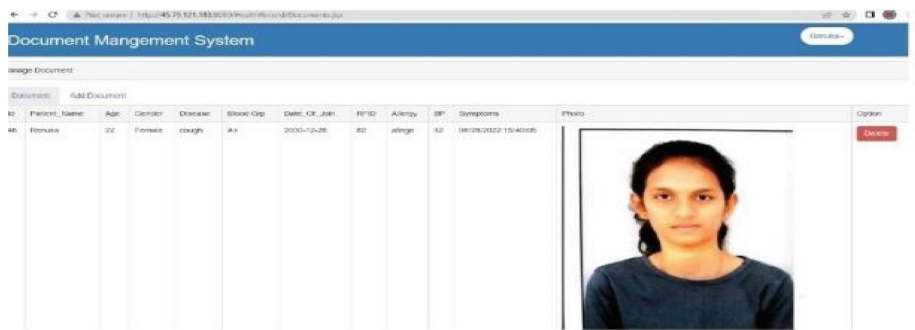


Fig. 8. Patient 1

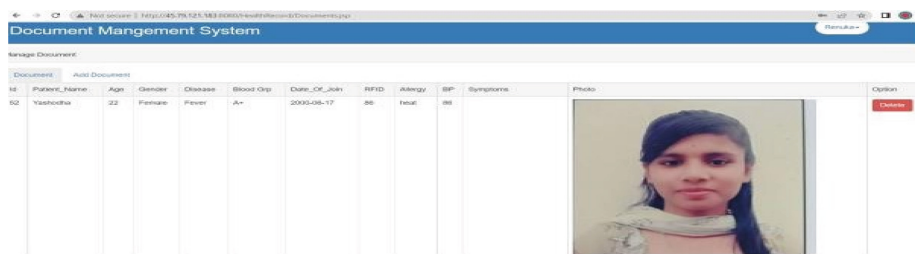


Fig. 9. Patient 2



Fig. 10. Patient 3

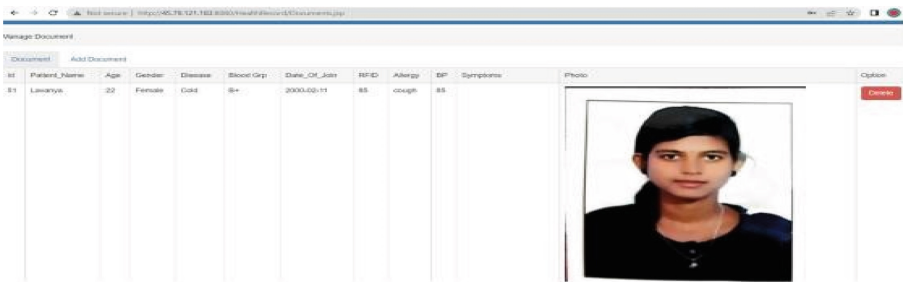


Fig. 11. Patient 4

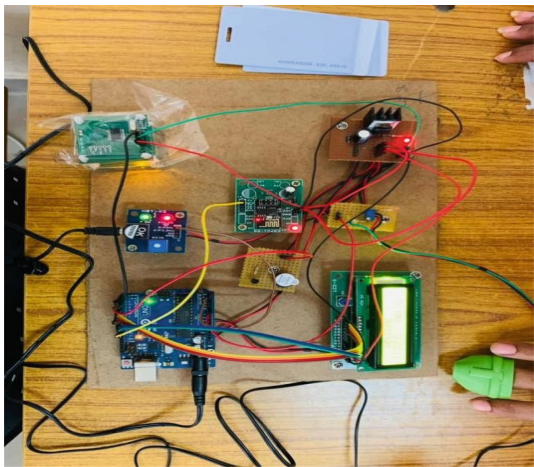


Fig. 12. This is displaying temperature and heartbeat rate

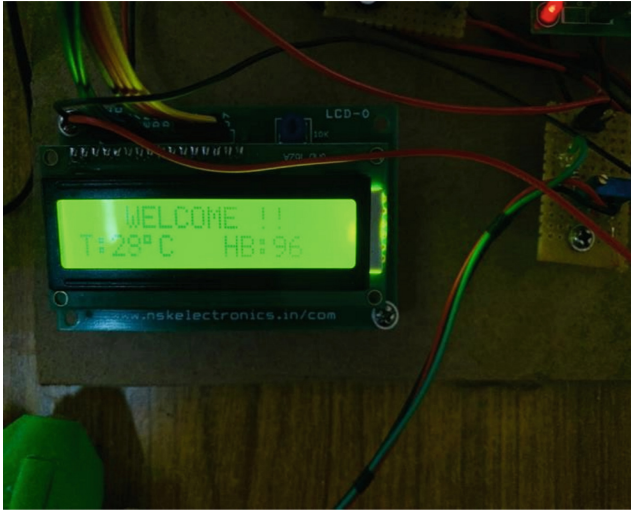


Fig. 13. This includes Temperature sensor, heartbeat sensor, Lcd display, buzzer, Wi-Fi

6 Conclusion

The objective of this initiative is to establish a system enabling government organizations and authorized private entities to access patient healthcare data. This initiative acts as a deterrent against corruption in diverse healthcare sectors. Although the implementation of this project demands substantial effort, its advantages are extensive and have enduring impacts on national healthcare systems. Electronic medical records, digital medical imaging, pharmacy records, and doctors' notes are seamlessly integrated and accessible in real-time. Doctors can analyze the well-organized cloud data, leading to improved treatment decisions, timely insurance programs, and the potential for personalized healthcare. To enhance security, there is a possibility of linking Aadhar card or passport information. Currently, our application is constrained to a specific number of patients, but there is a pressing need to expand its scope to encompass thousands of hospitals and hundreds of thousands of patients.

Acknowledgement. We express our gratitude to Professor Dr. Mohan S G and our supervisors for their patient guidance, enthusiastic support, and valuable feedback on this research project. We are also thankful to Mrs. Deepika K M for her guidance and assistance in keeping our development progress on course. Additionally, our heartfelt thanks go to the technicians at the department's laboratory for their support. We appreciate the technicians at the department's laboratory for providing the necessary resources required to run the application.

References

1. Kumar, R., Pallikonda Rajasekaran, M.: An IoT based patient monitoring system using raspberry Pi. In: 2016 International Conference on Computing Technologies and Intelligent Data Engineering (ICCTIDE 2016), pp. 1–4. IEEE (2016)

2. Kumar, M.A., Ravi Sekhar, Y.: Android based health care monitoring system. In: 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), pp. 1–5. IEEE (2015)
3. Patil, V., Thakur, S.S., Kshirsagar, V.: Health monitoring system using internet of things. In: 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 1523–1525. IEEE (2018)
4. Valsalan, P., Baomar, T.A.B., Baabood, A.H.O.: IoT based health monitoring system. *J. Critic. Rev.* 7(4), 739–743 (2020)
5. Shih, S.-T., Hsu, C.-M., Chao, C.-Y.: Self-healthcare management system based on RFID technology. In: The 7th International Conference on Networked Computing and Advanced Information Management, pp. 252–255. IEEE (2011)
6. Veena, T., Shakeel, F.: Monitoring health care system using internet of things-an immaculate pairing. In: 2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS), pp. 153–158. IEEE (2017)
7. Turcu, C.E., Turcu, C., Popa, V.: An RFID-based system for emergency health care services. In: 2009 International Conference on Advanced Information Networking and Applications Workshops, pp. 624–629. IEEE (2009)



Amalgamation of Semantic Web with IoT: Semantic Web of Things (SWoT)

Gaurav Jaglan¹(✉), Aman Jolly², Indrasen Singh³, Vikas Pandey⁴, and Shashikant⁴

¹ Semantics and AI Department, Knowledge Learning Infotech (Opc) Pvt. Ltd., Delhi, India
gaurav@klinfotech.in

² CSE Department KIET Group of Institutions, Ghaziabad, UP, India
aman.jolly@kiet.edu

³ Department of Embedded Technology, School of Electronics Engineering, Vellore Institute of Technology, Vellore, TN 632014, India
indrasen.singh@vit.ac.in

⁴ Electrical Engineering Department, Babu Banarasi Das University, Lucknow, UP, India

Abstract. The Semantic Web of Things (SWoT) paradigm boosts IoT system interoperability and intelligence. In this survey, we analyze SWoT, including important ideas, Semantic Web technologies in IoT, ontologies, alignment problems, benefits of integrating Semantic Web technologies, applications, use cases, and obstacles. The study shows how SWoT has been effectively deployed through use cases, bringing researchers and practitioners useful lessons and experiences. The study also covers SWoT difficulties including scalability, security, privacy, knowledge representation, and semantic reasoning, emphasizing the need for future research and development. Finally, SWoT difficulties include scalability, security, privacy, knowledge representation, and semantic reasoning. The report highlights the need for more research and development to overcome these difficulties.

Keywords: Semantic web of things · Semantic web · IoT · Ontology

1 Introduction

The Semantic Web of Things aims to address the cross-domain interoperability problems in various application scenarios, such as Industry 4.0. It provides an explicit, easy, and comprehensible description to express semantic objects and their data, which can be used to enable intelligent and autonomous interactions between devices, services, and users [1]. An emerging concept in the field of information and communication technology (ICT), the Semantic Web of Things (SWoT) brings together many of the most significant technologies from both the Web of Things (WoT) and the Semantic Web [1, 2]. The Semantic Web is an extension of the World Wide Web that gives a formal definition of concepts, terminology, and connections within a particular knowledge area. It makes it possible to gather linked data, structure that data, and retrieve it by using technologies that are specified as W3C standards. These technologies include Resource Description Framework (RDF), RDF Schema (RDFS), and JavaScript Object Notation for Linked Data (JSON-LD) [2].

The goal of the Semantic Web of Things is to solve the cross-domain interoperability issues that arise in a wide variety of application scenarios, including Industry 4.0. It offers a description that is plain, simple, and easy to comprehend in order to represent semantic objects and their data. This description may be utilized to allow intelligent and autonomous interactions between devices, services, and users [1]. To support the Semantic Web of Things, various tools and search engines have been proposed, such as the WOT Semantic Search Engine (WOTS2E) [3] and Thing Ontologies [4]. These tools and ontologies can help discover Linked Data and provide a JSON template representation of Thing properties, respectively. The Semantic Web of Things is an emerging vision that combines the technologies of the Web of Things and the Semantic Web to enable intelligent and autonomous interactions between devices, services, and users. It provides an explicit, easy, and comprehensible description to express semantic objects and their data, and various tools and ontologies have been proposed to support it. The Semantic Web of Things (SWoT) is an extension of the Internet of Things (IoT) that enhances communication by behaving intelligently [5]. It comprises a seven-layered architecture, with the perception layer being an important layer for collecting data from devices and communicating with its associated layer [5]. Figure 1 shows the components of Building the semantic web of things. Here are some of the key concepts related to SWoT:

- **Ontology construction:** Ontology construction is the core of SWoT, and it involves creating a formal representation of the knowledge in a particular domain [6].
- **Semantic annotation:** Semantic annotation is the process of adding metadata to data that describes its meaning and context [6].
- **Thing Description (TD):** TD is a JSON template representation of Thing properties, such as purpose, data types, and operations [4].
- **OpenAPI:** OpenAPI is a specification for documenting RESTful services so that a user or another service can comprehend their purpose and reuse them in applications [4].
- **Ontology mapping:** Thing descriptions can be semantically extended using ontologies or can be mapped to an ontology to allow a machine to better understand the inherent meaning of Thing descriptions and interact with them [4].
- **Discovery:** One of the challenges in SWoT revolves around the discovery procedures to find IoT devices within IoT infrastructures and whether the discovery performed is semantic or syntactic. Semantic discovery uses Semantic Web query languages (e.g., SPARQL) and reasoners (e.g., Pellet) to discover Things and check them for consistency or infer new properties [4, 7].
- **Compressive Sensing:** Compressive Sensing is a method for recovering and estimating sensory data from a low-rank structure to address data loss at the perception layer [5].
- **AI techniques:** AI techniques combined with recent advancements in IoT, Web of Things, and Semantic Web, referred to as the Semantic Web, promise to play an important role in Industry 4.0 [5]. The Semantic Web of Things for Industry 4.0 (SWeTI) platform uses these technologies to address Industry 4.0's challenges, facilitate cross-sector and cross-domain integration of systems, and develop intelligent and smart services for smart manufacturing [8].

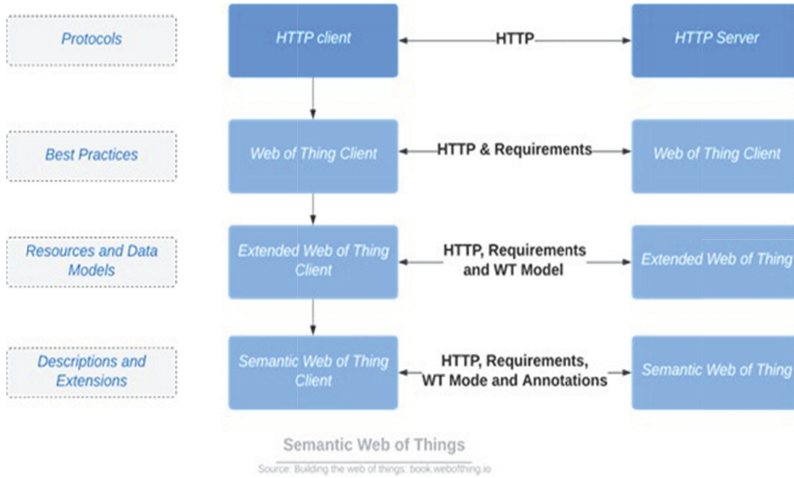


Fig. 1. Building the semantic web of things

2 Semantic Web Technologies Used in IoT Systems

Semantic Web technologies can be used in IoT systems to provide an explicit, easy, and comprehensible description to express semantic objects and their data. Figure 2 shows the conceptual Venn diagram of the evolution towards the Semantic Web of Things and global interoperability. Here are some specific Semantic Web technologies used in IoT systems.:

- 1) Resource Description Framework (RDF): RDF is a standard model for data interchange on the Web. It is used to represent information about resources in a graph-based format, where nodes represent resources and edges represent relationships between resources [1].
- 2) RDF Schema (RDFS): RDFS is a vocabulary for describing RDF resources. It provides a set of classes and properties that can be used to define the structure of RDF data [1].
- 3) JavaScript Object Notation for Linked Data (JSON-LD): JSON-LD is a JSON-based format to serialize Linked Data. It is designed to easily integrate into deployed systems that already use JSON and provides a smooth upgrade path from JSON to JSON-LD. It is primarily intended to be a way to use Linked Data in Web-based programming environments, to build interoperable Web services, and to store Linked Data in JSON-based storage engines [2].
- 4) Ontologies: Ontologies are formal descriptions of concepts and relationships within a given knowledge domain. They provide a common vocabulary for describing the structure of data and can be used to enable interoperability between different systems and applications [3].

Semantic Web technologies such as RDF, RDFS, JSON-LD, and ontologies can be used in IoT systems to provide a common framework for describing and sharing data.

These technologies enable interoperability between different systems and applications, making it easier to integrate data from various sources and formats.

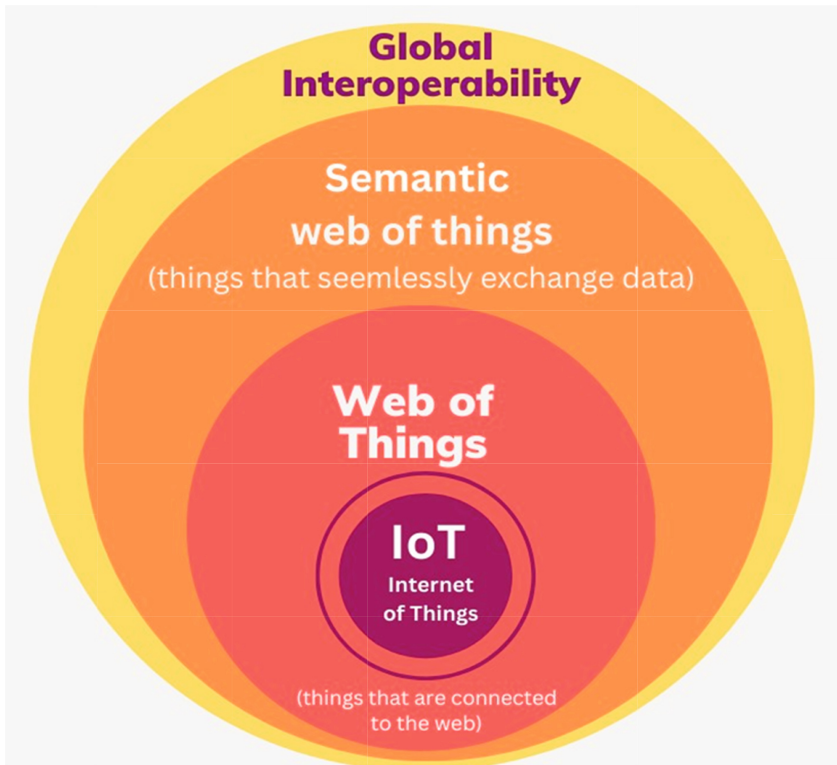


Fig. 2. The evolution towards Semantic Web of Things and global interoperability

3 Ontologies in SWoT

The Semantic Web of Things (SWoT) relies on ontologies to describe data structures and enable interoperability. SWoT ontologies compared. Table 1 shows significant features and priority areas for each ontology. The comparison shows how each ontology covers sensor data, smart appliances, IoT devices, and Web-based data description. These ontologies support interoperability and the Semantic Web of Things environment, notwithstanding their discrepancies.

3.1 Challenges in Aligning Ontologies in SWoT

Aligning ontologies is a crucial step in achieving semantic interoperability in the Semantic Web of Things (SWoT). However, there are several challenges that need to be addressed in order to align ontologies effectively. Here are some common challenges in aligning ontologies in SWoT:

- **Heterogeneity:** IoT systems generate data in various formats and structures, making it difficult to align ontologies that describe the same concepts [9].
- **Scalability:** As the number of IoT devices and applications grows, it becomes increasingly challenging to align ontologies at scale [10].
- **Complexity:** Ontologies can be complex and difficult to understand, especially for non-experts. This can make it challenging to align ontologies effectively [11].
- **Evolution:** Ontologies are not static and can evolve over time, making it challenging to maintain alignment between ontologies [9].
- **Lack of standards:** There is a lack of standardization in the development and use of ontologies, which can make it challenging to align ontologies across different domains [5].
- **Semantic heterogeneity:** Even when ontologies are aligned, there can still be semantic heterogeneity due to differences in the interpretation of concepts and relationships [6].

Addressing these challenges requires a combination of technical solutions and community efforts to establish best practices and standards for ontology development and alignment.

Table 1. Ontologies in SWoT.

Sr. No.	Ontology	Description	Key Features
1	SSN (Semantic Sensor Network) ontology	Describes sensors and sensor networks, including properties, capabilities, and observations	• Focuses on sensor data • Enables description of sensor properties and observations • Widely used in IoT applications
2	SAREF (Smart Appliances REference) ontology	Provides a common vocabulary for describing smart appliances, facilitating interoperability between different appliances and systems	• Specializes in smart appliance description • Enables interoperability between different appliances
3	WoT (Web of Things) ontology	Describes IoT devices and services, including properties, capabilities, and interactions, enabling semantic interoperability among IoT systems	• Focuses on IoT device and service description • Facilitates semantic interoperability among IoT systems

(continued)

Table 1. (continued)

Sr. No.	Ontology	Description	Key Features
4	SSN-XG (Semantic Sensor Network Incubator Group) ontology	Extends the SSN ontology to include additional concepts and relationships like time, space, and events, enabling advanced reasoning of sensor data	• Builds upon SSN ontology • Includes concepts of time, space, and events
5	W3C ontologies	Developed by the W3C as part of the Semantic Web technology stack, including RDF, RDFS, and OWL, providing a common framework for data on the Web	• General-purpose ontologies for describing and sharing data on the Web of Linked data • RDF provides a data model for describing resources • RDFS extends RDF with basic vocabulary for ontologies • OWL is a more expressive language for ontologies, enabling advanced reasoning and inference capabilities

4 Advantages of Integrating Semantic Web Technologies in IOT Systems

Semantic Web technologies can provide several benefits to IoT systems, including improved data integration and understanding, interoperability among resources and data providers, real-time decision-making and predictive maintenance, resource discovery and classification, and seamless connection between physical devices and information systems. Here are some specific benefits of using Semantic Web technologies in IoT systems:

- 1) Better integration and knowledge of data: Semantic Web technologies can give raw IoT data meaning and turn it into a data file with more information. This can help solve the problem of IoT systems not having interoperability, which is caused by how hard it is to share and reuse raw IoT data. [12]. By providing a common vocabulary for describing the structure of data, ontologies can enable interoperability between different systems and applications [13].
- 2) Interoperability among resources and data providers: Semantic Web frameworks can allow semantic interoperability between IoT systems. This means that data can be shared and used across services, business, and community boundaries [3]. Semantic technologies can be used to bring together the description and data of IoT devices and make it easier to connect with them, and explore their data [13].

- 3) Real-time decisions and predictive maintenance: Semantic Web tools can make it possible for IoT systems to make real-time decisions and perform predictive maintenance. By combining data from sensors, machines, and business systems, these frameworks can make discovery, automation, integration, and reuse across different apps more effective. [1].
- 4) Resource discovery and classification: Semantic Web technologies can be used in resource discovery and classification to provide better, domain-specific search engine capabilities [13]. By giving a formal description of concepts, terms, and relationships within a given area of knowledge, the Semantic Web makes it possible to collect, organize, and find linked data using W3C-standard technologies like Resource Description Framework (RDF), RDF Schema (RDFS), and JavaScript Object Notation for Linked Data (JSON-LD) [12].
- 5) Semantic Web technologies can provide a standard framework for defining and exchanging data, which enables a seamless link between physical devices and information systems. This is made possible by the fact that a seamless connection between physical devices and information systems can be established. This can be helpful in addressing the issue of lack of interoperability in IoT systems, which is created by the difficulty of sharing and reusing raw IoT data. This difficulty is driven by the fact that IoT data is typically stored in proprietary formats [12].

5 Applications and Use Cases of the Semantic Web of Things

A) Applications of the Semantic Web of Things

The Semantic Web of Things (SWoT) is an emerging vision that combines the technologies of the Web of Things and the Semantic Web to enable intelligent and autonomous interactions between devices, services, and users [14]. Here are some applications of the Semantic Web of Things:

- 1) Industry 4.0: SWoT applications focus on providing wide-scale interoperability that allows the sharing of IoT devices across domains [2]. It aims to address the cross-domain interoperability problems in various application scenarios, such as Industry 4.0 [15]. It provides an explicit, easy, and comprehensible description to express semantic objects and their data, which can be used to enable intelligent and autonomous interactions between devices, services, and users [2].
- 2) Smart Manufacturing: The Semantic Web of Things for Industry 4.0 (SWeTI) platform has been proposed to enable smart manufacturing [8]. Through realistic use case scenarios, the platform demonstrates how the Semantic Web of Things can be used to integrate data from various sources, including sensors, machines, and enterprise systems, to enable real-time decision-making and predictive maintenance.
- 3) Resource discovery and classification: Semantic Web technologies can be used in resource discovery and classification to provide better, domain-specific search engine capabilities [15]. By providing a formal description of concepts, terms, and relationships within a given knowledge domain, the Semantic Web enables the collection, structuring, and recovery of linked data through technologies that are specified as W3C standards, such as Resource Description Framework (RDF), RDF Schema (RDFS), and JavaScript Object Notation for Linked Data (JSON-LD) [2].

- 4) Data integration: Semantic Web technologies can be used in data integration, whereby data in various locations and various formats can be integrated in one, seamless application [15]. By providing a common framework that allows data to be shared and reused across application, enterprise, and community boundaries, the Semantic Web enables more effective discovery, automation, integration, and reuse across various applications [16].

The semantic Web of Things has various applications, including Industry 4.0, smart manufacturing, resource discovery and classification, and data integration. It provides an explicit, easy, and comprehensible description to express semantic objects and their data, which can be used to enable intelligent and autonomous interactions between devices, services, and users.

B) Use cases of SWoT

In Table 2, each use case is described, highlighting the specific application area and the benefits of incorporating Semantic Web of Things (SWoT) technologies. These use cases showcase the potential of SWoT in various domains, including smart cities, health-care, agriculture, fleet monitoring, energy management, manufacturing, and supply chain management.

Table 2. Use Cases of SWoT

Use Case	Description	Benefits
Smart Cities	Integration of SWoT in urban planning, traffic management, and environmental monitoring to provide meaningful insights	• Improved urban planning • Enhanced traffic management • Better environmental monitoring
Healthcare	Utilizing SWoT to enhance patient care through integration of data from medical devices and health records for personalized services	• Personalized healthcare • Improved patient care • Integrated health data management
Agriculture	Application of SWoT to optimize farming practices by integrating data from sensors, weather stations, and other sources	• Optimized farming practices • Improved crop health management • Efficient irrigation management
Fleet Monitoring	Utilizing SWoT technologies for dynamic dash- boarding applications to monitor vehicle fleets	• Real-time fleet monitoring • Enhanced visualization and analytics of fleet data • Web-based access to fleet information

(continued)

Table 2. (continued)

Use Case	Description	Benefits
Energy Management	Integration of SWoT in smart homes and buildings to optimize energy consumption and enable intelligent control systems	• Energy efficiency in homes and buildings • Intelligent control of heating, cooling, and lighting systems
Manufacturing	Application of SWoT to improve production processes by integrating data from sensors, machines, and supply chain systems	• Real-time insights for optimizing operations • Enhanced production efficiency • Improved supply chain management
Supply Chain Management	Utilizing SWoT for enhanced visibility and traceability in supply chains by integrating data from various sources	• Improved supply chain visibility • Real-time information on goods location and status • Enhanced traceability and logistics management

6 Challenges in Semantic Web of Things (SWoT)

The Semantic Web of Things” (SWoT) is a current investigation field intending to merge Semantic Web-based technologies with the Internet of Things (IoT) [17]. Even if technologies based on the Semantic Web have the potential to offer a number of benefits to IoT systems, there are also a number of obstacles that need to be solved. The following are some of the difficulties associated with SWoT:

- 1) **Standardization:** One of the major challenges in SWoT is standardization. There is a need for standardization of data models, ontologies, and protocols to enable interoperability among different IoT devices and systems [1]. Without standardization, it can be difficult to ensure that different devices and systems can communicate with each other effectively.
- 2) **Interoperability:** Interoperability allows IoT devices and systems to share data. Semantic Web tools can help solve this problem by characterizing and sharing data. IoT devices and systems are diverse, making interoperability difficult [18].
- 3) **Discovery:** Discovery means being able to find IoT products and services and use them. Semantic Web technologies can help solve this problem by giving people a standard language to use when talking about IoT devices and services. Because there are so many IoT products and services available, discovery can still be difficult [19].
- 4) **Security:** When doing SWoT, security is a primary consideration. Internet of Things devices are frequently susceptible to cyber assaults, and technologies related to the semantic web can create new security problems. For example, the usage of ontologies can make it simpler for malicious actors to take advantage of vulnerabilities in Internet of Things (IoT) systems [17, 19]. Therefore, it is important to ensure that Semantic Web technologies are used in a secure manner.

- 5) Scalability: Another problem that must be overcome with SWoT is scalability. It may become challenging to ensure that Semantic Web technologies can scale to fulfil the requirements of large-scale Internet of Things installations as the number of IoT devices and systems continues to increase. [20].

SWoT faces challenges in standardization, interoperability, discovery, security, and scalability. Using safe and scalable Semantic Web technologies and defining standards for data models, ontologies, and protocols to allow IoT device and system interoperability can solve these problems.

7 Conclusion

This study covers SWoT ideas, Semantic Web technologies utilized in IoT systems, ontologies, problems, benefits, applications, use cases, and future directions. This survey helps researchers and practitioners understand the Semantic Web of Things and its potential to advance IoT systems. SWoT uses RDF, RDFS, and OWL to standardize and communicate data in IoT systems. Ontologies give SWoT a standard language to define IoT devices, services, and data. SSN, SAREF, WoT, SSN-XG, and W3C ontologies are evaluated for characteristics and application. The diversity of IoT systems makes matching ontologies in SWoT difficult. It examines solutions and continuing research to solve these problems, enabling seamless semantic integration and querying across domains. Furthermore, the paper discusses real-world use cases of SWoT, offering concrete examples and insights into how SWoT has been applied in practical scenarios, providing valuable lessons and experiences for researchers and practitioners. The advantages of integrating Semantic Web technologies in IoT systems are highlighted, including improved interoperability, intelligent data processing, enhanced data integration and discovery, and semantic-driven decision-making. Real-world applications across various domains demonstrate the potential of SWoT in transforming smart cities, healthcare, agriculture, fleet monitoring, energy management, manufacturing, and supply chain management.

References

1. Zhang, J.: Literature review. In: Tono-types and Tone Evolution. FCL, vol. 11, pp. 15–34. Springer, Singapore (2020). https://doi.org/10.1007/978-981-33-4870-7_2
2. Thuluva, A.S., Anicic, D., Rudolph, S.: Semantic web of things for industry 4.0. in RuleML+RR (Supplement) (2017)
3. Kamilaris, A., Yumusak, S., Ali, M.I.: Wots2e: a search engine for a semantic web of things. In: 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), pp. 436–441. IEEE (2016)
4. Tzavaras, A., Mainas, N., Petrakis, E.G.: Thing ontologies for the semantic web of things. In: 2022 13th International Conference on Information, Intelligence, Systems & Applications (IISA), pp. 1–8. IEEE (2022)
5. Afzaal, R., Shoaib, M.: Data recoverability and estimation for perception layer in semantic web of things. PLoS ONE **16**(2), e0245847 (2021)
6. Xu, H., Sun, H.: Application of rough concept lattice model in construction of ontology and semantic annotation in semantic web of things. Sci. Program. **2022** (2022)
7. Cimmino, A., Garcí'a-Castro, R.: Wothive: enabling syntactic and semantic discovery in the web of things. Open J. Internet Things **8**(1), 54–65 (2022)

8. Patel, P., Ali, M.I., Sheth, A.: From raw data to smart manufacturing: Ai and semantic web of things for industry 4.0. *IEEE Intell. Syst.* **33**(4), 79–86 (2018)
9. Amara, F.Z., Hemam, M., Djeddar, M., Maimor, M.: Semantic web and internet of things: challenges, applications and perspectives. *J. ICT Standard.* 261–292 (2022)
10. Szilagyi, I., Wira, P.: Ontologies and semantic web for the internet of things-a survey. In: *IECON 2016–42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 6949–6954. IEEE (2016)
11. Bhatia, M., Kumar, A., Beniwal, R.: Swot analysis of ontology driven software engineering. *Indian J. Sci. Technol.* **9**, 38 (2016)
12. Chatzimichail, A., et al.: Semantic web and iot, Semantic IoT: Theory and Applications: Interoperability, Provenance and Beyond, pp. 3–33 (2021)
13. Kalatzis, N., et al.: Semantic interoperability for iot platforms in support of decision making: an experiment on early wildfire detection. *Sensors* **19**(3), 528 (2019)
14. Visual Paradigm. Swot analysis case studies. <https://online.visual-paradigm.com/knowledge/swot-analysis/swot-analysis-case-studies/>. Accessed 2023
15. WorldWideWebConsortium(W3C).RDFFAQ. <https://www.w3.org/RDF/FAQ>. Accessed 2023
16. SemanticWebInterestGroup. <https://www.w3.org/2002/07/swint>. Accessed 2023
17. Mishra, S., Jain, S., Rai, C., Gandhi, N.: Security challenges in semantic web of things. In: *Innovations in Bio-Inspired Computing and Applications: Proceedings of the 9th International Conference on Innovations in Bio-Inspired Computing and Applications (IBICA 2018) held in Kochi, India during December 17–19, 2018* 9, pp. 162–169. Springer (2019). 0.1007/978-3-030-16681-6_16
18. Silva, A.L.M., Pe´rez-Alca´zar, J.D.J., Kofuji, S.T.: Interoperability in semantic web of things: design issues and solutions. *Int. J. Commun. Syst.* **32**(6), e3911 (2019)
19. Ullah, I.: The role of semantic web technologies for IoT data in underpinning environmental science. Lancaster University (United Kingdom) (2018)
20. Hassan, B.A.: Towards semantic web: challenges and needs, arXiv preprint [arXiv:2105.02708](https://arxiv.org/abs/2105.02708) (2021)



Developing an Autonomous Fruit Picking and Sorting Robot for Vertical Farming

S. Santhosh, Pavitra kumari, J. Bindu Madhavi^(✉), and B. Mohan Kumar Naik^(✉)

Department of Mechatronics Engineering, Mangalore Institute of Technology and Engineering,
(Visvesvaraya Technological University, Belagavi), Moodbidri, Karnataka 574225, India
bindu@mite.ac.in, mohan72003@gmail.com

Abstract. The agriculture sector grapples with several challenges, such as the diminishing workforce on farms and the escalating expenses associated with manual fruit harvesting. Despite ongoing progress in automating agriculture for efficiency and large-scale production, manual labor remains prevalent in fruit harvesting. Addressing these issues necessitates a focus on labor-saving measures and the expansion of agricultural operations. To tackle these challenges, the development of an automated fruit harvesting and segregation robot emerges as a practical solution. This involves creating a robotic system that combines a low-cost vision camera and a robotic arm. The stereo-vision camera serves to identify the color, distance, and position of the fruit, while the robotic arm is employed for mechanically plucking and segregating the fruits based on ripeness.

Keywords: Fruit harvesting · segregation · open cv · r-cnn · yolo v3 · raspberry Pi

1 Introduction

In recent years, the agricultural sector has witnessed significant strides in automation, aimed at enhancing efficiency and facilitating large-scale farming. However, the realm of fruit harvesting largely remains reliant on manual labor. Addressing this challenge, the development of an automated fruit-harvesting robot emerges as a practical solution. This innovative approach involves two primary tasks: utilizing computer vision and sensors for the detection and localization of fruits on trees, followed by robotic arm motion to position the robot for precise harvesting without causing damage to the target fruit or the tree [2]. On the other side, it is also included with autonomous segregation, there are not a lot of implemented systems that contain both the harvesting and segregation unit, but this proposed system provides an additional feature, where the robot itself segregates the plucked fruits based on three major factors like size, color, and quantity. Fruit segregation plays an important role because there are some fruits where one rotted fruit can damage or rot other ripened fruits. So, to overcome all these challenges and to fully replace the manual plucking and harvesting system, this system is introduced in vertical farming. Vertical farming offers a potential solution to environmental challenges by enhancing food production with reduced resource consumption. By minimizing water

requirements through efficient water recycling, vertical farming addresses sustainability concerns. Furthermore, it ensures a consistent supply of products to demand centers, reducing the need for extensive storage and refrigeration. These factors underscore the compelling reasons driving the adoption of vertical farming practices.

2 Literature Review

The majority of the examined papers employ the Open-Source Computer Vision Library (OpenCV), a widely utilized library in robotics for the identification, tracking, and comprehension of the surrounding environment as captured by image sensors or cameras. Various image processing algorithms are introduced for object detection, including the Region-Based Convolutional Network (R-CNN), Single Shot Detector (SSD), Random Sample Consensus (RANSAC), Iterative Relief (I-RELIEF), K-means clustering, and the YOLO algorithm.

Kuang-Wen et al. [1] introduces the R-CNN algorithm for object detection. On the VOC 2010 dataset, the R-CNN achieves a mean average precision (mAP) of 53.7%. Notably, it demonstrates a substantial enhancement over the previous high of 24.3% by achieving a mAP of 31.4% on the 200-class ILSVRC 2013 object detection dataset. Despite this success, it's worth noting that this architecture poses challenges in learning, and the generation of test results on a single image from the VOC 2007 dataset takes 49 seconds.

Advaith and Jeet [2] have incorporated the OpenCV Deep Neural Network (DNN) concept to facilitate robot navigation and fruit picking. Following the completion of the setup, the robot effectively navigates to the fruit's location and successfully plucks it.

S. Ren et al. [4] devised an improved Faster R-CNN object detection system that eliminates the need for the selective search technique. In their proposal, Shaoqing Ren and his colleagues introduced an additional area proposal network within the Faster R-CNN framework. This network computes convolutional features directly from the convolutional network to identify objects, eliminating the need for a separate search process within the image.

The Single Shot MultiBox Detector (SSD) demonstrates significantly improved speed compared to two-shot RPN-based techniques. SSD enhances object recognition across various sizes by employing multiple grid sizes, a distinction highlighted in the SSD paper. In contrast, the YOLO algorithm achieves object detection in an image at an impressive rate of 45 frames per second, surpassing other detection methods such as R-CNN and SSD [5].

Furthermore, YOLO v3 represents an advancement over its predecessors in terms of speed. Running at 22 ms with a mean average precision of 28.2, YOLO v3 is three times faster than SSD. Notably, YOLO v3 excels in detecting small objects within an image. The methodology involves a set of algorithms known as YOLO- based convolutional neural networks to efficiently locate and identify objects [6].

Based on the conducted literature survey, diverse and unique implementations of robot designs for fruit plucking and segregation were identified. The customization of the robot design is a key aspect, tailored to the specific dimensions of the tomato tree and the tomatoes themselves.

In the implemented system, the workflow commences with the capture of video footage from a camera, which is subsequently transmitted to a Raspberry Pi. The Raspberry Pi undertakes the conversion of real-time video input into frames. To perform image processing tasks, the system utilizes the OpenCV library, enabling a spectrum of image processing operations in the Python programming language.

The image undergoes a pre-processing stage to remove noise and extract the region of interest, followed by the application of the YOLO algorithm to detect the shape of the tomatoes. Additionally, a color filter is applied to discern the color of the tomatoes in the image. Once the color and shape of the tomatoes are detected, the Raspberry Pi transmits this information to an Arduino as output.

The Raspberry Pi, in turn, issues commands to a robot mechanism, directing it to cut and place yellow and red tomatoes into different partitions. If the tomato is identified as green, the system proceeds forward without executing the cutting action. This integrated approach ensures efficient fruit plucking and segregation based on color and shape characteristics.

3 Proposed System

The primary objective of the system revolves around the efficient harvesting and sorting of ripened and deteriorated fruits while ensuring the overall quality of the produce. Employing the Raspberry Pi 4 Model B, a robust micro-computer, serves a dual purpose in processing real-time video and connecting to a stereo camera for live video capture. Additionally, the Raspberry Pi manages the control of servo motors responsible for the robot arm and cutter, as well as DC motors for the locomotion of the robot. The robotic assembly featured in this system is wholly constructed through the utilization of 3D printing technology.

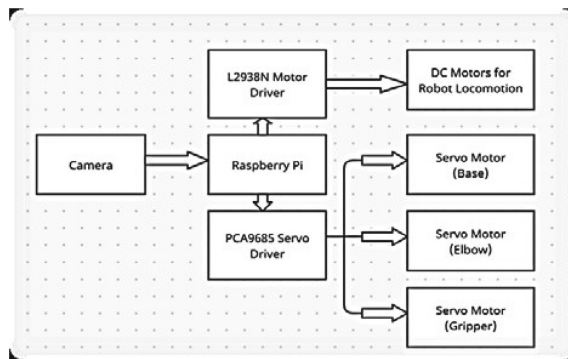


Fig. 1. Work flow of proposed system

In the envisioned system, a tomato fruit serves as the experimental focal point. The system comprises key components such as Raspberry Pi, motors, drivers, and a camera module, as illustrated in Fig. 1. The camera continuously captures video, and the

Raspberry Pi processes the real-time input video, converting it into frames. Subsequently, the system performs object detection on the images to identify the tomato's shape. Figure 2 depicts the additional step of applying a color filter to the tomato image, facilitating the detection of its specific color.

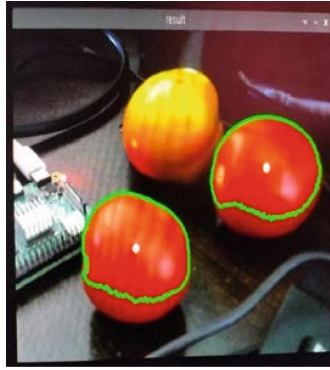


Fig. 2. Red Tomoto detection (Color figure online)

The captured video from the camera undergoes image processing using the OpenCV library, facilitating various image operations in Python. The pre-processing steps involve de-noising the image, extracting a region of interest, and identifying relevant features. The crucial task of classifying the tomato's color into green, yellow, or red is then performed. Subsequently, the Raspberry Pi processes this information and generates output detailing both the color and shape of the detected tomato. This output is then communicated as valuable information about the tomato to external systems or applications. Raspberry Pi sends these output signals to the motor driver circuit, when the driver circuit allows it to continue operation, The Raspberry Pi orchestrates motor commands based on tomato characteristics as follows when a red tomato is detected, the robot motor is instructed to approach the tomato, the servo motor to execute the cutting action, and the dropper motor to deposit the tomato in box2. Similarly, for yellow tomatoes, the robot motor advances towards the tomato, the servo motor performs the cutting task, and the dropper motor places the tomato in box1. In the case of green tomatoes, the command directs the robot motor to move forward.

4 Hardware Implementation

Interfacing Components with Raspberry Pi

In the proposed system, the functionality of the robotic arm is the major, so this arm is mainly made connected through five servo motors. PCA9685 16-channel servo driver for controlling multiple servos. The Servo Driver board communicates with the Pi4 through the I2C protocol, utilizing the set of pins positioned at either end of the board, connecting to the I2C pins on the Pi4. Operating at 3.3 V, the Servo Driver board requires a separate power supply for the servos, which operate at 5 V. This power supply must

be robust enough to provide sufficient current for the connected servos. Each board has the capacity to support up to 16 servos. In this project, three servos were used, each with a maximum current of 140 mA ($140 \text{ mA} \times 3 = 420 \text{ mA}$), necessitating a 1 A power supply (Fig. 3)

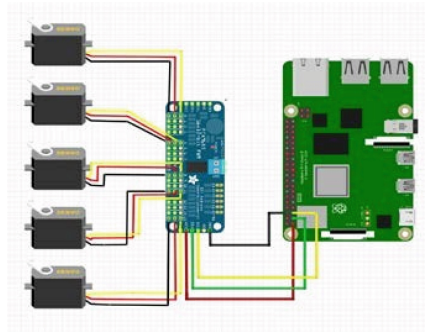


Fig. 3. Interfacing Servos through PCA9685 with Raspberry Pi.

Interfacing DC Motors Through L298N with Raspberry Pi

The L298 is recognized as a dual bidirectional motor driver, relying on a dual H-Bridge Motor driver IC. This circuit facilitates the independent control of two DC motors in either direction. Widely employed in prototypes and hobbyist projects, the L298 is favored for its simplicity in use and seamless integration with both Raspberry Pi and Arduino platforms. Beyond its streamlined design, the L298 includes an onboard 5 V regulator, offering a convenient power source for 5 V circuits (Fig. 4).

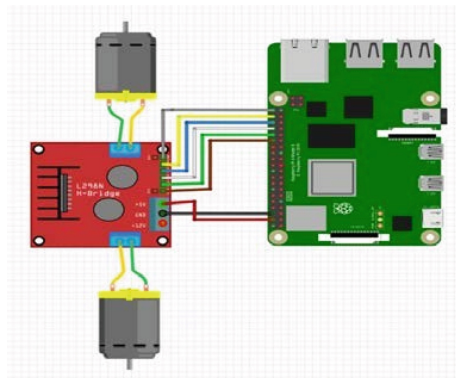


Fig. 4. Interfacing DC Motors through L298N with Raspberry Pi

Interfacing Camera with Raspberry Pi

Real-Time video capturing for object detection is achieved by interfacing the webcam with the USB port of Raspberry Pi. Once the webcam is interfaced with Raspberry Pi then in the configuration, it needs to enable the webcam access. In our proposed system, a Logitech 720 p web camera is used to obtain real-time object detection. This color of the tomato can be detected with the help of the OpenCV algorithm 0(Fig. 5).



Fig. 5. Interfacing Camera with Raspberry Pi

5 Software Implementation

Process algorithm and flow chart of the system (Fig. 6)

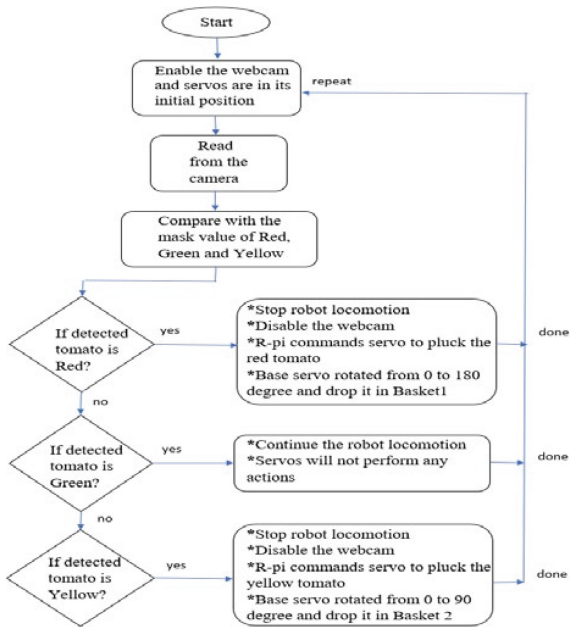


Fig. 6. Process flow of the system

To initiate webcam access on the Raspberry Pi, navigate to the configuration settings and enable the webcam functionality. Upon launching the main program, all servos will assume their initial positions. The program then engages in real-time object detection on tomatoes, analyzing their color and shape. As the webcam captures the object, it undergoes comparison with pre-defined mask values representing red, green, and yellow colors. When a red tomato is detected, the robot's locomotion halts, the webcam is disabled, the gripper servo plucks the tomato, and the base servo rotates its base from 0 to 90°, dropping it into basket 1. Subsequently, all servos return to their initial positions, and the camera is re-enabled for further tomato detection. If the detected tomato is yellow, a similar sequence occurs, with the base servo rotating from 0 to 180° and dropping the tomato into basket 2. Once again, all servos return to their initial positions, and the camera resumes detection. In the case of a green tomato, the robot's locomotion continues until either a red or yellow tomato is detected.

Initiating Video Capture from Camera Source

Python offers a multitude of libraries for image and video processing, with OpenCV standing out as a versatile option. OpenCV is an extensive library that offers a range of functions for image and video operations. Through OpenCV, it becomes possible to capture video from a camera. By creating a video capture object, it becomes convenient to capture webcam footage, allowing for subsequent operations and manipulations on the video data.

How to Record a Video

- To obtain a video capture object for the camera, use `cv2.VideoCapture()`.
- Using the object created above, set up an infinite while loop and read the frames using the `read()` method.
- To view the video's frames, use the `cv2.imshow()` function.
- Ends the cycle when a user hits a particular key.

6 Experimental Results

The Raspberry Pi is powered by a 5v adaptor which in turn powers up all the drivers, servos, and motors. Then Raspberry Pi automatically connects to Putty through which Raspberry Pi is accessed remotely, by entering the command `python main.py` (`main.py` is the program name) in the vim terminal the robot will perform its action based on the program, and the output of the detected objects which are appeared in the terminal is shown below (Fig. 7).

```
SSH is enabled and the default password for the 'pi' user has not been changed.  
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.  
  
pi@raspberrypi:~$ cd Collector-Bot/  
pi@raspberrypi:~/Collector-Bot$ python main.py  
-----main-----  
yellow tomato distance 6.844106403878327  
arm basket  
arm init  
red tomato distance 7.8109519311294495  
arm basket  
arm init
```

Fig. 7. Colour and distance of detected tomato is appeared in the terminal.

In the above figure when the robot detects the red and yellow tomato, the same thing reflects in the terminal with the distance between the gripper and the object. Once the robot drops the tomato in the basket, the robot again goes to its initial position. If a robot detects a green tomato, it is left as it is in the plant (Figs. 8, 9 and 10).



Fig. 8. Red colour tomato plucked by robot



Fig. 9. Segregating red tomato



Fig. 10. Plucking and Segregation of yellow tomato.

7 Conclusion

The implementation of autonomous fruit plucking and segregating robots presents a significant advancement over manual harvesting and segregation methods in vertical farming. This robotic system is capable of autonomously plucking red and yellow tomatoes and subsequently segregating them based on their shape and color. The Raspberry Pi serves as the main micro-processor, enabling real-time object detection.

The integration of both harvesting and segregating functionalities in this system represents a substantial improvement compared to traditional technologies. The 3D-printed robot arm is designed with precision to ensure that it performs plucking and segregation without causing harm to plants or fruits. This effective design contributes to the overall efficiency and non-intrusive operation of the robotic system in the vertical farming environment.

References

1. Hsieh, K.-W., Huang, B.-Y., Hsiao, K.-Z., Tuan, Y.-H., Shih, F.-P.: Fruit maturity and location identification of beef tomato using R- CNN and binocular imaging technology. *J. Food Measure. Charact.* **15**, 5170–5180 (2021). <https://doi.org/10.1007/s11694-021-01074-7>
2. Advait and Jeet: Design and Fabrication of the autonomous Fruit Plucking Robot. *Research Paper Engineering E-ISSN No: 2454-9916*, vol. 5, Issue: 8 (2019)
3. Zhao, Z., Wu, X., Xu, S., Zheng, P.: Object detection with deep learning: a review. *IEEE Trans. Neural Networks Learn. Syst.* **30**(11), 3212–3232 (2019)
4. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(6), 1137–1149 (2017)
5. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. *ResearchGate* (2018)
6. Zhao, L., Li, S.: Object detection algorithm based on improved YOLOv3. *Electronics* **9**(3), 537 (2020)
7. Onishi, Y., Yoshida, T., Kurita, H., Fukao, T., Arihara, H., Ayako: An automated fruit harvesting robot by using deep learning. *J. https://robomechjournal.springeropen.com/articles/https://doi.org/10.1186/s40648-019-0141-2*
8. Ya, X., Ge, Y., Grimstad, L., Pål, J.: An autonomous strawberry harvesting robot: design, development, integration, and field evaluation. <https://onlinelibrary.wiley.com/doi/full/https://doi.org/10.1002/rob.21889>

9. Wu, J., Zhang, B., Zhou, J., Xiong, J., Gu, B., Yang, X.: Automatic recognition of ripening tomatoes by combining multi- feature fusion with a bi-layer classification strategy for harvesting robots. <https://doi.org/10.3390/s19030612>. Received: 8 January 2019/Revised: 30 January 2019/Accepted: 30 January 2019/Published: 1 February 2019
10. Bautista, J.F., Oceña, C.D., Cabrerós, M.J., Alagao, S.P.L.: Automated sorter and grading of tomatoes using image analysis and deep learning techniques. <https://ieeexplore.ieee.org/document/9400055>



An Efficient System for Water Quality Monitoring in Sustainable Ground Water Supply Using IoT Technology

C. Sri Krishna Shastri^(✉) and M. C. Jayaprakash

Mangalore Institute of Technology and Engineering, Moodabidri, Karnataka, India
krishnashastri@gmail.com

Abstract. Water is an essential resource for all humans and living beings. All over the globe, people require pure water for drinking and cooking purpose. However, in present scenario it is very difficult to get pure water from natural structures such as river, lake or manmade wells due to the contaminated chemicals from Industry. The drinking water will mostly contaminated due to human errors or sometimes due to natural disasters such as flashfloods, tsunami etc. Considering the facts, supplying a fresh drinking water for the households is a major concern for the water supplying authorities in India. In this paper, an attempt is made to monitor different parameters such as pH value, TDS, Turbidity and temperature of water. Based on the obtained result, different methods are incorporated to clean the water at source itself before supplying it to the consumers. These parameters are measured using various sensors and an Arduino based system is designed for processing the sensed data.

Keywords: Turbidity · Ground water · ph value etc.

1 Introduction

Water is an extremely sacred resource for the existence of mankind on earth. It carries nutrients to all cells in our body & oxygen to our brain. About 71% of earth is covered with water, but only 2.5% of it is drinking water. With increase in pollution, population and climate change, it is expected that by 2025 we will experience eternal water shortages. But in the 21st century, research has proved the fact that people are reaching a point where water will be more expensive as compared to mineral oil. The main purpose of this paper is to reduce wastage of the most valuable resource, termed as “WATER”. This is one of the major aims of the Government of India, to supply safe drinking water to every household till the year 2024. The water must not be wasted as till now near about 14 crore households in India have water shortage. Hence the goal is not only to access clean and potable water to households and to raise our Country’s ranking in Water Quality Indexing but also safe drinking water availability. In a survey carried out recently, increase in human population caused major issues for water pollution. All living beings suffer from insufficient pure water supply for their day-to-day needs. Supply of pure

water to the daily needs has become a serious issue in recent days. In places like metro cities, some part of the city will get enough water whereas some other part may face scarcity of pure drinking water due to water supply mismanagement. This is mainly due to the failure in proper planning of water distribution. Hence by real time monitoring of water supply pipes using AI/ML technology one can resolve all these issues.

In various parts of the globe, people mainly depend on ground water for their domestic, agriculture and industrial need rather than surface water resource. Groundwater is one of the most important sources of fresh drinkable water which can satisfy the needs of the humankind. In traditional approach, the quality of water is examined by sending the water sample to the lab and manual process the quality measures are taken. Through this the chemical, physical and biological agents present in water are tested. But the process is time dependent and labor intensive.

The system is a model and instructions on nonstop and real time observing of water resource using mobile platform. Water resource with nonstop monitoring helps in a proper distribution so that one can have the data of obtainable amount of water in storage tank, inflow rate, abnormality in the supply system.

Depletion of ground water occurs primarily because of sustained groundwater pumping. Some of the issues caused by this are: Lowering of the Water Table, Increased Costs, Reduced Surface Water Supplies, Land Subsidence and Water Quality Concerns.

In this paper, the main focus is to monitor the quality of water in supply system continuously with the aid of IoT technology. The continuous monitoring of water quality helps in maintaining the proper record of the drinking water. In this paper we have used things speak as free server in which data is continuously pushed on the cloud so that one can see the information in real time operation. Using different sensors with a microcontroller and Arduino as a computer in system can monitor information and also control operation through wireless networks efficiently.

1.1 Background Work

Ahmed et al. (2020) used IoT systems for water quality monitoring in real-time and machine learning methods like ANN and SVM to predict WQI, and detect abnormal events, namely, purposeful pollution to allow real-time pollution detection and control operation. Their proposed method also applied regression, correlation analysis, hierarchical clustering, etc. to study multiple directions to include learning techniques to support decision-making.

Sokratis Kartakis, Weiren Yu, Reza Akhavan, and Julie A. McCann entitled “Adaptive Edge Analytics for Distributed Networked Control of Water Systems” This paper discusses the burst detection and localization approach that integrates low weight compression and anomaly identification with graph topology analytics for water distribution.

Paetz, J.L Villa (2018): here they have identified the Ecological observation in internal waters as a prime complication for enough growth and administration of such territories. A well designed Kalman filter was cast-off to estimate the positioning and location by means of the analyses of an inertial part combined with a gaussmeter and Global Positioning System.

Pranita Vijaykumar Kulkarni and M. S. Joshi have adopted Internet of Things (IoT) technology in which the work focuses on a solution for 'Water distribution and management' in urban and rural areas with the help of IoT. This work concentrates on the environmental issues like durability, affordability, prevention against leakage and maintenance problems. Hence the designed system prevents excess usage of water, water wastage and promises an efficient water management system.

E.V.de Souza, M.A Costa Da Silva (2021): The idea behind this paper is to implement and develop effective water loss strategies, the current paper aims at presentation of methodology for the improvement of water resources for efficiency in the water supply system.

Grunwald, A.; Schaarschmidt (2019): They have studied the effect of less usage of mobile and computer technology in water supply system. The authors have proposed too different use cases. In the first one, the temperature of a horse steady was closely watched and the data was transmitted via a RF channel. In the second use case, they have deployed a self-developed device on the agriculture land at a known depth to identify the soil properties. In addition, a server and gateway architecture having access to a cloud system for data processing purposes was proposed and in a second step, a less power prototype with dissimilar sensors for data gathering for the labelled use cases are established. The developed system has shown significance result in identifying the soil properties for the rural conditions.

Liu et al. (2019) focused on a prediction model of water quality which needed data of higher quality. They constructed and operated a smart system of water quality monitoring on the basis of the IoT, which generated too much big data at a higher speed and thus made water quality data complex. They designed and established a drinking-water quality time-series prediction model to predict water quality big data by means of state of the art deep learning (DL) method using LSTM. They obtained drinking-water quality data from the automatic water quality monitoring station of Guazhou Water Source of the Yangtze Riverin Yangzhou.

Muharemi et al. (2019) described many methods for detecting variations or abnormalities taking place in time series data of water purity. Their work too discussed and proposed solutions to overcome certain problems when handling time series data. They used logistic regression, deep neural network (DNN), linear discriminant analysis (LDA), long short-term memory (LSTM), support vector machines (SVM), ANN, and recurrent neural network (RNN) to apply over time series data and perform evaluation using F score.

Gowthami et al.: presented a low cost IoT-based solution to monitor quality, level, and quantity of water in overhead tanks. Only pH and turbidity were considered to monitor water quality. In addition, they also used a combination of ultrasonic sensor and a flow sensor to control water level and quantity in tanks, respectively. Technical specifications of sensors are not provided. The entire system is based on an Arduino Uno and ESP8266 Wi-Fi transceiver. machine learning methods were utilised for better decision making. In short, this system works as follows: The Arduino Uno reads sensors, does some local processing, and shifts data to a cloud server (not specified) via ESP8266 and a local Wi-Fi router. The cloud server is used to store data, visualise sensor data, and send an alert to end users via a smart phone in case any anomaly is detected. To summarise, this system is not well mature and should be consulted if auto-filling of tank and/or cleaning is desired only.

Muharemi, F., Logofatu, D. & Leon, F. (2019): Identifying the purity of water play key role for the success of aqua companies. Aqua related companies must ensure that they must supply pure and safe drinking water for mankind. Now sophisticated equipments with advanced sensors are used to monitor the water quality. This work represents the description of several approaches to identifying changes or anomalies occurring on water quality time series data. The results show that all algorithms are vulnerable although SVM, ANN and logistic regressions tend to be a little less susceptible, while DNN, RNN and LSTM are very susceptible.

S. Garuglieri, D. Madeo (2019): The authors have suggested an integrated system for the purpose of monitoring of the aquatic ecosystems. They have developed a low cost system, called Water Environmental Mobile Observer and standard components as the physical and chemical parameters are on condition that with a segmental collection of sensor are performed a bathymetry.

Muhammad, S. Y., Makhtar, M., Rozaimée, A., Abdul Aziz, A. & Jamal, A. (2015) According to the World Health Organization (WHO), approximately 2 billion people worldwide use drinking water sources that are contaminated with faeces. This is a serious issue since contaminated water may lead to certain waterborne diseases such as cholera, hepatitis A, dysentery, jaundice, and typhoid fever. Therefore, many researchers around the world are interested in studying the water quality. One of the most commonly used approaches is by using machine learning. Machine learning approach has grabbed the interest of many researchers since the last several years due to its power to compute complicated mathematical computations on big data analysis. Therefore, this study explored the correlation between different water quality parameters and Water Quality Index (WQI) in water quality studies that used machine learning by using a meta-analysis approach. This study used estimated variance, heterogeneity index, Chi-squared heterogeneity test and the random effects model. Based on the selected articles, pH, dissolved oxygen (DO) and biochemical oxygen demand (BOD) are the parameters commonly used in water quality studies which use a machine learning approach. This study found that pH is the best chemical factor which greatly affects the Water Quality Index since it has the highest mean correlation and lowest estimated variance due to sampling error.

1.2 Motivation for the Work

The conventional method for monitoring of the quality of water is such that the water sample is collected and sent to the laboratory to be tested manually by analytical methods. Although by this method the chemical, physical, and biological agents of the water will be analyzed, it has several drawbacks. Firstly, it is time consuming and labor intensive. Secondly, the cost for this controlled, displayed, and transferred. Compared to the conventional water quality testing techniques, sensor based water quality testing has many advantages such as accurate, high sensitivity, good selectivity, speed, fast response, low cost etc.

In the present century, there were enough inventions in technology to keep the drinking water pure and safe, but at the same time due to manmade pollution, Global warming and other environmental issues, the drinking water will be contaminated badly. Hence it is imperative to develop a better methodologies to monitor the water quality parameters in real time. The water quality parameters pH measures the concentration of hydrogen ions. It helps in identifying the water is acidic or alkaline. The pH value of pure water is 7 where as pH less than 7 is acidic and more than 7 is alkaline. The range of pH is varies between 0–14. The drinking water pH value must be between 6.5 to 8. Turbidity means the large number of suspended particles in water which is invisible to human eye. Higher value of turbidity results in health hazard like diarrhea, cholera etc. Hence the drinking water must have less value of turbidity.

2 Methodology

The IoT system integrated smart water quality system of household for sustainable ground water management uses tank to store water before the water supply. There is a Nitrate/Sulphate/Fluoride sensor and pH sensor that is embedded after the water flow sensor which checks the Nitrate/Sulphate/Fluoride value and decides if the water is potable (suitable for drinking). If the water is not potable, the sensor doesn't release the water further. The same would be attempting for other quality of water standards. Collecting all the required data to process continuously and push data through a Wireless signal to a device. Flow rate measured in Litres/sec/min/hour. By calculating the pulses from the output of the sensor, one could easily keep track the fluid movement. The IoT based model consists of node MCU for transmitting data over an app which can be accessed by the authorized end user. The final data is compared with manual tested data for further verification.

The circuit was designed according to the block diagram Fig. 1, Various sensors such as pH, temperature, turbidity and TDS are interfaced to Arduino for water quality system and further processing. Final output signal from Arduino is relayed through ESP8266 node MCU to the designated application for real time monitoring of water quality.

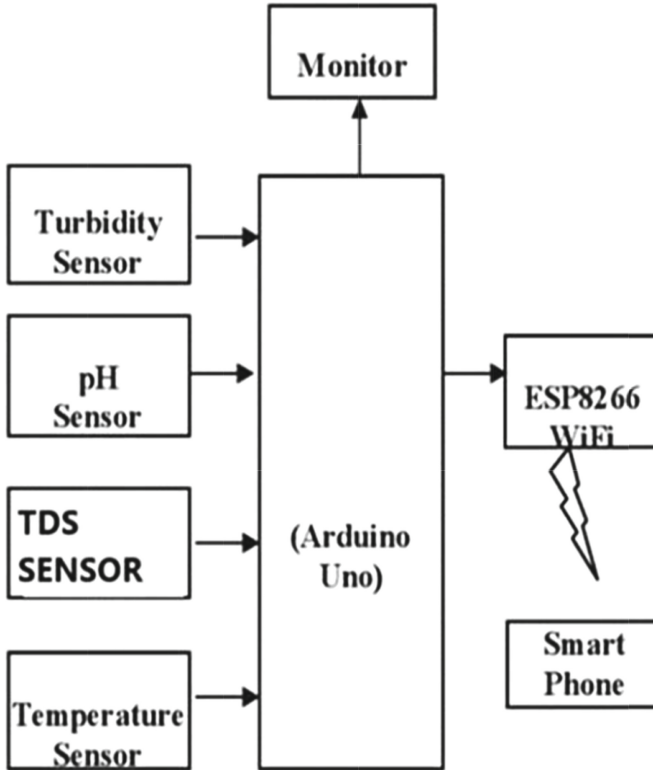


Fig. 1. System Block diagram

3 Results and Discussions

Water samples from different water sources were tested to build a reference on the parameters for each type of sample. The chosen sample of water belongs to tap water, surface water, pool water, etc. For safety reasons, the system is not installed in the specific areas of interest; instead water samples were collected and tested in a safe controlled environment. Readings were taken simultaneously as we tested nine different water Samples, according to that we got some results (Fig. 2).

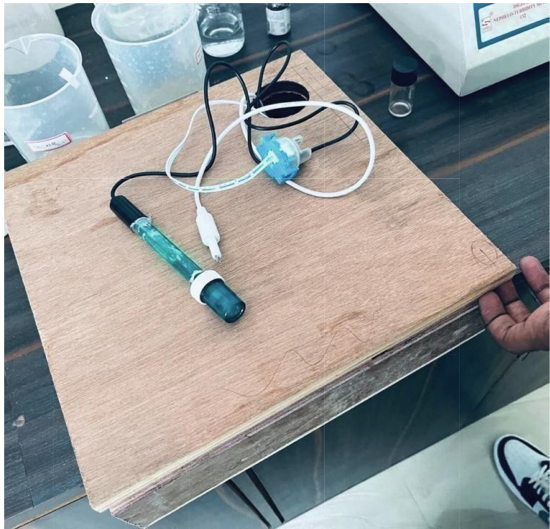


Fig. 2. Laboratory Set Up for testing water sample

The water samples from various sources is tested in the laboratory for quality purpose. We tested the samples for pH level, Turbidity, TDS and dissolved oxygen content with the aid of various sensors. The results obtained shows that the water from well/ pond found to be more suitable for drinking purpose. The water from bore well and tap are found to be lesser in quality as the results tabulated in Table 1 below (Figs. 3 and 4).

Table 1. Tested parameters of different water samples..

Sample	Ph	Turbidity	TDS	Muddiness
Well	7.23	2.6	168	22
Borewell	7.45	4.5	274	37
Tap water	7.9	3.8	235	34
River	7.52	3.2	224	33

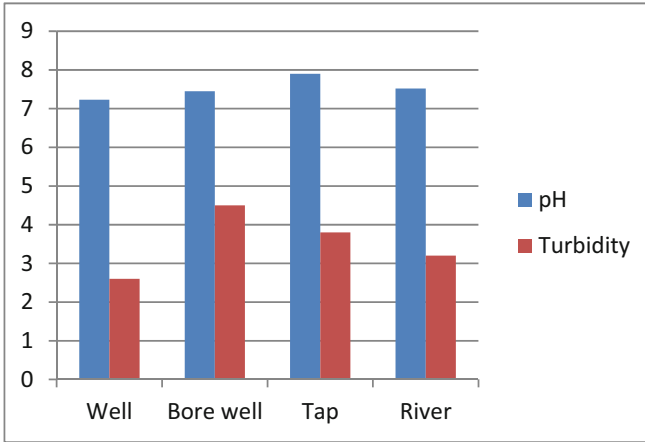


Fig. 3. Ph and Turbidity levels of water samples

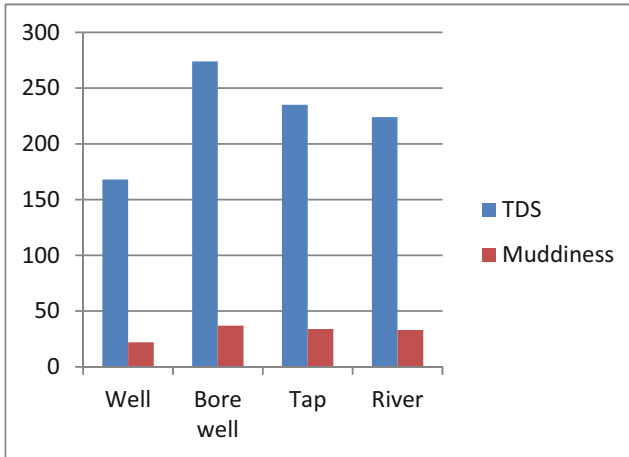


Fig. 4. TDS and Muddiness levels of water samples

4 Conclusion

The system was designed with an intent to help the mankind to decide the quality of water from various sources for dinking purpose. The designed system consists of microcontroller with different sensors for monitoring the parameters of water. The obtained results are displayed on a LCD display. Also using Iot technology, all the obtained parameters are uploaded to cloud server using Blynk application so that the user can access the data remotely sitting anywhere in the globe. The obtained results were compared with the standards specified by Word Health Organisation and validated with manual approach.

References

- Natividad, J.G., Palaoag, T.D.: 1.Isabela State University–Iligan Campus, City of Iligan, Isabela, Philippines, 2.University of the Cordilleras, Baguio City, Philippines IOPC on f. Series: Materials Science and Engineering 482 (2019)
- Tamilselvan, G.M., Ashishkumar, V., Jothi Prasath, S., Mohammed Yusuff, S.: *Int. J. Recent Technol. Eng.* **7**(4S), (2018) ISSN:2277-3878. IoT Based Sustainable Groundwater Quality System for Green India
- Kulkarni, P.V., Joshi, M.S.: *Int. J. Innov. Res. Sci. Eng. Technol.* (An ISO3297:2007 Certified Organization) **5**(9) (2016)
- Devasena, D., Ramya, R., Dharshan, Y., Vivek, S., Darshan, T.: *Int. J. Eng. Adv. Technol.* **8**(6) (2019) ISSN: 2249–8958 Kusuma, S.S., Anil, G.N.: *Int. Res. J. Eng. Technol.* **05**(02) (2018)
- Kusuma, S.S., Anil G.N.: *Int. Res. J. Eng. Technol.* **05**(02) (2018)
- Hussain, I., Das, M., Ahamad, K.U., Nath, P.: Water salinity detection using a smartphone. *Sens. Actuat. B Chem.* **239**, 1042–1050 (2017)
- Paez, J., Villa, J.L., Cabrera, J., Yime, E.: Implementation of an unmanned surface vehicle for environmental monitoring applications. In: *Proceedings of 2018 IEEE 2nd Colombian Conference on Robotics and Automation (CCRA)*, pp. 1–6. Barranquilla, Colombia, 1–3 November 2018
- Demetillo, A.T., Taboada, E.B.: RealTime water quality monitoring for small aquatic area using unmanned surface vehicle engineering. *Technol. Appl. Sci. Res.* **9**(2), 3959–3964 (2019)
- Kartakis, S., Yu, W., Akhavan, R., McCann, J.A.: *IEEE First International Conference on Internet-of-Things Design and Implementation*. 978-1-4673- 9948- 7/16 © 2016 IEEE (2016)
- Barabde, M., Danve, S.: *Real Time Water Quality Monitoring System, IJIRCCE*, vol. 3, June 2015
- Purohit, A., Gokhale, U.: Real time water quality measurement system based on GSM , IOSR (IOSR-JECE), vol. 9, Issue 3, Ver. V (2014)
- O’Connell, E., Healy, M., O’Keeffe, S., Newe, T., Elfed
- Kedia, N.: Water quality monitoring for rural areas- a sensor cloud based economical project. In: *1st International Conference on Next Generation Computing Technologies (NGCT-2015)* Dehradun, India, 4–5 September 2015. 978-1-4673-6809- 4/15/\$31.00 ©2015 IEEE (2015)
- Bhatt, J., Patoliya, J.: *Iot Based Water Quality Monitoring System, IRFIC* (2016)
- Michallom, ondrejpriby & miroslavsvitek, *Internet 4.0 as a part of smart cities*, 978-1-5090-116-2/16/\$31.00 ©2016 IEEE
- Sun, Z., Liu, C.H., ChatschikBisdikia_, Branch, J.W., Yang, B.: 2012 9th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks
- Nasser, N., Ali, A., Karim, L., Belhaouari, S.: 978-1-4799- 0792- 2/13/\$31.00©2013 IEEE
- Hari Sudhan, R., Ganesh Kumar, M., Udhaya Prakash, A., Devi, A., Sathiya, S.: *Arduino AtMEGA-328 microcontroller. Int. J. Innov. Res. Electric. Electron. Instrument. Control Eng.* **3**(4) (2015)
- Muhammad, S. Y., Makhtar, M., Rozaimee, A., Abdul Aziz, A. Jamal, A.: Classification model for water quality using machine learning techniques. *Int. J. Softw. Eng. Appl.* **9**(6), 45–52 (2015)
- Muharemi, F., Logofatu, D. Leon, F.: Machine learning approaches for anomaly detection of water quality on a real-world data set. *J. Inf. Telecommun.* **3**(3), 294–307 (2019)
- Garuglieri, S., Madeo, D., Pozzebon, A., Zingone, R., Mocenni, C., Bertoni, D.: An Integrated System for Real- Time Water Monitoring Based on Low Cost Unmanned Surface Vehicles. In: *2019 IEEE Sensors Applications Symposium, Sophia Antipolis*, pp. 1–6. France, 11–13 March 2019
- Liu, P., Wang, J., Sangaiah, A. K., Xie, Y., Yin, X.: Analysis and prediction of water quality using LSTM deep neural networks in the IoT environment. *Sustainability* **11**, 2058 (2019)



Stable Control of Underwater Target Search Robot Support Rescue and Rescue Work

Nguyen Duc Anh¹ , Ravishankara Kulamarva³, D. Suresha⁴,
and Nguyen Quang Vinh²

¹ University of Fire Prevention and Fighting, Ha Noi, Viet Nam

² Academy of Military Science and Technology, Ha Noi, Viet Nam
Vinhquang2808@gmail.com

³ Srinivas Institute of Technology, Mangaluru, India

⁴ AJ Institute of Engineering and Technology, Mangalore, India

Abstract. Robots that operate underwater (underwater robots) replace humans in searching for victims, weapons, and evidence in criminal cases... Normally, robots move underwater thanks to the Human control from a shore station, however, when detecting a suspected object as a search target, the robot needs to stabilize in one position to observe and salvage the object. The level of position stability and working posture of the robot greatly affects the quality of observed images and the ability to salvage objects. Therefore, the topic group chose history Use a gravity-compensated PID automatic controller to perform the above task, because this is a set The controller has a fast response time, suitable for the object of the problem. Simulation results on Matlab/Simulink show that the robot is quite stable at the reference position. Result Actual testing has proven the correctness of the simulation and meets the requirements of the problem.

Keywords: Stable control · PID · Underwater robots

1 Introduction

Currently, autonomous underwater vehicles (AUVs) are being widely used in many different civil and military tasks such as oil and gas exploration and exploitation [10], surveying oceanography, demineralization, and bathymetric data collection in marine and riverine environments [11]. In the process of performing the above tasks, people usually need to deploy orbit tracking or accurate path tracking techniques for AUV devices. With the focus on AUV motion control as mentioned above, the field of trajectory tracking control has received relatively more attention than path tracking control. The reason is that it is related to the design of control rules to make the device able to achieve and follow a trajectory based on time-varying parameters (i.e. a geometric path associated with a time rule) [12]. However, it will be very difficult and challenging to design an accurate trajectory tracking rule for AUVs. The reason is because the device's dynamics has highly nonlinear characteristics and strong disturbances leading to non-stationary problems [9]. Meanwhile, the presence of external influences that

change over time, which are factors that are difficult to measure or estimate, also makes designing a trajectory tracking controller a difficult task [1, 4, 7, 12]. Up to now, many different control methods have been proposed by engineers and researchers interested in the problem of designing precise motion control systems for underwater moving devices. PID controllers are often used to control devices that move underwater due to their relative ease of deployment. In Fossen [9], the author proposed to apply PID with compensation method restoring force (i.e. gravity and buoyancy), ensuring local stability of the closed-loop system, to adjust an ROV device to a set point. In [13], the author designed an unblocking PD control scheme, with fast deceleration and position error convergence, to control AUVs according to a set point. Have shown that the PID control can only adjust to the exact set point (i.e. keep it fixed) if the device is linear and the action is constant. But in the case of trajectory tracking, the PID controller cannot precise tracking both theoretically and practically, even for a quadratic linear device [14]. To control the device moving underwater in the case of high-precision trajectory tracking, several types of The controller applies the same understanding of device dynamics presented in [15]. These can give consistent performance and stability results both theoretically and in simulation. Authors [16], also stated that these types of controllers are difficult to implement experimentally, the reason being that it is impossible to determine the exact AUV dynamics in practical applications [6, 9].

There are no published projects on applying AUVs to rescue work. The authors present an AUV control method applied in rescue and prevention work. The achieved simulation and actual results have proven the correctness of the proposed algorithm and model.

2 Underground Robot Modeling

To describe the motion of underground robots, people use the Euler transformation matrix for a 6-degree-of-freedom mathematical model that represents the relationship between the coordinate system attached to the object and the coordinate system fixed to the earth's center. The general dynamic equation of the underground robot is described in [2, 3, 5] as follows:

$$\begin{cases} \dot{\eta} = J(n_2)v \\ \mathbf{M}\dot{v} + C(v)v + D(v)v + G_f(\eta) = \tau_A \end{cases} \quad (1)$$

In which $\eta = [\eta_1^T, \eta_2^T]^T$ and $\eta_1 = [x, y, z]^T$ location of the robot's center of mass in a fixed coordinate system; $\eta_2 = [\varphi, \theta, \psi]^T$ orientation angle between mounted coordinate system attached to a fixed coordinate system; $v = [v_1^T, v_2^T]^T$ and: $v_1 = [u, v, w]^T$ long velocity vector projected onto the attached coordinate system axis; $v_2 = [p, q, r]^T$ transport vector angular velocity projected onto the attached coordinate system; $J(n_2)$ Coordinate conversion matrix from the attached coordinate system to the fixed coordinate system, the values of the parameters depend on the rotation angles $\eta_2 = [\varphi, \theta, \psi]$;

$D(v)$: Damping matrix; $G_f(n)$: force and moment vectors caused by gravity and buoyancy; τ_A : Force and Moment generated by thrusters; $\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A$ in there $\mathbf{M}_A$

Inertial matrix of the enclosed water mass; MRB Inertia matrix; $\mathbf{C} = \mathbf{C}_{RB} + \mathbf{C}_A$ in there $\mathbf{C}_A(v)$: Coriolis radial matrix of the attached water mass; $\mathbf{C}_{RB}(v)$ Coriolis centripetal force matrix.

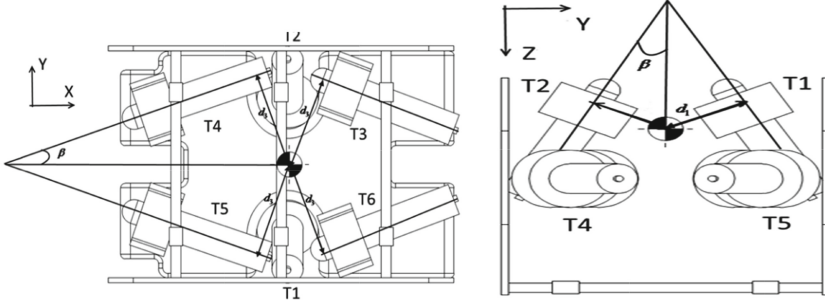


Fig. 1. Underground robot model

With the model shown in Fig. 1, the expression describing the relationship between the external force moment in all directions τ_A and the voltage signal matrix controlling the thrusters U is as follows [1, 8] (Figs. 2 and 3):

$$\tau_A = \begin{bmatrix} 0 & 0 & c\alpha & c\alpha & c\alpha & c\alpha \\ -s\beta & s\beta & -s\alpha & s\alpha & -s\alpha & s\alpha \\ c\beta & c\beta & 0 & 0 & 0 & 0 \\ -d_1 & d_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -d_2.c\beta & -d_2.c\beta & d_2.c\beta & d_2.c\beta \\ 0 & 0 & d_3 & d_3 & -d_3 & -d_3 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} \quad (2)$$

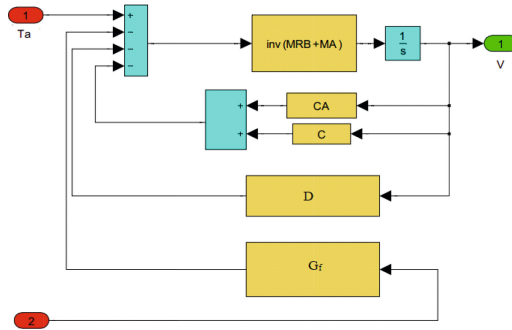


Fig. 2. Underground Robot model described in Matlab/Simulink

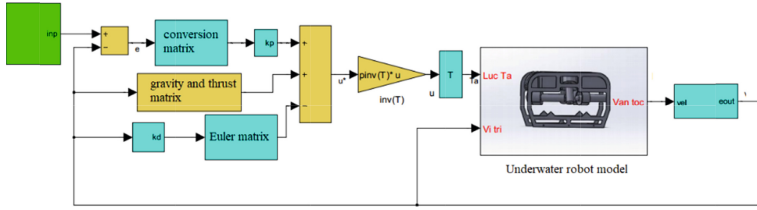


Fig. 3. PID controller model

3 Design an Automatic Controller to Stabilize the Robot

The controller's task is to automatically adjust the voltage levels supplied to the propulsion motors appropriately so that the robot is always stable in posture and position at the working point according to work requirements in a complex water environment. Complexity (changes in flow, environmental factors...). Select a gravity-compensated PID controller with the voltage to control the thrusters according to the following law:

$$\mu = T^{-1} [J^T(\eta) K_p e - J^T(\eta) K_d \dot{J}(\eta) + G_f(\eta)] \quad (3)$$

The calculation results show that the control law ensures that the system will be stable according to the target Lyapunov norm with Lyapunov function

$$V(v, e) = \frac{1}{2} (v^T \cdot M \cdot v + e^T \cdot K_p \cdot e)$$

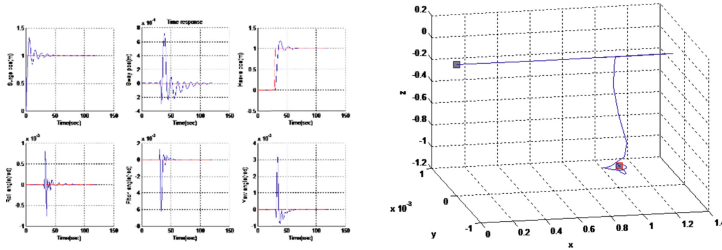


Fig. 4. Status of underground robot and motion trajectory in case 1

4 Simulation and Experiment Evaluate the Stability of the Robot When Operating

To ensure the controller works well, the robot implicitly provides a stable response, the research team conducted simulations and tested the controller in 2 cases:

4.1 Case 1

The robot is required to move from coordinates (0,0,0) to (1, 0, 0.7) according to the route: in the first 30 s, move along the x-axis (forward) 1m. 30 s later requires a z-axis displacement (dive) of 0.7m. The simulation process does not include environmental noise and errors in the underground robot model. The simulation results are shown in Fig. 4:

The simulation results in Fig. 4 show that the controller provides voltage to the thrusters so that the robot automatically moves, stops, and maintains exactly at the working point with coordinates (1, 0, 0.7). The movement trajectory is completely consistent with the designed route. The pitch angle, deflection angle, and tilt angle only fluctuate in a very small range, this ensures that there is no vibration during movement, the robot always maintains the posture and direction of observation, satisfying the requirements set for case 1.

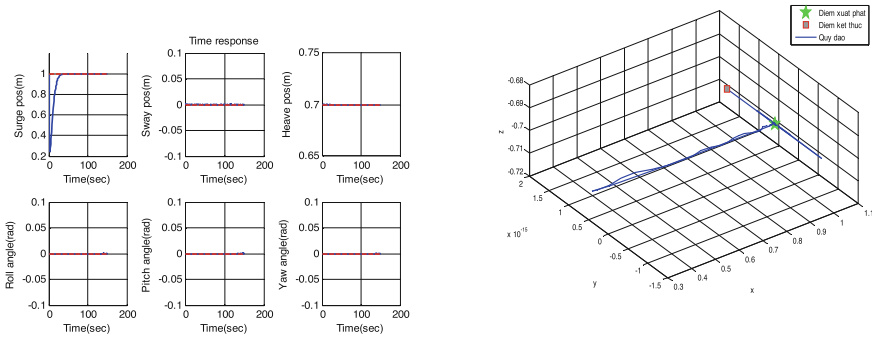


Fig. 5. Figure 5: State of the underground robot and motion trajectory when the flow is along the x-axis

4.2 Case 2

The robot stops at a point with coordinates (1, 0, 0.7) to observe. The simulation process adds environmental noise, flow velocity, and errors in the underground robot model. The controller's requirements must ensure that the robot maintains the reference coordinates and is stable with allowable vibration.

- When the flow is disturbed along the x-axis with a velocity of 0.5m/ s

The simulation results shown in Fig. 5 show that: when the flow impacts along the x-axis on the head of the underground robot, it will not greatly affect the stability of the system, the underground robot is pulled under the influence of the flow. Back 0.8 m along the x-axis, the controller immediately pulls the underground robot back to the working point, with a very small deviation from the initial position of about 10–15 m. The response of tilt angle, tilt angle, and deflection angle does not change, ensuring that the robot maintains the same direction and working posture.

- When the flow is in two directions along the x-axis with a velocity of 0.5 m/s and along the y-axis with a velocity of 0.1 m/s.

The simulation results in Fig. 6 show that the flow impacting the side of the underground robot harms the system, the robot tends to oscillate very small around the working point position. The response of the roll tilt angle is deviated from the set position by about 0.02 rad, because there is a force from the flow that always acts on one side of the robot, making the robot always tend to tilt at a small angle. The pitch angle and deflection angle fluctuate within the allowable range of about 0.1 rad. However, the controller still meets the requirements set out in this case.

Some image testing, assessment of authenticity in Fig. 7 and The actual testing results are shown in the video: <https://youtu.be/uBM4xI21ECI>.

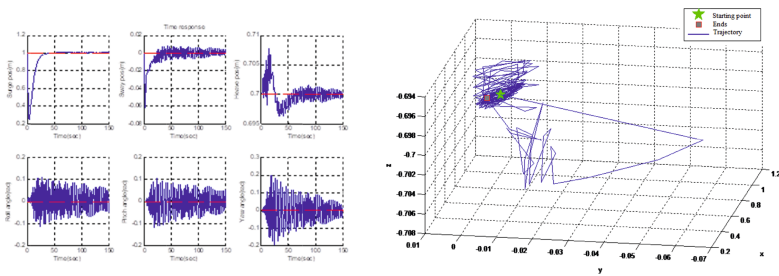


Fig. 6. Status and motion trajectory of the robot underground when the flow affects both the x and y axes

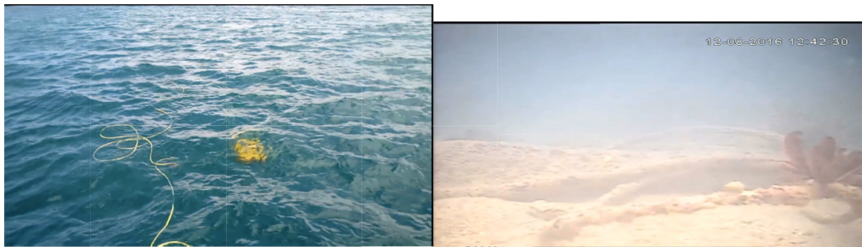


Fig. 7. Testing, assessment of authenticity

5 Conclusion and Future Research Directions

The simulation results accurately reflect the physical nature of the robot's operation in still water and flowing water environments. The designed gravity-compensated PID controller performs the set function well.

The above results are based on the project team's creation and completion of an experimental underground robot operating in real environments that meets the proposed requirements and demonstrates the simulation results.

Next, the research team will continue to conduct real-world testing with different scenarios to evaluate the stability of the system as well as the effectiveness of the algorithm.

References

1. Christ, R., Wernli Sr, R.: The ROV manual: a user guide for observation class remotely operated vehicles (2011)
2. Volden, Ø., Cabecinhas, D., Pascoal, A., Fossen, T.: Development and ex. *Ocean Eng.* **280**, 114675 (2023)
3. Lipko, I.: Identification of the horizontal movement of the underwater vehicle MiddleAUV. In: 2022 International Russian Automation Conference (RusAutoCon), pp. 820–825 (2022)
4. Wu, H., Wu, Y., Zhang, Y., Zhang, Y.: Intelligent encryption method for wireless sensing signal of underwater vehicles. *Int. J. Vehicle Inform. Commun. Syst.* **7**, 366–380 (2022)
5. Vinh, N., Van Phuc, P.: Control of the motion orientation of autonomous underwater vehicle. *Procedia Comput. Sci.* **150**, 69–77 (2019)
6. Vinh, N., Thanh, N.: Construction of a backstepping controller for controlling the depth motion of an automatic underwater vehicle. In: Research In Intelligent And Computing In Engineering: Select Proceedings Of RICE 2020, pp. 325–332 (2021)
7. Inzartsev, A.: Underwater vehicles. BoD–Books on Demand (2009)
8. Bahr, A., Leonard, J., Fallon, M.: Cooperative localization for autonomous underwater vehicles. *Int. J. Robot. Res.* **28**, 714–728 (2009)
9. Fossen, T.: Feedback error-state Kalman filter with time-delay compensation for hydroacoustic-aided inertial navigation of underwater vehicles. *Control Eng. Pract.* **138**, 105603 (2023)
10. Xiang, X., Lapierre, L., Jouvencel, B.: Smooth transition of AUV motion control: from fully-actuated to under-actuated configuration. *Robot. Autonom. Syst.* **67**, 14–22 (2015)
11. Paul, L.: Transformative experience. OUP Oxford (2014)
12. Yang, C., Li, Z., Cui, R., Xu, B.: Neural network-based motion control of an underactuated wheeled inverted pendulum model. *IEEE Trans. Neural Networks Learn. Syst.* **25**, 2004–2016 (2014)
13. Jang, J., Liccardo, D.: Small UAV automation using MEMS. *IEEE Aerospace Electron. Syst. Magaz.* **22**, 30–34 (2007)
14. Smallwood, D., Whitcomb, L.: Adaptive identification of dynamically positioned underwater robotic vehicles. *IEEE Trans. Control Syst. Technol.* **11**, 505–515 (2003)
15. Santhakumar, M., Asokan, T.: Power efficient dynamic station keeping control of a flat-fish type autonomous underwater vehicle through design modifications of thruster configuration. *Ocean Eng.* **58**, 11–21 (2013)
16. Fischer, N., Hughes, D., Walters, P., Schwartz, E., Dixon, W.: Nonlinear RISE-based control of an autonomous underwater vehicle. *IEEE Trans. Rob.* **30**, 845–852 (2014)

Communication Technology and Data Processing



Comparative Analysis of Network on Chip Topologies

Mondal Churantal[✉], C. S. Chandana, Sangavi Siddarth, V. Sanju,
Ramakrishnan Nithin, and Farhanaaz

School of Computer Science and Engineering, REVA University, Bengaluru, India
20010113696.churanta@cse.reva.edu.in, {2008239,sanju.v,
nithin.ramakrishnan,farhanaaz}@reva.edu.in,
R20200785.siddarth@cse.reva

Abstract. Network-on-chip (NoC) is a verbal exchange protocol for incorporated circuit and System-on-Chip (SoC) systems. It correctly integrates a packet-switched network of routers and switches. Compared to conventional communique structures, NoC offers scalability, modularity, high performance, flexibility, and capability performance. NoC is important in present-day chip layout as it addresses issues of performance, scalability, and strength intake. It improves overall performance via the usage of high bandwidth and occasional latency connections. It allows for scalability via validating more than one component and allows for destiny expansion. The purpose of the paper is to analyze the Network on-chip (NoC) layout and examine one-of-a-kind topologies for efficient on-chip verbal exchange.

Keywords: RiCoBiT · Network on Chip · NoC topology · Mesh · Torus

1 Introduction

Chip designs nowadays face many problems due to the fact they're getting more complicated. This happens due to the fact chips should have more parts, be smaller but filled with greater stuff, including exclusive varieties of elements, work faster, use less power, and be made faster. To solve these problems, we need to make smart choices about how we design chips, set up communication between parts, save energy, and make the design process smoother.

The usual way of connecting parts in a chip, called a bus, has some limitations. It can't easily grow to handle more parts, it uses a lot of energy, can be slow, isn't very flexible, and can't handle errors well. Network-on-Chip (NoC) [1] is a better approach because it can grow, work faster, use less power, be more flexible, and handle errors better. It's like having a better road system inside the chip.

This paper discusses NoC and sees how well it works compared to other ways of connecting parts in a chip.

2 Literature Review

Network-on-Chip (NoC) is a crucial communication infrastructure within integrated circuits and System-on-Chip (SoC) designs. It plays a vital role in enabling efficient data transfer and communication within the system. NoC consists of several key components, working together to facilitate this communication.

Routers are the central components of a NoC. They act as intermediaries for routing and switching data packets. When a router receives a packet from a source node, it examines the packet's destination address and decides the best way to forward it. Routers are responsible for establishing communication paths and ensuring that packets reach their intended destinations.

Links, which can be either wired or wireless, form the physical connections between routers in a NoC. These links carry both the data packets and control signals needed for communication between the routers. The quality and characteristics of these links have a significant impact on the overall efficiency and speed of data transmission within the NoC.

Network infrastructures serve as the bridge between the NoC and the processing elements (PEs) or IP cores within the system. They provide the necessary connections and protocol translations to facilitate communication between the NoC and external components. The network involves handling tasks such as sending and receiving packets and adapting protocols to ensure seamless integration of connected components with the NoC.

One distinct feature of NoCs is their packet-based communication paradigm. Unlike traditional systems that use a shared bus, NoCs break data into packets. Each packet contains a portion of the data being transmitted, along with additional control information. These packets are routed from the source to the destination through the NoC.

Compared to shared bus architecture, this packet-based approach offers several advantages. First, it provides enhanced scalability. As the number of components in a shared bus architecture increase, the bus can become a bottleneck. In contrast, NoCs offer multiple parallel communication channels, enabling efficient and concurrent data transfers between various components. This scalability supports the integration of a larger number of components and accommodates the growing complexity of chip designs.

Another benefit of the packet-based communication mode is improved performance. By dividing data into packets, the communication channel provides the facility to transmit multiple packets simultaneously, this helps to reduce the latency and improves overall system performance. The parallel nature of packet-switched communication in NoCs enables faster and more efficient data transmission, which is particularly advantageous for data-intensive applications that require high bandwidth and low latency.

NoCs also offer advantages in terms of modularity and reusability. Separating the communication infrastructure from functional blocks allows for easier integration and reconfiguration of IP cores. This modular design approach enables different components to be developed independently and then integrated into the NoC, facilitating faster design iterations and promoting the reuse of IP cores across different chip designs. It also allows for the customization of NoC topologies to meet specific application requirements.

Further, NoCs contribute to energy efficiency. In traditional bus-based architecture, all components connected to the bus commute for access result in increased passenger consumption. NoCs, on the other hand, use packet-switched communication, which reduces power consumption by minimizing wire length and associated capacitance. Additionally, NoCs enable dynamic routing and power management strategies, optimizing power efficiency and reducing energy consumption in chip designs.

3 Advancements in System-On-Chip Designs

The way computer chips are made has improved a lot over time. Now, we can put different types of cores (like the brain of a computer) on a single chip. But for these cores to work together, they need a good way to talk to each other. This talking is like a traffic system inside the chip. We want this traffic system to be flexible, fast, and able to handle lots of things at once. One solution for this is called “Network-on-Chip” or NoC. NoC is like the road system inside the chip. It helps the cores communicate efficiently. The layout and connections of NoC are called “NoC topology.” There are regular topologies that connect similar cores and irregular ones for different cores with different functions.

Before, we used systems outside the chip for communication, but they couldn’t keep up with the need for speed, size, and power efficiency. On-chip networks like SoC and MPSoC are better because they use less power and work faster than off-chip solutions.

Inside on-chip networks, there are different ways to connect the parts. Some use a shared bus, which is like a simple road, to connect everything. But as the chip gets more complex, this can cause problems like slow communication and using too much power. NoC is better because it uses a network of routers instead of a shared bus. This saves power and works more efficiently.

NoC makes it easy to connect different parts of the chip and ensures they talk to each other properly. It’s good for communication inside the chip, no matter how big or small it is. One common way to set up NoC is by using a mesh, which connects each router to some cores and other routers. This way, data can move efficiently. But when the chip gets really big, the mesh can get complicated. A torus is another way to set up NoC. It connects nodes in a smarter way, which can handle bigger chips, but it can also cause some delay.

In the end, improvements in chip technology have allowed us to use Network-on-Chip (NoC) in System-on-Chip (SoC) designs. NoC helps different cores communicate better, and it’s flexible, fast, and efficient. Different NoC setups like mesh and torus can handle communication and scalability issues. On-chip networks are becoming more popular because they use less power and work better than off-chip solutions. NoC-based designs are becoming important for modern applications and services.

4 Network on Chip Architecture

A typical modular NoC architecture consists of three essential components: network interfaces (NI), switches, and links [4].

The main function of network interfaces [8] is to translate the bus protocol used by the processing elements (PEs) to the network protocol used by the switch. Each IP core must connect to the NoC via NI. Which transforms transaction requests and responses into internal packets. These packets were then flit-sorted by the NI.

The network backbone consists of switches, which are responsible for forwarding packets from source to destination. Switches make it easy to create different topologies, provide arbitrary connections between multiple inputs and outputs, and thus connections to multiple PEs. Moreover, switches provide buffering resources to prevent congestion and increase performance.

Communication plays an important role in drawing connections between NIs and switches, as well as between switches. Connections serve more than physical lines; They can act to fill critical time. The internal structure of the NoC [9] is shown in Fig. 1, where PE stands for Processing Elements, NI stands for Network Interface, and RL stands for Routing Logic

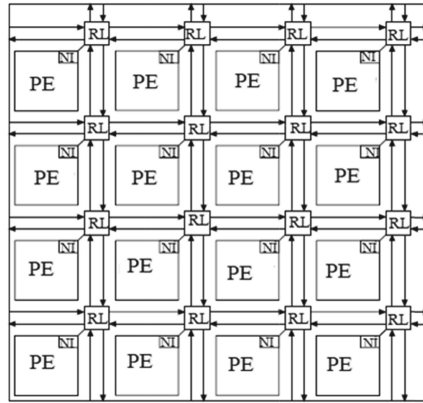


Fig. 1. Internals of NoC

5 Network on Chip Topologies

There are several notable NoC architectures, including mesh, torus, and RiCoBiT [2]. Each of these has unique characteristics that make them suitable for different applications.

5.1 Mesh Topology

Mesh topology [6] is a well-connected network where every node is directly connected to every other node. This type of network is commonly used in wide-area networks

(WANs) and peer-to-peer networks. It ensures that each node has a direct connection to every other node, which makes the network robust and enables multiple paths for communication between nodes. This redundancy enhances fault tolerance, as even if one path fails, communication can continue through other paths. Mesh topology allows for the easy expansion of networks by adding new nodes and connecting them to existing ones, making it scalable (Fig. 2).

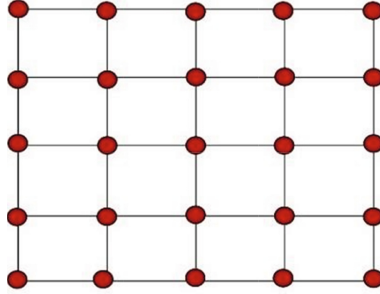


Fig. 2. Mesh Topology

One of the advantages of mesh topology is traffic distribution. Data can be sent simultaneously using different paths, which helps to manage network traffic efficiently. However, as the number of nodes increases, the complexity and cost of implementing mesh topologies also rise significantly. This complexity can be a concern, especially when the number of connections grows exponentially with the increasing number of nodes. There are two main types of mesh topologies: full mesh, where every node has its own connection with all other nodes and partial mesh, in which only some nodes have direct connections to each other.

5.2 Torus Topology

The torus architecture is an extension of the mesh structure, but it takes the network and wraps it around itself to form a toroidal shape. This configuration allows for even more connections and multiple paths between routers. It enhances fault tolerance and reduces congestion in the network. Torus architectures provide good scalability and support both short and long-distance communication.

However, the complexity of wiring in Torus topologies increases with the number of dimensions in the torus [6]. In other words, as you add more layers to the toroidal shape, the wiring becomes more intricate. Nonetheless, torus topologies are valuable for their ability to handle high communication demands and provide redundancy, making them suitable for a wide range of applications.

5.3 RiCoBiT Topology

RiCoBiT, which stands for Ring Connected Binary Tree [3], is a network topology used in Network-on-Chip (NoC) systems. It is a special kind of arrangement that combines the strengths of both ring and binary tree topologies.

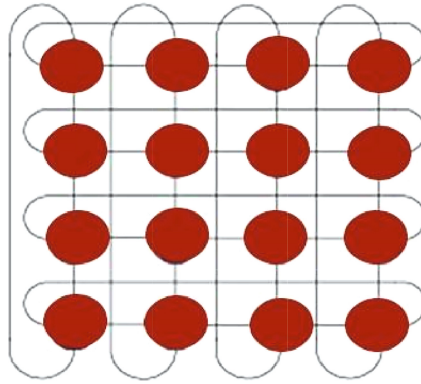


Fig. 3. Torus Topology

As you can see in Fig. 3, the nodes form a binary tree, with the root node connecting to all others. These nodes are then grouped into rings, each containing two nodes. The rings are connected to each other hierarchically.

This hierarchical setup is useful for efficiently routing data packets. RiCoBiT [7] has several advantages compared to other NoC topologies. It has a low maximum hop count, meaning that data packets can travel from one node to another in a relatively small number of steps [5]. It also has a low average hop count, which means that, on average, the distance between any two nodes is relatively short. This can significantly improve the performance of the NoC system (Fig. 4).

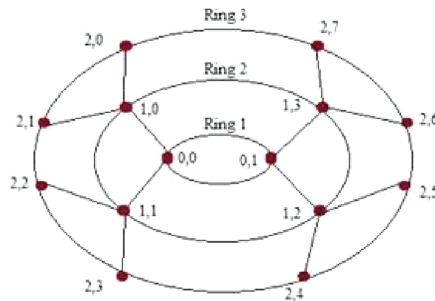


Fig. 4. RiCoBiT Topology with three rings

Moreover, RiCoBiT topology is scalable. If the number of nodes in the system grows, you can easily expand the topology by adding more rings. This makes it a great choice for NoC systems that need to support a large number of nodes.

RiCoBiT topology has been proven effective in various NoC applications, including high-performance processors, embedded systems, and wireless sensor networks [10]. It offers a balanced combination of efficiency and scalability, making it a valuable choice in the world of Network-on-Chip systems.

6 Flow Control Mechanisms for NoC

Developing an efficient routing algorithm for a Network-on-Chip (NoC) necessitates a radical draw close of various routing techniques and drift manage mechanisms well suited with NoC architectures. These mechanisms embody shop and forward routing, virtual reduce-via routing, and wormhole routing, all commonly employed in NoCs.

Store-and-ahead routing [4, 10] is the maximum sincere among the various go with the flow manipulate mechanisms in NoC architectures. It entails briefly storing data packets at every node earlier than forwarding them to the next. Despite its simplicity, this approach has drawbacks which include excessive latency and the requirement for substantial buffer ability.

To cope with the latency and buffer size troubles related to shop-and-ahead routing, the virtual cut-via method became devised for NoC architectures. This technique permits on the spot transmission of packets upon arrival at a router, eliminating the need for temporary garage. Consequently, latency is significantly reduced, inspite of a smaller buffer size, as packets are not held at intermediate nodes.

Wormhole routing is any other method involving the division of information packets into smaller gadgets known as “float control digits” or “flits” [5]. The transmission path is determined by way of the primary flit, and subsequent flits comply with the same course.

In assessment to other routing methods, wormhole routing offers numerous blessings. It allows extra efficient utilization of community assets, reduces congestion, and improves standard network overall performance by way of permitting parallel transmission of flits, allowing the simultaneous transmission of more than one packets.

7 Comparative Analysis

When assessing the efficiency of different topologies for NoC architectures, several factors such as the maximum hop count, average hop count, count of wire segments, and wire length are typically considered [4].

A. Maximum Hop Count

The maximum hop count is the greatest distance of all the shortest paths in a network, which is a key determinant of communication latency. As this count increases, latency also increases, leading to a decrease in performance.

B. Average Hop Count

The average hop count is calculated by taking the mean of the hop counts for all possible pair combinations. This provides a holistic view of the performance of the system.

C. Count of Wire Segments

An increase in the number of wire segments leads to an increase in the required routing area. This results in a decrease in efficiency, which negatively impacts the network’s ability to facilitate packet communication.

D. Wire Length

It is very important to note that both the length of the wire and the number of segments has a significant impact on network performance. The efficiency of packet crossing across the network is in inverse relation to each other.

In evaluating the performance of on-chip network topologies, we consider several parameters: maximum hop count, average hop count, wire length, and the number of wire segments. The maximum hop count refers to the number of intermediate nodes a packet needs to traverse in all the shortest paths to reach its destination [4]. This parameter is also indicative of communication latency. The average hop count is determined by the number of hops required for communication between all pairs of source and destination nodes. Other performance parameters include wire length and the number of wire segments. An increase in these parameters inversely affects the performance of the topology. According to a study, RiCoBiT outperforms mesh and torus topologies based on these performance parameters [9].

Table 1. Evaluation of Performance differences amongst Various Network Topologies Using Qualitative Analysis.

Performance Parameters	Topologies		
	<i>RiCoBiT</i>	<i>Mesh</i>	<i>Torus</i>
Maximum Hop Count	Minimum	Maximum	Moderate
Average Hop Count	Minimum	Maximum	Moderate
Wire length	Moderate	Minimum	Maximum
Number of Wire Segments	Same	Same	Same

Table 1 gives an examination of different community topologies, thinking about performance parameters together with maximum hop be counted, average hop depend, twine duration, and the range of wire segments. Through the carried out evaluation, it is able to be inferred that RiCoBiT outperforms different topologies in phrases of performance.

8 NoC Routing Algorithms

There are two primary types of routing algorithms for Network-on-Chip (NoC): adaptive and oblivious. This classification is illustrated in Fig. 5.

When it comes to routing packets in Network-on-Chip (NoC) systems, there are two most important styles of oblivious routing algorithms: deterministic and stochastic [4, 10]. Stochastic algorithms randomly ship packets, at the same time as deterministic ones always observe a predetermined route for delivery. Choosing among those relies upon at the particular desires of the utility and the preferred performance/complexity stability.

Minimal turn routing [5] is a honest yet effective oblivious routing algorithm in NoC architectures. It involves figuring out the direction for packet routing at each step, with examples like XY routing, pseudo-adaptive XY routing, and surrounding XY routing.

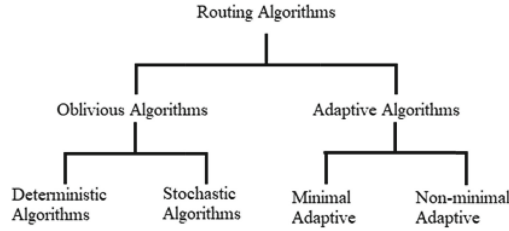


Fig. 5. Classification of routing algorithms.

On the opposite hand, stay lock-free routing algorithms, which include west-first routing, northlast routing, and poor first routing, goal to save you deadlocks, making sure that packets don't get caught in an infinite loop throughout routing.

Selecting a routing set of rules depends on different factors like community topology, site visitors patterns, and overall performance necessities. Careful attention ensures that the chosen technique aligns with the particular traits of the community, optimizing its performance and reliability.

In the NoC realm, deterministic routing algorithms are massive, supplying a established technique to packet routing [4, 5]. Examples consist of shortest course routing, source routing, vacation spot-tag routing, and topology adaptive routing. Shortest path routing targets to discover the maximum direct course between supply and vacation spot nodes, minimizing the packet's hops. Source routing empowers the supply node to chart the whole packet direction for unique manage. Destination-tag routing appends a tag to the packet specifying its intended destination, facilitating green and direct shipping.

Topology adaptive routing algorithms take a dynamic approach, adjusting the routing path in response to changing network conditions and traffic patterns. These adaptive strategies aim to optimize packet transmission while circumventing congestion issues. When choosing a deterministic routing algorithm for a given NoC, it's crucial to consider factors such as the network's size, the nature of traffic patterns, and the performance requirements. This thoughtful selection process ensures that the routing algorithm aligns with the network's unique characteristics and operational goals, thereby enhancing its overall efficiency and reliability.

The underlying premise of stochastic routing algorithms [4, 9] in NoC is that every packet will ultimately get to its destination. Stochastic routing, in contrast to deterministic routing, randomly chooses a path depending on traffic patterns and network conditions rather than adhering to a predetermined path. Although this method adds a degree of uncertainty to the packet transmission process, it may result in a more effective use of network resources.

A greater sophisticated shape of routing in Network-on-Chip (NoC) is obtainable with the aid of adaptive routing algorithms [3, 4], which take into account traffic patterns and community conditions whilst making routing choices. In situations where network situations are dynamic and continuously converting, adaptive routing algorithms show to be a better desire than oblivious routing. Examples of adaptive routing algorithms consist of hot potato routing, flip around routing, strange-even routing, minimal adaptive routing, and completely adaptive routing.

Fully adaptive routing dynamically adjusts the routing path based totally on packet precedence and congestion stages. In contrast, minimal adaptive routing selects the course with the fewest turns. Hot potato routing objectives to minimize latency via directing packets to the closest on hand node. Turn round and peculiar-even routing are built at the idea of fending off deadlocks.

The choice of particular adaptive routing algorithms is influenced with the aid of elements inclusive of community length, visitors patterns, and overall performance necessities. These algorithms recall the dynamic nature of the community and adjust their routing decisions accordingly, contributing to progressed efficiency in the face of changing situations.

9 Future Scope and Research Challenges

The primary objective of this assignment is to develop a faster processor. In our future plans, we aim to expand the capabilities of the Network-on-Chip (NoC) simulation tool to support a wider range of network designs. Additionally, we intend to adapt the simulator to be compatible with multi-core processors. To mitigate the delays associated with memory access, we plan to implement a technique known as multithreading. Multithreading enables the concurrent execution of multiple threads, which can significantly accelerate project execution and enhance overall performance. Our ultimate goal is to minimize latencies and boost the processor's speed through the implementation of multithreading.

10 Conclusion

In conclusion, our review and comparative analysis of Network-on-Chip (NoC) topologies have provided valuable insights into their unique characteristics, advantages, and trade-offs. Mesh and Torus architectures offer excellent scalability and low latency for short-distance communication. Tree architectures provide hierarchical organization and fault tolerance, while Binary Tree architectures offer higher bandwidth. These findings underscore the importance of selecting the appropriate NoC topology based on specific application requirements and design constraints.

NoC has emerged as a vital communication infrastructure for modern chip designs, addressing the limitations of traditional bus-based architectures. By employing packetbased communication, parallelism, and modular design approaches, NoCs offer scalable and efficient solutions. They enable the integration of a larger number of components, support faster data transfers, promote modularity and reusability, and optimize power consumption. The significance of NoC in facilitating the development of advanced and feature-rich chip designs cannot be overstated.

The choice of the right NoC topology has a significant impact on system performance and design efficiency. This decision directly affects crucial metrics such as scalability, latency, throughput, fault tolerance, and power consumption. A well-designed NoC topology can enhance overall system performance by minimizing communication latency, reducing contention, and maximizing the use of available bandwidth. It also streamlines design efficiency by facilitating modular design and component

integration. Therefore, thoughtful consideration of the NoC topology is essential for achieving optimal system performance, meeting design requirements, and ensuring efficient communication within chip or System-on-Chip (SoC) designs.

Acknowledgements. Our heartfelt thanks go out to the Chancellor, Vice Chancellor of REVA University, and the Director of the School of Computer Science and Engineering at REVA University in Bengaluru. Their invaluable support and guidance have played a pivotal role in ensuring the successful completion of this project.

References

1. Dally, W.J., Towles, B.P.: Principles and Practices of Interconnection Networks. Morgan Kaufmann (2004)
2. Baby, N., et al.: Network on chip simulator: Design, implementation and comparison of Mesh, Torus and RiCoBiT topologies. In: 2016 2nd International Conference on Next Generation Computing Technologies (NGCT). IEEE (2016)
3. Sanju, V., Chiplunkar, N.: RiCoBiT-ring connected binary tree: a structured and scalable architecture for network on chip based systems. In: 2015 International Conference on Computing and Network Communications (CoCoNet). IEEE (2015)
4. Ramakrishnan, N, Sanju, V.: Analysis of Network on Chip Topologies. In: 2023 International Conference on Applied Intelligence and Sustainable Computing (ICAISC). IEEE (2023)
5. Uma, R., Sarojadevi, H., Sanju, V.: Routing of flits in parallel input interface scenario in a generalized network-on-chip framework using wormhole flow control algorithm. In: Shetty, N.R., Patnaik, L.M., Nagaraj, H.C., Hamsavath, P.N., Nalini, N. (eds.) Emerging Research in Computing, Information, Communication and Applications. Lecture Notes in Electrical Engineering, vol. 790. Springer, Singapore (2022). https://doi.org/10.1007/978-981-16-1342-5_52
6. Sanju, V., et al.: A performance study of 2D mesh and torus for network-on-chip-based system. In: Proc. Int. Conf. Emerging Research in Computing, Information, Communication and Applications, Bangalore, India (Elsevier, Amsterdam, 2013) (2013)
7. Satish, J. A., et al.: RiCoBiT-A topology for the future multi core processor: a concept analysis and review of literature. In: 2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE). IEEE (2020)
8. De Micheli, G., et al.: Networks on chips: from research to products. In: Proceedings of the 47th Design Automation Conference (2010)
9. Uma, R., Sarojadevi, H., Sanju, V.: Network-on-chip (noc)-routing techniques: a study and analysis. In: 2019 Global Conference for Advancement in Technology (GCAT). IEEE (2019)
10. Kalita, A., Ray, K., Biswas, A., Hussain, M.A.: A topology for network-on-chip. In: 2016 International Conference on Information Communication and Embedded Systems (ICICES), pp. 1–7. Chennai, India (2016). <https://doi.org/10.1109/ICICES.2016.7518838>



Adaptive Handover Decision and Load Balancing Optimization in LTE Networks: A Fuzzy Logic System Perspective

Divya Mohan¹ , A. Geetha Mary² , K. Priyadharshini¹ , and Rashmi Naveen³

¹ Department of Computing Technologies, SRMIST Kattankulathur Campus, Chennai, India
divyamohan797@gmail.com

² School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India

³ NMAM Institute of Technology, Nitte, Udupi 574110, India

Abstract. LTE provides fast and continuous handover from one BS (Base Station) to another to achieve the user's desired Quality of Service (QoS). The decision must rapidly be taken between handover and load balancing to attain efficient QoS. Various algorithms are implemented into the system to make this decision, carrying out planning, processing and installation steps. This requires a maximum number of manual involvements and results in errors. Hence, to make the systems error-sensitive and to reduce manual involvement, LTE networks implement SON functions to operate decision-making automatically. Where Key factors of system performance include handover and load balancing. A new efficient optimization technique for LTE networks based on a fuzzy logic system is proposed in this study for decision-making. Fuzzy logic considers several network parameters and takes handover or load-balancing decisions. This decision is made according to network status and also considering its parameters. A fuzzy logic system requires an optimal solution at certain stages; hence, the system uses an optimization algorithm to optimize the handover and load-balancing decision. To satisfy the increased demand for a high data rate while maintaining the QoS, a dense Long-Term Evolution (LTE) cell is necessary. This network implements specific algorithms to the network for performing handover. This research proposes an efficient network fuzzy logic algorithm to achieve continuous handover. Also, this study analyses the possibility of a successful handover or load balancing made using this fuzzy logic system. When a fuzzy logic system cannot make an appropriate decision, it uses an optimization algorithm and overcomes the conflict, as mentioned above. The outcome of the proposed system model shows an increased system performance and mobility for users; hence, it hopefully reaches the desired QoS.

Keywords: LTE · BS · 3GPP · SON · Handover · Fuzzy logic · Load balancing

1 Introduction

Handover is a transfer of a network connection from one base station to another. While switching from one base station to another, the user should not experience any interruptions. This gives the user a continuous, seamless connection to the network without

knowing to which base station the user was connected [9]. Hence, LTE is utilised in 3GPP to balance the increased demand for network connection. This network provides services with the installation, planning, testing, monitoring, mitigation and evaluation [10]. Handover [11] is a mandatory process in the LTE/4G environment, precisely when a high-speed continuous connection is required [12]. 3GPP uses an LTE network to increase the capacity and speed of a network [13]. The complexity and size of the LTE networks get maximised due to the number of users. The users always require a continuous network connection.

The applications like video calls and streaming video require a breakless connection [14]. In turn, it needs a high throughput with the network connection. The user must be able to utilise the network connection even in a high-speed moving environment. Users must comfortably use the network's services at any location with increased mobility. This is the reason why the user node requires continuous handover.

Load balancing [15] is a reverse proxy method that shares the network and applies it across different base stations at the time of traffic. This technique shares the workload across n number of BS; this, in turn, decreases the overall burden of the base station. This technique distributes requests and hence saves the memory space. It also offers excellent flexibility in network connection. Load balancing [16] is applied to the system where increased concurrent user capacity and high overall reliability are required. Based on the network requirement, load balancing is done. To increase the response time for a BS, some additional functions are needed that include throughput management, i.e., when there is a rapid increase in traffic that provides for both outbound and inbound requests of a BS, load balancing handles it with an appropriate solution. The load from a BS must be distributed between nodes, which is necessary for balancing the traffic [17]. When it comes to a security concern, the node, when facing denial of service, which is a negative impact that includes unwanted requests or another call that compromises the use of the network, the load balancing will be invaluable. This gains an advantage with VPN solutions, Firewalls and many other features [1]. Continuous connectivity of networks can be achieved through load balancing that connects more BS continuously. This tends to have various problems, including limited connection between the BS and nodes [19]. The load balancing technique provides a solution by sending the request from a BS to the appropriate cell.

The previously proposed system models are complex and also require manual deployment. It becomes costly by nature and insensitive to error, resulting in unsatisfied customers. 3GPP utilises the SON network for cost reduction with improved planning, maintenance and deployment. Network planning, maintenance, optimisation and configuration are deployed in SON [20]. Load balancing and handover are the key assistants of a SON network, as the SON network serves the load from a congested one to neighbouring cells with load balancing. LTE network optimises parameters using two or more SON functions [21]. A negative impact on the network is caused by a conflict and dependency state during optimising a single parameter with two SON functions. The parameters, namely SNR (Signal-to-noise ratio), Throughput and Efficiency of the network, are analysed for handover optimization. These parameters are changed to increase the network's speed for the user. The key motivators for SON are the increasing number and complexity of network characteristics. The rapid expansion of wireless networks

has resulted in the parallel operation of 2G, 3G, and 4G infrastructures, and a constantly rising number of base stations must be configured and managed with as little human intervention as possible.

The SON process includes self-configuration, where an IP address is allocated to the new user. Gateway is configured for the new user so that the BS can exchange IP packets with other mobile nodes. The new Base station provides information, including type, hardware, etc., to the Self-configuration Subsystem for authentication. Necessary software and configuration data are downloaded from the Self-configuration Subsystem. The new connection is configured based on the transport and radio configuration data. The self-optimization mechanism uses base station and performance measurements to autotune the network. This process operates in an operational mode, which begins when the RF interface is turned on. The self-optimization procedure gathers measurement data from the base station. Then, with the assistance of an external optimization tool, it auto-tuned the configuration data to optimize the network. [22].

An efficient handover optimizations technique for LTE networks based on fuzzy logic algorithm [23]. The mean number of handovers with system delay and system throughput depends on user speed. The features desirable for handover are low speed and QoS. Overhead signaling should be minimized during handover. As a result, there is opportunity to minimize complexity and improve network quality of service (QoS) by minimizing call blocking and dropping. This research provides a unique approach for load balancing and handover optimization based on collaborative optimization. This technique is based on fuzzy logic and the Sea Lion Optimization technique (SLnO). The fuzzy logic action is based on previous actions, and its influence and performance on the network are monitored using KPI (Key performance indicators). To achieve the best results, these actions are optimized using a proposed method. Based on collaborative optimization, this approach minimizes the coordination complexity of two SONs.

In this study, Sect. 2 describes the literature review, Sect. 3 briefs on the problem statement and motivation of the research, Sect. 4 briefs about the proposed model, and Sect. 5 shows the simulation results. At last, Sect. 6 contains the overall result and conclusion.

2 Related Works

Akyildiz et al. [1] discussed an approach in LTE- advanced network with the backwards-compatible enhancement of LTE release eight, which is fully specified in 3GPP release 10. This paper describes the leading technologies in this network and validates its advantages and challenges. Hasan et al. [2] have proposed an alternative algorithm for load balancing and mobility in a small cell network. For enhancing the network performance, like the service throughput and drop rate. Lack of resources causes overloading of cells, which cannot select the required source even when the nearby cells can share the networks. However, the proposed algorithm does not have information on edge UEs first to hand for load balancing. Li et al. [3] have submitted a technique combining optical switching and wireless technology for transforming physically moving base stations into logically moving cells. The system's relay stations help improve the assist handovers and signal strength—however, frequent interference along with over-signalling is presented

among RSs, BSs and HSTs. Bejerano et al. [4] have intimated a model to achieve a better trade-off between HPO and LB. In this proposed model, a co-satisfaction indicator (RCSF) was presented that consists of an impact on the handover success and user satisfaction ratios. However, this model cannot be implemented practically and also, at the time of simulation, the optimisation values of HPO and LB are 30 s and 1 s, respectively. These results showed that the proposed system model was unable to implement.

Bejerano et al. [5] have proposed a design aspect of DMFO and its performance via extensive simulations and analytical simulations. Subsequently, it also presented the proposed technique lowers the overhead of the signal at the time of equating with the requirements of the QoS. With various mobility patterns, the given model results in failure. For example, the proposed method identifies the signal-to-noise ratio that increases the spectral efficiency for transferring signals, adding to a low reporting overhead. Nguyen et al. [6] proposed a distributed mobility robustness optimization algorithm for lowering handover failures during radio link failures, adjusting offset and TTT (Time to Trigger) parameters. However, an impact was created in all the nearby cells, so the algorithm could not differentiate between various cell environments. The proposed algorithm optimizes the parameters of the handover. However, the proposed method performed handover before the TTT window expired. The result validated that the performance of the proposed method degraded with the maximization of Handover when the TTT was little. Anyway, it was not helpful for LTE networks. Chen et al. [7] have proposed an approach for 3GPP LTE for increased system output and decreased packet delay. This proposed approach used two handover management notations to avoid the ping-pong effect. The algorithm proposed had different issues at the time of handling handover. However, there were limitations like high carrier interferences, high data loss and high disruption time. Awada et al. [8] have proposed an algorithm for optimizing inter-RAT handover. The proposed algorithm is simulated on both LTE and 3G networks, in which the parameters were adjusted. However, the proposed algorithm requires an extension for an improved handover offset.

3 Problem Statement and Motivation of the Research

3GPP utilize an LTE network to meet the maximized demand for wireless data service. SON is used to configure and optimize the wireless connection to reduce manual involvement. Adjusting the parameters depending on the network status increased system performance. On the other hand, it shows specific negative impacts. For instance, if a user moves from one location to another, the mobility performance and the handover optimization must be high, and load balancing must happen effectively. Handover is the technique for switching user network connection from one BS to another, and load balancing is sharing the network from congested BS to neighboring ones. Various existing techniques were used to make handover decisions and to implement load-balancing techniques. The previously used methods could not resolve high call drop rates, signal fading and ping pong effect. Moreover, implementing the existing algorithm also resulted in a high signal cost and increased memory usage. Thus, the previously proposed algorithms could not handle the issues while taking handover and load balancing decisions in the configured system model.

For increased connectivity, the LTE network combines two or more SON network functions and automates the handover and load balancing to reduce manual involvement and make the system error-sensitive. The decision-making process is automated using a fuzzy logic algorithm, which is efficient and gives relatively high system performance. This system model was proposed in previously existing methods. When a conflict criterion was raised, the fuzzy system could not decide to transfer the network using handover or load balancing. No optimal solution is provided to resolve this conflict; hence, the user may experience an interruption in the network connection. This proposed model makes use of an optimization algorithm to overcome this issue. Therefore, this system presents a novel solution for the conflict state in the fuzzy system and gives an optimal solution to decide on handover or load balancing.

4 Proposed Model

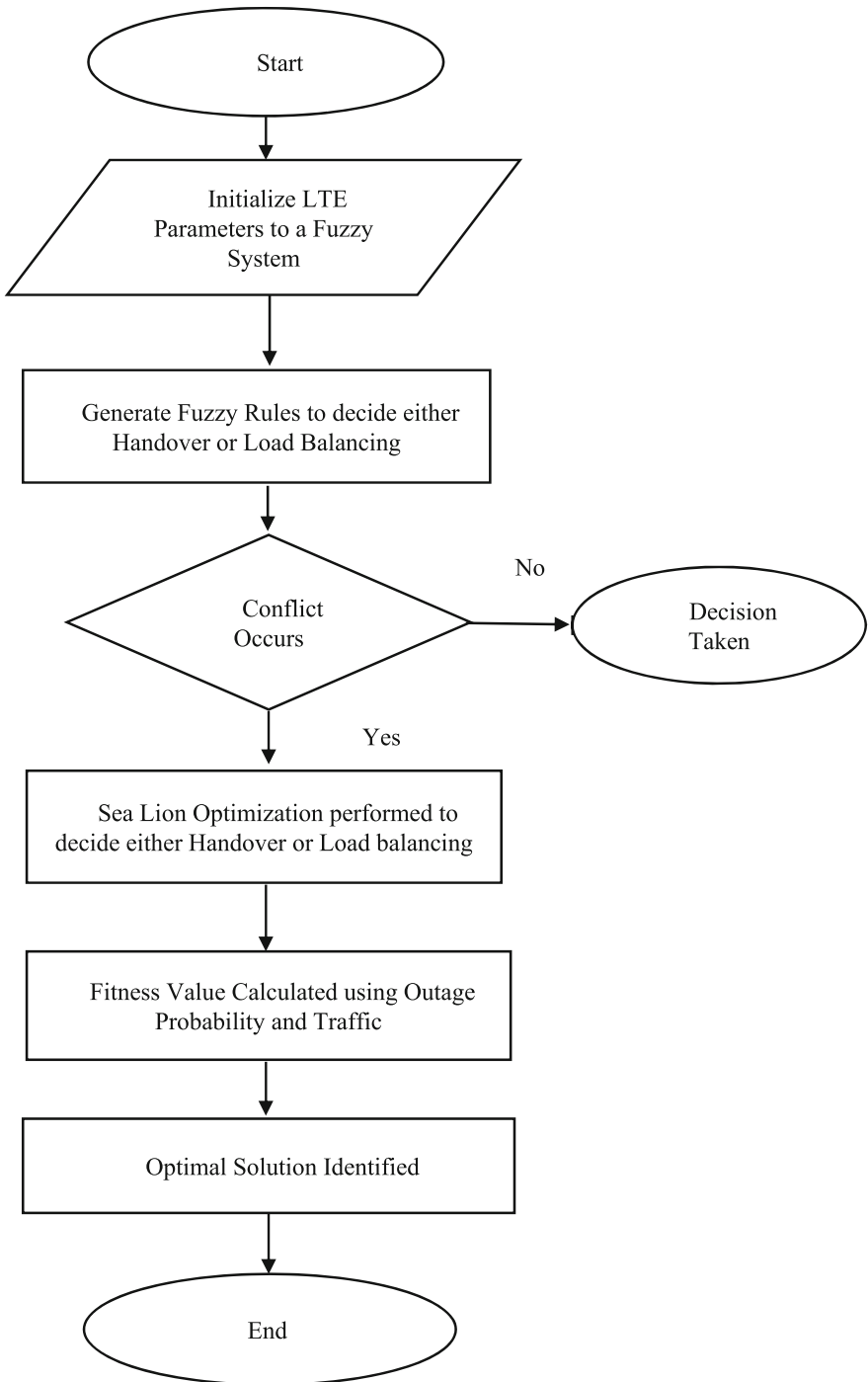
This study proposes a system model for making decisions between handover and load balancing using a fuzzy logic algorithm. A powerful setup is created for handoff procedures in LTE communication. The data can initially be obtained from the network service providers; this [24] data is analyzed based on our process. This dataset provides information on user handoff-related data attributes for analysis. In addition to this information, based on the generated rules, the fuzzy logic makes decisions even though a conflict state occurs at a particular rule. This happens while automating handover and load balancing using SON functions. This model provides optimized decision-making by using an algorithm that takes control of decision-making between handover and load balancing. The steps for executing the proposed system model algorithm are shown in the figure below (Fig. 1).

Figure 1 shows the flow chart of the proposed model where the parameters of the LTE network are initialized and fed to the input of the fuzzy system. Based on the rules generated by the fuzzy system, an output based on which handover decision or load balancing is made. The rules caused switches to either handover or load balancing state. A fuzzy system cannot find an optimal solution when specific rules give a conflict condition. The use of a Sea Lion optimization algorithm optimizes this state. The algorithm's fitness function takes outage probability and traffic of the network as input and gives an outcome. Based on the fitness value, the decision is made.

4.1 Proposed Fuzzy System

Fuzzy is a term that refers to values which are not fixed or vague. True or false statements cannot be determined when the situation is encountered. Fuzzy logic has more flexibility and can be utilized efficiently. In this proposed approach, a fuzzy system is used to decide whether to take a handover decision or load balancing. The fuzzy system makes decisions based on the generated rules. The central part of the fuzzy system is based on the generated rules. The dynamic behavior of the fuzzy system is more efficient in decision-making. The fuzzy system is rule-based as an IF-THEN strategy.

The proposed work model is based on fuzzy logic considering CBR (Call Block Rate), BS location, Load and speed to prevent the connection and decide on handover or

**Fig. 1.** Flow chart of the proposed model

load balancing. The single output from the fuzzy system generates rules that connect the network to handover or load balancing. Where H_y Handover optimization value output and L_y denotes load balancing offset value. For seamless connectivity of the network, the handover value must be high. The number of rules is based on a fuzzy input set. The actions are generated according to the fuzzy set in the table below (Fig. 2).

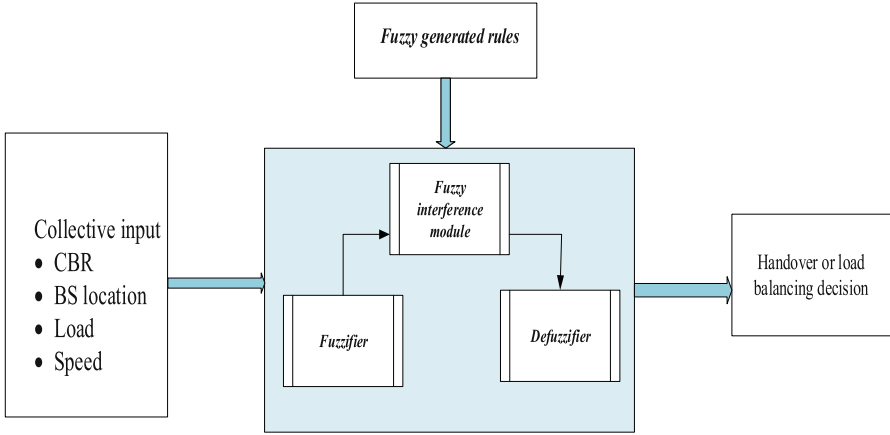


Fig. 2. Architecture of the fuzzy logic system in the proposed system model

From the Table 1, it is understandable that,

- From Rule 2, if CBR is LOW, BS location is LOW, Load is LOW, and Speed is LOW, then HANDOVER Appears.
- From Rule 3, if CBR is LOW, BS location is LOW, Load is HIGH, and Speed is LOW, then HANDOVER Appears.
- From Rule 9, if CBR is HIGH, BS location is LOW, Load is LOW, and Speed is LOW, then LOAD BALANCING Appears.
- From Rule 10, if CBR is HIGH, BS location is LOW, Load is HIGH, and Speed is LOW, then LOAD BALANCING Appears.
- From Rule 15, if CBR is HIGH, BS location is HIGH, Load is HIGH, and Speed is LOW, then LOAD BALANCING Appears.

The fuzzy system faces a conflict state when executing Rules 8, 12 and 16. At this stage, the fuzzy system could not decide between handover and load balancing. Also, CBR is low at the handover decision and high at the time of load balancing. When BS location does not impact the handover decision and the load balancing decision. When the Speed and load are high, and CBR is low, a conflict state occurs for the first time. Similarly, conflict occurs for the second time when load and speed are high, and BS location increases. Likewise, conflict occurs when the load and speed are high, whereas CBR and BS locations are also high. At the conflict state, the connectivity of the network decreased. Hence, this study proposed an optimization algorithm to find the decision for this conflict state.

Table 1. Rules generated by fuzzy system

Rule No	Input1	Input 2	Input 3	Input 4	Output
	Call Block Rate	BS location	Load	Speed	Decision making
1	Low	Low	Low	Low	Handover
2	Low	Low	Low	High	Handover
3	Low	Low	High	Low	Handover
4	Low	Low	High	High	Handover
5	Low	High	Low	Low	Handover
6	Low	High	Low	High	Handover
7	Low	High	High	Low	Handover
8	Low	High	High	High	Conflict
9	High	Low	Low	Low	Load Balancing
10	High	Low	Low	High	Load Balancing
11	High	Low	High	Low	Load Balancing
12	High	Low	High	High	Conflict
13	High	High	Low	Low	Load Balancing
14	High	High	Low	High	Load Balancing
15	High	High	High	Low	Load Balancing
16	High	High	High	High	Conflict

4.2 Optimization Technique

This proposed model uses a novel Sea Lion Optimization algorithm method to overcome the conflict state in the fuzzy system. This algorithm was proposed based on the Sea Lion's hunting behaviours. One of the most intellectual creatures is the sea lion. The sea lions live in large colonies with thousands of animals. [25]. The ability of sea lions to adapt swiftly to fish movements is their most important attribute [27]. In other words, they can detect fish and quickly react to attract them to shallow water along the beach or the surface of ocean. Furthermore, these creatures possess remarkable senses which enable animals to locate prey, especially fish, even in the dark of the sea. Their gazes are forward-facing, letting them to focus on their prey easily. They can, for example, expand their pupils to let in plenty of visible light for good undersea vision [28]. When there are a lot of fish around, hunting in groups of sea lions increases the odds of getting more prey. Sea lions typically chase each other as gathering their meal into small balls and capturing each other's prey on the margins [28–30]. The attributes of the sea lion's tracking, encircling, and attacking prey are mathematically described in the proposed model. This method is used to settle conflicts and make decisions about handover and load balancing.

4.2.1 Initialization

In the initialization phase, the input parameters are outage probability and throughput. Outage probability is the receiver power signal obtained, and the throughput is the quantity of data transferred from one BS to another during congestion. It is measured in Gbps (Gigabytes per second). The input parameters are assigned as O_{pty} as Outagerobability and O_t is denoted as Throughput.. Also, the output from the fuzzy logic system is given to the fitness value calculation (Fig. 3).

4.2.2 Fitness Function

The fitness function is calculated by taking outage probability and traffic measure or throughput as input, and the output value decides whether the state is handover or load balancing. The fitness function retains a maximum output value to decide handover or load balancing. If not, then the function repeatedly calculates within a time slot towards a goal of reaching maximum value. It can be calculated from the formula given below:

$$F_n = \frac{O_{ptry}}{O_T} \quad (1)$$

Here, the outage probability can be derived from the equation below,

$$O_{pty} = \left(1 - \left(\frac{1}{\Sigma(SNR)} + 1 \right) \right) \quad (2)$$

SNR is Signal to noise ratio, which can be defined as the signal's power to the noise signal's ratio. From Eq. (3), the SNR value can be calculated and using the SNR value, the outage probability can be calculated by using Eq. (2)

$$SNR = \frac{U_l}{P_l} \times m \times N \quad (3)$$

Here, U_l Denotes uplink value and P_l denotes Packet loss, and N represents initial Noise Average value. Throughput can be measured as the number of data units processed in a given time slot. This is implemented widely in systems ranging from various aspects of the computer and System to network Organizations. Throughput here is termed for traffic of the network and can be calculated as

$$O_T = \Sigma(B_{sl} \times B_W \times \text{Log}2(1 + B_{sl} \times SNR)) \quad (4)$$

B_{sl} is the notation for finding the exact BS location and *represents* B_W represents Bandwidth. The fitness value calculates the handover or load balancing state decision. If the decision is made, the function gets terminated, or the fitness function gets repeatedly executed.

4.2.3 Updating the Current State of the Fitness Solution

The fitness function output updates the current state from conflict to the handover or load balancing state. The updation of the current process needs a common feature used

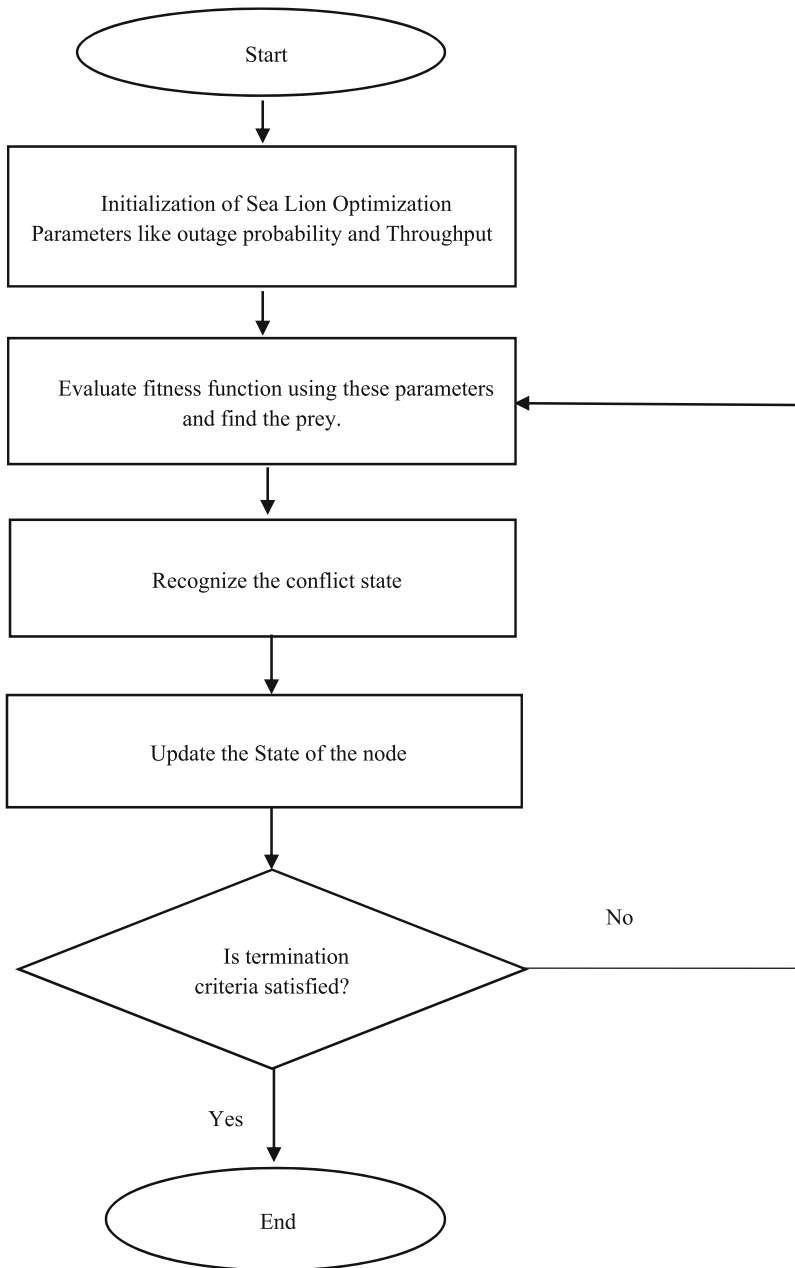


Fig. 3. Flow chart model of Sea Lion optimization algorithm for proposed work

in this algorithm. This is divided into two stages: exploitation and exploration. The suggested algorithm is a mathematical model inspired by sea lion hunting behavior. These organisms can exist alone or in groups. Furthermore, these sea lions have different

phases of hunting behavior such as tracking and racing the prey with their whiskers, summoning other subgroup members, pursuing and surrounding the prey, and attacking the prey. SLnO, like other optimization approaches, comprises two phases: exploitation and exploration.

Exploration is defined as a comprehensive look for the best solutions. Furthermore, exploitation was linked to less searching.

Exploitation Phase

In the phase of exploitation, the joined members modify their locations in light of the search agent's best position. In the exploration phase, on the other hand, the locations of the joined members (search agents) are updated in accordance with the position of the selected sea lion that has been chosen randomly.

It consists of connecting a constrained area of searching space. This hopes to give a good solution termed as 'S'. This phase then intensifies the search in the neighbouring BS to find the optimal solution. It can also be termed a local search. At first, the search agents update their current position about a frequently chosen search agent. Also, from the second instant onwards, the search agent updates their position concerning the optimal solution as far as possible. A rapid search agent totals the value $|C| < 1$ set to the exploitation phase. The existing phase's position was updated using the optimisation technique in this phase.

$$\overrightarrow{Dist} = abs(2\overrightarrow{B}.\overrightarrow{SLrnd}(t) - \overrightarrow{SL}(t)) \quad (5)$$

where $\overrightarrow{SLrnd}(t)$ is used here to point a randomly selected sea lion from the present population. It should be stressed that when the vector " $\overrightarrow{C}$ " is bigger than one, this equation is used for detecting the global optimal solution. Because of that, this algorithm is considered as being a global optimizer.

$$\overrightarrow{SL}(t+1) = \overrightarrow{SLrnd} - \overrightarrow{Dist}.\overrightarrow{C} \quad (6)$$

The existing phase's position was updated using the optimisation technique in this phase.

$$\overrightarrow{W} = \left| \overrightarrow{D}.\overrightarrow{y} * (t) - \overrightarrow{y}(t) \right| \quad (7)$$

where t is the iteration time, $\overrightarrow{y}(t)$ is the current position and D is the coefficient vector, the value of $\overrightarrow{y} * (t)$ gets updated with each iteration.

$$\overrightarrow{D} = 2.r \quad (8)$$

Here, r is the random vector. The value of r varies between 0 and 1. The vector value decreases from 2 to 0.

Exploration Phase

It consists of connecting a much greater area of the search space and the expectation of identifying another encouraging solution that can also be refined. This process then

differentiated the search which helped avoid getting trapped in a local optimum. In the wild, sea lions hunt for prey by using their whiskers and swimming zigzagging. As a result, in this investigation, is used with random values. If is higher than one or less than one, sea lions are forced to migrate away from the intended prey and the sea lion's leader. As a result, sea lions are forced to seek for alternative prey. Moreover, it was similar to the exploitation phase. A rapid search agent selects the more excellent $\vec{C} > 1$ then the exploration phase is triggered. This phase mainly uses bubble net attacking criteria.

The vector " $\vec{C}$ " in this equation is reduced linearly from "2" to "0" throughout the expanse of iteration steps for the reason that this reduction drives the leader of the sea lions into moving in the direction of the current target prey and encircle them.

- If the value of " $\vec{C}$ " is less than one, then the search agent is moving in the direction of the prey and the other search agents should adjust their locations according to that.
- If the value of " $\vec{C}$ " is greater than one, then the search agent is moving away from the prey.
- If the value of " $\vec{C}$ " is equal to zero, then the optimal solution has attained which means that the algorithm terminates at this point. However, if the value of " $\vec{C}$ " is greater than one or less than a negative one, the search agents will move obliquely away and search for a new cluster to join it.

$$(t + 1) = \begin{cases} \vec{y} * (t) - \vec{D} \vec{E} & \text{if } \vec{C} < 1 \\ \vec{D} . e^{bl} . \cos 2\pi l + \vec{y} * (t) & \text{if } \vec{C} \geq 1 \end{cases} \quad (9)$$

5 Simulation Analysis

This segment contains the analysis output of the proposed model. The application is operated in the MATLAB platform, having an Intel(R) Core (TM) i5-3570S CPU processor with a speed of 3.10GHz. The input parameters for comparison are Call block rate, Outage probability, Spectral Efficiency, SNR and Throughput.

The optimization metrics analyzed in the proposed model are Call Block Rate (CBR), SNR, Throughput and Outage Probability. The metrics used in the proposed system model are defined in Table 2 below.

The metrics used to evaluate simulation results are defined in the table above. These metrics are used for comparison to show the better system performance of the proposed one than the previous one.

6 Call Block Rate

Call Block Rate (CBR) is compared with load and speed parameters to show the system performance. The value of CBR decreases with an increase in speed. This graph represents the comparison of CBR in the range of -80 to -100, and the speed ranges from 50 to 500 kmph (Figs. 4, 5).

Table 2. Definition of metrics from the proposed model

Evaluation Metrics	Definition
CBR	It helps to balance the load and to preserve the connection quality
SNR	Measurement of signal strength concerning noise
Throughput	Maximum number of requests per unit of time
Outage probability	The probability of a communication channel that cannot be terminated

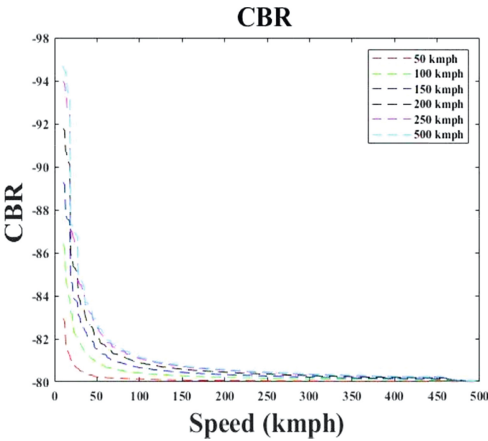


Fig. 4. Comparison between CBR and Speed

The simulation results show that the CBR value reduced with the load increase in the proposed model. CBR value ranges between -80 to -100 in the graph and the value of load changes from 0 to 500 kb/. From Fig. 6, we can see that the graph leniently goes down concerning the increase in load.

7 SNR

Signal to Noise Ratio is the ratio of the desired signal to the noise background. It can be calculated using Signal Power and Noise Power. It is expressed in dB. (decibels).

From this analyzed result, it is found that the SNR value tends to increase with time comparatively.

Figure 6 shows that the previous algorithms, like WOA, ABC AND PSO were compared to the proposed Sea Lion optimization. From this analyzed result, it is found that the SNR value tends to increase with time comparatively.

Figure 7 shows that the SNR value decreased with the increase in load. SNR varies in the 0 to 300 decibels range, and Load varies from 0 to 500 kb/S.

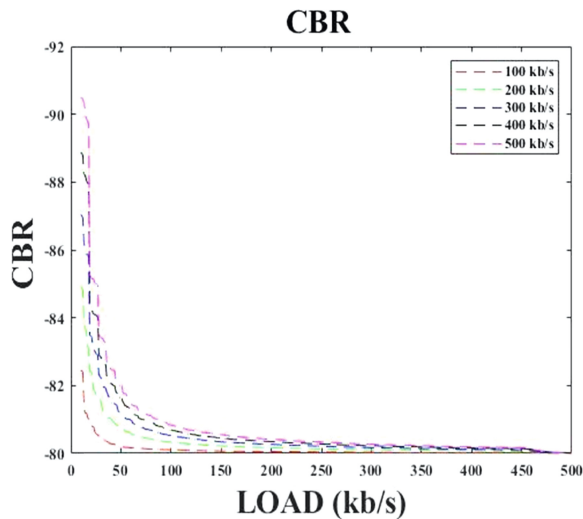


Fig. 5. Comparison between CBR and Load

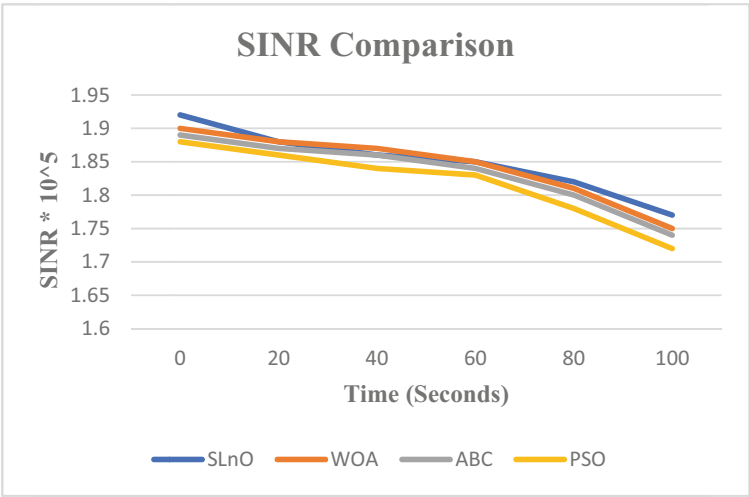


Fig. 6. Comparison of SNR value in previous algorithms with the varying Time

Figure 8 shows the graph which compares the SNR value to the speed; the analysis results in a decreased SNR value with an increase in speed. SNR ranges between 0 to 1000 decibels, and speed varies from 0 to 500 kmph.

8 Throughput

A value indicates the maximum number of transactions per second and the number generated within a time slot—Fig. 9.

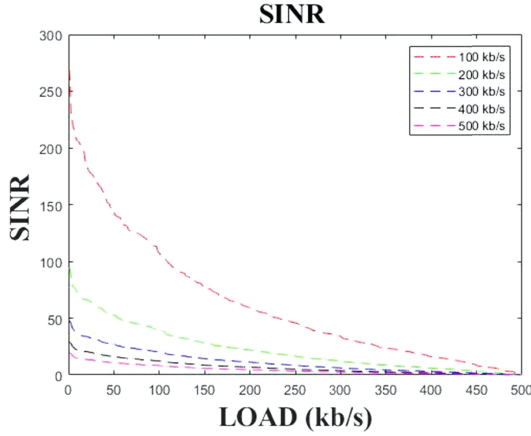


Fig. 7. Graph that shows a comparison of SINR value with the Load

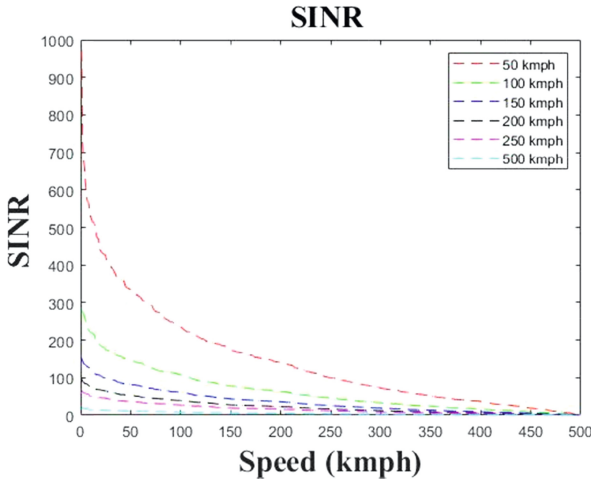


Fig. 8. SINR and Speed comparison

The analyzed result of the Graphical representation of throughput value concerning the time in previous algorithms showed. The proposed model's throughput value is increased from the previous algorithms. The throughput value ranges from 1.206 to 1.215, and the time value varies between 0 to 100 s.

9 Outage Probability

It is defined as the instant when the user node power decreases below the threshold; it is said that the user goes out of the coverage area at this instant.

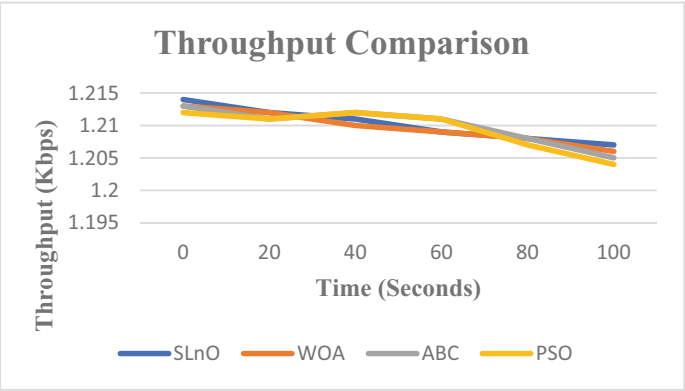


Fig. 9. Graphical representation of Throughput to Time (in seconds)

It shows that the outage probability varies up to 0.999946 in the proposed algorithm concerning the change in time. In contrast, the previous algorithms show a lower outage probability concerning an increase in time.

10 Overall Result

The result shows that the proposed system model gives an optimized decision. The model also reduces signal cost and reduces the usage of memory.

Table 3. Listing of values obtained by calculating the performance parameters

Performance Parameters	Obtained Values
Call Block Rate Varying Speed	432 kmph
Call Block Rate Varying Load	394 Kbps
Spectrum Efficiency	63 s
SINR By Varying Time	55 s
SINR By Varying Load	345 Kbps
SINR By Varying Speed	468 kmph
Outage Probability	58 s
Throughput	60 s

CBR, throughput, spectral efficiency and outage probability are compared with Time, speed and load. From the existing algorithms, the proposed one is performing better. It gives increased mobility and system performance.

Hence, Table 3 represents the measure of performance parameters that showed an optimum critical performance of the proposed system.

11 Conclusion

The main idea of this model is to optimize the decision-making for an efficient and continuous connection. This study uses fuzzy logic for making decisions; the decision was made while analyzing parameters like SNR, throughput and Spectral efficiency. The fuzzy logic system generates rules by taking CBR, Load, Speed and BS location as input. The decision for handover or load balancing is made based on this generated rule. Examining these parameters using two SON functions results in a conflict state. In this conflicted state, the fuzzy system could not perform the decision-making either handover or load balancing. At this stage, the fuzzy logic system uses the whale optimization algorithm to find the exact decision that should be made for a continuous network connection. Sea Lion optimization algorithm (SLnO) uses a fitness function which takes Outage probability and throughput as input and gives an output value. This value takes control of decision-making and proceeds with handover or load balancing. However, the centralized coordination scheme is prioritization of SON function. This causes the risk of engrossing by high prioritization. Signal fluctuations may occur in the LTE network because of conflict conditions, so the network does not obtain optimal solutions. Two or more SON functions are integrated to overcome these issues in the LTE network by maintaining proper handover or load balancing decisions.

References

1. Akyildiz, I.F., et al.: The evolution to 4G cellular systems: LTE n-advanced. *Phys. Commun.* **3**(4), 217–244 (2010)
2. Hasan, M.M., Kwon, S., Na, J.H.: Adaptive mobility load balancing algorithm for LTE small-cell networks. *IEEE Trans. Wireless Commun.* **17**(4), 2205–2217 (2018)
3. Zhou, Y., Ai, B.: Handover schemes and algorithms of the high-speed mobile environment: a survey. *Comput. Commun.* **47**, 1–15 (2014)
4. Li, W.Y., et al.: A novel dynamic adjusting algorithm for load balancing and handover co-optimization in LTE SON. *J. Comput. Sci. Technol.* **28**(3), 437–444 (2013)
5. Bejerano, Y., et al.: DyMo: dynamic monitoring of large-scale LTE-multicast systems. *IEEE/ACM Trans. Netw.* **27**(1), 258–271 (2019)
6. Nguyen, M.T., Kwon, S., Kim, H.: Mobility robustness optimization for handover failure reduction in LTE small-cell networks. *IEEE Trans. Veh. Technol.* **67**(5), 4672–4676 (2017)
7. Chen, X., Kim, M.J., Yoo, S.H., Park, N.Y., Youn, H.Y.: Efficient and prompt handover in LTE-based systems by predicting the target eNodeBs. In: 2014 International Conference on Cyber-enabled Distributed Computing and Knowledge Discovery, pp. 406–413 (2014)
8. Awada, A., Wegmann, B., Viering, I., Klein, A.: A SON-based algorithm for the optimization of inter-RAT handover parameters. *IEEE Trans. Veh. Technol.* **62**(5), 1906–1923 (2013)
9. Algeri, N., Boukerche, A.: An efficient handover trigger scheme for vehicular networks using recurrent neural networks. In: Proceedings of the 15th ACM International Symposium on QoS and Security for Wireless and Mobile Networks, pp. 85–91 (2019)
10. Algeri, N., Boukerche, A.: A two-tier machine learning-based handover management scheme for intelligent vehicular networks. *Ad Hoc Netw.* **94**, 101930 (2019)
11. Castro-Hernandez, D., Paranjape, R.: Optimization of handover parameters for LTE/LTE-A in-building systems. *IEEE Trans. Veh. Technol.* **67**(6), 5260–5273 (2017)
12. Hashim, H.A., Abido, M.A.: Location management in LTE networks using multi-objective particle swarm optimization. *Comput. Netw.* **157**, 78–88 (2019)

13. Hussein, Y.S., Ali, B.M., Rasid, M.F.A., Sali, A., Mansoor, A.M.: A novel cell-selection optimization handover for long-term evolution (LTE) macrocell using fuzzy TOPSIS. *Comput. Commun.* **73**, 22–33 (2016)
14. Silva, K.C., Becvar, Z., Cardoso, E.H., Francês, C.R.: Self-tuning handover algorithm based on fuzzy logic in mobile networks with dense small cells. In: 2018 IEEE Wireless Communications and Networking Conference (WCNC), pp. 1–6 (2018)
15. Singh, A., Juneja, D., Malhotra, M.: Autonomous agent-based load balancing algorithm in cloud computing. *Procedia Comput. Sci.* **45**, 832–841 (2015)
16. Wan, J., Chen, B., Wang, S., Xia, M., Li, D., Liu, C.: Fog computing for energy-aware load balancing and scheduling in intelligent factory. *IEEE Trans. Industr. Inf.* **14**(10), 4548–4556 (2018)
17. Ghomi, E.J., Rahmani, A.M., Qader, N.N.: Load-balancing algorithms in cloud computing: a survey. *J. Netw. Comput. Appl.* **88**, 50–71 (2017)
18. Pedrycz, W.: Heterogeneous fuzzy logic networks: fundamentals and development studies. *IEEE Trans. Neural Netw.* **15**(6), 1466–1481 (2004)
19. Gupta, Y., Saini, A., Saxena, A.K.: A new fuzzy logic based ranking function for efficient information retrieval system. *Expert Syst. Appl.* **42**(3), 1223–1234 (2015)
20. Iacoboaiea, O.C., Sayrac, B., Jemaa, S.B., Bianchi, P.: SON coordination in heterogeneous networks: a reinforcement learning framework. *IEEE Trans. Wireless Commun.* **15**(9), 5835–5847 (2016)
21. Sun, S., Kadoch, M., Ran, T.: Adaptive SON and cognitive smart LPN for 5G heterogeneous networks. *Mob. Netw. Appl.* **20**(6), 745–755 (2015)
22. Zhou, Q., Wu, W., Liu, D., Li, K., Qiao, Q.: Estimation of corrosion failure likelihood of oil and gas pipeline based on fuzzy logic approach. *Eng. Fail. Anal.* **70**, 48–55 (2016)
23. Korenevskiy, N.A.: Application of fuzzy logic for decision-making in medical expert systems. *Biomed. Eng.* **49**(1), 46–49 (2015)
24. Mohan, D., Geetha, M.: LTE KPI for data mining and machine learning. *Mendeley Data*, V3 (2022). <https://doi.org/10.17632/czkn9c4wk6.3>
25. Pitcher, B.J., Harcourt, R.G., Charrier, I.: Individual identity encoding and environmental constraints in vocal recognition of pups by Australian sea lion mothers. *Anim. Behav.* **83**(3), 681–690 (2012)
26. Folkens, P. A., Reeves, R. R.: Guide to marine mammals of the world (No. Sirsi) i9780375411410). National Audubon Society (2002)
27. Pitcher, T.J., Kalikoski, D., Short, K., Varkey, D., Pramod, G.: An evaluation of progress in implementing ecosystem-based management of fisheries in 33 countries. *Mar. Policy* **33**(2), 223–232 (2009)
28. Schusterman, R.J.: Behavioral capabilities of seals and sea lions: a review of their hearing, visual, learning and diving skills. *Psychol. Rec.* **31**(2), 125–143 (1981)
29. NOAA Fishers. https://www.westcoast.fisheries.noaa.gov/protected-species/marine_mammals/pinnipeds/california_sea_lions.html. Accessed 15 Mar 2019
30. NOAA National Ocean Service. <https://search.noaa.gov/search?dc=1199&utf8=%E2%9C%93&affiliate=oceanservice.noaa.gov&query=sea+lion>. Accessed 15 Mar 2019



A Comprehensive Review of Optimization Techniques for Automatic Voltage Regulator Controllers

Rumana Ali^{1,2,3} and Vinayambika S. Bhat^{2,3}(✉)

¹ Electronics and Communication Engineering, Mangalore Institute of Technology and Engineering, Moodbidri, India

² Electrical and Electronics Engineering, Adichunchanagiri Institute of Technology, Chikmagalur, India

vinayambika@mite.ac.in

³ Visvesvaraya Technological University, Belagavi 590018, Karnataka, India

Abstract. The article presents a latest literature review on issues related to the automatic voltage regulators (AVR), conventional methods for voltage regulation, application, and advancement in Voltage regulation techniques. The article also includes review of optimization of the AVR with different controllers used by researchers for efficient performance of the power system. Modern-day power system networks are intricate plus prone to a number of risks. The networks are vulnerable to instability because operating a power system is complex and unreliable. One of these problems is rotor-angle instability, which, if correctly resolved, could cause system failure. AVR is vital for the operation of power systems. Automatic voltage regulators are utilised in power systems to keep an alternator's terminal voltage at a set point. AVR controls the evenness of the terminal voltage by adjusting the generator's exciter voltage. A power system stabilizer (PSS) is essentially a power oscillation damping (POD) regulator that is used to increase rotor angle stability. AVR PSSs must be properly designed, which is a difficult yet necessary process. Several approaches are put out in the literature for the effective blueprint of PSSs or POD controllers. This provision includes an up-to-date overview of the AVR and PSS as well as optimisation methods. Based on this review, it can be imparted that there is plenty of room for developing AVR and PSS furthermore in a effective and reliable way, pondering quite a lot of uncertainties that could appear while operating current power systems.

Keywords: PID controller · FOPID · PSO · AI · CS · HDO · SD · HCO · TLBO · optimization · automatic voltage regulator

1 Introduction

Voltage regulation is an important part of power generation because it ensures that voltage levels provided to end consumers are steady and within acceptable ranges. Synchronous generators are critical components of power plants, transmuting mechanical energy into electrical energy. Technological breakthroughs have resulted in the development of numerous strategies and systems to optimise voltage control in synchronous

The original version of the chapter has been revised. Author corrections have been made. A correction to this chapter can be found at https://doi.org/10.1007/978-3-031-75608-5_28

generators throughout the years. This article investigates cutting-edge technology for voltage control for synchronous generators, highlighting formal applications and its advancement. Field excitation systems are vital for managing synchronous generator output voltage. These structures have conventionally used direct current (DC) excitation techniques, such as the brushless excitation structure. Excitation transformer provides excitation current to rotor windings, and voltage regulation is accomplished by adjusting the field current. The article is structured as mentioned. Section 1.1 concisely states about the conventional methods for voltage regulation. Section 2 portrays the advancements and mathematical modeling of AVR. Section 3 presents the literature review and insights of various optimisation techniques. Section 4 presents conclusions drawn from various research articles.

1.1 Conventional Methods for Voltage Regulation

Antiskid AVRs are a means to stipulate constant voltage output from generators. They are critical for supplying a continuous and safe power supply to electrical equipment and appliances [1, 2]. Electromechanical Voltage Regulators are the oldest and prevalent types of AVRs. To change the generator's field current, they frequently integrate voltage sensors, microcontrollers, and power electronics like thyristors or IGBTs. PID control techniques, which stand for proportional-integral-derivative, are universally exploited in AVR systems. These methods compare the desired and actual voltages to compute an error signal, and then they modify the field current using proportional, integral, and derivative terms. Voltage stabilisers are utilised as additional layer of voltage regulation in addition to AVR systems. These devices can react to voltage changes fast and maintain a constant output voltage even while the load is changing. Multiple generators may be run in parallel under specific circumstances, and their aggregate output is controlled to maintain a steady voltage. In large-scale power producing plants, this strategy is typical [3]. The unique application, the kind of generator or transformer, the desired level of precision, and the available budget all play a role in the decision of which AVR system to choose. Solid-state and microcontroller-based solutions are frequently used in modern systems because of their adaptability, effectiveness, and simplicity of digital control and monitoring system integration. Power stations synchronous generators frequently have AVRs installed. A generator AVR's major job is to control the generator's output voltage to a fixed set point, making sure it stays within reasonable bounds, often 5% of the nominal voltage or less. AVRs for generators make it easier for connected generators to run smoothly and preserve the voltage stability of the power supply. The most popular controller used for AVR systems is the PID controller, but there is a necessity for added precise and robust control due to increased demand and elevated dispersion of renewable energy sources. Table 1 presents the applications of AVR.

2 Advancements in AVRs

Advancements in AVRs have been driven by technological innovations, improved control algorithms, and the increasing complexity and demands of modern power systems. Digital control allows for more sophisticated control algorithms and faster response times to

Table 1. Applications of AVR system

Application	Description
Voltage Stability	AVRs assist in preventing voltage variances that exceed permitted limits [3]
Reactive Power Control	In addition to regulating voltage, some advanced AVR systems also control the reactive power output of generators [4]
Frequency Control	While not directly related to voltage regulation, some AVR systems may also monitor and control the generator’s frequency [5]
Integration with Control Centers	AVR data is usually assimilated into supervisory control and data acquisition (SCADA) systems, energy management systems (EMS). This integration allows for centralized monitoring, control, and optimization of the complete power grid

voltage fluctuations and load changes. AVR systems are integrated into Wide Area Monitoring and Control systems, which provide a holistic outlook of the entire power grid. This allows for coordinated voltage control and optimization across multiple generators and substations in a wide geographical area [1]. AVR systems can be remotely monitored and controlled through communication networks. This enables operators to make real-time adjustments and diagnose issues without physical access to the equipment technology has evolved to work seamlessly with energy storage systems, namely batteries and supercapacitors. These structures provide rapid response to voltage fluctuations and enhance grid stability. Due to the expanding reliance on digital control and communication, AVR systems have also seen advancements in cybersecurity to safeguard from cyber threats and ensure the integrity and availability of the control system. As more distributed energy resources, are integrated into power systems, AVRs are adapting to manage the challenges of voltage control in decentralized generation environments. AVR systems are designed to be more energy-efficient, minimizing losses in the voltage regulation process [2, 4]. This aids to lessen the operational costs and environmental impact. In the context of grid modernization initiatives, AVRs are evolving to support the integration of smart grid know-hows, comprising of advanced metering, demand response, and automation of the grid, for improved grid resilience and efficiency. Some AVR systems are endowed with predictive maintenance capabilities, utilizing data analytics and condition monitoring to predict and prevent equipment failures, ensuring high reliability. Advanced protection features, such as fault detection and rapid shutdown, are integrated into AVR systems to prevent damage to equipment during voltage disturbances or faults. Artificial intelligence (AI) and machine learning are utilised to create power system dynamics prediction models. These models help PSS predict grid events, such as voltage instability or oscillations, and adjust control parameters in real-time to mitigate potential issues. Figure 1 displays a single line model of a single machine infinite bus (SMIB) system. An excitation system is included with the generator under

consideration, neglecting line resistance. The load_angle $\Delta\delta$, rotor_speed $\Delta\omega$, internal_voltage of the generator E_q , field_voltage E_{fd} , internal_voltage of the generator E_d and electrical_torque T_e can be used to represent the fourth-order system comprising of a generator and exciter.

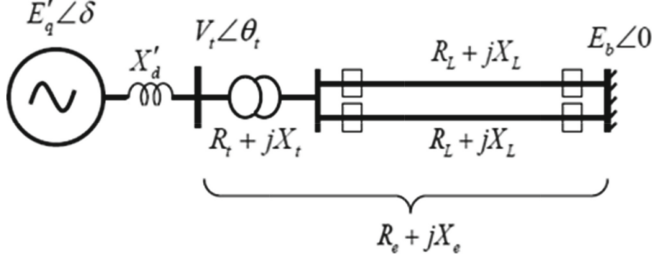


Fig. 1. Single line representation of a SMIB system [1]

The nonlinear dynamical equations for a synchronous generator in IEEE Model 1.1 are presented as [1].

$$\delta' = \omega_B S_m \quad (1)$$

$$S'_m = \frac{1}{2H} [T_m - T_e - DS_m] \quad (2)$$

where, S_m is the slip speed of the rotor, with H as inertia constant, T_e and T_m are electrical_torque and mechanical_torque corresponding, and generator's damping coefficient is denoted by D .

Generator Electrical Dynamics:

$$\frac{dE'_q}{dt} = \frac{1}{T_{do'}} \left[-E'_q + (x_d - x'_d) i_d + E_{fd} \right] \quad (3)$$

$$\frac{dE'_d}{dt} = \frac{1}{T_{qo'}} \left[-E'_d + (x_q - x'_q) i_q \right] \quad (4)$$

where, $T_{do'}$ and $T_{qo'}$ means d_axis and q_axis time constants of transient open-circuit, correspondingly. x_d and x_q means d_axis and q_axis reactance of the steady state, correspondingly. x'_d and x'_q denote d_axis and q_axis reactances of transient. E'_d and E'_q denote q_axis and d_axis transient voltages of generator. T_e and T_m means electrical_torque and mechanical_torque, respectively. Linear I-order excitation structure, dynamical equations can be identified as:

$$\frac{dE'_{fd}}{dt} = (V_t - V_r) \frac{K_E}{T_E} - \frac{K_E}{T_E} E_{fd} \quad (5)$$

E_{fd} symbolises voltage of field_excitation. K_E represents gain of excitation system, T_E represents time constant. V_t represents terminal, V_r is the reference voltage of the synchronous generator.

Electrical calculations:

$$i_d = \frac{E_b \cos \delta - E'_q}{x_e + x'_q} \quad (6)$$

$$i_q = \frac{E_b \sin \delta + E'_d}{x_e + x'_q} \quad (7)$$

$$v_q = E'_q + x'_d i_d \quad (8)$$

$$v_d = E'_d - x'_q i_q \quad (9)$$

$$V_t = \sqrt{V_q^2 + V_d^2} \quad (10)$$

$$T_e = E'_d i_d + E'_q i_q + (x'_d - x'_q) i_d i_q \quad (11)$$

where i_q and i_d symbolises q_axis and d_axis currents correspondingly, v_q and v_d stands for stator voltages q_axis and d_axis correspondingly. Terminal voltage is denoted by V_t and dependent control E_{fd} .

$$K_1 = \frac{E_b E'_{qo} \cos \delta_o}{X'_{de}} - \frac{E_b E'_{do} \sin \delta_o}{X'_{qe}} + \frac{X'_d - X'_q}{X'_{de} X'_{qe}} E_b^2 \cos 2\delta_o$$

$$K_2 = \frac{E_b \sin \delta_o}{X'_{de}}$$

$$K_3 = \frac{X'_{de}}{X_{de}}$$

$$K_4 = \frac{X_d - X'_d}{X'_{de}} E_b \sin \delta_o$$

$$K_5 = - \left(\frac{V_{do}}{V_{to}} \frac{X'_q}{X'_{qe}} E_b \cos \delta_o + \frac{V_{qo}}{V_{to}} \frac{X'_d}{X'_{de}} E_b \sin \delta_o \right)$$

$$K_6 = \frac{V_{qo}}{V_{to}} \frac{X_e}{X'_{de}}$$

$$K_7 = \frac{X_q - X'_q}{X'_{qe}} E_b \sin \delta_o$$

$$K_8 = K_{11} = 0$$

$$K_9 = \frac{X'_{qe}}{X_{qe}}$$

$$K_{10} = \frac{E_b \cos \delta_o}{X'_{qe}}$$

$$K_{12} = \frac{V_{do} X_e}{V_{to} X'_{qe}} \quad (12)$$

Two axis model of synchronous machine as in Fig. 2 is considered. The linear constants K_1 to K_{12} are mentioned in Eq. (12). Transfer function given in Fig. 3 depicts first order excitation system.

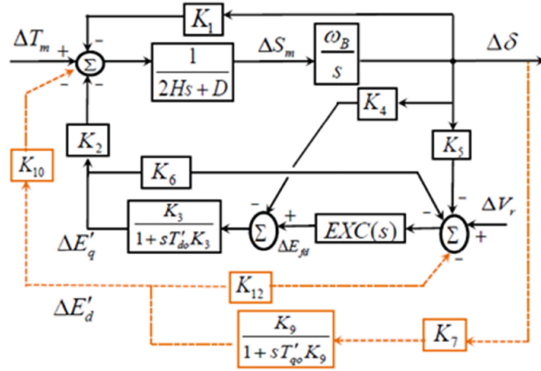


Fig. 2. Linearized illustration of SMIB power system of Model 1.1 IEEE [2].

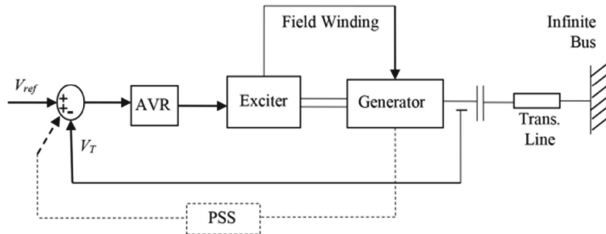


Fig. 3. Linearized illustration of SMIB power system of IEEE Model 1.1 [2]

At a specific operational point, the AVR and PSS control schemes make use of linear control theory. When there is a significant disruption, linear control methodologies may be futile or even endanger the system's stability. Since design of nonlinear controllers processes are independent of operating points, this calls for the employment of nonlinear control methods as a good solution. The two primary nonlinear techniques in power system works are centered on Lyapunov functions and normal practices [4, 5].

3 Optimization Techniques for AVR

The Proportional Integral Derivative (PID), recurrently chosen controller and is renowned for its design edifice ease for a vast choice of functioning environments [2], it is also one of the configurations presented in the literature to control the AVR system.

[3] has introduced an AVR systems PID plus II order derivative (PIDD2) controller. With the aid of population-based algorithms or optimisation techniques, the controller gains are determined to be at their optimal values. A PSO [4] tuned PID controller for AVR systems. [5] has created the Artificial Bee Colony (ABC) comparative performance evaluations. For the objective of tuning a traditional PID controller, Teaching Learning-Based Optimisation (TLBO) is applied [6]. [7] displays a comparison of two synchronous machine models (in investigations of transient stability with and without the Power System Stabilizer (PSS)). It offers a look at how power systems recover from disturbances of various kinds and resume stable or regular performance. In the AVR system, [8] introduced a unique tuning scheme for the PID controller utilizing the algorithm based on flocking of Cuckoo. [9] proposed a novel performance metric to identify PID controller designs that are elite for an AVR system by means of CS. With the help of population-inspired algorithms, these cutting-edge swarm intelligence techniques, individuals work together to produce intelligent behaviours. For optimizing PID controller gain, [10] proposed an improved whale optimisation algorithm (IWOA) to analyze stability. TSA bound PID controller [11] adopted to retain the terminal voltage stability of synchronous generator automated voltage regulators. In comparison to other methods reported by the researchers for the AVR system, the suggested TSA bound PID controller imparts better tracking performance and robustness against uncertainties in system parameters. [12] explained how to compute the best PID factors for an AVR using a unique optimisation technique called Simulink Design Optimisation (SDO). [13] introduced a novel technique named Taguchi mixed genetic algorithm to upgrade step responsiveness of voltage. [14] determines PID controller gains in the AVR system utilizing a novel parameter tuning approach centred on Harris Hawk's Optimisation (HHO). The AVR system, which is utilised in power system utilities, has been discussed in [15], along with the difficulties in attaining steady state. Through an enhanced kidney inspiration algorithm along a goal function, it illustrates a tuning design. Table 2 presents the insights of the various research papers which are implanted using PID controllers along with latest optimization techniques.

Ant Lion bound optimisation method is exercised to adjust PIDD2 gains. An AVR system's application and examination of a fractional order PID (FOPID) controller demonstrates how well it works to increase system robustness [16]. [17] examines the employment of a FOPID controller based on swarm intelligence in AVR systems. [18] presents the design of a PID controller with fuzzy logic bound voltage control task in the AVR system. Compromise between the dynamic response features of the [19] supply an objective function. [20] suggested a novel design technique that combines the use of genetic algorithms. In order to concurrently achieve global asymptotic stability and voltage regulation, [21] proposed a revolutionary Lyapunov-based excitation control (LEC). [22] discusses the design of AVR Controller for diesel-based power plants using Routh's Criteria Stability Analysis, which suggests method to regulate voltage in response to an increase in reactive power, [23] illustrates the construction of an artificial neural network (ANN) based AVR controller.

[24] offers a method that is the most effective for creating a PID-PSS that is reliable SMIB power system. To create an ideal PID-PSS for enhancing the functioning, the research also offers straightforward and novel closed loop stability requirements for the

Table 2. AVR system designed with PID controller and various optimization techniques

Application	Description
[8]	CS-based PID controller has goal function that significantly enhances the PID tuning optimisation method
[10]	The IWOA AVR system is demonstrates superior transient response and stability metrics. It also performs robustness analysis of the optimized AVR system by examining the system response when the generator, exciter, and amplifier's time constants are varied
[11]	Treeseed Algorithm optimized PID improves the AVR tracking performance as the load varies and the controller performed better provision for different cases
[12]	By introducing disturbances to the amplifier parameter uncertainty and generator field voltage, the suggested controller's resilience was confirmed. To validate the stability of the AVR that is discovered by the conventional investigation, work suggests a unique stability analysis known as disk-based stability analysis
[13]	The Taguchi approach, which makes use of analysis of means, was used to discover the close to optimal values for the design variables. The two most important design variables were chosen using an analysis of variance
[14]	put forth a fresh method of parameter tuning based on Harris Hawks Optimisation (HHO) to figure out PID improvements
[15]	Self-adaptive algorithm named improved kidney-inspired algorithm (IKA) for fine regulating the PID, with strong convergence properties and seeks to optimize the AVR system's transient responsiveness

same. To replicate the ideal PSS features in the rotor mode, a special non-linear objective is taken into account [25]. Table 3 presents the insights of the various research papers which are implanted using FOPID controllers along with latest optimization techniques.

[27] evaluated the working of the proposed technique on 23 benchmark functions, suggested an improved crow search algorithm (ECSA). [28] offers a PSS and AVR intelligent coordinator structure. The sine-cosine algorithm, a novel evolutionary method, is deployed to optimize the scaling factors. To increase the AVR system's resistance to quite severe disturbances induced together by the load and the set point sides, the author suggests a state feedback PI-controller with 2DOF [29]. [30] compares and contrasts five meta-heuristic algorithms to resolve the best way to tune PSSs in a multi-machine smart power system. Latest optimization approaches have been presented [31]. [32] explains a novel artificial intelligence (AI) technique for controlling an AVR that makes use of a neural network enhanced by the slime mould algorithm (SMA). [33] uses a reformed edition of the grey wolf optimizer and fitness function, the proposed controller's settings were built and optimized for AVR. It demonstrated robustness in the presence of externally induced system uncertainty. The suggested controller's performance was tested by simulations utilizing a real-world AVR system. [34] investigates a nonlinear excitation system based on normal form for SMIB power system's IV order synchronous generator model and adopts a SMIB power system to do nonlinear time domain simulation and eigenvalue analysis. [35] uses a soft computing approach based

Table 3. AVR system designed with FOPID controller and various optimization techniques

Reference	Insights
[16]	Ant Lion Optimizer algorithm (ALO), a brand-new population-based optimisation algorithm, is suggested as a solution to the FOPID controller tuning issue
[26]	In order to enable the designer to assess importance of performance criteria, the study examines one of the time domain objective functions that has been utilised in literature
[20]	The suggested method has several advantages, such as simple implementation, consistent convergence characteristics, good computing efficiency, superior solution to enhance the AVR system's transient responsiveness
[22]	The study emphasizes the significance of advancements in automated control theory and practise, easier to get performance from dynamic systems and improve quality
[23]	Controller's potential for enhancing power system transient stability by swiftly eliminating oscillations has been demonstrated
[24]	provides new, straightforward closed-loop stability requirements for SMIB interval systems, which are then utilised to develop the best PID-PSS for enhancing SMIB system performance
[25]	technique exhibits the ideal PSS features via eigenvalue analysis and exhibits the robustness of this technique in compared to conventional controllers

on the neural network parasitism-predation algorithm (PPA) to adapt the AVR's parameter acquisition. [36] offered fractional order predictive control methodology for a linear time-invariant system. To enrich the dynamic output and stability of voltage of many generators in wide-area multigeneration systems, a modified version of Q-learning technique has been exhibited [37]. [38] examines the use of a FOPID controller in the AVR, which gives more accurate control action and control flexibility than a traditional PID controller.

[39] proposes a unique design approach that uses the Coyote optimisation algorithm (COA) to optimize a novel objective function and factors of a $PI^\lambda DND^2N^2$ controller. AVR using the predation parasitism (PP) with the forward backpropagation neural network cascaded is exhibited [40]. [41] uses Artificial Ecosystem Optimisation (AEO), a cutting-edge optimisation technique, to locate the PID controller's ideal design parameters for an AVR. AEO has demonstrated its capacity to obtain ideal solutions quickly and effectively in terms of precision and time needed for each iteration. Fuzzy Inference System is adopted to control the AVR system's parameters is trained, suggested NeuroFuzzy Inference System with adaptive technology, customizing a hybrid optimisation scheme to produce the appropriate dataset required for the fuzzy inferential system's training [42]. New Water Cycle Algorithm based on PID regulator employing Integral Square Error criteria was proposed by [43]. Table 4 presents the insights of the various research papers which are implanted using AVR and PSS along with latest optimization techniques.

Table 4. AVR system and PSS designed with FOPID controller and various optimization techniques

Reference	Insights
[28]	To keep the power system from entering an unstable area during severe disturbances, the suggested intelligent coordinators dynamically revise the gain values of AVR and PSS
[44]	Stability, time-domain, frequency-domain, and two distinct robustness tests demonstrate the ascendancy of the controller proposed
[34]	controller meets the requirements of an ideal voltage regulator by noting that the rotor and voltage profiles are uncoupled with it in contrast to traditional voltage regulators
[35]	Noted that higher speed and rotor angle performance, with a decrease in overshoot, of the suggested PPA-CFBNN technique
[36]	Using numerical simulations to compares the performance of the proposed FO-MPC with the traditional MPC. Assess the suggested FO-MPC method's effectiveness and superiority in managing the AVR system
[38]	Demonstrates the improved performance of the suggested technique, it is compared to various AI-based systems found in recent research
[40]	Enhances the performance of the AVR on a single machine system, the study combines PPA approaches utilising a cascade-forward backpropagation neural network
[41]	There is no comparison of the suggested AEO algorithm's performance or speed with that of other widely used algorithms in the study
[43]	The adoption of Communication Time Delays with AVR is strongly advised by this research for improved voltage control and durability. In order to identify the dominance, NWCA PID performance is paralleled with PSOPID, GWOPID, and MFOPID approaches

4 Conclusion

These advancements in AVR technology are crucial for ensuring the stability and reliability of power systems in an era of increasing complexity, renewable energy integration, and digitalization. They enable power utilities and operators to better manage voltage regulation and respond to the changing dynamics of modern electrical grids. In the latest review papers on optimization techniques used in automatic voltage regulator (AVR) systems, several significant conclusions can be drawn. These papers showcase the wide range of optimization techniques applied in AVR systems, including classical methods like PID control and more advanced approaches such as genetic algorithms, fuzzy logic, and artificial neural networks. Review papers often highlight trade-offs between different optimization techniques. Due to the increasing assimilation of renewable energy resources, the role of optimization in AVR becomes even more critical. Optimization algorithms can help manage the variability of renewable generation and maintain grid stability. These reviews typically suggest future research directions, including the need

for hybrid optimization approaches, improved modeling of power systems, and the development of intelligent AVR systems capable of self-learning and adaptation. The choice of optimization techniques for AVR can depend on various factors, including the specific application, system complexity, and performance requirements. PID controllers are extensively incorporated into AVR due to their simplicity and effectiveness. Optimal Control Techniques, Genetic Algorithms are used for tuning controller parameters in AVR systems. They are particularly effective when dealing with nonlinearities and complex system dynamics. ANNs, including deep learning approaches, can learn complex patterns from data and adapt to various operating conditions. Combining multiple optimization techniques can provide robustness and performance improvements.

Acknowledgments. The authors express their gratitude to Adichunchanagiri Institute of Technology, Chikmagalur, and the Mangalore Institute of Technology and Engineering, Moodbidri, for establishing an environment conducive to research and development.

References

1. Kumar, A.: Ideal AVR and PSS: a theoretical study. *IET Gener. Transm. Distrib.* **14**(25), 6250–6258 (2020). <https://doi.org/10.1049/iet-gtd.2020.1247>
2. Prasad, S., Kumar, A., Mandal, R.K.: Extended ideal AVR: a theoretical study. In: Asia-Pacific Power Energy Engineering Conference, APPEEC (2021)
3. Chaib, L., Choucha, A., Arif, S.: Optimal design and tuning of novel fractional order PID power system stabilizer using a new metaheuristic Bat algorithm. *Ain Shams Eng. J.* **8**(2), 113–125 (2017)
4. Vamsi Krishna, A.K., Tyagi, T.: Improved whale optimization algorithm for numerical optimization. In: Gao, X.Z., Tiwari, S., Trivedi, M., Mishra, K. (eds.) *Advances in Computational Intelligence and Communication Technology*. AISC, vol. 1086, pp. 59–71. Springer, Singapore (2021). https://doi.org/10.1007/978-981-15-1275-9_6
5. Micev, M., Čalasan, M.P., Ali, Z.M., Hasanien, H.M., Abdel Aleem, S.H.E.: Optimal design of automatic voltage regulation controller using hybrid simulated annealing – Manta ray foraging optimization algorithm. *Procedia Technol.* **21**, 295–302 (2015)
6. Moradi Zirkohi, M.: An efficient optimal fractional emotional intelligent controller for an AVR system in power systems. *J. AI Data Min.* **7**(1), 191–200 (2019)
7. Sahib, M.A.: A novel optimal PID plus second order derivative controller for AVR system. *Eng. Sci. Technol. Int. J.* **18**(2), 194–206 (2015)
8. Panda, S., Sahu, B.K., Mohanty, P.K.: Design and performance analysis of PID controller for an automatic voltage regulator system using simplified particle swarm optimization. *J. Franklin Inst.* **349**(8), 2609–2625 (2012)
9. Idir, A., Canale, L., Bensafia, Y., Khettab, K.: Design and robust performance analysis of low-order approximation of fractional PID controller based on an IABC algorithm for an automatic voltage regulator system. *Energies* **15**(23) (2022)
10. Rao, R.V., Savsani, V.J., Vakharia, D.P.: Teaching-learning-based optimization: an optimization method for continuous non-linear large-scale problems. *Inf. Sci. (NY)* **183**(1), 1–15 (2012)
11. Morab, A.N., Jinde, A.P., Narra, J., Kokane, O.: Comparative study of synchronous machine, Model 1.0 and Model 1.1 in transient stability studies with and without PSS. *Int. J. Eng. Res.* **V4**(05), 300–306 (2015)
12. Sikander, A., Thakur, P., Bansal, R.C., Rajasekar, S.: A novel technique to design cuckoo search based FOPID controller for AVR in power systems. *Comput. Electr. Eng.* **70**, 261–274 (2018)

13. Bingul, Z., Karahan, O.: A novel performance criterion approach to optimum design of PID controller using cuckoo search algorithm for AVR system. *J. Franklin Inst.* **355**(13), 5534–5559 (2018)
14. Mokeddem, D., Mirjalili, S.: Improved whale optimization algorithm applied to design PID plus second-order derivative controller for automatic voltage regulator system. *J. Chinese Inst. Eng. Trans. Chinese Inst. Eng. A* **43**(6), 541–552 (2020)
15. Kose, E.: Optimal control of AVR system with tree seed algorithm-based PID controller. *IEEE Access* **8**, 89457–89467 (2020)
16. Gopi, P., et al.: Dynamic behavior and stability analysis of automatic voltage regulator with parameter uncertainty. *Int. Trans. Electr. Energy Syst.* **2023**, 1–13 (2023)
17. Hasanien, H.M.: Design optimization of PID controller in automatic voltage regulator system using Taguchi combined genetic algorithm method. *IEEE Syst. J.* **7**(4), 825–831 (2013)
18. Abubakr, H., Vasquez, J.C., Hassan Mohamed, T., Guerrero, J.M.: The concept of direct adaptive control for improving voltage and frequency regulation loops in several power system applications. *Int. J. Electr. Power Energy Syst.* **140**, 108068 (2022)
19. Ekinici, S., Hekimoglu, B.: Improved kidney-inspired algorithm approach for tuning of PID controller in AVR system. *IEEE Access* **7**, 39935–39947 (2019)
20. Bourouba, B., Ladaci, S., Schulte, H.: Optimal design of fractional order $PI\lambda D\mu$ controller for an AVR system using ant lion optimizer. *IFAC-PapersOnLine* **52**(13), 200–205 (2019)
21. Khan, I.A., Alghamdi, A.S., Jumani, T.A., Alamgir, A., Awan, A.B., Khidrani, A.: Salp swarm optimization algorithm-based fractional order PID controller for dynamic response and stability enhancement of an automatic voltage regulator system. *Electronics* **8**(12), 1472 (2019)
22. Al Gizi, A.J.H., Mustafa, M.W., Al Zaidi, K.M.A., Al-Zaidi, M.K.J.: Integrated PLC-fuzzy PID simulink implemented AVR system. *Int. J. Electr. Power Energy Syst.* **69**, 313–326 (2015)
23. Alghamdi, S., et al.: Optimal PID controllers for AVR systems using hybrid simulated annealing and gorilla troops optimization. *Fractal Fract.* **6**(11), 682 (2022)
24. Al Gizi, A.J.H., Mustafa, M.W., Al-Geelani, N.A., Alsaedi, M.A.: Sugeno fuzzy PID tuning, by genetic-neutral for AVR in electrical power generation. *Appl. Soft Comput. J.* **28**, 226–236 (2015)
25. Liu, H., Su, J., Qi, J., Wang, N., Li, C.: Decentralized voltage and power control of multi-machine power systems with global asymptotic stability. *IEEE Access* **7**, 14273–14282 (2019)
26. Mangera, P., Sumbung, F.H., Parenden, D.: Automatic voltage regulator (AVR) controller design based on routh's crution stability analysis in diesel-based power plants, January 2018
27. Memon, A.P., Sattar Memon, A., Akhund, A.A., Memon, R.H.: Multilayer perceptrons neural network automatic voltage regulator with applicability and improvement in power system transient stability. *Int. J. Emerg. Trends Electr. Electron.* **9**, 30–38 (2013)
28. Vijaya, V.L., Manyala, R.R., Mangipudi, S.K.: Design of a robust PID-PSS for an uncertain power system with simplified stability conditions. *Prot. Control Mod. Power Syst.* **5**(1) (2020). <https://doi.org/10.1186/s41601-020-00165-9>
29. Bhullar, A.K., Kaur, R., Sondhi, S.: Enhanced crow search algorithm for AVR optimization. *Soft Comput.* **24**(16), 11957–11987 (2020). <https://doi.org/10.1007/s00500-019-04640-w>
30. Khezri, R., Oshnoei, A., Yazdani, A., Mahmoudi, A.: Intelligent coordinators for automatic voltage regulator and power system stabiliser in a multi-machine power system. *IET Gener. Transm. Distrib.* **14**(23), 5480–5490 (2020). <https://doi.org/10.1049/iet-gtd.2020.0504>
31. Blondin, M.J., Sanchis, J., Sicard, P., Herrero, J.M.: New optimal controller tuning method for an AVR system using a simplified ant colony optimization with a new constrained Nelder-Mead algorithm. *Appl. Soft Comput. J.* **62**, 216–229 (2018)
32. Sabo, A., Odoh, T.E., Shahinzadeh, H., Azimi, Z., Moazzami, M.: Implementing optimization techniques in PSS design for multi-machine smart power systems: a comparative study. *Energies* **16**(5) (2023)

33. Rais, M.C., Dekhandji, F.Z., Recioui, A., Rechid, M.S., Djedi, L.: Comparative study of optimization techniques based PID tuning for automatic voltage regulator system. *Eng. Proc.* **14**(1), 1–6 (2022)
34. Aribowo, W.: Slime mould algorithm training neural network in automatic voltage regulator. *Trends Sci.* **19**(3) (2022)
35. Verma, S.K., Devarapalli, R.: Fractional order $PI\lambda D\mu$ controller with optimal parameters using modified grey wolf optimizer for AVR system. *Arch. Control Sci.* **32**(2), 429–450 (2022)
36. Prajapati, Y., Kamat, V., Patel, J., Kher, R.: A comprehensive survey on use of soft computing and optimization techniques for load frequency control 3. *Different Soft Comput. Tech.* **8**(2), 64–70 (2020)
37. Deghboudj, I., Ladaci, S.: Automatic voltage regulator performance enhancement using a fractional order model predictive controller. *Bull. Electr. Eng. Informatics* **10**(5), 2424–2432 (2021)
38. Abegaz, B.W., Zarrabian, S.: Modified q-learning method for automatic voltage regulation in wide-area multigeneration systems. *Int. Trans. Electr. Energy Syst.* (2022)
39. Jumani, T.A., et al.: Jaya optimization algorithm for transient response and stability enhancement of a fractional-order PID based automatic voltage regulator system. *Alexandria Eng. J.* (2020)
40. Moschos, I., Parisses, C.: A novel optimal $PI\lambda DND2N2$ controller using coyote optimization algorithm for an AVR system. *Eng. Sci. Technol. an Int. J.* **26**, 100991 (2022)
41. Aribowo, W., Suprianto, B., Buditjahjanto, I.G.P.A., Widyartono, M., Rohman, M.: An improved neural network based on the parasitism – predation algorithm for an automatic voltage regulator. *ECTI Trans. Electr. Eng. Electron. Commun.* **19**(2), 136–144 (2021)
42. Čalasan, M., Micev, M., Djurovic, Ž., Mageed, H.M.A.: Artificial ecosystem-based optimization for optimal tuning of robust PID controllers in AVR systems with limited value of excitation voltage. *Int. J. Electr. Eng. Educ.* (2020). <https://doi.org/10.1177/0020720920940605>
43. Lawal, M.J., Hussein, S.U., Saka, B., Abubakar, S.U., Attah, I.S.: Intelligent fuzzy-based automatic voltage regulator with hybrid optimization learning method. *Sci. African* **19**, e01573 (2023)
44. Rao, G.S., Kalyan, C.N.S., Kumar, C.V., Goud, B.S., Kumar, M.K., Reddy, C.R.: Automatic voltage regulator using global optimization algorithms based on traditional controller. In: 2022 International Conference on Intelligent Controller and Computing for Smart Power, ICICCSP 2022, July 2022



Implementation of UWB Receiver Front-End in FPGA

Aravinda Koithyar^(✉), S. Arun, and K. Bhaskar

Department of Electronics and Communication, New Horizon College of Engineering,
Bengaluru, India
aravindak@newhorizonindia.edu

Abstract. Nowadays most commonly used technology for communication is the narrowband technology, which got less accuracy over tracking of precise location and more chances of signal interference with the same frequency due to lesser bandwidth. Usage of UWB technology can overcome the above mentioned problems. This article discusses the implementation of UWB filter bank receiver front-end in FPGA, which serves the purpose in short range communication and accurate ranging. The receiver architecture consists of a filter bank followed by a down conversion stage that relaxes the analog-to-digital converter requirements in terms of input analog bandwidth. The methodology followed here is to design receiver's architecture in Simulink, conversion of Simulink model to HDL code, and verifying the same using Xilinx. Finally the HDL code is downloaded in FPGA, which imitates the functionality of filter bank receiver architecture. Further, output of receiver can be fed to ADCs, upon performing some DSP on ADC output, location of an object or a person can be tracked.

Keywords: UWB filter bank · down conversion · Simulink model · HDL code · Xilinx · FPGA · ADC · DSP

1 Introduction

Ultra-wideband, or UWB, is a short-range radio frequency technology that can be used for wireless communication and is extremely capable of accurate detection of asset, device, and person location. UWB is a communication system that uses radio waves to transfer data between devices, just as Bluetooth and Wi-Fi. It accomplishes this by using sub-nanosecond pulses across “ultra-wide” range of frequencies. A UWB signal is defined as the one that has a bandwidth of more than 20% of its center frequency or more than 500MHz. Because the signal has a greater bandwidth, a UWB system is more resistant to multipath propagation and narrowband interferences, which only affect a small portion of the spectrum. This technology has a very fine time resolution, making it a strong contender for localization and range applications. In 2002, the American Federal Communications Commission (FCC) set a frequency spectrum for UWB communications ranging from 3.1 to 10.6 GHz [6].

Ultra-Wideband (UWB) communications are steadily gaining traction around the world, and several nations' governments have established frequency bands and operating

guidelines. Two of the various technologies available on the market to obtain a UWB signal are largely favored in recent standardization work, namely Orthogonal Frequency Division Multiplexing (OFDM) and also the Impulse Radio (IR).

On the one hand, IEEE 802.15 is a standard that defines a UWB signal that is supported by OFDM, which consists of a large number of digitally-generated thin sub-carriers. On the other hand, the IEEE 802.15.4a standard defines a UWB signal that supports short-duration pulses, a technique known as IR. Apart from the technologies covered by standards, various modulation strategies for UWB are being proposed, such as frequency modulated UWB or IR transmittal, as well as even shorter, sub-nanosecond pulses.

UWB is currently attracting a lot of attention and is a hot topic in business and academia. UWB technology enables high-speed interconnects in devices throughout the digital home and workplace to have the convenience and quality of wireless communications. Wireless personal space networks are designed for short-term use (WPANs), UWB is the dominant technology for freeing people from wires, allowing for the wireless connection of many devices for the transmission of video, audio, and other high-resolution data.

It has applications in Communication and sensors which includes high data rate, low data rate and Home network applications [7]. Also, its major application is in Positioning and tracking, Radar, 3D location sensing and WBAN [7–9]. In this work, IR based technique is employed for pulse generation.

Short-range radio technology such as UWB enhances longer-range radio technologies such as Wi-Fi, WiMAX, and cellular wide-area communications. It is used to send data from a number of devices to other devices in the immediate vicinity (up to ten meter or thirty feet). When technical standards are met, UWB devices will function in spectrum currently used by existing radio services without causing interference, allowing restricted spectrum resources to be utilized more effectively. UWB has the potential to be either a substitute for old-fashioned communication or the apex of an older technology, and both possibilities are possible (Fig. 1).

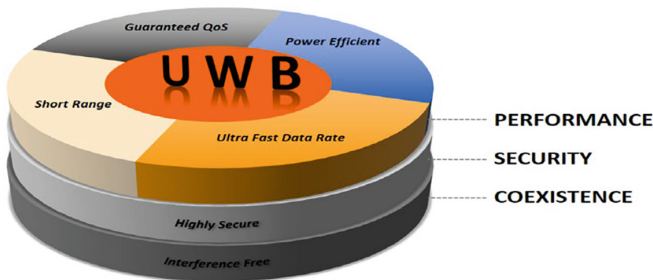


Fig. 1. UWB specification

UWB positioning systems use methods that disperse radio energy (across a large frequency range) with a very low power spectral density. They also rely on transmitting extremely short pulses. By sending more pulses per second, UWB leverages high pulse repetition rates to boost data throughput and localization accuracy. The technology is appropriate for radars and imaging because the lowering of pulse repetition rates enhances range capability.

There are examples of ultra-wideband technology in various sectors; today, ultra-wideband technology is predicted to become a highly essential component and has had a far-reaching impact in computers and several mobile phone firms. Ultra-Wideband (UWB) technology uses sub-nanosecond pulses. The UWB signal is substantially wider in the frequency domain with these pulses than with standard communications, which use narrowband frequency-multiplexed signals.

Instead of broadcasting on discrete frequencies as narrowband wireless transmission does, UWB communication spreads signals across a very wide bandwidth range. Although the continuous sinusoidal wave based narrowband transmission is today the most widely used type of wireless communication, the initial wireless transmission in 1894 was really an impulse based UWB transmission (Fig. 2).

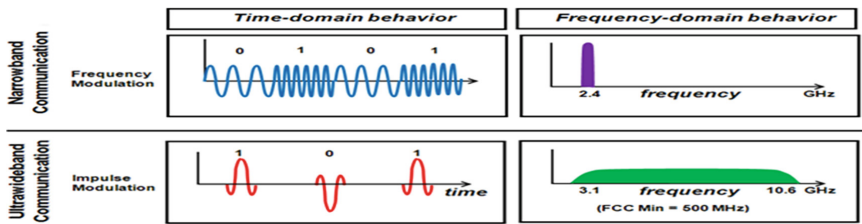


Fig. 2. UWB signal versus traditional narrowband signal

The receiver noise will feature in the low noise amplifier stage, which can give the required gain for medium range communications. Now, there is a brief history of ultra-wideband technology in this study, followed by a summary of ultra-wideband technological aspects that have gained favor in industry.

UWB transmissions cover a significant amount of the spectrum. The FCC defines power emission restrictions in different frequency bands for various types of UWB applications to avoid substantial interference with existing customers. Emission restrictions are measured in equivalent isotopically radiated power (EIRP). Under Part 15 of the commission's guidelines, it depicts the spectral mask for indoor and outdoor uses in the United States. Between 3.1 GHz and 10.6 GHz, a large spectrum up to 7.5 GHz is available at the EIRP limit of -41.3 dBm/MHz, which becomes the major spectrum target for most UWB systems.

When we think of wireless property technology, we usually think of Bluetooth and Wi-Fi. These technologies, on the other hand, lack the precision, location capabilities, and frequency security that UWB provides. Although other wireless property technologies may provide locomote, UWB's performance is far superior. UWB also works in a very different part of the spectrum, a pair of complete rings crowded around 4GHz. Furthermore, UWB will be compatible with other popular wireless technologies such as Wi-Fi, Bluetooth, and near-field communication (NFC).

One of the most obvious advantages of UWB is that data packets are sent and received using a larger portion of the physical layer (PHY). This layer, which is currently defined by IEEE 802.15.4z, provides an important security extension not available in other technologies, allowing security measures such as cryptography and random range generation to be used to prevent intruders from accessing UWB communications.

UWB is still a replacement technology, which means that companies working on it are just getting started. Samsung could be a member of the FiRa Consortium, which is focused on encouraging UWB use and developing revisions to UWB standards and certification procedures to verify capability (Fig. 3).

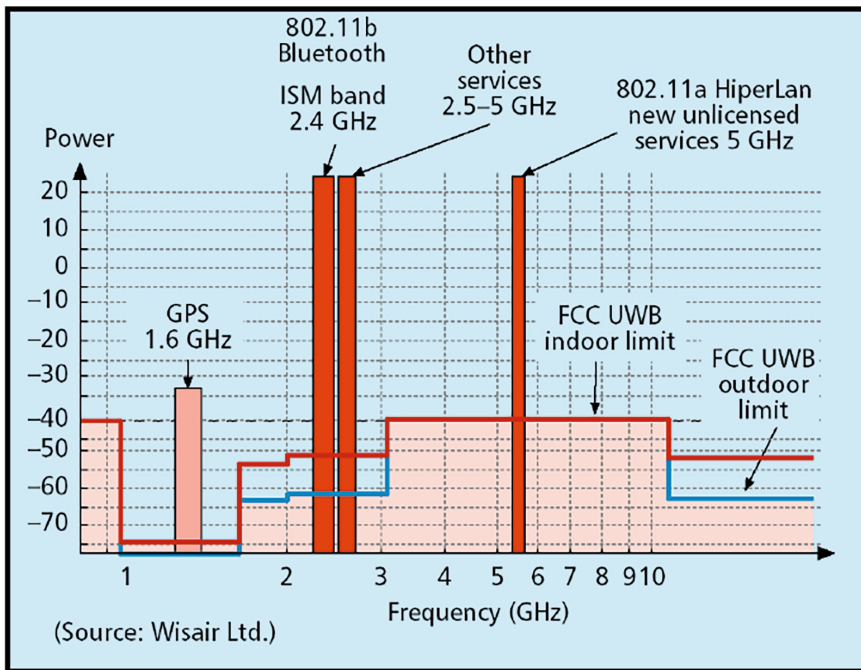


Fig. 3. FCC spectral mask for UWB systems

There are already a few of fascinating UWB-enabled applications in the Samsung Galaxy line-up. The use of UWB, for example, improves the new near Share app for device-to-device file transfers. Near Share automatically lists another UWB-equipped device at the top of your sharing panel when you simply point your phone at it.

UWB has also been included into Samsung's SmartThings realize application. People will find alternative UWB-equipped devices using augmented reality (AR) and intuitive directions. And, as Samsung announced at its unpacked event on January 14, 2021, the new Galaxy SmartTag+ supports UWB and can be attached to practically any item, allowing one to track items that don't have their own inherent affiliation.

"With UWB and AR finder, the Galaxy SmartTag+ can help you discover your items with unrivalled accuracy by guiding you to their location," said Elizabeth Lindstrom of Samsung's international Direct to Client Centre. People may also use AR finder on these UWB-enabled smartphones to drop virtual messages that let other Galaxy smartphone users know that one requires assistance.

The particular work discussed in this article had been intended for the practical implementation of a receiver front-end, and ultra-wideband is used for testing the low data rate communication, and is also used for calculating the ranging. The receiver architecture comprises of a filter bank, followed by a downconverter stage for converting high frequency signals to low frequency signals, and finally an analog-to-digital (ADC) converter with input analogue bandwidth requirements. A four-way Wilkinson power divider and a series of four Gaussian filters were used to create the filter bank.

The article mainly focuses on the implementation of a UWB receiver front-end based on a filter bank topology [1]. The RAKE receiver is the optimum choice for receiver architecture in multipath AWGN channels [10, 11, 12]. It combines a finite number of signal routes in a coherent manner to improve the quality of the received signal in terms of bit error rate (BER). However, in medium range UWB applications, the amount of signal energy captured by the RAKE receiver greatly depends on the amplitude distribution of incoming signal paths and the number of fingers in the RAKE.

In outdoor UWB applications, the received energy is dispersed over a vast number of routes, each of which has a significant amount of energy. This results in a poor performance of the RAKE receiver for a moderate number of fingers. However, as mentioned in [2], filter bank architectures are able to capture more energy from transmitted signals than RAKE receivers in very dense multi-path environment, boosting the SNR and as a result, lowering the BER.

The filter bank approach reduces the sampling frequency requirements because of the bandwidth of each filter and the input analog bandwidth requirements, by using a down conversion stage.

To examine the functioning of the said architecture, it is implemented in FPGA. This process involves creating the Simulink model and converting it to HDL code [4, 15] followed by downloading the design to FPGA [4, 13]. Thereby implementing the architecture in FPGA which is cost effective.

2 Literature Survey

[1] Xavier Artiga described the practical implementation of a receiver front end in an impulse radio ultra-wideband testbed for low data rate communications and range in detail in his article. The receiver architecture he proposed consists of a filter bank followed by a down conversion stage that lowers the ADC's input analog bandwidth restrictions. A four-way Wilkinson power divider and a group of four Gaussian filters

make up this filter bank. Prior to the filter bank, a low noise amplifier stage limits the receiver noise figure and provides the desired gain for medium range communications.

[2] In his study, Antonio Mollfulleda demonstrated that in dense multi-path channels, a filter bank receiver design for Ultra-wideband impulse radio systems is suitable. This method is based on the short time Fourier transform's time-frequency representation of the received signal. The receiver is made up of an analog component with bandpass filters, followed by an ADC bank. Under the assumption of synchronized reception, the filter bank receiver is evaluated in terms of error probability over multi-path channels, resulting in a closed form formulation of the bit error probability. Even with a moderate number of filters, simulation results for non-line of sight channels demonstrate that the proposed receiver's performance is on the verge of the matched filter bound. The author had demonstrated a testbed that had been created to provide a practical demonstration of the FB receiver's capabilities. The results of the channel estimate measurements demonstrate that dispersive channels are well captured.

[3] Qin Zhou in his paper he had discussed about the different types of UWB signaling scheme, which are Impulse Radio UWB and MB-OFDM UWB. In his paper he had mainly focused on IR UWB signaling and he also explained different modulation techniques required for UWB signaling. He explained about applications and advantages of the UWB technology in comparison with other technologies. The IR-UWB receiver based on the energy detection (ED) principle is investigated in the first part of his dissertation. A low-power ED receiver with multi-mode operation and flexibility is proposed. In 90 nm CMOS, the receiver prototype for the 3–5 GHz band is implemented.

[4] Norliza Othmanon field programmable gate arrays (FPGA), her work presents a simulation analysis of nonlinear dynamic of cardiac excitation based on the Luo Rudy Phase I (LR-I) model toward numerical solutions of ordinary differential equations (ODEs) responsible for cardiac excitation. She had designed LR-I model using Simulink blocks and converted it to HDL code. Verified the code using Co-simulation and her model was created utilizing the XC6VLX240T FPGA Xilinx Virtex-6 development board, which is ideal for solving higher-order ODEs and real-time applications while consuming little power.

[5] In his work, Tlelo-Cuautle demonstrates procedure for co-simulation. He explains in detail by giving some examples of co-simulation between: Active-HDL/Matlab-Simulink, Xilinx System Generator/Matlab-Simulink, and Altera DSP Builder/Matlab-Simulink.

3 Receiver Architecture

The receiver architecture in [1] has Amplification chain, Power splitters, Filter bank and Down conversion stage. In our architecture, we had not included amplification stage, as we assumed that the received signal is not degraded because in our case, the signal not transmitted through any medium. Since, the filter bank is designed in Simulink, power splits equally to all branches, hence power splitter is also not included. Therefore, our architecture consists of filter bank followed by down conversion stage as shown in a Fig. 4.

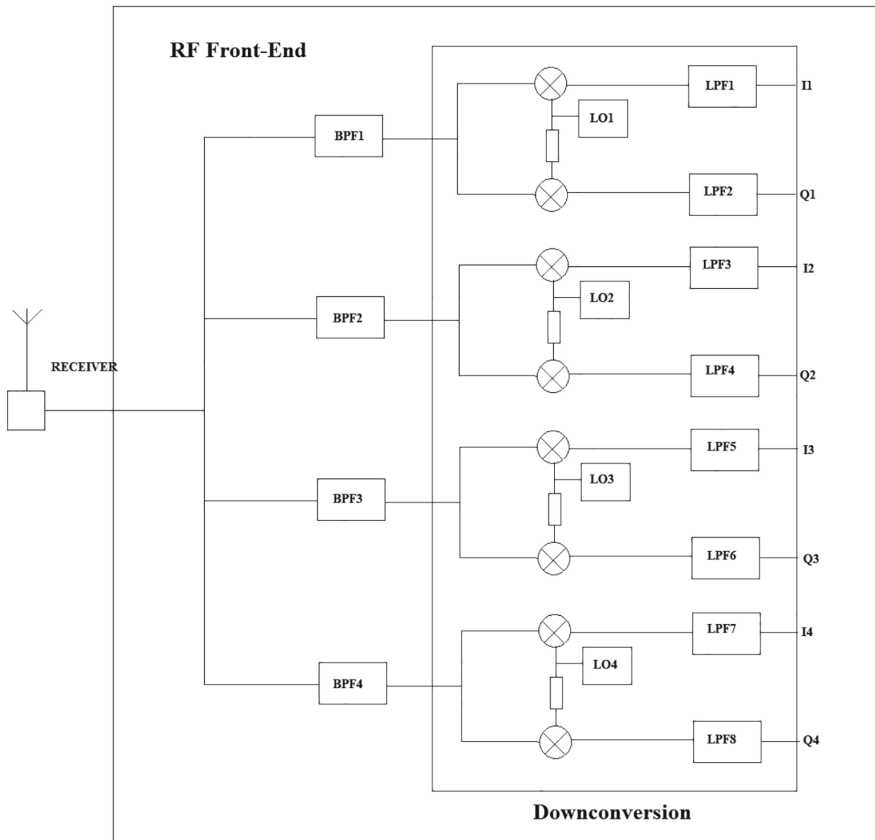


Fig. 4. Receiver block diagram

3.1 Filter Bank

The received signal is spread over a large band of frequencies since the sent signal is an impulse signal of sub-nanoseconds pulse. Filter bank is used in order to split the large band of frequencies to sub smaller bands. It is made up of four band pass filters, where

each filters having different center frequency with same bandwidth. These filters were chosen because they can concentrate energy the most in both the time and frequency domains, allowing us to get very near to meeting the orthogonality criteria in both domains. The four filters for this receiver should have the same group delay, allowing the set of ADCs to sample all of the channels' pulses at the same time.

3.2 Down Conversion

The sampling frequency of ADC's are low when compared to the frequency of filtered signals. So down conversion is done in order to make the filtered signals suitable for ADC requirements. An envelope detector, such as the one used in [14] is the simplest option for adapting the incoming signal to the ADC requirements. A diode and a capacitor are all that is required. However, because the phase information is lost in this instance, non-coherent energy detection techniques must be used. We eventually opted to employ a coherent direct down conversion to evaluate both coherent and non-coherent detection algorithms. We use four local oscillators, four mixers which results in eight separate signals (in-phase and quadrature components for each channel), and the main disadvantage of the proposed method is its complexity and power consumption. All these signals are fed to Low pass filter to meet the conditions of ADC.

4 Implementation

See Fig. 5.

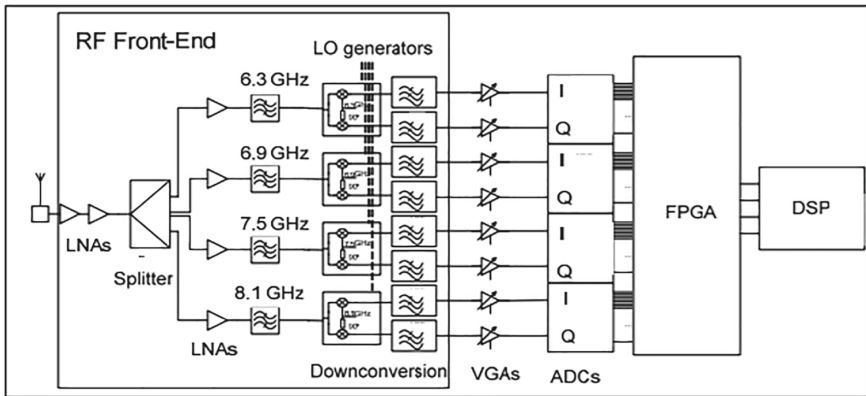


Fig. 5. Block diagram of the implementation

4.1 Creating a Simulink Model

See Fig. 6.

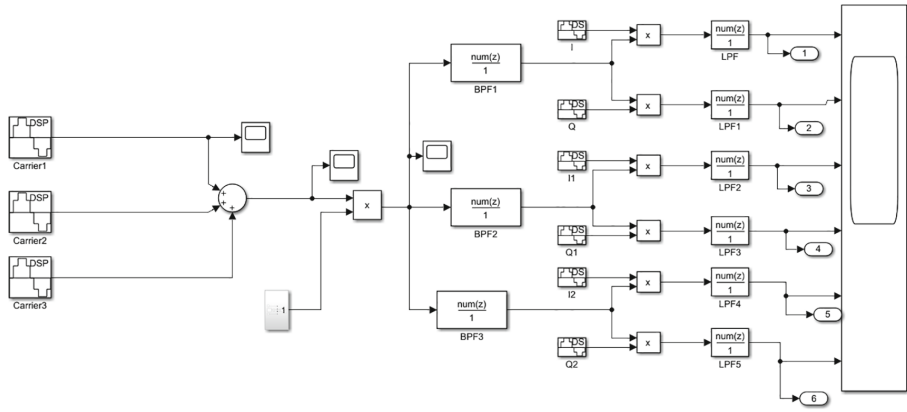


Fig. 6. UWB Filter bank architecture design in Simulink

We had designed the transmitter, which consists of carrier signal and a pulse generator. The carrier is the result of sum of sinusoidal signals of different frequencies; in the model, four DSP sinusoidal signals were examined. The sub-nanosecond pulse generated from pulse generator is modulated with carrier. In the model, multiplier circuit acts as a product modulator. Output of multiplier is the transmitted signal.

The received signal is fed to bank of band pass filters, each filters designed to have same bandwidth but with different centre frequencies. All the bandpass filters should be designed in such a way that, frequency domain representation of each when it is placed adjacent to each other, the overall bandwidth should be equal to bandwidth of pulse signal. In contrast, the filter bank is designed to split the bandwidth of UWB signal. In the Simulink model in order to convert it into HDL code [4], the block must be digital in nature. So, the filters are designed as digital filters. Each band pass filters output is down converted by mixer using local oscillator whose frequency is same as the centre frequency of BPF. In the model, filters output is given to mixer, instead of using LO DSP sine and cosine signals are used to provide inphase and Quadrature phase components. The I and Q signals further sent through low pass filter (Fig. 7).

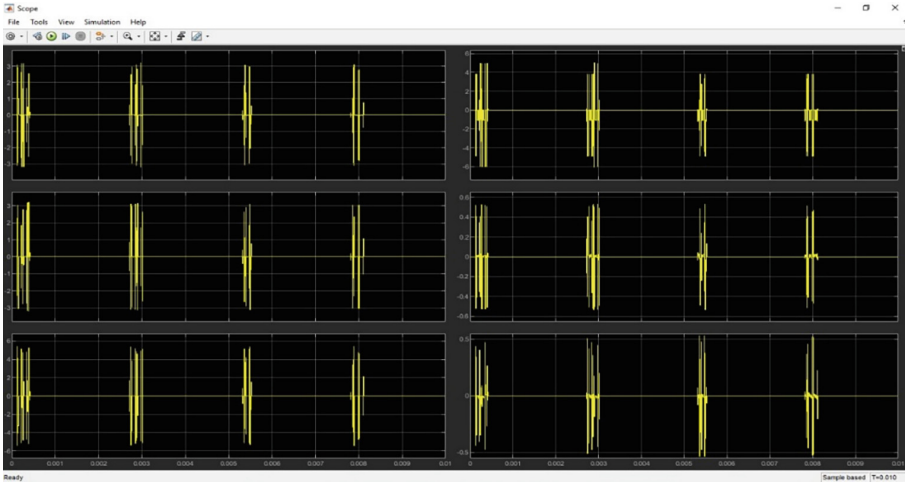


Fig. 7. Output of Simulink Model

4.2 Simulink Model to HDL Code Conversion

Each and every block used to create a Simulink model must and should support the HDL code conversion. All the analog blocks are continuous in nature which will not support the HDL code conversion. Hence, it is not appropriate to use analog blocks in the model. Therefore, the blocks used in the model should be discrete or digital. In the Simulink model, carrier used at transmitter is the discrete sinusoidal signal. All the filters used at receiver end are digital. Using HDL coder application, which is available in latest version of MATLAB, Simulink model can be converted to VHDL code [4, 15]. In the process of code generation, initially each block used in Simulink model are converted to VHDL code as separate modules. And interconnection between the blocks is converted as a main module, which includes instantiation of modules (blocks in the Simulink).

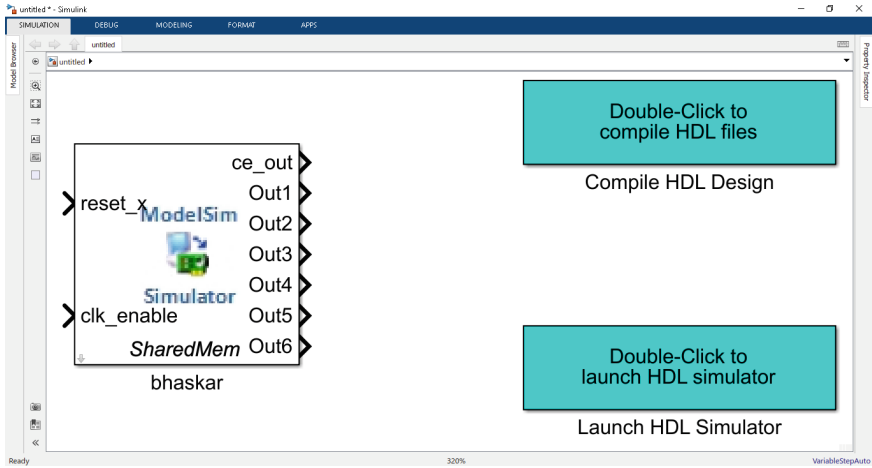


Fig. 8. Modelsim blocks generated by Co-Simulation Wizard

4.3 Generated HDL Code Verification

See Fig. 9.

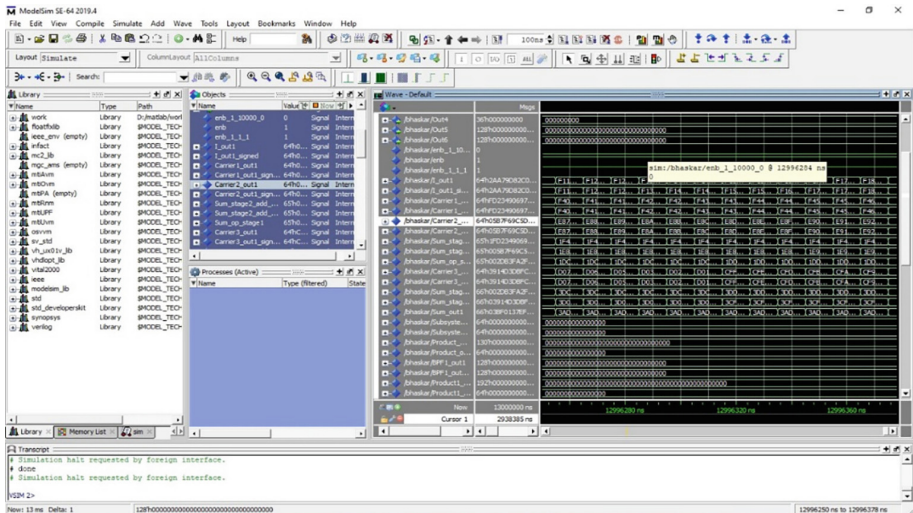


Fig. 9. Behavioral output of Modelsim

The code generated from HDL coder can be verified using Co-Simulation [4, 5, 15, 16]. This application can link the server of Matlab to HDL verifier application. In our project we had used Modelsim as an HDL simulator. Using Co-Simulation Wizard, we can create Modelsim blocks where VHDL code is instantiated inside the block. Modelsim block contains inputs, clock, reset and output pins as shown in Fig. 8. In addition to these blocks, two other blocks are generated for compiling and simulating

the code. Designed model includes the input signal, hence it is also converted to VHDL code. Therefore, the Modelsim block doesn't have input pin externally other than clock and reset. The inputs can be given externally to the Modelsim block and outputs can be viewed using scope.

4.4 Synthesizing HDL Code Using Xilinx

Generated VHDL code is successfully verified using Co-Simulation [4, 15]. The output of Simulink model matches with Modelsim. Therefore, we can conclude that the generated code is functional and can be implemented in FPGA. The HDL file is simulated and synthesized using Xilinx [17]. After synthesizing code has been mapped to RTL. RTL schematic is shown in the Fig. 10. Based on the number of LUTs required to implement the model, FPGA type can be selected.

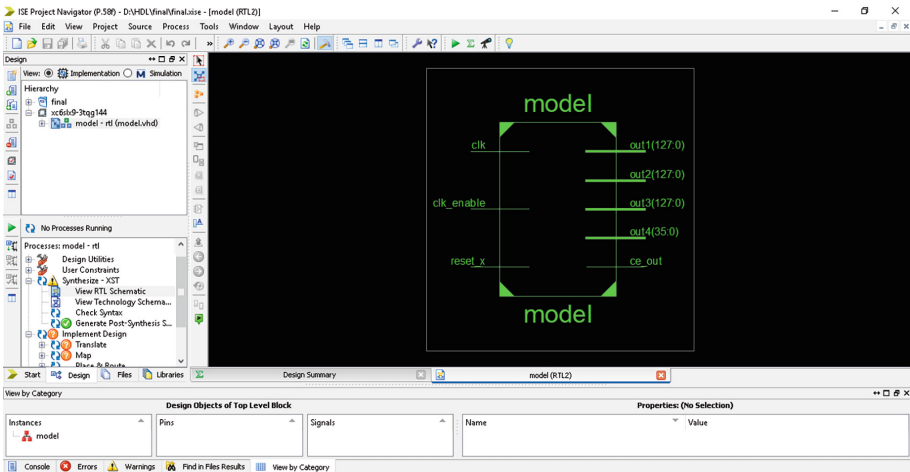


Fig. 10. RTL schematic of Filter bank generated by Xilinx

5 Result and Conclusion

The UWB filter bank architecture designed in Simulink, and it is converted to HDL code. We can conclude that, the Simulink output matches Modelsim output after using the Co-Simulation to verify the code. At the end, VHDL code is synthesized using Xilinx. Figure 11 shows the output of HDL code which signifies that the code is suitable for downloading in FPGA. Thereby implementing the Filter bank architecture in FPGA.

This implementation can be used for tracking or positioning the objects by means of transmitting pulses of sub-nanoseconds with help of carrier, this transmitted signal hits an object and reflects back. The reflected signal is fed to above filter bank architecture followed by ADC's. By performing some DSP for the ADC output exact location of the object can be tracked. In the same way it can be used for Short-range communication where transmitted signal is fed to filter bank architecture.

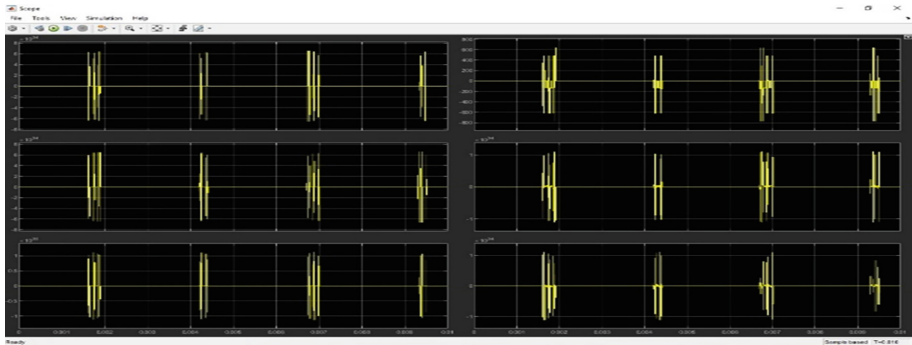


Fig. 11. Output of converted HDL code

6 Future Scope

The UWB filter bank receiver was constructed up to Low pass filters only, due to the Simulink block limitations. This can be improved further by adding ADCs to the Low pass filter's output. As a result, the digital output can be stored in FPGAs as shown in architecture in [1]. Using some DSP algorithms, this data can be utilized to analyze the received signal and also to understand the properties of the received signal. We can trace the exact distance of the object from which the transmitted signal is reflected back by performing some DSP on the stored data. It has many varied applications as mentioned in [7].

References

1. Artiga, X., Berenguer, L.: A filter bank based ultra-wideband receiver front-end for ranging applications. In: IEEE MTT-S International Microwave Workshop on Wireless Sensing, Local Positioning, and RFID, IMWS 2009, Croatia (2009)
2. Mollfulleda, A., Ibars, C., Leyva, J., Berenguer, L.: Practical demonstration of filter bank receiver for ultra-wideband radios. In: 9th European Conference on Wireless Technology, September 2006
3. Zhou, Q.: Sub-Nyquist Sampling Impulse Radio UWB Receivers for the Internet-of-Things. Doctoral Thesis in Information and Communication Technology School of Information and Communication Technology, KTH Royal Institute of Technology Stockholm, Sweden, 2016
4. Othman, N., Jabbar, M.H., Mahamad, A.K., Mahmud, F.: Luo Rudy Phase I excitation modeling towards HDL coder implementation for real-time simulation. In: 2014 5th International Conference on Intelligent and Advanced Systems (ICIAS). <https://doi.org/10.1109/icias.2014.6869495>
5. Tielo-Cuautle, E., Rangel-Magdaleno, J., De la Fraga, L.: Matlab-simulink co-simulation. In: Engineering Applications of FPGAs. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-34115-6_3
6. Federal Communications Commission: Revision of Part 15 of the Commission's Rules Regarding Ultra-Wideband Transmission Systems. First Report and Order, ET Docket 98-153, FCC 02-48, April 2002. <http://www.fcc.gov>
7. Ullah, S., Ali, M., Hussain, Md.A., Kwak, K.S.: Applications of UWB technology. [arXiv: 0911.1681](https://arxiv.org/abs/0911.1681)

8. Werfelli, H., Chaoui, M., Lahiani, M., Ghariani, H.: Receiver design for UWB communications. In: 1st International Conference on Advanced Technologies for Signal and Image Processing - ATSIP'2014, 17–19 March 2014
9. Jiang, S., Skibniewski, M.J., Yuan, Y., Sun, C., Lu, Y.: Ultra-wide band applications in industry: a critical review. *J. Civ. Eng. Manag.* **17**(3), 437–444 (2011)
10. Goyal, V., Dhaliwal, B.S.: Analysis of UWB multiple access modulation scheme using pulse position modulation. *GESJ Comput. Sci. Telecommun.* (2015). ISSN 1512-1232
11. Chen, R., Chiu, P.-L., Yang, H.-L.: Design and performance analysis of DS-UWB rake receiver. In: 2006 IEEE International Symposium on Circuits and Systems (ISCAS), p. 4 pp.- (2006). <https://doi.org/10.1109/ISCAS.2006.1693683>
12. Thomos, C., Kalivas, G.: FPGA-based architecture and implementation techniques of a low-complexity hybrid RAKE receiver for a DS-UWB communication system. *Telecommun. Syst.* **52**, 2115–2132 (2013). <https://doi.org/10.1007/s11235-011-9489-1>
13. Orabi, H., Elnawawy, M., Sagahyroon, A., Aloul, F., Elwakil, A.S., Radwan, A.G.: On the Implementation of a Rotated Chaotic Lorenz System on FPGA, pp. 417–422. <https://doi.org/10.1109/APCCAS47518.2019.8953183>
14. Gregmini, O., Porcino, D.: Field trials of an accurate UWBranging system. In: IET Seminar on Ultra-Wideband Systems, Technologies and Applications, April 2006
15. Othman, N., Mahmud, F., Mahamad, A.K., Hairol Jabbar, M., Adon, N.A.: Cardiac excitation modeling: HDL coder optimization towards FPGA stand-alone implementation. In: 2014 IEEE International Conference on Control System, Computing and Engineering (ICCSCE 2014), pp. 507–511 (2014). <https://doi.org/10.1109/ICCSCE.2014.7072771>
16. Gupta, A.K., Ahmad, A., Samuel, P.: HDL co-simulation of single phase Z-source inverter. In: Students Conference on Engineering and Systems (2013)
17. Sarker, M.A.L., Lee, M.H.: Synthesis of VHDL code for FPGA design flow using Xilinx PlanAhead tool. In: International Conference on Education and e-Learning Innovations, pp. 1–5 (2012). <https://doi.org/10.1109/ICEELI.2012.6360614>



Design and Development of a Virtual Laboratory Using the Python Web Framework Django to Carry Out Experiments in the Stream of ML, DSP, and Digital Communication

Shivaprasad^(✉) and A. S. Dinesha

Department of ECE, Vivekananda College of Engineering and Technology, Puttur, D.K,
Karnataka 574203, India

shivaraol01@gmail.com, dineshaas.ece@vcetputtur.ac.in

Abstract. Hands-on laboratories are crucial for enhancing students' career prospects by enabling them to gain practical experience. However, students often encounter challenges accessing laboratory facilities outside of regular college hours. This paper suggests the development of a virtual laboratory (V-lab) to enhance students' observational and learning abilities, better preparing them for the professional environment. V-lab offers comprehensive instructions for conducting experiments, incorporating authentic client inputs. This platform enables students to engage in experiments related to ML (Machine Learning), signal processing, and DCOM (Digital Communication). The intelligent web server, designed for use by 3rd year ECE (Electronics and Communication) students at VCET, Puttur, has been frequently utilized for conducting experiments in the Advanced Communication Laboratory and DSP (Digital Signal Processing) Laboratory. This website proves beneficial for students working on projects in the field of ML, offering comprehensive information on Classification reports, Confusion matrices, and Accuracy in a single step. The V-lab incorporates various Machine Learning algorithms applied to publicly available datasets such as iris flowers and lung cancer to assess their utility and effectiveness. The results indicated exceptional performance by all available ML algorithms on these datasets. To further improve and expand V-lab services, user feedback is actively collected. Importantly, the website does not need sign-up credentials and is free to use.

Keywords: VLAB · DSP · D-COM · ML

1 Introduction

In the majority of engineering colleges, practical laboratories typically do not operate around the clock. Occasionally, an engineering enthusiast may feel motivated to delve into research or experiments to augment their learning. However, time constraints often hinder their ability to do so. This limitation extends to the institute, as there may not be a teacher or instructor available to assist and guide students with ongoing lab work beyond regular college hours.

In standard laboratory sessions, students might hesitate to clarify their doubts due to language barriers. Occasionally, students may struggle to grasp technical concepts related to a specific experiment due to a lack of prerequisite skills. Furthermore, teachers may inadvertently overlook important topics that need to be covered due to time constraints.

Students are required to adhere to stringent laboratory etiquette when entering and exiting the laboratory. As part of the protocol, students must record their entry and exit times in the attendance ledger maintained for each laboratory session. Additionally, students are obligated to submit their observations, records, and responses to viva questions related to the designated experiments regularly.

Ultimately, students face a drawback in terms of exposure to real-world practical issues. The absence of hands-on experience can pose a challenge for students in capitalizing on opportunities at esteemed institutions.

To address the challenges mentioned in most engineering colleges, a system (server) has been developed. This system accepts valid inputs from students, that are experiment-dependent, executes them, and provides graphical representations for documentation. Built on the widely recognized “web development framework Django”, the system handles tests across various streams with Python programming. Integration of HTML, CSS, and JavaScript enhances the system’s visual appeal and incorporates animation features. Each experiment in the system encompasses detailed steps, theoretical background, and instructions supplemented with practical usage examples. Additionally, the server incorporates instructional videos demonstrating the proper utilization of available tests with valid inputs.

1.1 Features of Lakshya V-Lab

1. Accessible resources anytime and anywhere, contingent on reliable internet connectivity.
2. Enables concurrent experimentation by multiple users.
3. Simplicity and user-friendliness allow for independent experiment execution without the need for instructors.
4. Facilitates swift completion of experiments due to the immediate availability of all required resources and materials.
5. Requires no prior programming experience.
6. Streamlines the application of Machine Learning algorithms to any datasets.
7. Teachers can introduce new experiments to the server with prior approval from the system administrator.
8. Particularly beneficial for students who are uncomfortable posing questions to teachers due to language barriers or feelings of inferiority.
9. Lacks the conventional dos and don’ts associated with physical laboratories.
10. Students enjoy unrestricted access and usage rights for all available experiments

1.2 Mandatory Tools

Mandatory tools used in the design, development, and deployment of the web server into production include.

User Interface (UI) framework:**HTML and CSS**

To generate forms, buttons, navigation bars, links to external websites, and experiment sections on a web browser, HTML scripts are employed. CSS scripts are then utilized to structure HTML elements, alter font styles, separate the page both horizontally and vertically, resize image objects, manipulate background and foreground colors, etc.

JavaScript

JavaScript is applied to incorporate animations into HTML objects, validate the contents of forms, and create dynamic web content, among other functionalities.

Backend framework:

Python: In the system, client inputs for each experiment are managed, processed, and then presented back on the browser through the use of Python programming.

Django: A robust and dependable web framework used for facilitating communication between the client and server through request and response mechanisms.

NumPy: An effective Python package for data processing, equipped with libraries to compute Fourier transforms, determine inner and outer products between scalars and vectors, and design linear algebra problems.

SciPy: A Python package for scientific calculations on data, capable of reading .wav files and designing both FIR (“Finite Impulse Response”) and IIR (“Infinite Impulse Response”) filters.

Scikit-learn (Sklearn): A Scikit learn package employed for applying various Machine Learning classification algorithms on available datasets.

Pandas: A Python package used to read .csv files and manage missing attributes in datasets.

Matplotlib: A Python package utilized for data representation, organization, and visualization.

Mpld3: A Python package employed to display Matplotlib figures on HTML pages, enhancing the visualization experience.

Web Server:

Apache: Open-source HTTP cross-platform web server to deploy the development server project onto production.

mod-wsgi: Apache module to communicate with Django wsgi applications with Apache server for web-hosting.

2 Literature Review

Guerra et al. in [1] discussed the significance of Virtual Laboratories in the realm of DSP. The article outlines a solution approach for remotely performing DSP tests for both professors and students. The approach involves the utilization of TMS320C6713 DSK

and TMDX5505EZDSP boards. However, it's noted that the system, being hardware-driven, exhibited a response time of 30 s during algorithm loading. Additionally, in the event of errors requiring debugging, the recorded response time was 90 s.

Kholoshnia et al. in [2] detailed their objective which was the development of a virtual laboratory designed for instructing different ML algorithms, complete with visualizations of the execution process. In this laboratory, students have the autonomy to construct a map of an area in both 2D as well as 3D, dynamically modify it, and experiment with various algorithms to identify the shortest path. The Q-learning reinforcement learning and NEAT genetic algorithm are employed for determining the shortest path. The programming languages used for implementing the shortest path-finding method on a 2D map are C++ and Python. It's worth noting that the system lacks ML classifiers, which are fundamental components of machine learning algorithms.

Wahyono et al. detailed the development of an intelligent Virtual Laboratory (VL) in [3], which possesses the capability to track the proficiency levels of its users. Students can utilize this Virtual Laboratory based on their achievements in theoretical aspects of electric machines. VLC-UM employs an AI (Artificial Intelligence) backward chaining method to assess the abilities of students preparing for practical coursework. The determination of practicum modules accessible to students is accomplished through the utilization of a forward chaining algorithm. Machine learning techniques are applied to classify learning materials on the basis of student's abilities. The reported accuracy of the system in finding an appropriate module for users was 80%, and in categorizing the learning materials based on user ability, it achieved an accuracy of 90%.

Alexiadis et al., as described in [4], developed the Virtual Laboratory (V-lab) for multimedia applications called MASTERS. This V-lab incorporates 20 GUIs ("Graphical User Interfaces") to facilitate teaching activities related to different DSP notions. The MASTERS V-lab utilizes "MATLAB" programming on the server side to handle requests from clients. However, it's noted that the system incurred a high initial investment due to its reliance on licensed MATLAB tools. Furthermore, the MASTERS V-lab is specifically focused on algorithms within the domain of DSP and does not extend its coverage beyond this area.

Huang et al., as outlined in [5], introduced a Virtual Laboratory for DSP utilizing LabVIEW. The objective of the paper was to establish serial communication between TMS320LF2407 DSP and LabVIEW. The SCI ("Serial Communication Interface") of TMS320LF2407 DSP and the Serial Communication Control Toolbox of LabVIEW were employed to facilitate this communication. It's important to note that the tests carried out with these tools were limited to the realm of DC motor speed control.

Bean et al., as presented in [6], developed the virtual laboratory LaViCAD for analog and digital communications. LaViCAD serves as a valuable tool for verifying the performance of various communication systems and visualizing signal processing techniques commonly covered in electrical engineering degree programs. It's noted that the system is hardware-driven, resulting in delays and incurring a high initial investment cost. The LaViCAD lab specifically focuses on algorithms within the domain of communication systems.

Zohar Dvir, as described in [7], proposed a method for performing DSP that integrates on-site and distance learning environments. The methodology involves the design and

implementation of remote-controllable experiments using an oscilloscope and waveform generator with network-enabled support through the TCP/IP protocol. In particular, the Analog Devices EZKIT LITE board is triggered by the Tavor Electronics WW2572 waveform generator. By using an RS232 to LAN adapter to create connectivity, students may enter test software programs into the EZ-KIT LITE board from remote locations. However, it's noted that the ILM (Integrated Learning Methodology) had limitations on the number of participants who could use it simultaneously.

To address the shortcomings identified in the survey papers mentioned earlier, such as delays in loading algorithms, debugging delays, high initial investment costs, a limited number of experiments, and a narrower domain, the Lakshya V-lab was issued.

3 Methodology

In the present section, we explore the explanation behind picking the “Django Web framework” to develop the V-lab, flowchart, and block diagram of the suggested system.

3.1 Django Web-Framework

1. Django's syntax is designed to be clear and straightforward, which contributes to the readability and maintainability of the code. This is particularly advantageous in collaborative projects and when dealing with complex algorithms.
2. Django comes with robust debugging tools that can help developers identify and fix issues efficiently. Debugging is a crucial aspect of the development process, and having effective tools can save significant time and effort.
3. Django is optimized for performance, providing a fast and reliable request/response cycle. This is essential for web applications that may experience a high volume of traffic or demand quick response times.
4. Django follows the Model-View-Controller architectural pattern, which enhances code organization and separation of concerns. This can simplify the development and maintenance of complex applications.
5. Django includes robust HTTP libraries, allowing developers to handle HTTP requests and responses effectively. This is essential for building web applications that need to communicate seamlessly with clients.
6. Middleware in Django allows developers to process requests globally before they reach the view and responses before, they are sent to the client. This provides flexibility and customization options for handling various aspects of the application.
7. Leveraging Python's power is a significant advantage. Python is recognized for its readability, ease of learning, and versatility. It is generally utilized in scientific computing and data analysis, making it well-suited for domains like DSP, DCOM, and ML.

3.2 Block Diagram of the Suggested V-Lab Server

Figure 1 below depicts the block diagram of the Django web framework applied to design the V-lab server.

The user (client) initiates the interaction by sending a request to the web server for the home page of the V-lab. The URL module of the Django framework collects and processes incoming requests. It determines the appropriate view (handler) on the basis of the authenticated match. The View module processes the request and acts as an intermediary between the client and the database (Model). It communicates with the database to fetch the necessary data. The View module identifies the required templates (JS, CSS, HTML) for rendering and sends the response back to the client. The Model interacts with the database to retrieve or update data as needed by the View. It plays a vital role in managing the application's data and business logic. The client receives the response from the server, displaying the V-lab home page.

Experiment Interaction: The client enters details to conduct experiments through the provided interface on the home page. Each test on the server homepage has corresponding technique handlers, which are Python-based functions. These handlers obtain data, process it, and send back corresponding solutions to the client. The Template module includes HTML, CSS, and JavaScript codes. Python functions render these templates to generate dynamic content for the web pages. Data is sent from the Python environment to the HTML page using jinja-based code snippets. This enables dynamic content rendering based on data processed by Python functions.

Python program-based figures are shown on the HTML page by using the Mpld3 package. This allows for the integration of interactive and dynamic visualizations generated by Python onto the web interface [8].

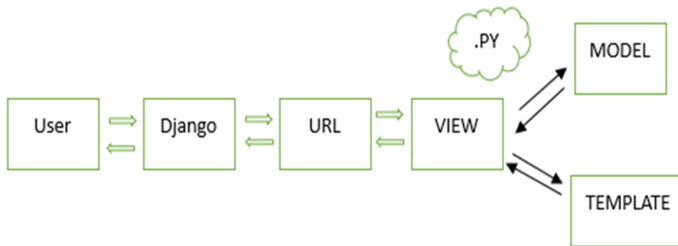


Fig. 1. Django Web framework-based V-Lab

3.3 The Proposed Method's Flow Chart

The V-Lab Server/Client system flowchart is shown in Fig. 2 below. The client sends the server a request for HTTP. The server's URL module uses regular expression programming to authenticate the match request. The view module processes the sent authenticated request, generating the required templates, aggregating the models, and forwarding them to the client. Until the application is closed, the request/reply cycle is repeated.

4 Methods and Computation

This segment presents a selection of valuable algorithms that are seldom integrated into web servers for data processing.

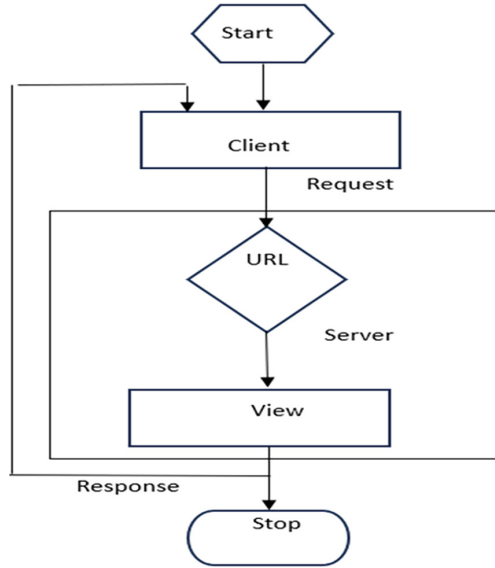


Fig. 2. V-lab server/client system

4.1 Eight-Point Radix2 Decimation in Time Fast Fourier Transform Method

- (i) Break down the 8-point time domain sequence into two 4-point sequences based on even & odd indexing. Express the 8-point Discrete Fourier Transform (DFT) as derived from two 4-point DFTs as outlined below

$$Z(r) = X_1(r) + T_N^r X_2(r) \quad (1)$$

$$Z(r+4) = X_1(r+4) - T_N^r X_2(r+4) \quad (2)$$

where $X_1(r)$ and $X_2(r)$ are four-point Discrete Fourier Transform.

- (b) Break down the 4-point sequences into two 2-point sequences using even and odd indexing. Illustrate the 4-point Discrete Fourier Transform (DFT) by combining two 2-point DFTs, as depicted below.

$$X_1(r) = X_3(r) + T_{\frac{N}{2}}^r X_4(r) \quad (3)$$

$$X_1(r+2) = X_3(r+2) - T_{\frac{N}{2}}^r X_4(r+2) \quad (4)$$

$$X_2(r) = X_5(r) + T_{\frac{N}{2}}^r X_6(r) \quad (5)$$

$$X_2(r+2) = X_5(r+2) - T_{\frac{N}{2}}^r X_6(r+2) \quad (6)$$

- (iii) Finally determine two-point DFT's $X_3(r)$, $X_4(r)$, $X_5(r)$ and $X_6(r)$ [9].

4.2 Overlap Add Method Algorithm

Inputs available $x(n)$, $h(n)$, and block size N .

- (i) Add $L-1$ zeros to the sequence $h(n)$ to extend its length to N , here $N = L + M - 1$, and M is the length of $h(n)$.
- (ii) Break down the sequence $x(n)$ into $x_1(n)$, $x_2(n)$, $x_3(n)$, and so forth. To derive $x_1(n)$, extract the initial L samples of $x(n)$ and append $(M-1)$ zeros at the end. Likewise, obtain $x_2(n)$ by taking the next L samples of $x(n)$ and padding $(M-1)$ zeros at the end. Repeat this process until all samples of $x(n)$ have been covered.
- (iii) Compute the circular convolution between $x_1(n)$ and $h(n)$ and denote the result as $y_1(n)$.
- (iv) Iterate over (iii) to compute $y_2(n)$, $y_3(n)$ and so forth.
- (v) Sum up the last $(M-1)$ samples of each output sequence with the first $(M-1)$ samples of the succeeding output sequences. Arrange the remaining samples sequentially to obtain the final sequence [9].

4.3 DWT Level-1 Decomposition Using Db1 Wavelet

- (i) The time-domain sequence $x_1(m)$ undergoes filtering with both Low Pass and High Pass filters. Low Pass filtering is accomplished using a first-order moving average, denoted as $lpf(m)$, while High Pass filtering is achieved through a moving difference filter, denoted as $hpf(m)$. The respective responses of these filters are indicated below [10].

$$y_{lpf}(m) = x_1(m) * lpf(m) \quad (7)$$

$$y_{hpf}(n) = x_1(m) * hpf(m) \quad (8)$$

- (b) The filters' responses are down-sampled by a factor of two to obtain the final approximation coefficients and detailed coefficients [10].

$$apr(m) = \text{Downsample}(y_{lpf}(m)) \quad (9)$$

$$det(m) = \text{Downsample}(y_{hpf}(m)) \quad (10)$$

4.4 Most Specific Hypothesis Determination Algorithm

Let S signify the hypothesis space, where the hypothesis space encompasses a set of hypotheses (denoted as s) that are consistent with the target value. I represent the instance space, where i 's are instances of I .

- (i). Assign the hypothesis set s to the most specific hypothesis in H .
- (ii). As long as every positive instance in i
 - As long as every attribute value v_i in s
 - If the attribute value v_i is consistent with i
 - No changes in s
 - Otherwise, update v_i in s by the more common attribute value that is consistent with i
- (iii) List the hypothesis s [12].

5 Experiment Conduction Using V-Lab

This segment furnishes a detailed, step-by-step procedure for conducting experiments in the domains of DSP, DCOM, & ML.

5.1 Basic DSP Experiments

- (1) Visit Lakshya vcetputtur website, Fig. 3 shows the server home page. Select DSP exp, swipe to Basic DSP, and choose Convolution.



Fig. 3. V-lab server homepage

- (B) Under the convolution user interface section, provide the sequences $x_1(m) = [1, 1, 1]$ and $x_2(m) = [1, 1, 1]$ as revealed below in Fig. 4 and hit the submit button.
- (C) The outcome of linear and circular convolution with the essential graph is furnished back to the user for verification and observations, as shown below in Fig. 5 [13–16].

CONVOLUTION

Inputs

Enter the first sequence

1 1 1

Example: 1 2 3

Enter the second sequence

1 1 1

Example: 1 2 3

Submit

Example for Linear Convolution

$$\sum_{k=-\infty}^{\infty} x(k)h(n-k)=y(n)$$

Eg: let $x(n)=\{1,2,3\}$ and $h(n)=\{1,1,1\}$

$$y(0)=\sum x(k)h(-k)=x(0)h(0)=1$$
$$y(1)=\sum x(k)h(1-k)=x(0)h(1)+x(1)h(0)=3$$
$$y(2)=\sum x(k)h(2-k)=x(0)h(2)+x(1)h(1)+x(2)h(0)=6$$
$$y(3)=\sum x(k)h(3-k)=x(1)h(2)+x(2)h(1)=5$$
$$y(4)=\sum x(k)h(4-k)=x(2)h(0)=3$$

Example for Circular Convolution

$$\sum_{k=0}^{N-1} x(k)h((n-k))_N=y(n)$$

Eg: let $x(n)=\{1,2,3\}$ and $h(n)=\{1,1,1\}$

$$y(0)=\sum x(k)h((-k))_3=x(0)h(0)+x(1)h(2)+x(2)h(1)=6$$
$$y(1)=\sum x(k)h((1-k))_3=x(0)h(1)+x(1)h(0)+x(2)h(2)=6$$
$$y(2)=\sum x(k)h((2-k))_3=x(0)h(2)+x(1)h(1)+x(2)h(0)=6$$

Fig. 4. Convolution user interface

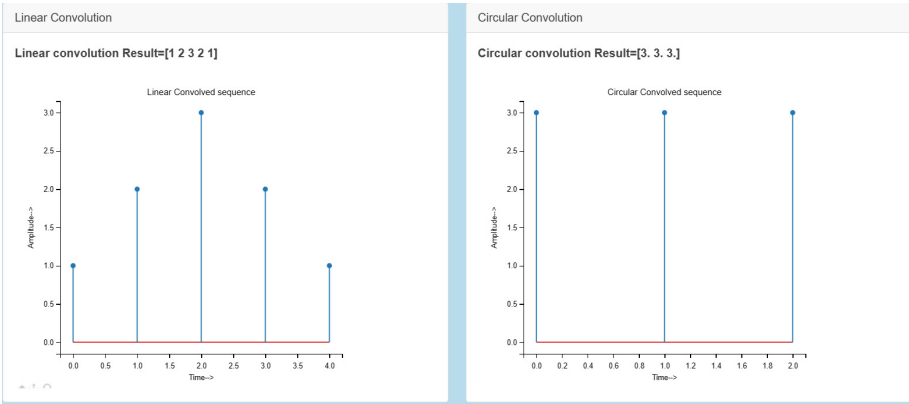


Fig. 5. Linear and Circular convolution outcomes

5.2 Advanced DSP Experiments

- (1) Under the homepage of Lakshya, mouse over on DSP exp, swipe over to Advance DSP, and choose MFCC.
- (2) Under the Mel frequency cepstrum coefficients user interface segment shown in Fig. 6, provide the information about the speech file to manipulate, block size, the proportionality of overlap in the block, and total filter banks to process the data.
- (C) The outcome of the Mel frequency cepstrum coefficients algorithm would be furnished with a graph, as revealed in Fig. 7. This test shows speech, its information with respect to time and frequency and Mel frequency cepstrum coefficients [11, 14–17].

5.3 Digital Communication Experiment Conduction Procedure

- (1) Under the Lakshya homepage, mouse over DCOM experiments, and choose PSK (“Phase Shift Keying”) modulation.

Inputs

Select wav file for Feature extraction

for1.wav

Enter the Frame size(Window size)

Example:512

Enter the Percentage of overlap in frame size

Example:0.5

Enter the No.of filter banks

Example:20

MFCC algorithm usage guidelines

1. Provide .wav file only.
2. wav sample files are available for download [here](#).
3. These algorithm returns Original speech signal and MFCC features.
4. The dimension of MFCC features depends on filter bank size, provided by the user.
5. Data furnished by these algorithm may used in Speech recognition system for validation

Fig. 6. Mel frequency cepstrum coefficient user interface

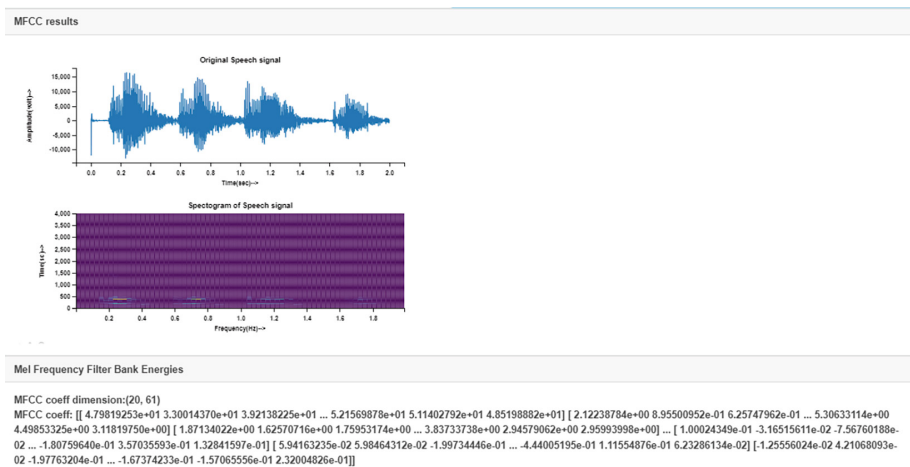


Fig. 7. Speech envelope, Time-frequency information of speech, and Mel frequency coefficients

- (2) Under the PSK modulation user interface segment shown in Fig. 8, provide the data including F_s , F_c , T_b , and total run time, and submit the form.
- (3) Phase shift keying modulated waveform together with input data and modulating data is furnished as shown in Fig. 9 [14–16, 18].

5.4 ML Classifier Experiment Conduction Procedure

- (1) Under the server homepage, mouse over on ML Classifiers, and choose Linear discriminant analysis.
- (2) Server consists of useful datasets for classification. Choose any dataset file, the file should be of.csv file type, and enter submit to initiate processing as shown in Fig. 10 below.



Fig. 8. Phase shift keying user interface

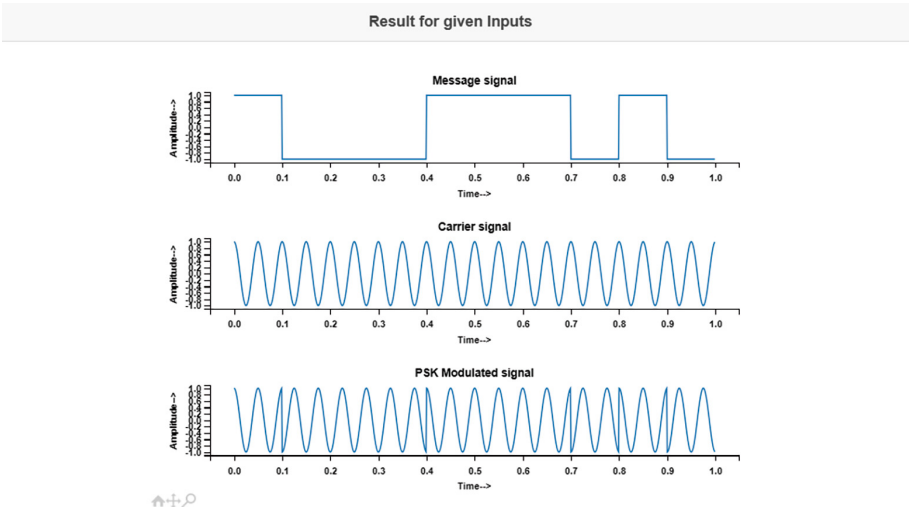


Fig. 9. Data, Modulating data, and Phase shift keying data

- (C) Accuracy scores including precision, recall, and f1-score are displayed. The lung cancer dataset is used for classification, Fig. 11 below shows the accuracy report provided by the server
- (D) Confusion Metrics and Testing data set accuracy utilized on the lung cancer Dataset, redirected by the server is revealed in Fig. 12 [19, 20].

6 Findings and Discourse

This section examines how several ML techniques available in VLab are compared using datasets related to cancer and iris.

LINEAR DISCRIMINANT CLASSIFIER

Inputs

Select csv file for Classification

Choose File

No file chosen

Enter the test dataset size

Example:0.25

Submit

Linear Discriminant Analysis(LDA) Classifier usage guidelines

1. Provide .csv file only.

2. Remove column and row names from the csv file if existing.

3. Last column of the CSV file should have class label in integer form.

4. No. of components for classifier is to default= min(no of features,no of classes-1).

5. CSV sample files are available for download [here](#).

Fig. 10. LDA user interface

Classification report				
	precision	recall	f1-score	support
NO	0.750000	0.333333	0.461538	9.000000
YES	0.918919	0.985507	0.951049	69.000000
accuracy	0.910256	0.910256	0.910256	0.910256
macro avg	0.834459	0.659420	0.706294	78.000000
weighted avg	0.899428	0.910256	0.894567	78.000000

Fig. 11. Accuracy metrics on Lung cancer Dataset

Confusion Matrix			
		0	1
0	8	2	
1	3	65	
Testing Dataset Accuracy: 0.9358974358974359			

Fig. 12. Confusion Matrix and Accuracy of Lung cancer data

6.1 Utilization of Diverse ML Algorithms on the Iris Dataset for the Purpose of Comparisons

In the segment, the V-Lab machine learning classifiers are employed on the iris dataset, and the performance metrics provided using ML classifiers are documented. Petal width, petal length, sepal length, and sepal width are the four features of the iris flower that are represented in the 150 cases that make up the dataset. The dataset can be obtained for download [21]. The dataset has 140x4 = 600 values in total. The training and testing sets of the dataset are divided, and the ratios of testing to training are set at 80:20, 75:25, and 70:30, respectively [19].

Table 1 and Fig. 13 Illustrate the accuracy of the testing dataset, computed with different ML classifiers used for the iris flower dataset, with varying sizes for the testing dataset.

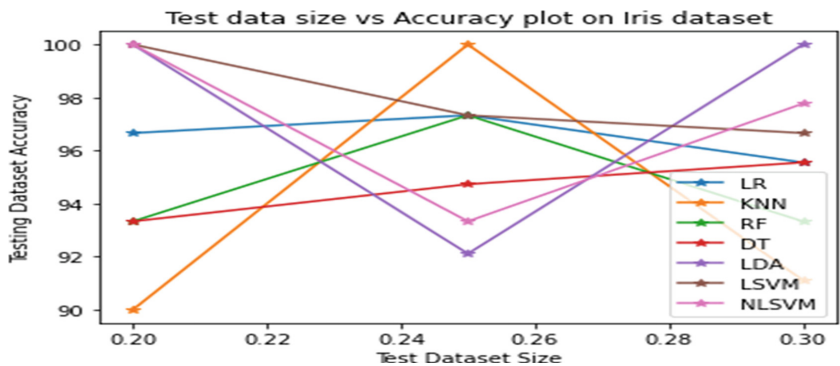


Fig. 13. Sample size versus accuracy viewed on Iris Dataset

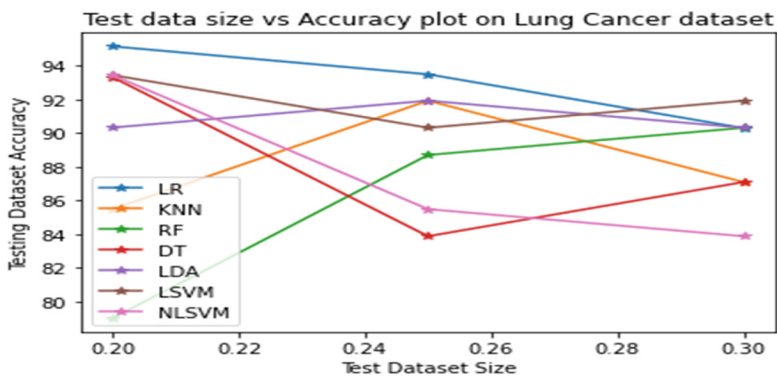
6.2 Utilization of Diverse ML Algorithms on the Lung Cancer Dataset for the Purpose of Comparisons

In this segment, V-Lab Machine learning classifiers are deployed on a dataset related to factors influencing lung cancer. The dataset is accessible for download, and the performance metrics provided by ML classifiers are documented [22]. Comprising 15 attributes, the dataset encompasses 309 instances. Among these instances, 270 pertain to patients with cancer, while 39 instances represent patients without cancer. The dataset is partitioned into training and testing sets, with the training-to-testing ratio configured at 80:20, 75:25, and 70:30, respectively [19].

The accuracy of the testing data set, as determined by applying different machine learning classifiers to the lung cancer dataset with varying testing dataset sizes, is shown in Table 2 and Fig. 14 below.

Table 2. Performance evaluation of ML Algorithms applied to the lung cancer dataset

ML Algorithm	Accuracy Sample strength (0.2)	Accuracy Sample strength (0.25)	Accuracy Sample strength (0.3)
LR	95	93	90
K-Nearest neighbor	85	92	87
RF	79	89	90
DT	93	84	87
Linear Discriminant Analysis	90	92	91
SVM (linear)	93	90	92
SVM (RBF)	93	85	84

**Fig. 14.** Sample size versus accuracy viewed on the Cancer dataset

6.3 Student Evaluation and Impact Analysis

In this segment, an evaluation is conducted on students' feedback regarding the V-lab tool and its effect on the VTU (Visvesvaraya Technological University) end-semester exam for the Digital Signal Processing (DSP) course during the academic year 2021–22.

The primary objective was to solve the DSP course and DSP Lab. V-Lab comprehensively covered all the tests outlined in the DSP Lab curriculum. Furthermore, it included essential DSP algorithms integral to the course, such as the overlap-add/save technique, DIF-FFT, Circular convolution, Radix-2 DIT-FFT, and Linear convolution using DFT & IDFT.

The noteworthy outcome of this implementation is the attainment of a 100% pass rate in both the DSP course and DSP Lab in the academic year 2021–22. This success suggests a positive impact of V-Lab on the student's understanding and performance in DSP-related studies.

Table 3 and Fig. 15 contain questions and corresponding ratings given by third-year Electronics and Communication (EC) students during the academic year 2021–22. A

Table 1. Performance evaluation of ML Algorithms applied to the iris flower dataset

ML Algorithm	Accuracy Sample strength (0.2)	Accuracy Sample strength (0.25)	Accuracy Sample strength (0.3)
LR	97	97	96
K-Nearest neighbor	90	100	91
RF	93	97	95
DT	93	95	96
Linear Discriminant Analysis	100	92	100
SVM (“Support Vector Machine”) (linear)	100	97	97

(continued)

Table 1. (continued)

ML Algorithm	Accuracy Sample strength (0.2)	Accuracy Sample strength (0.25)	Accuracy Sample strength (0.3)
SVM (RBF)	100	95	98

total of 32 pre-final students participated in V-lab sessions and contributed feedback to enhance the tool. This feedback from students is valuable for ongoing improvements and optimizations of the V-lab tool to better meet the educational needs and expectations of the users.

7 Conclusions

The suggested V-lab server to enhance the practical knowledge of students in the Electronics and Communication (EC) stream. It allows students to engage with the tool beyond regular laboratory hours, providing a flexible and accessible platform for practical learning. All experiments pertaining to Digital Signal Processing (DSP), Digital Communication (DCOM), and Machine Learning (ML) streams can be efficiently conducted within an hour, yielding prompt results.

The V-lab not only facilitates quick experimentation but also motivates students to undertake projects in the field of Machine Learning. It offers essential analytical metrics, including the Confusion Matrix, Classification Report, Actual Class vs. Predicted Class, along Testing Dataset Accuracy. This feature enables students to assess the effectiveness of various ML classifiers without requiring a strong programming background.

Furthermore, to ensure the preservation of data for future analysis, the V-lab allows users to store critical information such as the Classification Report and Actual versus Predicted Label Classes in formats like Excel or PDF.

Table 3. Questions versus Rating.

Questions	1	2	3	4	5	AVG
(i) The features of the V-Lab GUI are designed to be user-friendly and straightforward, ensuring ease of use for the users	-	-	-	3	29	4.93
(ii) Experiments concerned DSP Lab are covered effectively	-	-	-	1	31	4.93
(ii) Your ability to learn and observe was improved by V-Lab	-	-	3	1	28	4.81
(iv) The incremental results obtained through the overlap-add and overlap-save methods played a crucial role in identifying and correcting errors that may have occurred during manual calculations	-	-	1	3	28	4.81
(v) The incremental results obtained through the Radix-2 DIT and Rasix-2 DIF FFT methods played a crucial role in identifying and correcting errors that may have occurred during manual calculations	-	-	2	2	28	4.78

“1-Very Poor; 2-Poor 3-Satisfactory; 4-Good; 5-Excellent”

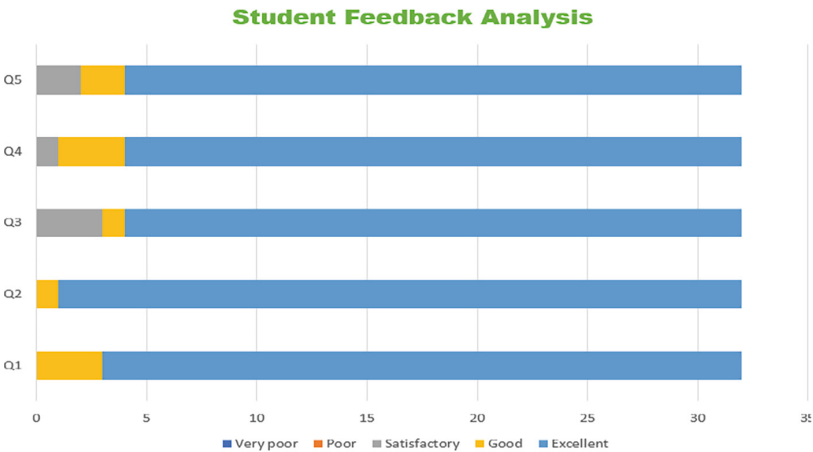


Fig. 15. Feedback report

The effect of the developed tool was assessed by collecting feedback from users, and the responses indicate that students who utilized the V-lab for experimentation found it highly useful for their curriculum. This positive reception suggests that the tool effectively contributes to the learning students experience in the EC stream.

References

1. Guerra, J.A., Dominguez-Cid, S., Guerrero, J.I., Garcia, A., Larios, D.F., Leon, C.: Virtual laboratory for digital signal processing. In: 2020 XIV Technologies Applied to Electronics Teaching Conference (TAEF), pp. 1–6 (2020). <https://doi.org/10.1109/TAEF46915.2020.9163709>
2. Kholoshnia, V.D., Boldyreva, E.A.: The virtual machine learning laboratory with visualization of algorithms execution process. In: Uskov, V., Howlett, R., Jain, L. (eds.) Smart Education and e-Learning 2020. Smart Innovation, Systems and Technologies, vol. 188, pp. 221–230. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-5584-8_19
3. Wahyono, I.D., Putranto, H., Asfani, K., Afandi, A., Sunarti: VLC-UM: a novel virtual laboratory using machine learning and artificial intelligence. In: 2019 International Seminar on Application for Technology of Information and Communication (iSemantic), Semarang, Indonesia, pp. 360–365 (2019). <https://doi.org/10.1109/ISEMANTIC.2019.8884288>
4. Alexiadis, D.S., Mitianoudis, N.: MASTERS: a virtual lab on multimedia systems for telecommunications, medical, and remote sensing applications. IEEE Trans. Educ. **56**(2), 227–234 (2013). <https://doi.org/10.1109/TE.2012.2210224>
5. Huang, F.H., Li, Y.: Research of DSP virtual laboratory based on LabVIEW. Adv. Mater. Res. **503**, 236–240 (2012). Trans Tech Publications, Ltd. <https://doi.org/10.4028/www.scientific.net/amr.503-504.236>
6. Bean, M., Sugranes, A., Berzosa, C., Berzosa, F.: Virtual analog and digital communications laboratory: LaViCAD: online interactive tool for learning communications systems, pp. 429–434 (2010). <https://doi.org/10.1109/EDUCON.2010.5492544>
7. Dvir, Z.: Web-based remote digital signal processing (DSP) laboratory using the integrated learning methodology (ILM). In: Badica, C., Paprzycki, M. (eds.) Advances in Intelligent and Distributed Computing. SCI, vol. 78, pp. 257–262. Springer, Berlin, Heidelberg (2008). https://doi.org/10.1007/978-3-540-74930-1_27
8. Django Software Foundation (2019). Django. <https://djangoproject.com>
9. Proakis, J.G., Manolakis, D.G.: Digital Signal Processing Principles, Algorithms, and Applications, 3rd edn. Prentice-Hall, Englewood Cliffs (1996)
10. Parkale, Y.V., Nalbalwar, S.L.: Application of 1-D discrete wavelet transform-based compressed sensing matrices for speech compression. Springerplus **5**, 2048 (2016). <https://doi.org/10.1186/s40064-016-3740-x>
11. Han, W., Chan, C.-F., Choy, C.-S., Pun, K.-P.: An efficient MFCC extraction method in speech recognition. In: 2006 IEEE International Symposium on Circuits and Systems (ISCAS), Kos, Greece, p. 4 pp.- (2006). <https://doi.org/10.1109/ISCAS.2006.1692543>
12. Mitchell, T.M.: Machine Learning (1997)
13. Ingle, V.K., Proakis, J.G.: Digital Signal Processing Using MATLAB, 3rd edn. CL-Engineering (2011)
14. Harris, C.R., Millman, K.J., van der Walt, S.J., et al.: Array programming with NumPy. Nature **585**, 357–362 (2020). <https://doi.org/10.1038/s41586-020-2649-2>
15. Hunter, J.D.: Matplotlib: a 2D graphics environment. Comput. Sci. Eng. **9**(3), 90–95 (2007)
16. “Bringing Matplotlib to the Browser”. github.com. <https://mpld3.github.io>. Accessed Sept 2023

17. Virtanen, P., et al.: SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**(3), 261–272 (2020)
18. Haykin, S.: *Communication Systems*. John Wiley & Sons, New York (2008)
19. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O.: Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)
20. Mckinney, W.: *pandas: a Foundational Python Library for Data Analysis and Statistics*. Python High-Performance Science Computer (2011)
21. Fisher, R.A.: Iris. UCI Machine Learning Repository (1988). <https://doi.org/10.24432/C56C76>
22. Stacy Robert “Survey Lung cancer”. <https://data.world/sta427ceyin>. Accessed Jan 2023

Image/Video Analysis and Processing



Swift Detection of Human Fall Events in Compressed Videos

S. M. Praveenkumar^(✉), Prakashgoud Patil, and P. S. Hiremath

Department of Master of Computer Applications, KLE Technological University,
Hubballi 580031, Karnataka, India
{praveenkumar_sm, prakashpatil, pshiremath}@kletech.ac.in

Abstract. Herein, a novel methodology MVCNN for real-time human fall event detection in the compressed domain of videos using motion vectors (MV) and revamped YOLOv3 using CNN. The videos in MPEG-4 and H.264 compression are considered for the present study. Any video source without any prior setup could be considered by adapting the proposed method to various video codecs and camera settings. Existing algorithms for human fall event detection in a compressed domain video have some limitations in this regard such as (i) requirement of keyframes at a fixed interval, (ii) usage of P frames only, and (iii) support for a single codec only. These limitations are overcome in the proposed method by using arbitrary keyframe intervals, using both P and B frames, and supporting more than one codec, e.g. MPEG-4 and H.264 codecs. The experimentation is carried out using the benchmark datasets, namely, Le2i, UR, and Multiple Cameras Fall Event Datasets. The fall event detection accuracy of the proposed method in the compressed domain is found to be comparable to that observed in raw video data by using other recent methods. The proposed method outperforms other methods in the literature on the compressed domain by significantly improving the fall event detection inference in the compressed domain.

Keywords: Convolution Neural Network · Video Processing · Data Association · Motion Vector

1 Introduction

Fall events during human activities are of common occurrence with significant concern. Injuries caused during the fall events are often related to death of senior citizens and some patients [1]. Fall event detection is vital for the safety of the children, elderly and patients in solo indoor and crowded outdoor scenes. Video surveillance of such scenes is one of the effective technological means of dealing with this kind of security concerns.

Video analytics is one of the promising computer vision approaches for human activity detection. Videos provide rich information, including time-based steadiness in successive frames, which are utilized in groups for enriching data analysis. In the video analytics field, two popular network architectures, namely, two-stream ConvNet [2–4] and 3D-CNN [5–7], are often employed for video processing. To analyze the temporal

information, the two-stream ConvNet uses decoded RGB video frames and optical flow. The task of extracting optical flow consumes more time out of the overall pre-processing time. In the 3D-CNN, the temporal information is analyzed using spatio-temporal filters, but the large number of parameters involved in the 3D convolution operation lowers the inference speed. To overcome these limitations, the recent tendency in analyzing video is to use the transformers [8–12], which achieve better results.

In recent years, video analytics in the compressed domain has received much attention of several researchers [13–17], which established the advantages of taking video input directly from the compressed domain. These methods work on motion vectors and residuals, which have half the time complexity in comparison with optical flow methods and yet yield viable results [14]. Wu et al. [16] have developed a method, namely, CoViAR, for action recognition based on deep learning in compressed videos. Herein, three different 2D-CNNs are used, one each for training with each of the three types of data, namely, the I-frame, and the P-frame with the residuals and motion vectors, in the compressed domain. In order to enhance performance, optical flow is also accounted, which essentially decodes the video for the purpose of extracting the desired optic flow, and thus defeats the main advantage of operating directly in the compressed domain itself. Further, it enhances the computational cost in terms of memory space and time, since each optical flow frame computation time required is in the range 20–80 ms. In [18, 19], the knowledge learnt in optical flow is transferred by correlating the optical flow and motion vectors. In [9], the DMC-Net, which is an improvement over CoViAR, is proposed for motion vector noise reduction by employing a lightweight generator and capturing finer motion details, wherein optical flow computations are avoided and thus significantly reduce the computation time. As in CoViAR, the DMC-Net also uses multiple 2D-CNN models, one each for each of the different modalities.

In the literature, several methods for detection of human fall event in the compressed domain videos have been reported. The limitation of these methods arises due to the usage of a keyframe interval of fixed length and only the P frames, which consequently reduce the video processing rate and visual quality in the compression domain [20]. Moreover, setting a specific codec, length of keyframe interval, or type of frame might not be allowed in some video sources.

With a motivation to overcome the observed limitations of the existing methods, it is proposed to develop a human fall event detection in a compressed domain video data streaming by employing deep learning techniques. The videos in MPEG-4 and H.264 compression format are considered for the present study. The proposed method comprises the spatio-temporal fall event detection using motion vectors, and event tracking by revamped YOLO. In every k th frame, the object detector of YOLO produces bounding boxes for detected objects. As in MVint[18], these boxes are circulated in-between frames, which depends on the mean or median of motion vectors bounded by the preceding boxes.

Major improvement in the present work on methodology for fall event detection in compressed videos over previous works is the flexibility provided in the support for compressed video analytics with the usage of: (i) key frame intervals of varying size, and (ii) P as well as B frames. This technique allows for video compression format variations and thus making it more amenable for application to a variety of real-world scenes. The

validation of the experimental results of the proposed method is done by performing detection of fall events in the videos taken from the benchmark video datasets, namely, Le2i, UR, and Multiple Cameras Fall Event Datasets. The analysis of its performance is done for different codecs, namely, MPEG-4 and H.264, and frame types, namely, P and B frames.

The main novel contributions in the present work are summarized as following:

- A fast real-time fall event detection algorithm in the compression domain (MPEG-4 or H.264) using motion vectors and revamped YOLO v3, which supports (i) keyframes at intervals of variable length, and (ii) P as well as B frames.
- The validation of the experimental results of MVCNN is done by performing Fall event action recognition in the videos taken from the benchmark video datasets, namely, Le2i, UR, and Multiple Cameras Fall Event Datasets, in terms of parameters, GFLOPS, accuracy and speed.
- Ablation study has been done to analyze the performance of proposed method for different codecs, namely, MPEG-4 and H.264, and MVCNN variants in comparison with the state-of-the-art methods.

The paper is organized into five sections. The Sect. 2 contains a survey of the recent related work. In the Sect. 3, the proposed fall event detection algorithm is described. In the Sect. 4, the experimental results and the algorithm performance analysis are presented. Finally, in the Sect. 5, the conclusions are given.

2 Related Work

Automatically detecting human fall events of the senior citizens and kids using surveillance cameras is one of the significant issues for home/public health care in solo scenes and crowded scenes. Present fall event detection methods depend on specifically designed wearable or portable devices, which could be in convenient for elderly people to wear or carry on their person all the time. One of the promising techniques for visual surveillance applications is computer vision. These techniques are widely used in many computer-assisted visual tasks such as object tracking, event detection, personal identification, and behavior understanding and description, etc., due to the affordable video cameras. Research efforts have been carried out for achieving computer vision-based detection of fall events and other events in compressed video analytics [21–24]. In [22], a method was developed to detect fall events in a compressed domain using a portion of the video, assuming events are happening in discontinuities in a motion field. In [23], object tracking in a video captured by an Omni-directional camera is done using the Gaussian mixture model (GMM) for fall events classifier.

2.1 Video Compression

Digital video compression achieves reduction in the number of bits per frame of a video by removing spatial as well as temporal redundancies and thus transforming the video to a form suitable for its efficient storage and transmission. In video codecs, a video is split into a sequence of frames, which comprises an Intra frame (I-frame) followed by a

sequence of inter frames (P- or B-frames). Herein, I-frame is a self-contained RGB frame with full visual representation, while P-frames or B-frames only represent the changes that occur with respect to the preceding frame. The data representation in P-frames are the motion vectors and in B-frames the residuals. Motion vectors denote the shift of a block of pixels in a frame to the corresponding region in its next frame. The residual denotes the error of the motion prediction.

2.2 Solitary Scene Fall Event Detection

Most existing methods have used handcrafted features and traditional machine learning techniques for fall event detection till now [1, 26]. In comparison with other events, fall events are characterized by significant changes in the posture of the falling object and higher speed of these changes during occurrence of the event. Vishwakarma et al. [42] devised a method based on posture changes in the object based on the concept of finite state machine. Herein the aspect ratio is used for state transition in the finite state machine, the dip angle and gradient of bounding boxes are used for detection of the fall event. But, more the dependency on the aspect ratio, poorer the robustness of the method. Chua et al. [43] used angle and distance information to detect the fall event, wherein the posture state was represented by extracting three critical points that determine the angle of the connection line and distance relation. Change in posture state is used for final classification of event. The depth image-based method was proposed by Stone and Skubic [44], in which the confidence in a fall event is calculated based on five quantities: velocity, acceleration, ground projection rate, mean and change in occlusion. This method has the limitation that the operative range of the depth camera is narrow. Rougier et al. [45] devised a method that uses shape features and Motion History Image (MHI) for detection of a fall event, which is characterized by the intensity of the motion based on MHI in combination with the direction of the fitting ellipse. Herein, good results are obtained because of rich motion history associated with a particular posture.

Fall event detection based on neural network features has been attempted by some researchers. In [40], Principal Component Analysis Net (PCANet) is developed for fall event detection. To extract features and obtain image class labels, Chan et al. [46] used the PCA network and SVM, and the labels are then employed as the features in the fall detection process. Wang et al. [41] proposed a fall detection method utilizing features of HOG, LBP and neural networks, and achieved improvement over PCANet features.

2.3 Detection of Abnormal Event in Crowded Scene

Recently, the focus of researchers is on complex scene abnormal event detection in a crowded environment [26, 27]. Complex scene-based fall event detection is critical because of the difficulty in identifying abnormal events in a crowded and complex scene due to occlusion of the falling object. The main challenge in the crowded and complex scene is identifying and tracking an individual pedestrian. GAN (Generative Adversarial Network) based method for abnormal event detection has been proposed by Ravanbakhsh et al. [30], wherein image difference between the original image and the matching generated image of different areas was used for abnormal event detection. In

this method, it can only detect uncommon events caused by nonhuman objects like cars, bicycles, etc., but the events are not classified. The global features-based method was proposed by Marsden et al. [29], wherein features such as mean-motion speed, crowd conflict, collectiveness of crowd, and density of group are used to detect crowd events detection.

3 Proposed Methodology

3.1 Overview

The proposed approach MVCNN for human action recognition associated with human fall event in a compressed video is based on revamped YOLO v3 [31] and motion vectors. The human fall event considered is the fall of a pedestrian. The compressed domain video codecs MPEG-4 and H.264 are considered.

Proposed Approach: It has three main stages: fall event action detection, event box prediction and data association, as shown in the Fig. 1. The process of object tracking involves two steps: prediction and update. During the event box detection stage, new action A^t at time t is identified by utilizing the previous action A^{t-1} at time $t-1$ and motion vectors M^t at time t . In the update step, that is executed on each frame in intervals of fixed length K , resets the accumulated error and handles the appearing and disappearing person objects. During update step, fall event action detector, namely, revamped YOLO v3, detects the action D^t . Detected events having confidence scores above the threshold θ_c are matched with predicted event boxes in A^t in data association stage, where in all pair wise intersection-over-union (IoU) distances are computed and Hungarian method [22] for assignment of detected event action to predicted event action through motion vectors is implemented. Two event boxes are said to be matched if IoU distance between them is above the threshold value θ_{IoU} . The revamped YOLO v3 is trained on compressed videos, namely, MPEG-4 and H.264.

Revamped YOLO v3: For the fall event action detection step, YOLO v3 framework [32] is modified by using a compressed VGG network [33] as the front network of the framework, which realizes the network complexity reduction from 65 Bflops to 39 Bflops. The main target of the algorithm is human fall event detection in a video input. The usage of very small anchor settings does not affect performance of the detection algorithm and reclustering the size of anchors is done by using K-Means algorithm. Further, the complexity is reduced to 34 Bflops by removing the final FPN structure of the YOLO v3 network because of the targets of larger scale in the front-down camera. To avoid severe target occlusions, soft-NMS method is used, which performs better than the traditional non-maximum suppression (NMS) [34]. In soft-NMS method, the bounding box with IOU greater than the threshold and lower confidence is not removed directly, instead its confidence is reduced. Thus, in a scene of dense targets, it ensures the removal of false positives of detected targets correctly and also the reduction of false negatives.

3.2 Motion Vectors in Compressed Domain

To reduce the byte size for storage or transmission of a video in compressed domains, the encoded video contains only a few key frames in full form. Each of the non-key frames, that are between any pair of consecutive key frames, is segmented into macroblocks and then motion vectors are computed which point to macroblocks having similar appearance in the subsequent frames (depicted as red arrows in Fig. 2). Thus, reuse of information from adjacent frames leads to an efficient encoding of the content in a video. In MPEG-4, each macroblock is of size 16×16 pixels, while in H.264, macroblock is of size varying between 4×4 and 16×16 pixels. Between two consecutive key frames, there is a sequence of video frames which could be of type P-frame or B-frame. A non-key frame is of type P-frame when motion vectors have reference only to preceding frames, whereas it is of type B-frame when motion vectors have references to both preceding and succeeding frames. The non-key frames in MPEG-4 are only P frames, whereas the H.264 utilizes P as well as B frames. The motion vectors, which imply the object motions in a video, assist in visual human action recognition.

The motion vectors are computed in compressed domain videos, say MPEG-4 or H.264 [20]. In the set $M^t \subset Z^{10}$ of motion vectors computed in the frame F^t at time t , the number $|M^t| = j^t$ of motion vectors may vary with change in frame. For the key frames, $M^t = \emptyset$. Each element $m_j^t \in M^t$ represents a motion vector given by the ten-tuple

$$m_j^t = (s_j^t, v_{j,w}^t, v_{j,h}^t, x_{j,src}^t, y_{j,src}^t, x_{j,dst}^t, y_{j,dst}^t, v_{j,x}^t, v_{j,y}^t, v_{j,s}^t) \quad (1)$$

where the vector's reference frame $F^{t+s_j^t}$ has the offset s_j^t with respect to the current frame F^t . In case of a P vector, s_j^t is negative, while it is positive for a B vector. The $v_{j,w}^t$ and $v_{j,h}^t$ denote the width and height, respectively, of the vector's macroblock. As illustrated in the Fig. 3, each motion vector in the current frame starts at the center of its corresponding macroblock $(x_{j,dst}^t, y_{j,dst}^t)$ and points to the center of the macroblock $(x_{j,src}^t, y_{j,src}^t)$ of similar appearance in the reference frame. The $v_{j,x}^t$ and $v_{j,y}^t$ denote x- and y-components of motion vector, after scaling by $v_{j,s}^t$ which satisfy giving values for the components.

$$x_{j,src}^t = \lfloor x_{j,dst}^t + v_{j,x}^t / v_{j,s}^t \rfloor, y_{j,src}^t = \lfloor y_{j,dst}^t + v_{j,y}^t / v_{j,s}^t \rfloor \quad (2)$$

3.3 Handling of Key Frames

In earlier methods for action recognition in the literature, the update step is performed on key frames only, while action recognition step is on non-keyframes only, since motion vector computation, that is needed for action recognition, is not done in key frames. Thus, it requires key frames at intervals of fixed length. However, the proposed method adopts a novel strategy for action recognition and updation steps, which are scheduled in regular intervals irrespective of the frame type that is encountered. When a key frame is encountered in an action recognition step, the motion vectors of its preceding frame are considered, i.e., set $Mt = Mt - 1$, and then action recognition is performed on it. The

underlying assumption is that there is not much change in the action performed by the object in between two consecutive frames, and thus, $\mathbf{M}^t - 1 \cong \mathbf{M}^t$ and it could be used for action recognition in case of a key frame.

3.4 Event Box Prediction

In the event action box prediction stage, the algorithm predicts event action $\mathbf{A}^t$ at time t based on action $\mathbf{A}^{t-1}$ at instant $t-1$ and motion vectors $\mathbf{M}^t$ at time t . The algorithm computations, which are designed to handle the case of P frame only or the case of both P and B frames, comprises the following steps:

- a) Construct the subset $\tilde{\mathbf{M}}^t \subset \mathbf{M}^t$ of motion vectors having non-zero magnitude:

$$\tilde{\mathbf{M}}^t = \left\{ m_j^t \in \mathbf{M}^t \mid ((v_{j,x}^t)^2 + (v_{j,y}^t)^2)^{1/2} > 0 \right\} \quad (3)$$

- b) Rescale the x- and y-components of each motion vector m_j^t in $\tilde{\mathbf{M}}^t$ by the temporal distance s_j^t :

$$\bar{v}_{j,x}^t = v_{j,x}^t / s_j^t, \bar{v}_{j,y}^t = v_{j,y}^t / s_j^t \quad (4)$$

- c) Apply motion scale $v_{j,s}^t$ to obtain the floating-point resolution:

$$\tilde{v}_{j,x}^t = \bar{v}_{j,x}^t / v_{j,s}^t, \tilde{v}_{j,y}^t = \bar{v}_{j,y}^t / v_{j,s}^t \quad (5)$$

- d) Compute the new reference points $(\tilde{x}_{j,src}^t, \tilde{y}_{j,src}^t)$ of the vectors with respect to rescaled x- and y-components:

$$\tilde{x}_{j,src}^t = x_{j,dst}^t + s_j^t \tilde{v}_{j,x}^t, \tilde{y}_{j,src}^t = y_{j,dst}^t + s_j^t \tilde{v}_{j,y}^t \quad (6)$$

- e) For each action $a_{i-1}^{t-1} = (x_i^{t-1}, y_i^{t-1}, w_i^{t-1}, h_i^{t-1})$ in $\mathbf{A}^{t-1}$, obtain the subset $\hat{\mathbf{M}}_i^t$ of motion vectors in $\tilde{\mathbf{M}}^t$ corresponding to new reference points $(\tilde{x}_{j,src}^t, \tilde{y}_{j,src}^t)$

$$\hat{\mathbf{M}}_i^t = \left\{ m_j^t \in \tilde{\mathbf{M}}^t \mid x_i^{(t-1)} - (w_i^{(t-1)})/2 \leq \tilde{x}_{j,src}^t \leq x_i^t - 1 + (w_i^{(t-1)})/2 \right. \\ \left. \Delta y_i^{(t-1)} - (h_i^{(t-1)})/2 \leq \tilde{y}_{j,src}^t \leq y_j^{(t-1)} + (h_i^{(t-1)})/2 \right\} \quad (7)$$

- f) For each $\hat{\mathbf{M}}_i^t$, evaluate aggregates (mean or median) $\mu_{i,x}^t$ and $\mu_{i,y}^t$ of the components $\tilde{v}_{i,x}^t$ and $\tilde{v}_{i,y}^t$ of all vectors $m_j^t \in \hat{\mathbf{M}}_i^t$.

- g) Predict each new event bounding box $a_i^t = (x_i^t, y_i^t, w_i^t, h_i^t)$ in $\mathbf{A}^t$ by performing a shift operation on the center of the corresponding previous event box a_{i-1}^{t-1} based on the aggregates $\mu_{i,x}^t$ and $\mu_{i,y}^t$:

$$x_i^t = x_{i-1}^{t-1} + \mu_{i,x}^t, y_i^t = y_{i-1}^{t-1} + \mu_{i,y}^t, w_i^t = w_{i-1}^{t-1}, h_i^t = h_{i-1}^{t-1} \quad (8)$$

Also, in the case when both frame types P and B are used, the above procedure is adopted after rescaling motion components as in the step (b), wherein the Π -rotation operation is performed on vectors in P so as to make them point in the same direction as that of the vectors in B. Thus, the vectors in P and B frames are treated in the same manner during further computations in the subsequent steps.

4 Results and Discussion

4.1 Datasets

The proposed method (MVCNN) for fall event action detection is experimented using the publicly available benchmark datasets, namely, Le2i [34], UR [36], and Multiple Cameras [35] Fall Event Datasets, which have the characteristics as given in the Table 2. Le2i Fall Detection Dataset has 143 and 48 video clips with fall and without fall events, respectively, acquired in four types of scenes: home, cafe, office and lecture hall. Multiple cameras fall dataset has 22 fall events and 2 mixed daily events. The multi-view synchronization videos can be treated individually as normal fall events dataset as well as a dataset for 3D scene reconstruction. Unlike these two datasets, the videos in UR Fall Detection Dataset are acquired through the kinect sensor and, thus, it comprises synchronized RGB videos and depth videos. Also, UR provides accelerator data as well as TST data. During these experiments, the evaluation of recognition speed is done in terms of number of frames per seconds (fps). The dataset is divided in to two sets in ratio 80:20, one for training and other for testing. The proposed algorithm MVCNN is executed on the processor Intel core i7 @ 3.7 GHz with 64 GB RAM using Python implementation on Ubuntu 20.04 platform. During training, each sample video is split into 20-frame clips following the previous related works. For the MVCNN network, each frame is resized to 255 X 255. The videos in MPEG-4 domain have, on average, 11 P-frames for every I-frame. The videos in H.264 domain have, on average, 11 P/B frames for every I-frame.

During inference, each set of random samples comprises 15 clips with 12-frame sequence (i.e., 1 I-frame followed by 11 P-frames) of MPEG-4 encoded video, while it comprises 15 clips with 12-frame sequence (i.e., 1 I-frame followed by 11 P/B-frames) of H.264 encoded video, to test the MVCNN network. Each of these clips is randomly cropped from the videos in benchmark dataset considered.

Computation Evaluation

For estimating computational requirements in video Fall event action recognition, the number of GFLOPS involved in one prediction (consisting of multiple data passes) is determined. For each clip, the proposed MVCNN requires 34 GFLOPS. Since each video is sampled for 15 clips, the total number of GFLOPS is 34. The time spent for loading video frames and extracting motion vectors is $O(1.5 \text{ ms /frame})$.

4.2 Ablation Study

The ablation study is conducted by considering four aspects as following:

- i) Mean and Median of motion vectors
- ii) Two different codecs, namely, MPEG-4 and H.264
- iii) Comparison of the proposed MVCNN model with the state-of-the-art
- iv) Performance analysis of MVCNN by considering three benchmark datasets, namely, Le2i, UR and Multiple Cameras

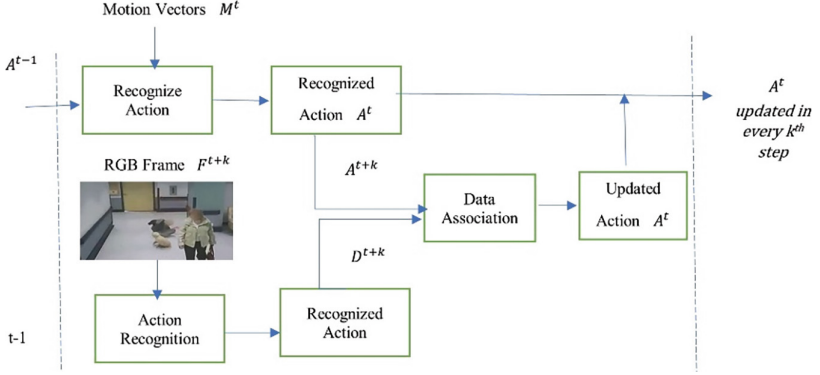


Fig. 1. The Framework of the proposed MVCNN. During Action Recognition, Action A^t is recognized using the previous Action A^{t-1} and motion vectors M^t of frame F^t . The update step done on every k^{th} frame involves Action recognizer running on RGB frame F^{t+k} . Data association is done using the Hungarian method which matches recognized Action D^{t+k} from the RGB frame with predicted Action A^{t+k} from motion vectors.

Comparison of mean and median variants of MVCNN

The performance of two variants of the proposed method (MVCNN) implemented on UR, and Multiple Cameras and Le2i Fall Detection Datasets in MPEG-4 compressed domain, wherein the first variant uses the arithmetic mean and the second one uses the median of the vector components, is compared in the Table 2. It is observed that both the variants perform closely.

Comparison of MVCNN variants in different compression domains

The performance of action recognition of MVCNN (Mean and Median) on UR, and Multiple Cameras and Le2i Fall Detection Datasets is evaluated for (i) MPEG-4 with P-frames, (ii) H.264 with P-frames only, and (iii) H.264 with both P and B-frames, using the metrics, namely, detection F-score (%) and speed (FPS). The results presented in Table 2 show that the proposed algorithm performs significantly better in MPEG-4 domain with higher event detection F-score and speed than that in H.264 compressed domain. However, in case of H.264, event detection F-score is marginally improved when both P and B frames with the median variant are handled rather than only the P frame, but at the expense of a loss in speed. The detection of fall event action on MPEG-4 videos is done with higher speed than that on H.264, since the macroblocks in each frame of H.264 have smaller size than that of MPEG-4 which results in computation of a higher number of motion vectors in each frame. These observations from results hold in cases of all the three datasets experimented there upon.

Comparison to with the state-of-the-art

In Tables 3, 4, and 5, the proposed MVCNN method is compared with the state-of-art on UR, and Multiple Cameras and Le2i Fall Detection Dataset. It is noted that the accuracy of an fall event action recognition model depends strongly on (i) the architecture, (ii) the pretraining dataset, and (iii) the clip size. The robustness of the proposed MVCNN action recognizer in terms of accuracy, parameters, GFLOPS, and speed is demonstrated by the tabulated experimental results.



Fig. 2. Motion vectors in (a) MPEG-4 frame, and (b) H.264 frame. Borders of macroblock are white, while motion vectors in P and B frames are red and green, respectively. Macroblocks with no motion vectors are intra-coded.

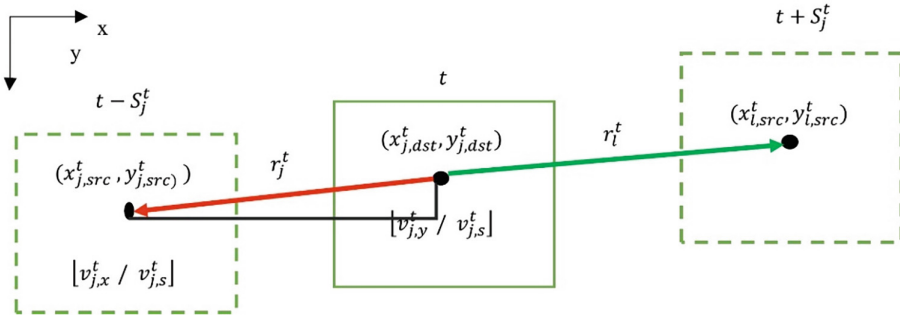


Fig. 3. Inter-coded macroblock (with solid boundary) in a P frame (MPEG-4/H.264) at time t has one motion vector v_j^t , which points to a sub image with a similar appearance (with dashed boundary) in the previous frame at time $t - S_j^t$. In case of H.264, macroblock has motion vector that points to a sub image with a similar appearance in the next frame at $t + S_l^t$.

Several observations are drawn based on the results presented in Tables 3, 4, and 5 using only the mean variant of MVCNN on the two different datasets. In comparison with MEWMA [37] architecture, MVCNN provides an Average Precision gain of +1.09% on UR. In comparison with Bhandari et al. [38] architecture, MVCNN provides an Average Precision gain of +4.19%, Average recall gain of +2.14 and F-score gain of +4.05 on UR. In comparison with Qi Feng et al. [39] architecture, MVCNN provides an Average Precision gain of +2.89, Average recall gain of +7.34 and F-score gain of +5.95 on UR. In comparison with Wang S, et al. [40] architecture, MVCNN provides an Sensitivity gain of +9.46 on Multiple Cameras. In comparison with Wang K, et al. [41] architecture, MVCNN provides an Sensitivity gain of +4.66 and Specificity gain of +5.2 on Multiple Cameras. In comparison with Qi Feng, et al. [39] architecture, MVCNN provides an Sensitivity gain of +6.76 and Specificity gain of +3.7 on Multiple Cameras. In comparison with A. Kumar [48] architecture, MVCNN provides an Average Precision gain of +9.41 on Multiple Cameras. In comparison with J. Redmon [49] architecture, MVCNN provides an Average Precision gain of +1.64 on Multiple Cameras. In comparison with L. Zhang [50] architecture, MVCNN provides an Average Precision gain of +1.47 on Multiple Cameras. In comparison with Peng Geng [51] architecture, MVCNN

provides an Average Precision gain of $+0.19$ on Multiple Cameras. The sample results of fall event action detection using proposed algorithm in compressed domains MPEG-4 and H.264 using UR fall detection dataset, which contains solitary fall event scenarios, are shown in the Figs. 4 and 5. In the Fig. 4, the ‘Walking’ action is identified along with motion vectors also being extracted. In the Fig. 5, ‘Fallen’ event action is identified along with motion vectors also being extracted. The sample results of fall event action detection using proposed algorithm in compressed domains MPEG-4 and H.264 using Multi Camera Fall Detection dataset, are shown in the Figs. 6 and 7. In the Fig. 6, the ‘Walking’ action is identified along with motion vectors also being extracted. In Fig. 8 ‘Fallen’ event action is identified along with motion vectors also being extracted.

In all the cases, it is observed that the proposed MVCNN has outperformed other state of the art methods in terms of enhanced Average Precision, Sensitivity, Specificity, F-score as well as speed and significantly reduced computational cost in terms of number of parameters[M], and GFLOPS

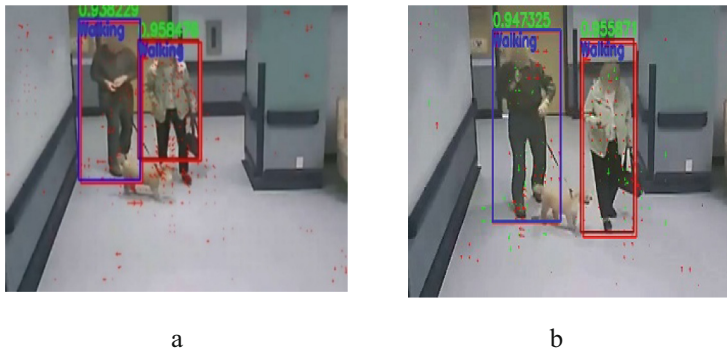


Fig. 4. Sample results for UR Fall Detection dataset with ‘walking’ action and corresponding vectors in (a) MPEG-4 domain and (b) H.264 domain.

Table 1. Characteristics of Datasets, namely, Lie2 Fall [34], Multiple Camera Fall [36] and UR Fall [35].

Video specs	Le2i [34]	Multiple Camera [36]	UR [35]
Number of Videos	191	192	70
Number of fall videos	143	176	30
Resolution	320 X 240pixels	720 X 480pixels	640 X 480pixels
Frame rate(fps)	25	30	25
Filming scenes	solitary	solitary	solitary

Table 2. Performance of two variants of the proposed method MVCNN implemented on Le2i Fall, Multiple Camera Fall and UR Fall datasets in MPEG-4 and H.264 domains Multiple Camera Fall and UR Fall datasets in MPEG-4 and H.264 domain

Dataset	Proposed Method Variants	F- Score	Speed (FPS)
UR Fall Detection Dataset	MVCNN (Mean, MPEG-4, P)	97.05 %	422.0
	MVCNN (Median, MPEG-4, P)	95.0%	276.5
	MVCNN (Mean, H.264, P)	90.25%	179.9
	MVCNN (Median, H.264, P and B)	91.0%	162.4
Multiple Camera Fall Dataset	MVCNN (Mean, MPEG-4, P)	95.25 %	319.0
	MVCNN (Median, MPEG-4, P)	92.85%	233.0
	MVCNN (Mean, H.264, P)	90.25%	243.0
	MVCNN (Median, H.264, P and B)	91.3%	224.0
Le2i Fall Detection Dataset	MVCNN (Mean, MPEG-4, P)	91.06 %	408.0
	MVCNN (Median, MPEG-4, P)	88.25%	318.0
	MVCNN (Mean, H.264, P)	86.45	145.0
	MVCNN (Median, H.264, P and B)	87.21%	135.0



Fig. 5. Sample results for UR Fall Detection dataset with ‘Fallen’ and ‘Walking’ actions and corresponding vectors in (a) MPEG-4 domain and (b) H.264 domain.

Table 3. Performance comparison of proposed method for fall event action detection with other methods using UR dataset.

Methods	Average Precision	Average recall	F-score
Harrou et al. [37] (MEWMA)	96.6	–	–
Bhandari et al. [38] (optical flow)	93.5	96.6	95.0
Qi Feng et al. [39] (Attention guided model)	94.8	91.4	93.1
Proposed (Motion vectors)	97.69	96.74	97.05

Table 4. Performance comparison of proposed method for fall event action detection with other methods using Multiple Cameras dataset.

Methods	Sensitivity	Specificity
Wang S, et al. [40] (PCANet)	88.9	98.9
Wang K, et al. [41](combination of features)	93.7	92.0
Qi Feng, et al. [39] (Attention guided model)	91.6	93.5
Proposed (MVCNN)	97.36	96.2

Table 5. Performance comparison of proposed method for fall event action detection with other methods using Multiple Cameras dataset

Methods	Domain	Params. [M]	GFLOPS	Average Precision (%)	Speed (FPS)
A. Kumar [48] (YOLOv1)	Non-Compressed	–	–	96.33	30
J. Redmon [49] (YOLOv3)	Non-Compressed	–	–	97.1	30
L. Zhang [50] (SSD)	Non-Compressed	–	–	97.27	30
Peng Geng [51] (YOLOv3 + Attention weight)	Non-Compressed	–	–	98.55	30
Proposed (Motion vectors + Revamped YOLOv3)	Compressed	7.66	34.0	98.74	422

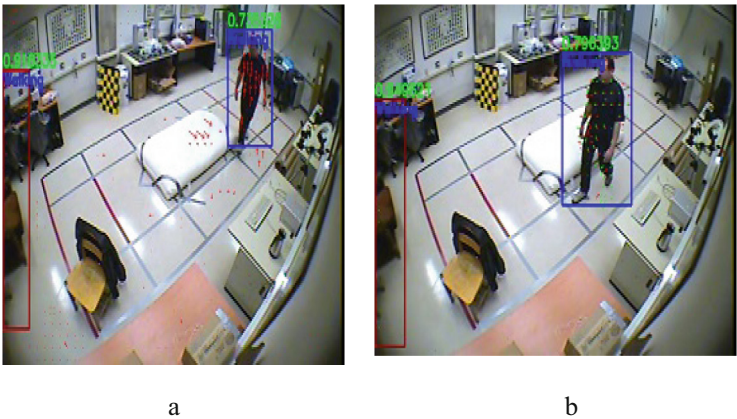


Fig. 6. Sample results for Multi Camera Fall Detection dataset with ‘walking’ action and corresponding vectors in (a) MPEG-4 domain and (b) H.264 domain.

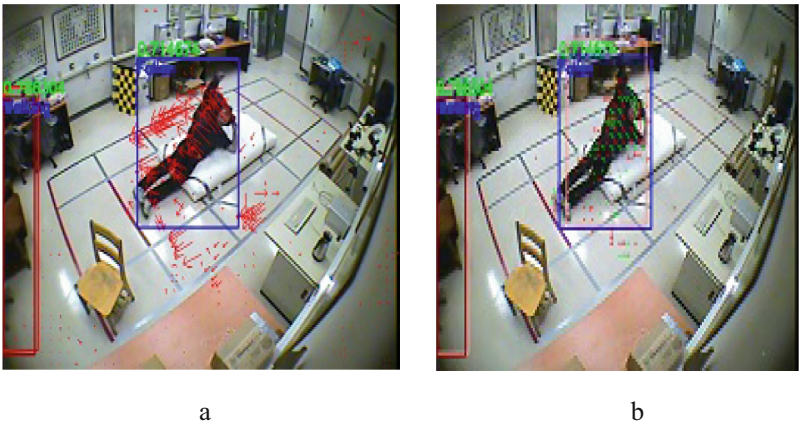


Fig. 7. Sample results for Multi Camera Fall Detection dataset with ‘Falling’ action and corresponding vectors in (a) MPEG-4 domain and (b) H.264 domain.

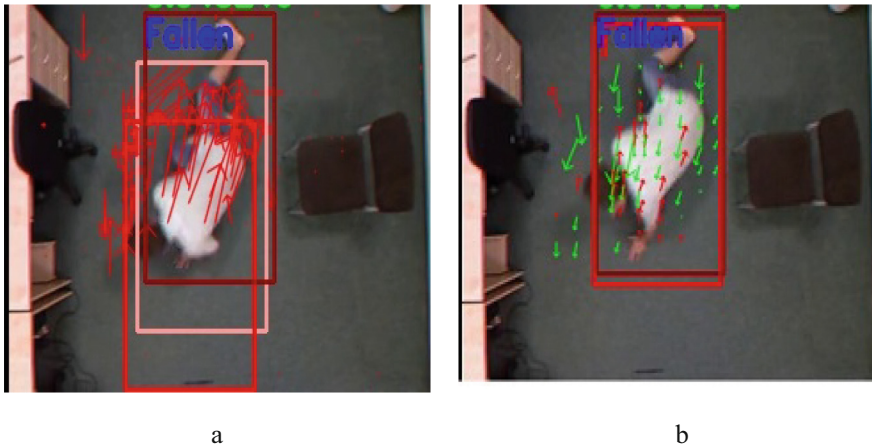


Fig. 8. Sample results for Le2i Fall Detection dataset with Fallen actions and corresponding vectors in (a) MPEG-4 domain and (b) H.264 domain

5 Conclusion

Herein, a novel method MVCNN is developed for real-time detection of fall event action in the compressed domain videos, namely, MPEG-4 and H.264, which has flexibility to deal with videos having key frames with non-uniform intervals and also to consider both P and B frames. The proposed algorithm is based on revamped YOLO v3 and motion vectors for target detection and tracking. The proposed algorithm is experimented on three benchmark datasets, namely, Le2i, UR, and Multiple Cameras Fall Event Datasets. The analysis of results indicate that the proposed algorithm performs with higher event detection F-score and speed, which makes it more effective in real-time fall event action detection in compressed domain as compared to other state of the art methods. Limitation of the proposed algorithm is the poor fall event action detection performance for H.264 domain in comparison with MPEG-4; and, this issue would be addressed in future work.

References

1. Muhammad, M., Shao, L., Seed, L.: A survey on fall detection: principles and approaches. *Neurocomputing* **100**, 144–152 (2013). <https://doi.org/10.1016/j.neucom.2011.09.037>
2. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 6201–6210. IEEE (2019)
3. Feichtenhofer, C., Pinz, A., Zisserman, A.: Convolutional two-stream network fusion for video action recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1933–1941. IEEE Computer Society (2016)
4. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pp. 568–576 (2014)
5. Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(1), 221–231 (2013)

6. Taylor, G.W., Fergus, R., LeCun, Y., Bregler, C.: Convolutional learning of spatio-temporal features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) *Computer Vision – ECCV 2010*. ECCV 2010. Lecture Notes in Computer Science, vol. 6316, pp. 140–153. Springer, Berlin, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15567-3_11
7. Tran, D., Bourdev, L.D., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 4489–4497. IEEE Computer Society (2015)
8. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lucic, M., Schmid, C.: Vivit: A video vision transformer. *arXiv preprint arXiv:2103.15691* (2021)
9. Weissenborn, D., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. OpenReview.net (2021)
10. Fan, H., et al.: Multiscale vision transformers. *arXiv preprint arXiv:2104.11227* (2021)
11. Liu, Z., et al.: Video swin transformer. *arXiv preprint arXiv:2106.13230* (2021)
12. Zhang, H., Hao, Y., Ngo, C.-W.: Token shift transformer for video classification. In: *MM '21: ACM Multimedia Conference, Virtual Event, China, October 20–24, 2021*, pp. 917–925. ACM (2021)
13. Huang, L., Liu, Y., Wang, B., Pan, P., Xu, Y., Jin, R.: Self-supervised video representation learning by context and motion decoupling. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13886–13895 (2021)
14. Shou, Z., et al.: DMC-net: Generating discriminative motion cues for fast compressed video action recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1268–1277 (2019)
15. Wang, S., Lu, H., Deng, Z.: Fast object detection in compressed video. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 7103–7112. IEEE (2019)
16. Wu, C.-Y., Zaheer, M., Hu, H., Manmatha, R., Smola, A.J., Krähenbühl, P.: Compressed video action recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6026–6035 (2018)
17. Yu, Y., Lee, S., Kim, G., Song, Y.: Self-supervised learning of compressed video representations. In: *Proceedings of the International Conference on Learning Representations (ICLR)* (2021)
18. Zhang, B., Wang, L., Wang, Z., Qiao, Y., Wang, H.: Real-time action recognition with deeply transferred motion vector CNNs. *IEEE Trans. Image Process.* **27**(5), 2326–2339 (2018)
19. Zhang, B., Wang, L., Wang, Z., Qiao, Y., Wang, H.: Real-time action recognition with enhanced motion vector CNNs. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016)
20. Bommers, L., Lin, X., Zhou, J.: MVmed: fast multi-object tracking in the compressed domain. In: *2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 1419–1424 (2020). <https://doi.org/10.1109/ICIEA48937.2020.9248145>
21. Haritaoglu, I., Harwood, D., Davis, L.S.: W4: real-time surveillance of people and their activities. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(8), 809–830 (2000)
22. Yoon, K., Dementhon, D.F., Doermann, D.: Event detection from MPEG video in the compressed domain. In: *Proceedings of the IEEE International Conference on Pattern Recognition, Barcelona, Spain (2000)*
23. Nait-Charif, H., McKenna, S.J.: Activity summarisation and fall detection in a supportive home environment. In: *Proceedings of the IEEE International Conference on Pattern Recognition*, vol. 4, pp. 23–26. Cambridge, UK (2004)
24. Lin, C.-W., Ling, Z.-H., Chang, Y.-C., Kuo, C.J.: Compressed domain fall incident detection for intelligent home surveillance. In: *Proceedings of the IEEE International Symposium on Circuits and Systems, Kobe, Japan (2005)*

25. Solbach, M.D., Tsotsos, J.K.: Vision-based fallen person detection for the elderly. arXiv preprint [arXiv:1707.07608](https://arxiv.org/abs/1707.07608) (2017)
26. Cui, X., Liu, Q., Gao, M., Metaxas, D.N.: Abnormal detection using interaction energy potentials. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3161–3167. IEEE (2011)
27. Qiao, M., Wang, T., Li, J., Li, C., Lin, Z., Snoussi, H.: Abnormal event detection based on deep autoencoder fusing optical flow. In: 2017 36th Chinese Control Conference (CCC), pp. 11098–11103. IEEE (2017)
28. Marsden, M., McGuinness, K., Little, S., O'Connor, N.E.: Holistic features for real-time crowd behaviour anomaly detection. In: Proceedings of the IEEE International Conference on Image Processing (ICIP), pp. 918–922. IEEE (2016)
29. Ravanbakhsh, M., Nabi, M., Sangineto, E., Marcenaro, L., Regazzoni, C., Sebe, N.: Abnormal event detection in videos using generative adversarial nets. arXiv preprint [arXiv:1708.09644](https://arxiv.org/abs/1708.09644) (2017)
30. Praveenkumar, S.M., Patil, P., Hiremath, P.S.: Real-time multi-object tracking of pedestrians in a video using convolution neural network and deep SORT. In: Tuba, M., Akashe, S., Joshi, A. (eds.), ICT Systems and Sustainability, Lecture Notes in Networks and Systems, vol. 321. Springer, Singapore (2022). https://doi.org/10.1007/978-981-16-5987-4_73
31. Redmon, J., Farhadi, A.: Yolov3: an incremental improvement. arXiv preprint [arXiv:1804.02767](https://arxiv.org/abs/1804.02767) (2018)
32. He, Y., Zhang, X., Sun, J.: Channel pruning for accelerating very deep neural networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, October, vol. 2, pp. 22–29 (2017)
33. Bodla, N., Singh, B., Chellappa, R., Davis, L.S.: Soft-NMS—improving object detection with one line of code. In: Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October, pp. 5562–5570 (2017)
34. Gasparini, S., et al.: Proposal and experimental evaluation of fall detection solution based on wearable and depth data fusion. In: Loshkovska, S., Koceski, S. (eds.) ICT Innovations 2015 . ICT Innovations 2015. Advances in Intelligent Systems and Computing, vol. 399, pp. 99–108. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-25733-4_11
35. Auvinet, E., Rougier, C., Meunier, J., St-Arnaud, A., Rousseau, J.: Multiple cameras fall dataset. DIRO-Université de Montréal (2010)
36. Kwolek, B., Kepski, M.: Human fall detection on embedded platform using depth maps and wireless accelerometer. *Comput. Methods Programs Biomed.* **117**(3), 489–501 (2014)
37. Harrou, F., Zerrouki, N., Sun, Y., Houacine, A.: Vision-based fall detection system for improving safety of elderly people. *IEEE Instrum. Meas. Mag.* **20**(6), 49–55 (2017)
38. Bhandari, S., Babar, N., Gupta, P., Shah, N., Pujari, S.: A novel approach for fall detection in home environment. In: Proceedings of the IEEE 6th Global Conference on Consumer Electronics (GCCE), pp. 1–5. IEEE (2017)
39. Feng, Q., Gao, C., Wang, L., Zhao, Y., Song, T., Li, Q.: Spatio-temporal fall event detection in complex scenes using attention guided LSTM. *Pattern Recogn. Lett.* **130**, 242–249 (2020)
40. Wang, S., Chen, L., Zhou, Z., Sun, X., Dong, J.: Human fall detection in surveillance video based on PCANet. *Multimedia Tools Appl.* **75**(19), 11603–11613 (2016)
41. Wang, K., Cao, G., Meng, D., Chen, W., Cao, W.: Automatic fall detection of human in video using combination of features. In: 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 1228–1233. IEEE (2016)
42. Vishwakarma, V., Mandal, C., Sural, S.: Automatic detection of human fall in video. In: Ghosh, A., De, R.K., Pal, S.K. (eds.) Pattern Recognition and Machine Intelligence. *PRMI 2007. Lecture Notes in Computer Science*, vol. 4815, pp. 616–620. Springer, Berlin, (2007). https://doi.org/10.1007/978-3-540-77046-6_76

43. Chua, J.L., Chang, Y.C., Lim, W.K.: A simple vision-based fall detection technique for indoor video surveillance. *SIViP* **9**, 623–633 (2015)
44. Stone, E.E., Skubic, M.: Fall detection in homes of older adults using the Microsoft Kinect. *IEEE J. Biomed. Health Inform.* **19**, 290–301 (2015)
45. Rougier, C., Meunier, J., St-Arnaud, A., Rousseau, J.: Fall detection from human shape and motion history using video surveillance. In: 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07), pp. 875–880. IEEE (2007)
46. Chan, T.H., Jia, K., Gao, S., Lu, J., Zeng, Z., Ma, Y.: PCANet: a simple deep learning baseline for image classification. *IEEE Trans. Image Process.* **24**, 5017–5032 (2015)
47. Vishnu, C., Datla, R., Roy, D., Babu, S., Mohan, C.K.: Human fall detection in surveillance videos using fall motion vector modeling. *IEEE Sens. J.* **21**(15), 17162–17170 (2021). <https://doi.org/10.1109/JSEN.2021.3082180>
48. Kumar, A., Srivastava, S.: Object detection system based on convolution neural networks using single shot multi-box detector. *Procedia Comput. Sci.* **171**, 2610–2617 (2020)
49. Redmon, J., Farhadi, A.: An incremental improvement. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, pp. 459–453. IEEE Computer Society (2018)
50. Zhang, L., Lin, L., Liang, X., He, K.: Is faster R-CNN doing well for pedestrian detection?. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *Computer Vision – ECCV 2016*. ECCV 2016. Lecture Notes in Computer Science(), vol. 9906, pp. 443–457. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_28
51. Geng, P., Xie, H., Shi, H., Chen, R., Tong, Y.: Pedestrian fall event detection in complex scenes based on attention-guided neural network. *Math. Problems Eng.* 4110246, 10 (2022). <https://doi.org/10.1155/2022/4110246>



Automatic Music Control Using Image Processing and MediaPipe

Sudheer Shetty¹(✉), R. Rakshitha², S. Arundhathi Bhat², Lathesh², Ravish Acharya², Suresh Kumar Peddoju³, and Hemanth Kumar Nichenametla⁴

¹ Department of ISE, Sahyadri College of Engineering & Management, Mangaluru Affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India
sudheershetty06@gmail.com

² Department of ISE, Alvas Institute of Engineering and Technology, Moodbidri Affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India

³ Department of Information Science Technology, Cullen College of Engineering, University of Houston, Houston, TX, USA

⁴ Engineering Data Science, Cullen College of Engineering, University of Houston, Houston, TX, USA

Abstract. Recognizing particular hand gestures and utilize them to transfer information or control things is one of the main objectives of gesture recognition system. The proposed system mainly focuses on situations where we multitask, i.e., work on various applications or run multiple desktop programs at the same time while listening to background music playing in one of the windows. Occasionally, we have to execute particular movements during a stop or change the song, likewise opening the music window and carrying out the necessary task. This method requires considerable time and work to finish. With the music controller, all we have to do is wave or make a simple hand gesture in front of the webcam to change or pause a specific piece of music that is playing. This saves time and helps us stay focused on the task at hand, without requiring switching between multiple multimedia players.

Keywords: Hand gesture · music controller · convolutional neural network · recurrent neural network

1 Introduction

With the use of gestures captured by the camera, users may now control the playback of music automatically. Automatic music control is the process of controlling music playback using hand gestures captured by the camera [1]. This device will recognize hand motions and adjust its actions accordingly, using machine learning and computer vision algorithms. Physical buttons or controls are not necessary since image processing and machine learning algorithms provide a new method to engage with music playback.

Using the webcam, this technology captures hand gestures from the user, which are subsequently processed to recognize certain hand movements. These gestures are then

classified by machine learning algorithms that have been trained on a dataset of hand movements. When motion is recognized, the music playback is adjusted by the system. Having been considered, autonomous music control through the use of machine learning and image processing techniques is a promising technology that may change the way we interact with music players. Using a camera to capture an image of the surroundings, the system can then apply image analysis to extract crucial information such as the number of individuals in the area, ambient lighting, and overall brightness. The optimal volume for the music is then chosen once machine learning algorithms have analyzed these characteristics.

This system can be developed with a variety of machine learning algorithms, such as support vector machines, decision trees, and artificial neural networks. The system can also be customized to meet user preferences. For example, the volume level can be adjusted based on the type of music being played or the time of day. Overall, an automatic music control system makes it easy and effective to guarantee that music is played at the right volume, providing a comfortable listening experience for everyone nearby. Automatic music volume control integrates image processing with MediaPipe to ensure that the music is always played at the right volume, which can enhance the user experience.

This technology has the potential to significantly change how music is played in a variety of settings, which would also increase listeners' interest and experience. Human-Computer Interaction (HCI) hand movement recognition could help achieve the required naturalness and convenience. When interacting with others, hand gestures—which range in complexity from simple to complex—serve the function of conveying information. With a few basic hand motions, one may utilize this technology to play, pause, skip, and adjust volume. It does this by utilizing sensors and algorithms to detect and interpret hand movements. Gesture-based music control allows us to effortlessly manage our music playback while on the go without having to take our phone out of our pocket or bag.

This technology is a great example of how it may enhance our daily lives and make them simpler and better for us. Gesture-based music control is changing the way we interact with technology and our surroundings in this way. Gesture recognition is one way for a machine to understand body language, and it fosters a deeper relationship between humans and machines than conventional user interfaces, or even GUIs. When playing music, the play, pause, stop, and next controls are commonly utilized to highlight creative user interfaces and interaction techniques. Furthermore, a range of capabilities for controlling music playback have been demonstrated and tested utilizing gesture recognition algorithms.

2 Related Work

The most popular techniques for recording hand gestures and carrying out the associated action are discussed in this section.

Using a real-time webcam display and musical audio as input, Bayd et al. [2] suggested an accurate and efficient playful technique for the automatic marking of the synchronization of hand gestures and musical rhythms. This makes it possible to check

a musical piece's degree of synchronization by tapping fingers in time with the rhythm. Additionally, an experiment was carried out using four levels of difficulty, varied mixing levels, and beat pulses added to the music. The findings demonstrate that as difficulty increases, the synchronization score falls in line with difficulty.

In an effort to determine if indoor nature exploration activities for children aged three to five could benefit from the use of a program that recognizes the shape of a hand as an image and runs a virtual mouse, Han et al. [3] conducted an experiment. It was established in the test on the program's adult developers that the virtual mouse's clicks and motions were naturally processed in accordance with hand movements, but when the same procedure was done to actual youngsters, an unexpected circumstance arose. The accuracy dropped as the youngster taking the test's distance from the camera rose because the adult and child's hand sizes were not taken into account. This is due to the fact that the characteristics of children roaming freely in a large space were not taken into consideration, in contrast to adults who conduct tests in place. Nonetheless, it was established that the accuracy decreased with the child's distance from the camera.

Altayeb [4] presented a method for detecting and recognizing hand movements using artificial intelligence based on image processing techniques. The robot arm was tested under various settings (finger movements, distance, and background colour) in order to assess accuracy. Each finger was given distinct motions during the finger actions, such as opening and shutting. Conversely, the hand was detected at various distances in order to conduct the distance test. The goal of the last test was to identify the hand based on the various background colours. It should be mentioned that ten repetitions of each test condition were conducted in order to assess the system's correctness. According to experimental findings, the suggested robot arm model using AI-based image processing methods made possible by the MediaPipe and OpenCV libraries has a very good accuracy.

The findings of the Gupta et al. [5] study offer compelling evidence of the model's amazing capacity to predict human hand movements with high accuracy thanks to the cooperative efforts of MediaPipe and OpenCV algorithms. The training set comprised a large dataset of more than 3000 images with various letter combinations (A to Z) and number combinations (1 to 9). The algorithm is important for people with many kinds of disabilities, such as those who are deaf, silent, or have other difficulties. Through the use of hand gesture detection, the algorithm provides a useful means of improving connection and communication for people with special needs. Additionally, the algorithm's adaptability makes it possible to use it in public places like train stations, airports, and other gathering places, where it can help create a more welcoming and accessible atmosphere. Expanding and developing this technology further has the potential to greatly improve the quality of life for the deaf and mute community. For example, the technique allows mathematical operations to be visualized as animated hand motions. The animated hand gestures offer a useful alternative for people who might have trouble with conventional forms of communication in addition to facilitating understanding and engagement. By enabling people with special needs to navigate and engage more skillfully in public settings, such technology can improve their overall experience and independence.

By adjusting the position of their fingertips and selecting their preferred instrument (conventional instruments like pianos or human voice instruments), users can perform

music using MoMusic, as demonstrated by Bian et al. [6]. The rhythms and notes produced by the finger movements are continuously changing to create an exciting melody because of the uncertainty of the finger trajectory, the variability of the corresponding sounding area, and the randomness of the chord and note combinations that adhere to the Probabilistic Chord Progress Model. MoMusic is a motion-based system for creating music that has a virtual human voice model and a probabilistic chord progression model. MoMusic allows users to create melodic music with the right finger trajectories. It is expressive and engaging.

Two models for sign language recognition are proposed in the work of Sreemathy et al. [7]: SVM with media-pipe and YOLOv4. Using Gramformer, the trained models were expanded to recognize continuous gestures with great accuracy. Each of these models has benefits and drawbacks. Since MediaPipe is used to extract hand key points for gesture detection, the SVM model with MediaPipe is trained quickly but is susceptible to noise. Although YOLOv4 demands a lot of processing resources, transition gesture recognition is no longer an issue. Additionally, the suggested model outperformed the other cutting-edge models when compared to other models, according to comparisons with other models. Additionally, an expert system that combines the best features of both models is suggested in order to generate real-time findings that are even more accurate than they would be taken alone. This system can be put into place in many public areas to facilitate easy communication between people with disabilities and the general public.

A hand gesture recognizer is made using OpenCV and Python in the Hand Gesture Recognition project by Mopidevi et al. [8]. They employed the MediaPipe and Tensorflow frameworks, respectively, for the detection and gesture recognition processes. They have utilized the knowledge of file management, widely used image processing techniques, neural network foundations, etc., here. The project is a simple example of how CNN may be applied to solve computer vision problems with high accuracy.

To improve communication between hearing and deaf people, Moustafa et al. [9] utilize deep learning to detect ArSL from deaf people's hands in real time. This allows them to convert signals into alphabets. First, a personal dataset is constructed by gathering data. Subsequently, a preprocessing phase is implemented, wherein the hand is cropped from the image and its shape is extracted. Lastly, they present four different CNN models, three from scientific papers and one from their own, each with a distinctive architecture. In order to compare them and select the best model, they applied them to their datasets.

A system that offers users a smooth and pleasurable music listening experience was proposed by Shetty et al. [10]. Through the integration of sophisticated facial expression recognition algorithms and real-time mood assessment, the system can precisely build playlists that align with the user's emotional state, promoting relaxation and alleviating stress. Given that an increasing number of individuals are using music as a stress and anxiety reliever, the system's capacity to enhance mental health and wellbeing renders it an invaluable asset to the music streaming market. Here is where the emotion-based music recommendation system was created, which effectively blends modern technology with music therapy to offer people appropriate and beneficial music recommendations depending on their moods. With an accuracy rate of roughly 75%, the model can detect and recognize seven different moods: anger, disgust, fear, happy, sad, surprise, and neutral.

Using a low-cost ESP8266 microcontroller chip, Yang et al. [11] enabled hand gesture detection for smart home appliance control. In order to determine the finger angle, this study will employ MediaPipe hand tracking to extend the fingers according to the palm and add vector angle formulas. There were ten defined hand movements. Ultimately, the developed system may use hand gestures to operate household appliances with a promising level of accuracy and recognition speed. To verify the integrity and soundness of the system design framework, WoPeD, a Petri net tool, is used for modeling and analysis. The bending angle of fingers was calculated using the vector method to significantly enhance the ability to recognize numbers 0 through 9. This technology can assist consumers in conveniently and comfortably operating their household appliances. For instance, users can turn on and off their home appliances without getting up. The process of turning on and off the electricity only requires them to connect their mobile devices to Wi-Fi infrastructure.

In order to assist guitarists, the work by Emad et al. [12] examines the various techniques used for note labeling and finger detection. Furthermore, this work presents a hybrid technique integrating audio processing and note labeling with finger motion detection using MediaPipe, which is projected to deliver improved accuracy in guitar learning systems. Guitarists' technique is projected to improve with the use of MediaPipe's hand/finger motion detection feature, as it records their hand movements in addition to the notes they play.

Using the DSL10-Dataset, Samaan et al. [13] demonstrated Dynamic Sign Language Recognition using three RNN models: GRU, LSTM, and BILSTM. The features extraction process made advantage of the MediaPipe framework. Two tests were carried out to demonstrate DSL identification from two perspectives, and the outcomes demonstrate an improvement in both. Three steps could be taken to implement the suggested strategy. First, compile a dataset of videos with uniform frame rates. Subsequently, provide the input data to the MediaPipe framework, enabling it to extract the stance, hand, and facial keypoints from every video frame. Lastly, load the retrieved keypoints into GRU, LSTM, or BILSTM, one of the ready-made RNN models, for the training phase. The accuracy of the test and train when face key points are included and when they are not are almost equal. The prediction and learning times are adversely affected by the fact that the key points containing the face are six times larger than the key points without the face.

In their study of the problem of mid-air hand pose control and visible interaction for a smart music speaker, Xu et al. [14] offer a unique pose space encoding and visualization model that combines a PointNet-based gesture categorization network with a fully-connected autoencoder. To be more precise, a fresh dataset of mid-air gestures is gathered in order to assess and refine the suggested approach for mid-air gesture control and recognition. In order to combine the posture space and the music space for interactions, the proposed autoencoder embeds gestures into low-dimensional spaces that are appropriate for visualization and interaction. Furthermore, the auxiliary classification network retains exceptional classification performance while progressively refining the gesture clustering in pose space. Additionally, the suggested interaction strategy lowers interaction latency by requiring a small number of gesture frames (2-frame sequence) of input to obtain a continuous control that the user can explore.

In order to examine the advantages and disadvantages of four cutting-edge Human Pose Estimation (HPE) libraries for processing image and video HPE, Chung et al. [15] conducted a comparative analysis of these libraries: OpenPose, PoseNet, MediaPipe Pose, and MoveNet. The only subject of this study was 2D single-person HPE. The evaluation criterion employed in this study to assess the HPE libraries' quality was PDJ.

The work by Natarajan et al. [16] advances the creation of a deep learning framework for the recognition, translation, and generation of sign language from beginning to end. It used the suggested H-DNA framework to address the issues that remained with previous SL recognition and video creation techniques. A number of quality measures are employed to qualitatively assess the suggested H-DNA framework. Compared to previous techniques, they acquired a relatively higher recognition rate and producing performance. Using human assessors, the assessment of coherence, relevance, and realism aspects yields positive outcomes in real-world situations.

Using MediaPipe and the LSTM model, Shinde [17] developed a translator for sign language to multilingual text and speech systems. This is a potential method for enhancing accessibility and communication for the deaf and hard-of-hearing community. The MediaPipe framework was used in the system's architecture to identify sign language gestures, and the LSTM model was used to convert those gestures into written and spoken words. Many software components were used in the system's construction, and testing revealed that the system could recognize and translate sign language with a high degree of efficiency and accuracy. The system has a wide range of possible uses, including communication, healthcare, and education. More inclusive and accessible communication can result from the system's ability to close the communication gap between sign language users and non-users.

Through the use of hand gesture recognition, a new deep-learning architecture was proposed by Peral et al. [18] that enables human-robot communication. The suggested approach recognizes human commands that are predefined in a library of hand gestures just by using visual information. Experiments are conducted to obtain an over-performing method that outperforms the accuracy of state-of-the-art methods. Additionally, a batch approach that successfully recognizes the motions made in front of the robot IVO is used to evaluate the system in real-life scenarios where movements are continuous. Collaborative robots will greatly benefit from the communication structure. The tests demonstrate that the framework can decipher gesture-based human commands.

A lightweight 3DGCN architecture was presented by Al-Hammadi et al. [19] for sign language recognition. In order to prevent the over-smoothing effect that is frequently seen in deep GCN architectures—which is caused by a high number of message repeats between graph nodes—the suggested architecture makes use of a few 3DGCN layers. By using this shallow design, a graph representation of the signer body is created using the most pertinent MediaPipe landmarks. The recognition challenge is transferred to graph classification in this embedding step. The suggested design is assessed using many datasets with disparate attributes and contrasted with cutting-edge systems. The results show how well the spatial attention layer works to improve the efficiency of this kind of lightweight 3DGCN architecture. Additionally, the suggested design performs exceptionally well and generalizes effectively across a range of datasets.

Ruiz et al.'s study [20] presented an SLR methodology for word level that makes use of a limited number of manually created features. Further analysis is done on non-manual qualities when compared to related work. Spatiotemporal characteristics from body posture, facial expressions, hands region, head attitude, and eye gaze estimation were retrieved following a ROI and tracking phase. Following a phase of data augmentation and dimensionality reduction, BiLSTM and transformer models were employed in the recognition phase. The preservation of non-manual features during dimensionality reduction indicates that features of this kind hold pertinent data.

3 The Proposed Approach

In order to precisely identify the user's gesture, the gesture recognition model makes use of machine learning technique MediaPipe. Establishing a communication link between the music player and the system is part of music player integration. By gathering information on the user's gestures, the system is trained to recognize the user's movements accurately during calibration. Ultimately, a smooth user experience and real-time response depend on the system's performance.

Having been considered, automatic music control system architecture is a promising technology that gives users a more natural and intuitive way to control their music playback. The system architecture can be further enhanced to enable more sophisticated gestures and commands, offering a more convenient and personalized user experience, with ongoing advances in machine learning and image processing.

The system architecture for autonomous music control has the potential to fundamentally change how people use their devices. The technology can be further enhanced to provide a more smooth and natural user experience with more research and development. Potential uses for the system design exist in other fields as well, like virtual reality and gaming. A more realistic and natural user experience can be had by using gesture recognition.

The music player receives commands from the music control component to handle gesture recognition. For example, the user can gesture to raise the volume, and the music player will receive the signal from the system to do so. The user interface part lets the user interact with the system and gives them feedback. The gesture that has been identified may be shown visually through the user interface, for example, by being highlighted on the screen. In order to let the user know that the system has detected the gesture, the user interface may additionally include audio feedback.

The system architecture of the proposed system is shown in the Fig. 1.

- **Hand Detection.** Develop a "detection mode" interface that uses a webcam to continually record a predetermined number of frames in order to extract the user's hand motion track from those frames. The implementation of recognition algorithms along the route, such as hands movement tracking, hands performance evaluation shape detection, and hands distance detection should be worked out.
- **Preprocessing.** Making decisions, or deciding which instruction needs to be decoded in order to correlate to the monitored gesture, comes after tracking the valid gesture. A number of signatures for hand motion patterns during pre-processing will be created.

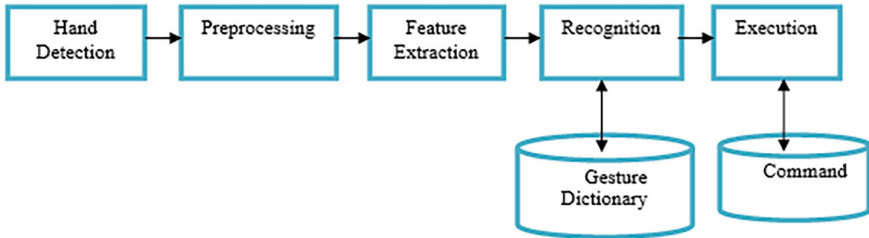


Fig. 1. System Architecture of Automatic Music Control

Then, various features of the open music player will be activated based on these signatures.

- **Feature Extraction.** Take out just the hand gestures from the picture. Eliminate the distracting areas and sounds. Create a binary image by processing the extracted picture and to create a feature set for classification, extract the recognizable, important features from the processed image.
- **Recognition.** The tracking of legitimate gestures is done continuously. When a legitimate gesture is tracked, it is identified as such and forwarded to further processing.
- **Execution.** Following the command's decoding, the gesture's corresponding task—which is nothing more than the gesture's output and basic interpretation—is carried out. This gesture was originally intended and defined.

4 Methodology

A proposed solution reduces the need for human intervention by allowing us to control the music with various hand gestures. The user's gesture will be captured by the webcam once the necessary libraries have been imported. The fingers count can then be determined using the image. In addition to counting fingers, orientation detection will occur, and the music will be adjusted accordingly. The conceptual design of a system, which describes its behaviour, structure, and other viewpoints, is referred to as a model. An architecture description is a formal description and representation of a system designed to make it easier to analyze its behaviours and structures.

The block diagram of the various stages of automatic music control system is depicted in Fig. 2.

- **Importing Libraries.** The required libraries, including cv2, MediaPipe, pyautogui, and time, are initially imported by the code.
- **Defining a Function.** Code specifies a function named `count_fingers()` that returns the number of fingers recognized in the gesture given a single input, 'lst', which represents the list of hand landmarks detected.
- **Initializing Video Capture.** The `cv2.VideoCapture()` function is used by the code to initialize a video capture object and read frames using a webcam.
- **Setting up MediaPipe.** In addition to setting up a MediaPipe drawing utilities module to visualize the landmarks that are detected, the code also sets up a MediaPipe pipeline to detect hands using the `mp.solutions hands` module.

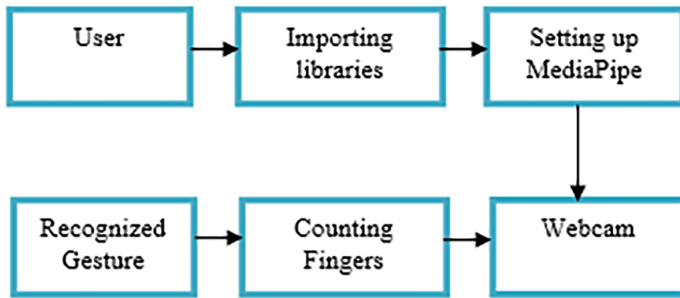


Fig. 2. Stages of Automatic Music Control System

- **Importing Libraries.** The required libraries, including cv2, MediaPipe, pyautogui, and time, are initially imported by the code.
- **Initializing Variables.** The code sets up a number of variables, including start_init, start_time, and prev, in order to track the timing and state of hand gestures.
- **Counting Fingers.** To count the number of fingers detected in the current hand gesture, the code invokes the count_fingers() function with the hand_keyPoints parameter.
- **Displaying the Video.** The current frame with the identified landmarks and the count of hand gestures is displayed by the code using cv2.imshow().

The user's image or gesture is captured by the web camera and subsequently processed through image processing techniques. In order to identify the user's gesture, the processed image is assessed using machine learning techniques. The system can then regulate the playing of the music after recognizing the user's gesture. Sending instructions to a music player, like skipping a track or changing the volume, is one example of this.

Additionally, the system might have a user interface that would let the user interact with it or display information about the music being played. Voice commands or a graphical user interface might be part of this. The overall goal of the system architecture for automatic music control through image processing is to give users a more convenient and user-friendly method of controlling their music playback. The technology lets users control their music playing with basic hand gestures by recognizing their movements in place of conventional buttons or touchscreens.

Next, the gesture that was identified is utilized to manipulate the music playback, such as muting a track or altering the volume. Numerous parts of the system architecture work together to deliver a flawless user experience. The main input device that records the user's image or motion is the camera. In order to extract features that correlate with the user's gesture, the image processing component analyses the acquired image. Machine learning methods are used by the gesture recognition component to precisely identify the user's gesture.

Based on the recognized gesture, the music control component instructs the music player to control the playback of the music. For instance, the user can gesture to raise the volume, and the music player will receive the signal from the system to do so. The user interface part of the system lets the user engage with it and gives them feedback.

5 Experimental Results

MediaPipe, image processing, and automatic music control have shown encouraging results. With the use of hand motions captured by the camera, users may now control the playback of music. A novel method of interacting with music playback is made possible by the use of mediapipe and image processing, which does away with the need for physical buttons or controls.

Figure 3 shows the comparison of the proposed approach and other existing approaches such as edge detection method, HSV & cross-correlation and background subtraction.

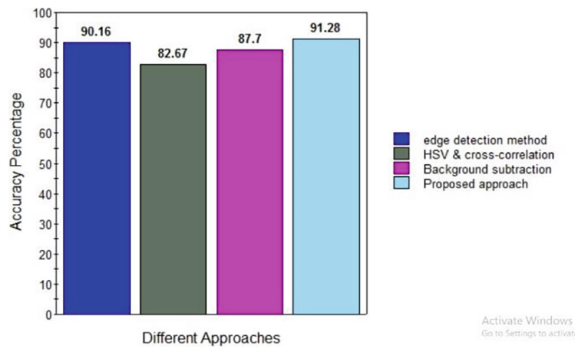


Fig. 3. Comparison Graph of the Proposed Approach and other Approaches

In comparison with other static hand gesture methods, the proposed approach is much more effective, offering up to 91.28% of recognition rate accuracy. The suggested strategy has several advantages over the other approaches, including higher accuracy in hand gesture recognition and the ability to be used in the presence of background noise due to its dynamic nature.

The various gesture signals utilized by the automatic music control system for its numerous functions are depicted in the images below. The next piece of music is played using finger count 2, while finger count 5 is used to play and pause the current piece. 4 finger counts are used to raise volume, and 3 finger counts are used to decrease it. We can turn off the music by using finger count 1 (Figs. 4, 5, 6, 7, 8, and 9).

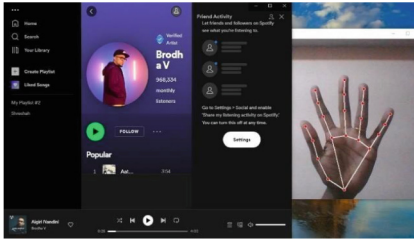


Fig. 4. Pause Music when Finger Count = 5

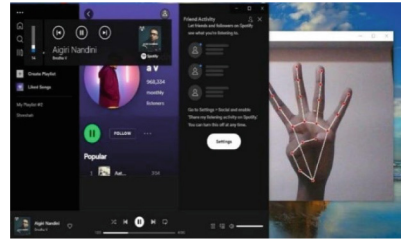


Fig. 5. Increase the Volume when Finger Count = 4

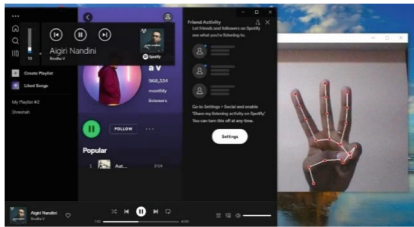


Fig. 6. Decrease the Volume when Finger Count = 3

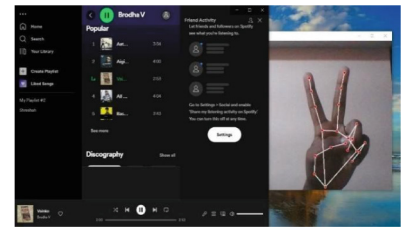


Fig. 7. Play Next Music when Finger Count = 2

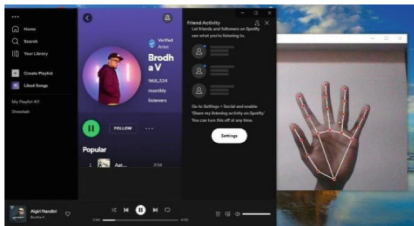


Fig. 8. Play Music when Finger Count = 5

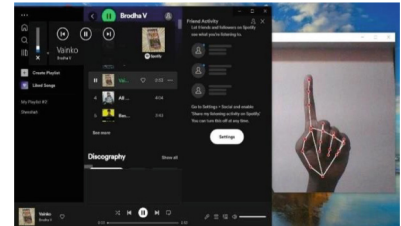


Fig. 9. Mute the Music when Finger Count = 1

6 Conclusion

The system's objective is to introduce technology into man's busy life by utilising improved image processing techniques to run a digital music player through hand gestures. The music player will carry out the designated action when a gesture is made in front of the camera. Every action of the music player is controlled by a distinct gesture; hence, should a gesture be presented incorrectly or without any motion at all, the system will respond by displaying an error message stating that the gesture input is invalid. Thus, it can be said that using the music player is quite simple and successful.

One potential improvement in the future would be to use 3D hand tracking technology. This would enable the system to identify particular hand movements in three

dimensions, which could be utilised to regulate various elements of the music, including volume, tempo, and track selection. Furthermore, by training the system to identify particular hand movements and link them to particular musical genres, moods, or themes—for example, “play some upbeat music”—the system can be trained to concurrently manage the volume and the song selection. For example, the system may be trained to play music or artists that are similar to the one it is currently playing when it detects a thumbs-up signal. Furthermore, the system might produce a more natural and user-friendly user interface by combining voice instructions with hand movements.

References

1. Lee, J.Y., Jung, J.E., Kim, H.J.: A real-time hand gesture recognition technique and its application to music display system. *J. Autom. Control Eng.* **4**(2) (2016)
2. Bayd, H., Guyot, P., Bardy, B., Slangen, P.R.L.: Automatic scoring of synchronization from fingers motion capture and music beats. In: Mazzeo, P.L., Frontoni, E., Sclaroff, S., Distant, C. (eds.) *Image Analysis and Processing. ICIAP 2022 Workshops. ICIAP 2022. Lecture Notes in Computer Science*, vol. 13373, pp. 235–245. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-13321-3_21
3. Han, J.S., Lee, C.I., Youn, Y.H., Kim, S.J.: A study on real-time hand gesture recognition technology by machine learning-based mediapipe. *J. Syst. Manag. Sci.* **12**(2), 462–476 (2022)
4. Altayeb, M.: Hand gestures replicating robot arm based on mediapipe. *Indonesian J. Electr. Eng. Inform. (IJEEI)* **11**(3), 727–737 (2023)
5. Gupta, A., Chawla, N., Jain, R., Thakur, N., Devi, A.: Gesture-based touchless operations: leveraging mediapipe and OpenCV. *NEU J. Artif. Intell. Internet Things* **1**(2) (2023)
6. Bian, W., et al.: MoMusic: a motion-driven human-AI collaborative music composition and performing system. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 13, pp. 16057–16062 (2023)
7. Sreemathy, R., Turuk, M.P., Chaudhary, S., Lavate, K., Ushire, A., Khurana, S.: Continuous word level sign language recognition using an expert system based on machine learning. *Int. J. Cogn. Comput. Eng.* **4**, 170–178 (2023)
8. Mopidevi, S., Biradhar, S., Bobberla, N., Buddati, K.S.: Hand gesture recognition and voice conversion for deaf and dumb. In: *E3S Web of Conferences*, vol. 391, p. 01060. EDP Sciences (2023)
9. Moustafa, A.M.A., et al.: Integrated mediapipe with a CNN model for arabic sign language recognition. *J. Electr. Comput. Eng.* (2023)
10. Shetty, A.A., Pranay, D., Shamanth, K.S., Vignesh, C.G.: Euphoria FM: music for every emotion. *J. Electr. Comput. Experiences* **1**(1), 11–17 (2023)
11. Yang, C.Y., et al.: Smart control of home appliances using hand gesture recognition in an IoT-enabled system. *Appl. Artif. Intell.* **37**(1), 2176607 (2023)
12. Emad, J., Sobeah, H., Serag, A., Yehia, A., Khaled, K., Mohamed, K.: Note frequency recognition and finger motion capture of a guitarist: a survey and challenges ahead. *J. Comput. Commun.* **1**(2), 69–79 (2022)
13. Samaan, G.H., et al.: Mediapipe’s landmarks with rnn for dynamic sign language recognition. *Electronics* **11**(19), 3228 (2022)
14. Xu, S., Kaul, C., Ge, X., Murray-Smith, R.: Continuous interaction with a smart speaker via low-dimensional embeddings of dynamic hand pose. In: *ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE (2023)

15. Chung, J.L., Ong, L.Y., Leow, M.C.: Comparative analysis of skeleton-based human pose estimation. *Future Internet* **14**(12), 380 (2022)
16. Natarajan, B., et al.: Development of an end-to-end deep learning framework for sign language recognition, translation, and video generation. *IEEE Access* **10**, 104358–104374 (2022)
17. Shinde, P.D.: A Translator for Sign Language to Multilingual Text and Speech (Doctoral dissertation, California State University, Northridge) (2023)
18. Peral, M., Sanfeliu, A., Garrell, A.: Efficient hand gesture recognition for human-robot interaction. *IEEE Robot. Autom. Lett.* **7**(4), 10272–10279 (2022)
19. Al-Hammadi, M., et al.: Spatial attention-based 3d graph convolutional neural network for sign language recognition. *Sensors* **22**(12), 4558 (2022)
20. Ruiz, D.S., Olvera-López, J.A., Olmos-Pineda, I.: Word level sign language recognition via handcrafted features. *IEEE Lat. Am. Trans.* **21**(7), 839–848 (2023)



Study of Lung Nodules Using CT Scan Image

P. V. Raja Suganya^(✉), T. Devi, and V. Joshita

Velammal Engineering College, Chennai, India
rajasuganya@velammal.edu.in

Abstract. Respiratory illness is the primary factor behind fatality associated with cancer globally, and timely identification is imperative for preserving lives. However, analyzing CT scans can be a burdensome task for radiologists and observer fatigue can lead to reduced performance. Timely identification of nodules is critical for effective treatment and improved survival rates. As a result of advancements in medical imaging technology, CT scans have become the primary approach for identifying lung nodules during the diagnostic phase. The proposed system utilizes an advanced convolutional neural network (CNN) architecture, trained using a vast dataset of annotated lung CT images, to accurately detect and classify lung nodules. The system incorporates various image pre-processing techniques to increase CT scan image quality and improve CNN performance. The suggested system attains notable precision and responsiveness in detecting lung nodules, potentially contributing to the timely identification and management of lung cancer. This report explores a project centered around the detection of lung nodules through the utilization of deep learning techniques.

Keywords: Deep Learning · Correlation Based Feature selection · Data Visualization · Recurrent Neural Networks Transfer Learning

1 Introduction

The significance of early detection of lung disease, which ranks among the primary causes of global mortality, cannot be overstated. Diagnostic modalities are employed to identify potential anomalies in lung tissue. However, visually analyzing these images for abnormalities is challenging and time-consuming, given the large amount of information that must be processed within limited time constraints. Consequently, there is a growing inclination in the medical field towards the utilization of computational intelligence techniques, including data mining and image processing, to automatically analyze patient records and images, gaining increasing recognition. By capitalizing on these methodologies, the undertaking strives to significantly enhance the discernment of pulmonary nodules, while maintaining a low rate of false positive results.

The project employs the LIDC/IDRI dataset, comprising approximately 1500 CT scans, to showcase the efficacy of the approach. The primary objective of this project is to develop a prognostic model that accurately ascertains the cancerous nature of lesions in the lungs of individuals at high risk. To accomplish this objective, the project

will leverage various techniques in the realms of data mining and pattern recognition. Addressing classification quandaries involves focal points such as feature manipulation, clustering feature extraction, feature engineering, and machine learning models. For predictive modeling, features are extracted from raw data and transformed into relevant features. In the case of lung cancer prediction, features could include the shape, size, and density of lung nodules or lesions observed in CT scans.

2 Literature Survey

Prior research in the field of lung nodule detection and classification has made significant advancements, employing various computational techniques and medical imaging technologies to aid in the early diagnosis and prognosis of lung cancer. Early studies often utilized traditional image processing methods, such as thresholding and feature extraction, to identify lung nodules in computed tomography (CT) scans. More recent works have shifted towards machine learning and deep learning approaches, leveraging convolutional neural networks (CNNs) and recurrent neural networks (RNNs) for improved accuracy. A notable development in this domain was the establishment of benchmark datasets, such as the Lung Image Database Consortium (LIDC) and the National Lung Screening Trial (NLST) dataset, which allowed for standardized evaluations of nodule detection algorithms. Despite these advancements, there remain challenges, including distinguishing between benign and malignant nodules, handling variations in nodule size and shape, and optimizing computational efficiency. This paper contributes to this growing body of research by presenting an innovative approach to lung nodule detection and classification, addressing some of these persisting challenges and demonstrating the potential for enhanced clinical outcomes.

Some of the published paper are mentioned below:

1. Title: "The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A completed reference database of lung nodules on CT scans."
 Abstract: This paper presents the creation of the Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI), a comprehensive reference database for lung nodules in CT scans. The database facilitates research in lung cancer detection and evaluation.
 Author: Armato, S. G., McLennan, G., Bidaut, L., McNitt-Gray, M. F., Meyer, C. R., Reeves, A. P.,... & Clarke, L. P. Year: 2011
2. Title: "Automated detection and quantification of solitary pulmonary nodules on high-resolution computed tomography scans."
 Abstract: This paper likely discusses the development of automated methods to detect and measure solitary pulmonary nodules in high-resolution CT scans, which is crucial for early lung cancer detection and diagnosis. It aims to improve the accuracy and efficiency of pulmonary nodule assessment in medical imaging. Author: Revel, M. P., & Gagnadoux, F. Year: 2015
3. Title: "Computer-aided diagnosis: how to move from the laboratory to the clinic."
 Abstract: this paper may discuss the challenges and strategies for transitioning computer-aided diagnosis (CAD) systems from experimental research settings to practical clinical use, potentially addressing the implementation of CAD in real-world medical settings.

Author: van Ginneken, B., Schaefer-Prokop, C. M., Prokop, M., van Rikxoort, E. M., & Scholten, E. T. Year: 2013

4. Title: “Automatic detection of subsolid pulmonary nodules in thoracic computed tomography images.”

Abstract: the development of automated methods for identifying subsolid pulmonary nodules in thoracic CT scans, which is essential for early lung cancer diagnosis and management, possibly highlighting the application of advanced image analysis techniques.

Author: Jacobs, C., van Rikxoort, E. M., Twellmann, T., Scholten, E. T., de Jong,

P. A., Kuhnigk & van Ginneken, B. Year: 2014

5. Title: “Clinical applications of deep learning for the thorax and chest: a literature review.”

Abstract: discusses the clinical applications of deep learning techniques in thoracic and chest imaging, exploring how these methods are used in medical diagnosis and analysis.

Author: Kim, H., Goo, J. M., Lee, K. H., & Lee, S. Year: 2019

3 Existing System

A traditional approach involves setting a threshold value, morphological operations, and region-based segmentation. These techniques are used to differentiate between pulmonary nodules and non-nodules in CT scan images. Thresholding relies on establishing a specific pixel density threshold to categorize regions within the images as potential nodules. However, this method can be limited by its sensitivity to image noise and artifacts, which can lead to inaccurate results. Morphological operations, such as erosion, dilation, and closing, are employed to enhance the shape and size of potential nodules, aiding in their identification. Nevertheless, these operations may not fully capture the complexity of nodule features, especially in cases involving irregularly shaped or textured nodules. Region-based segmentation involves dividing the images into distinct regions based on various features like texture, shape, and intensity. Subsequent analysis is performed within these regions to identify nodules.

This approach may encounter challenges when dealing with intricate nodule patterns and variations in size and shape. In essence, the existing system seeks to balance sensitivity and specificity in lung nodule detection, aiming to accurately identify nodules while minimizing false positives and false negatives in CT scan images.

4 Proposed System

The proposed system is designed to enhance the precision of lung cancer detection in CT scans while minimizing false positive diagnoses. It utilizes a sophisticated combination of deep artificial neural networks, specifically Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). The CNN is employed for pixel-level feature extraction, effectively distinguishing and analyzing pixels within CT scan images. This enables the capture of intricate details and patterns associated with lung nodules. The

RNN, on the other hand, is utilized for feature classification and the identification of pulmonary nodules. It excels in understanding sequential data and discerning complex patterns within the images. The system adopts a multi-step detection strategy. In the first stage, a sequence of CNNs identifies potential nodule candidates with high sensitivity. In the subsequent stage, specialized CNNs verify and refine the initial detections, ensuring a thorough and accurate assessment of lung nodules in CT scans. In the current landscape of lung nodule detection, deep learning-based techniques are recognized as the most effective. However, improvements are needed to enhance their accuracy, efficiency, and applicability in real clinical settings. The proposed system addresses these challenges, offering a comprehensive approach to lung nodule detection and classification in medical imaging, ultimately contributing to improved clinical outcomes.

Comparative analysis of the proposed system stands out by addressing the limitations of traditional approaches, such as sensitivity to image noise and difficulty with complex nodule patterns. It introduces deep learning techniques that are recognized for their effectiveness in medical imaging. The combination of CNNs and RNNs enhances both feature extraction and classification. This results in a more precise and reliable lung nodule detection system, minimizing false positives and false negatives. In contrast, traditional methods are less adaptable and may not achieve the same level of accuracy and efficiency. The proposed system represents a significant step forward in lung cancer detection and classification, offering the potential for improved clinical outcomes and a more reliable diagnostic process.

5 Module Implementation and Methods

5.1 Data Collection

To obtain a dataset of CT scan images that include images of lung nodules. The need to prepare the data involves cleaning the data and organizing it into a format. To acquire a dataset of CT scan images containing lung nodules, it is necessary to procure a collection that comprises such images. In our project, we used Chest CT-Scan images – Dataset from kaggle.com. Additionally, data preprocessing is required, which may involve adjusting the image size or normalizing pixel values. Preprocessing of the CT images is essential to enhance image quality and eliminate any disturbances. This entails isolating lung tissue from the remaining portions of the image by applying segmentation techniques like thresholding, region growing, and edge detection.

The need to prepare the data involves cleaning the data and organizing it into a format that can be easily read by deep learning frameworks. Deep learning framework's data loading functionality to import the dataset into your deep learning model. You may need to specify the path to the dataset and any other necessary parameters. This method allows you to train a model on one set of data, validate it on another, and test it on a third. The preprocessing techniques may include cropping, resizing, normalization, and filtering.

Figure 1 displays sample images from the lung CT dataset used in this study, illustrating different types of lung nodules. The four sub-figures (a, b, c, and d) provide visual examples of various pathological conditions found in the lung. (a) represents adenocarcinoma, (b) shows squamous cell carcinoma, (c) exhibits metastatic lung cancer, and (d) displays large cell carcinoma. These images are representative of the diverse nodule

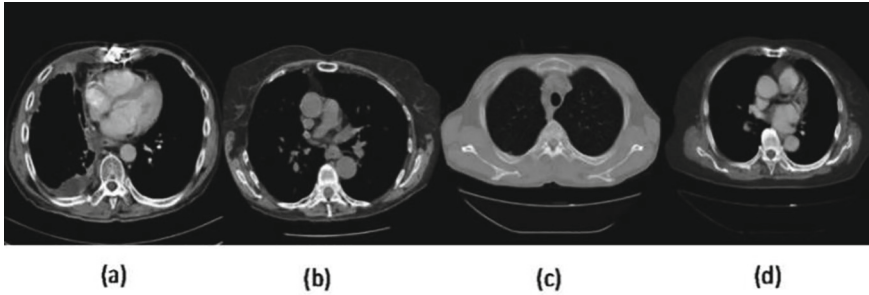


Fig. 1. Lung CT dataset samples: (a) adenocarcinoma, (b) squamous cell carcinoma, (c) metastatic lung cancer, and (d) large cell carcinoma

types and characteristics encountered in the dataset, which is essential for training and testing the proposed detection and classification system. These visual samples offer a glimpse into the complexity of lung nodule identification and classification in CT scans, emphasizing the importance of the research presented in this paper.

This task can be accomplished by employing deep learning methodologies like convolutional neural networks (CNNs) or alternative machine learning algorithms. The nodule detection process encompasses the identification of potential nodules within the segmented lung area, followed by the classification of these nodules as either nodules or non-nodules. Post-processing techniques can be applied to the detected nodules to remove false positives and to refine the segmentation and characterization of the nodules. This can be done using various morphological operations, adaptive thresholding, and region-growing techniques. The final step is to generate the output of the image processing system, which may include the location, size, and characteristics of the detected nodules.

5.2 Feature Selection and Engineering

This involves selecting a representative dataset that includes a variety of nodule sizes and shapes. The next step is to extract the regions of interest around the nodules. This involves cropping the CT scan images to include only the nodule and its surrounding tissue. Preprocessing the ROI images improves their quality and removes noise. The preprocessing techniques may include cropping, resizing, normalization, and filtering. Extract the features from the ROIs using various techniques such as gray-level co-occurrence matrix (GLCM) analysis, histogram analysis, texture analysis, and shape analysis. The extraction of features can be conducted at various scales to encompass the diverse attributes of the nodules. This approach allows for the identification and selection of the most pertinent features for nodule detection. This can be done using various feature selection techniques such as correlation-based feature selection, and embedded feature selection. This can be done using various techniques such as principal component analysis (PCA), nonlinear mapping, or kernel methods. The model can be trained using various techniques. The performance of the deep learning model should be evaluated using a set of metrics. A comparison can be made between predicted and

ground-truth labels for the test data in order to calculate these metrics. Based on the results of the performance evaluation, the feature extraction and engineering process can be iteratively refined until the desired level of performance is achieved.

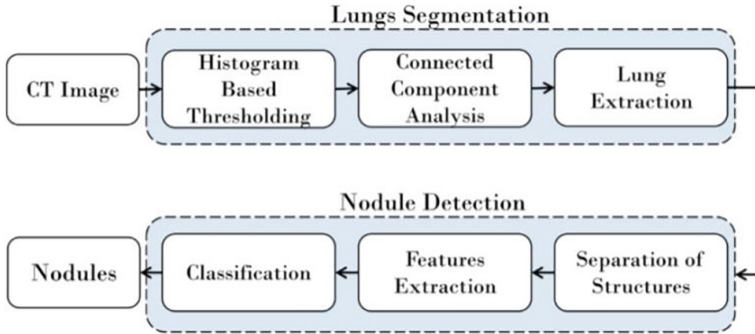


Fig. 2. Flow diagram of the proposed algorithm

Figure 2 presents the flow diagram of the proposed algorithm for lung nodule detection and classification. The algorithm comprises several essential stages. It begins with lung segmentation, where the CT images are processed to isolate the lung structures. This is followed by histogram-based thresholding to separate the lungs from the chest scan effectively. Connected component analysis is then applied to identify regions of interest within the lung. Next, nodule extraction is performed to isolate potential nodules from the image. The algorithm proceeds with the separation of structures, which refines the nodule extraction process. Feature selection is employed to choose the most relevant characteristics for classification. Finally, classification is executed to categorize the identified structures, whether they are nodules or non-nodules. This flow diagram provides an overview of the systematic steps involved in the algorithm, demonstrating the process for detecting and classifying lung nodules in CT images.

5.3 Data Preprocessing

Import the CT scan images and their corresponding annotations into the memory of the system. Convert the RGB images to grayscale, as the model will only need to process one channel. Resample the images to a uniform resolution to ensure consistency in the data. This may involve increasing or decreasing the resolution of the images, depending on the initial resolution and the requirements of the model. Normalize the pixel values to ensure that they are within a standardized range. This may involve rescaling the pixel values to a range of $[0,1]$ or $[-1,1]$. In Fig. 3, a visual representation of various anatomical structures within the human lung is observed on a CT scan. These structures are vital for understanding the context of lung nodule detection and classification. The figure highlights key components such as the airways, upper lobe, chest wall, horizontal fissure, middle lobe, primary tumor, oblique fissure, lower lobe, and vessels, and distinguishes between the right lung and left lung. This illustration provides to identification

and comprehension of the anatomical landmarks within the lung when considering the localization and classification of nodules in subsequent sections.

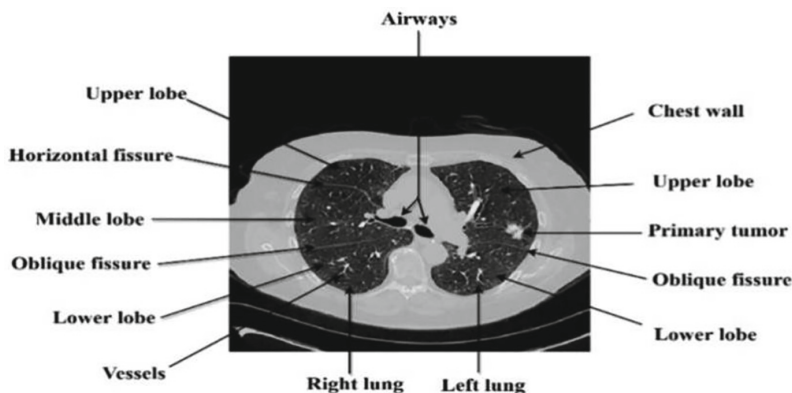


Fig. 3. Lung structures on the CT scan

Common filters used in CT scan preprocessing include Gaussian filters, median filters, and edge enhancement filters. Crop the images to remove any unnecessary regions that are not relevant to lung nodule detection. Resize the images to a standardized size that can be processed by the model. This may involve random rotations, flips, and scaling of the images. Split the data into train and test sets using a stratified sampling technique to ensure that both sets have a similar distribution of nodule sizes and shapes.

Data preprocessing methods in Lung Cancer Detection from CT Scans includes:

Image Acquisition and Collection: The initial step involves the acquisition of CT scan images, typically from medical institutions or databases. These images may vary in terms of resolution, orientation, and quality.

Image Standardization: To ensure uniformity, the acquired CT scan images are standardized to a consistent format. This includes adjustments for resolution, orientation, and pixel spacing. This step aims to remove variations that can affect subsequent analysis.

Noise Reduction: Image noise can adversely impact the accuracy of nodule detection. Various noise reduction techniques, such as filtering or denoising algorithms, are applied to enhance image quality and clarity. This step is particularly important for reducing false positives.

Contrast Enhancement: Enhancing the contrast in CT scan images can help in highlighting subtle features, such as nodules. Histogram equalization or contrast stretching techniques are commonly employed to improve image contrast.

Resampling: CT scans may have different slice thicknesses, which can affect the analysis. Resampling is often performed to ensure uniform slice thickness, making it easier to identify nodules consistently across the dataset.

Lung Segmentation: One critical preprocessing step is lung segmentation. This involves separating the lung region from the rest of the chest scan. Various methods, including thresholding and region-growing algorithms, are used to isolate the lungs.

Nodule Candidate Detection: To identify potential nodule candidates, techniques like region-growing, thresholding, or connected component analysis are employed. This step is crucial for narrowing down regions of interest in the lung.

Region Labeling: Once nodule candidates are identified, each candidate region is labeled and analyzed to assess its characteristics. This may involve further noise reduction and size filtering to eliminate false candidates.

Feature Extraction: Features are extracted from the labeled regions, which help in distinguishing between nodules and non-nodules. Common features include shape, texture, and intensity characteristics of the regions.

Normalization: The extracted features are often normalized to make them comparable and to prevent any single feature from dominating the classification process.

5.4 Model Selection

Select a suitable deep learning framework that is capable of identifying lung nodules from the available options, which encompass Convolutional Neural Networks (CNNs), U-Net, Mask R-CNN, and various other options. This process involves training the model multiple times with different parameter values to obtain the best-performing set. The procedure entails making predictions on the testing dataset and then evaluating the results by comparing them to the ground truth annotations. This evaluation enables the computation of performance metrics such as accuracy, sensitivity, specificity, F1-score, and the area under the receiver operating characteristic curve (AUC-ROC). Optimize the model performance by analyzing the performance metrics and tuning the hyperparameters or adjusting the model architecture accordingly. Once the model with performed, deploy it in a production environment to detect lung nodules in new CT scan images.

In Fig. 4 illustrates the neural network architecture employed in the proposed method for nodule classification. The figure outlines the various components of the network, including X_1 , X_2 , X_{16} , and X_{17} , which represent input features. These features are passed through the input layer and subsequently through the pattern layer and summation layer for further processing. The network is designed to classify nodules into two categories: Benign nodules and Malignant nodules. The final decision is made in the output layer of the network. This diagram provides an overview of the neural network structure used for the classification of lung nodules, which plays a crucial role in the proposed methodology.

5.5 Data Splitting

Load the dataset of CT scans with annotated nodule regions into memory. Divide the shuffled dataset into two parts - a training set and a testing set. A commonly used split is 80% training and 20% testing. To ensure that both the training and testing sets have

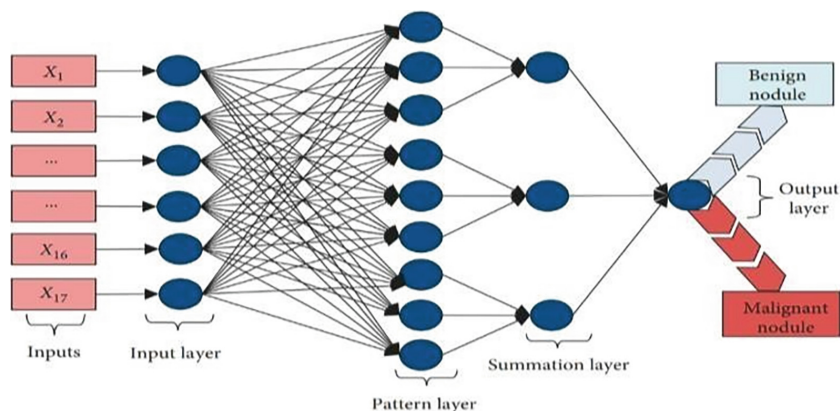


Fig. 4. Neural network architecture used in the proposed method for nodule classification.

a similar distribution of nodule sizes and shapes, it is recommended to use stratified sampling. The proportion of samples in each class (e.g., nodule vs. non-nodule) is the same in both the train and test sets. Save the train and test datasets to disk as separate files. The files can be saved in a format that can be easily loaded into the deep learning framework of choice, such as HDF5 or NumPy format. Data augmentation techniques include random rotations, flips, and scaling of the images. Before training the model, it is important to preprocess the images to ensure that they are in the correct format and are normalized. Typical preprocessing methods involve adjusting the image dimensions to a standardized size, normalizing the pixel values, and employing filters to eliminate noise.

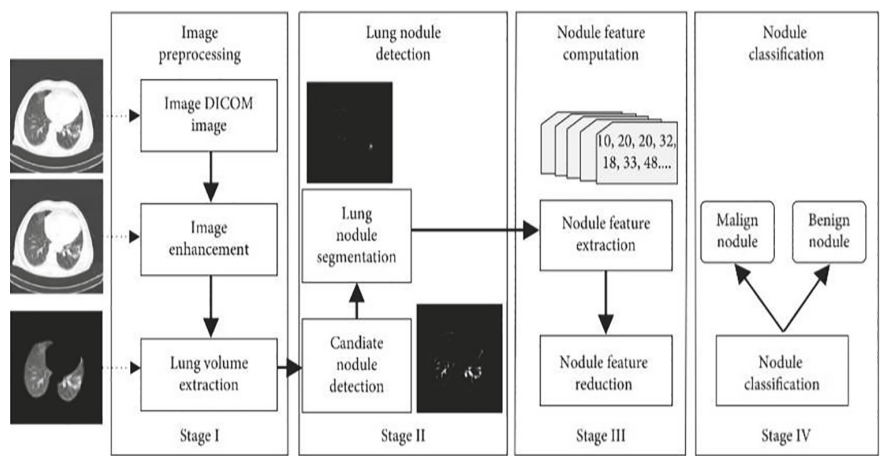


Fig. 5. Workflow of the designed pipeline to detect lung nodules. The system consists of four stages.

Figure 5 provides an overview of the workflow of the designed pipeline for the detection of lung nodules, which is organized into four distinct stages. In Stage I, the process begins with Image preprocessing that involves operations on DICOM images. This is followed by Image enhancement and Lung volume extraction. Subsequently, the focus shifts to Lung nodule detection, where candidate nodules are identified. In this stage, Nodule feature computation plays a crucial role, with specific values listed as “10, 20, 20, 32, 18, 33, 48.” Moving on to Stage II, the workflow encompasses Lung nodule segmentation and Candidate nodule detection. In stage III, the pipeline also includes Nodule feature extraction and Nodule feature reduction as part of the feature engineering process. Stage IV is dedicated to Nodule classification, where the system distinguishes between Malignant nodules and Benign nodules. The nodules undergo a classification process, contributing to the system’s diagnostic capabilities.

5.6 Model Training and Testing

Initialize the deep learning model with random weights or pre-trained weights from a pre-trained model. Select a suitable loss function to optimize the model. Common loss functions for lung nodule detection include binary cross-entropy and Dice loss. Select a standard for evaluating the model’s performance throughout the training procedure. Common measures for detecting lung nodules include precision, recall, selectivity, F1-measure, and the area under the receiver operating characteristic curve (AUC-ROC). Fine-tune the hyperparameters of the model, such as the learning rate, optimizer, batch size, and regularization, to maximize the model’s performance. Employ a training loop that progressively updates the model parameters based on the loss function and back-propagation. Continuously assess the model’s performance on a validation set to prevent overfitting and modify the hyperparameters accordingly. Save the trained model and its parameters for later use in inference and deployment.

5.7 Accuracy and Prediction

Load the trained deep-learning model into memory. Load the test data into memory, including the CT scan images and their corresponding annotations. Preprocess the test data using the same steps as for the training data, including resampling, normalization, filtering, and cropping/resizing. Convert the preprocessed test data into tensors using identical procedures as those employed for the training data, incorporating data augmentation and standardization. Employ the test data and the trained model to assess the model’s accuracy. This process may entail calculating metrics such as precision, recall, F1-measure, and the area under the receiver operating characteristic curve (AUC-ROC). Utilize the trained model to generate predictions on novel CT scan images that have not been utilized for training or testing purposes. This may involve preprocessing and transforming the new data in the same way as for the test data, and using the model to generate a prediction for each image. Evaluate the accuracy of the model predictions by comparing them to ground truth annotations, either manually or using automated tools. This process may require calculating measures such as precision, recall, selectivity, accuracy, and positive predictive value (PPV).

Figure 6 presents the confusion matrix for the proposed model, which is a critical component in assessing the performance of the model’s predictions. The matrix displays both predicted labels and true labels for various categories, including LUAD, LUSC, meta, and normal. In this figure, LUAD refers to Lung adenocarcinoma, LUSU refers to lung squamous cell carcinoma and meta refers to metastatic lung cancer which has distinct characteristics and is important for the classification and diagnosis of lung nodules in medical imaging and research.

		Predicted label				Recall(%)
		LUAD	LUSC	meta	normal	
True label	LUAD	21.6	5.0	1.5	2.0	72
	LUSC	1.5	32.0	0	0.5	94
	meta	1.2	0	22.3	1.5	89
	normal	4.0	1.0	0.2	36.8	88
Precision(%)		76	84	93	90	Accuracy 84%

Fig. 6. Confusion matrix for the proposed model

5.8 Data Visualization

Load the predicted data into memory, which may include the CT scan images, the predicted nodule locations, and the confidence scores for each prediction. Load the original CT scan images into memory, which may be in the form of DICOM files or other medical image formats. Use the predicted nodule locations to generate visualizations of the predicted regions of interest (ROIs) on the original CT scan images. This may involve overlaying the predicted locations on the original images or generating heatmaps that highlight the regions with the highest confidence scores. Compare the predicted locations with ground truth annotations, which may have been manually annotated by radiologists or generated by automated tools. This may require calculating measures such as accuracy, sensitivity, specificity, and positive predictive value (PPV). Produce heatmaps that accentuate the incorrect positive and incorrect negative predictions to aid in identifying areas where the model can be enhanced. Generate a receiver operating characteristic (ROC) curve to visualize the balance between the true positive rate (sensitivity) and the false positive rate (1-specificity) across various decision thresholds. Visualize the model confidence scores for each prediction, to help identify cases where the model may be uncertain or where additional human review may be necessary.

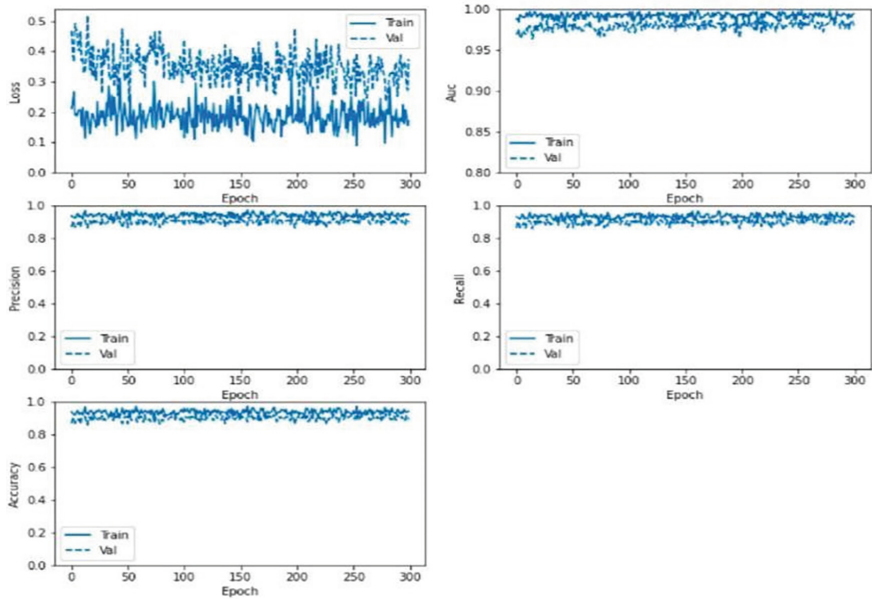


Fig. 7. Loss, AUC, precision, recall, and accuracy between the training and validation phases

Figure 7 presents the Loss, AUC, precision, recall, and accuracy between the training and validation phases. The overall accuracy of the model's predictions, measuring how well it correctly classified instances across all categories. In this case, the accuracy is 84%, indicating the proportion of correct predictions relative to all predictions. Precision is a measure of how many positive predictions were actually true positives. It is provided as percentages for different categories, such as 76% and 84%, reflecting the precision of the model's predictions in those specific categories. Recall, also known as sensitivity, gauges how many true positive instances were correctly identified by the model. It is presented as percentages, with values like 93% and 90% for different categories.

5.9 Results

Accuracy and Performance Metrics: The performance of the nodule classification model is evaluated using various metrics, including accuracy, precision, recall, F1-score, and the area under the receiver operating characteristic curve (AUC-ROC). These metrics provide a comprehensive understanding of the model's ability to correctly classify nodules.

Confusion Matrix: A confusion matrix is presented to visualize the model's true positive, true negative, false positive, and false negative predictions. This matrix allows for a deeper analysis of the model's performance in distinguishing between benign and malignant nodules.

Discussion and Analysis: The results are analyzed to assess the model's ability to accurately classify lung nodules. Any challenges or limitations encountered during the

classification phase are addressed and discussed. Insights are provided into the model's strengths and areas for potential improvement.

Clinical Implications: The clinical significance of the classification results is highlighted. Emphasize how accurate nodule classification can aid in early lung cancer diagnosis and prognosis.

6 Conclusion

The research paper introduces an automated system designed to detect lung nodules in CT images. The system comprises two phases: lung segmentation and nodule detection. A technique called threshold estimation is proposed to automatically separate the lungs from the chest CT scan. This technique employs structural characteristics for classification purposes. To accurately classify numerous regions of interest extracted from the lung segmentation, CNN and RNN techniques are utilized. The classification challenge is addressed by incorporating shape features and the presence of nodules in consecutive slices. The proposed system achieved impressive outcomes, with a sensitivity of 0.8785 and an accuracy of 0.85. Comparative analysis with other computer-aided diagnosis systems demonstrates the effectiveness of the proposed approach. The system offers valuable support in the early stages of lung cancer diagnosis, as it efficiently analyzes a large number of images, minimizing the risk of human error.

The system can serve as an initial diagnostic tool, where identified cases can undergo further medical analysis and investigations for confirmation. This approach can alleviate the strain on public hospitals, which may face a shortage of specialists and a high influx of patients. Moreover, the system is cost-effective as it can be deployed on standard computers readily available in hospitals and clinics. The proposed system acts as a valuable aid to radiologists, enabling them to swiftly and accurately detect lung nodules. By doing so, it contributes to alleviating the workload on hospitals and enhancing the early detection of lung cancer.

References

1. The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A Completed Reference Database of Lung Nodules on CT Scans. <https://pubmed.ncbi.nlm.nih.gov/?term=Armato%20SG%5BAuthor%5D>
2. Kanipriya, M., Hemalatha, C., Sridevi, N., SriVidhya, S., Jany Shabu, S.: An improved capuchin search algorithm optimized hybrid CNN-LSTM architecture for malignant lung nodule detection. *Biomed. Signal Process. Control* **78**, 103973 (2022). <https://www.mdpi.com/2313-433X/6/12/131>
3. Gao, X.W., James-Reynolds, C., Currie, E.: Analysis of tuberculosis severity levels from CT pulmonary images based on enhanced residual deep learning architecture. *Neurocomputing* **92** (2019). https://www.researchgate.net/publication/332611077_Analysis_of_Tuberculosis_Severity_Levels_From_CT_Pulmonary_Images_Based_on_Enhanced_Residual_Deep_Learning_Architecture
4. Alsharman, N., Jawarneh, I.: GoogleNet CNN neural network towards chest CT coronavirus medical image classification. *J. Comput. Sci.* **16**(5), 620–625 (2020). <https://thescipub.com/abstract/jcssp.2020.620.625>

5. Gayathri, J.L.: A computer-aided diagnosis system for the classification of COVID-19 and non-COVID-19 pneumonia on chest X-ray images by integrating CNN with sparse autoencoder and feed-forward neural network. *Comput. Biol. Med.* **141**, 105134 (2022). <https://pubmed.ncbi.nlm.nih.gov/34971978/>
6. Golan, R., Jacob, C., Denzinger, J.: Lung nodule detection in CT images using deep convolutional neural networks. In: *Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, Vancouver, BC, Canada (2016). <https://ieeexplore.ieee.org/document/7727205>
7. Tomassini, S., Falcionelli, N., Sernani, P., Burattini, L., Dragoni, A.F.: Lung nodule diagnosis and cancer histology classification from computed tomography data by convolutional neural networks: a survey. *Comput. Biol. Med.* **146**, 105691 (2022). <https://www.sciencedirect.com/science/article/pii/S0010482522004747>
8. Qiao, K., et al.: A neuro-heuristic approach for recognition of lung diseases from X-ray images. *Expert Syst. Appl.* (2019). <https://doi.org/10.1016/j.eswa.2019.01.060> https://www.researchgate.net/publication/330995650_A_neuro-heuristic_approach_for_recognition_of_lung_diseases_from_X-ray_images
9. Ma, J., Song, Y., Tian, X., Hua, Y., Zhang, R., Wu, J.: Survey on deep learning for pulmonary medical imaging. *Front. Med.* **14**, 450–469 (2020). <https://link.springer.com/article/https://doi.org/10.1007/s11684-019-0726-4>
10. Rakshit, S., Saha, I., Wlasnowolski, M., Maulik, U., Plewczynski, D.: ‘Deep Learning for Detection and Localization of Thoracic Diseases Using Chest X-Ray Imagery. In: *Artificial Intelligence and SoftComputing*, pp. 271–282 (2019). https://www.researchgate.net/publication/333410212_Deep_Learning_for_Detection_and_Localization_of_Thoracic_Diseases_Using_Chest_X-Ray_Imager



Mathematical Formulation and Genetic Algorithm for Permutation Flow Shop Scheduling with Release Date to Minimize Makespan

Sachin Karadgi¹  and P. S. Hiremath²

¹ Department of Automation and Robotics, KLE Technological University,
Hubballi 580031, India

sachin.karadgi@kletech.ac.in

² Master of Computer Applications, KLE Technological University, Hubballi 580031, India

Abstract. Manufacturing enterprises employ a series of production or assembly operations to manufacture products. These operations are performed in a particular sequence. It is crucial to establish the job sequence that will undergo the series of operations; this is known as the Permutation Flow Shop Scheduling (PFSS) problem. Mixed Integer Linear Programming (MILP) as a mathematical model is elaborated to address the PFSS problem subjected to jobs with release date constraints and minimize makespan. This scheduling problem belongs to the NP-hard optimization problem. Subsequently, a Genetic Algorithm (GA) approach has been designed and implemented. This GA approach proved highly effective in both computational efficiency and optimization performance.

Keywords: Mixed Integer Liner Programming · Genetic Algorithm · Flow Shop Scheduling · Makespan

1 Introduction

In many manufacturing enterprises, products are produced through a series of production and assembly operations. The products often undergo these production and assembly operations in the same sequence without skipping any operations. This requires setting up the production and assembly machines in a series known as a flow shop scenario [22], as illustrated in Fig. 1. Furthermore, scheduling is critical to determine the sequence of products (or jobs) operated on machines under certain constraints [16].

The flow shop scheduling problem refers to the task of scheduling n jobs on m machines subject to certain constraints as listed below:

1. At time zero, machines are available, but jobs can be processed after their release date r_j .
2. Each job shall be operated on all machines, and the job sequence should not be altered between machines, i.e., permutation (*pmu*).

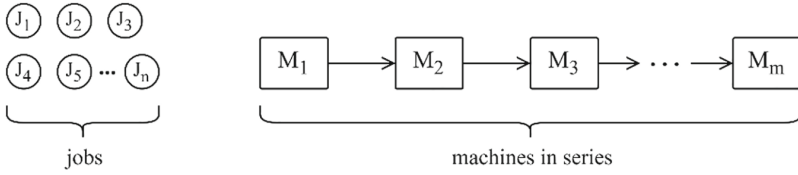


Fig. 1. Flow shop scenario with operations carried out in the same sequence.

3. A machine, at any given time, can operate only one job.
4. A job, at any given time, can be operated by a machine.
5. Interruption of a job is not permitted.
6. Except on the first machine, an operation on a job can be initiated after its operation on the predecessor machine.
7. There exists no prior priority among jobs.

There exist numerous objective functions [23]. However, the utilization of resources is critical. Hence, the article minimizes makespan, C_{max} , and indicates when the last job completes processing on machine m . The previously mentioned problem statement shall be considered as a Permutation Flow Shop Scheduling (PFSS) problem, which shall be represented as $F_m | r_j, pmul | C_{max}$ with m machines considering release date r_j of job j and job sequence remains unchanged with an aim is to minimize C_{max} . Research has been carried out to address $F_m | pmu | C_{max}$ and has proposed mathematical formulation [3, 12, 14, 22, 28, 30], but many of these formulations assume that a set of possible job sequences Π are already defined [12, 14, 28, 30]. Furthermore, researchers have also proposed numerous heuristic [23] and meta-heuristics algorithms [16, 23, 27, 28]. However, the literature is missing handling the release date r_j constraints.

A mathematical formulation of the PFSS problem mentioned above is presented, and this mathematical formulation is solved using the Genetic Algorithm (GA) approach to obtain a near-optimal fitness value. This article is arranged as follows. Section 2 briefly discusses a literature review of mathematical formulations and meta-heuristic algorithms concerning the PFSS problem. A comprehensive explanation of a mathematical formulation and the corresponding GA approach is provided in Sect. 3. The computational results of the Integer Programming (IP) solver and GA approach, including comparison for numerous data groups and parameters, are discussed in Sect. 4. In conclusion, the article discusses the problem of PFSS and various approaches to solving it in Sect. 5.

2 Literature Review

Researchers proposed heuristic techniques to solve PFSS problems. Among them, NEH (Nawaz-Enscore-Ham) [21], CDS (Campbell, Dudek, and Smith) [1], and slope [22] heuristic techniques exist. The ordered flow shop scheduling problem was formulated as a mathematical model to minimize makespan [9]. Furthermore, Pyramidal-NEH, pair-insert, and iterated local search algorithms were proposed and validated [9]. A hybrid metaheuristic algorithm based on GA was proposed [30]. In much of the literature, few individuals of the initial population are generated using well-known heuristic techniques (e.g., NEH, CDS).

A Mixed Integer Linear Programming (MILP) formulation was described to address the PFSS problem and minimize C_{max} for two machines [3], which was further extended to m machines [24, 29]. Also, a firefly algorithm was presented to solve the mathematical formulation [24]. Also, a mathematical formulation was proposed to solve the PFSS problem to minimize C_{max} subjected to machine blocking and sequence-dependent setup times [18], which was further solved with GA.

Population-based techniques, like Evolutionary Algorithms (EA), are utilized to determine the fitness functions/objectives of optimization problems [4]. For example, GA, Particle Swarm Optimization (PSO), evolutionary programming, and so forth are a few of the EA techniques. Being NP-Hard, the permutation flow shop problem is solved using various EA approaches. For instance, the problem was solved using GA [5, 16], artificial bee colony [12], PSO [14], and firefly [24] algorithms. A hybrid GA and tabu search algorithm was proposed, where the outcome of GA was treated as the initial solution in tabu search [27]. Likewise, a combination of GA and simulated annealing algorithms was proposed, where the GA schedule was further utilized to find a better schedule by simulated annealing algorithm [28].

The current article employs the GA approach to address the PFSS problem. Algorithm 1 specifies the different steps of GA as a pseudocode.

Algorithm 1 Pseudocode of Genetic Algorithm (GA)

- 1: Define solution coding;
- 2: Define T as the maximum generation counter and t as the generation counter;
- 3: Set $t = 1$ and T ;
- 4: Generate population with size N' as $P(t = 1)$;
- 5: Assign fitness to $P(t = 1)$ after the evaluation of objective and constraints;
- 6: **while** $t \leq T$ **do**
- 7: From $P(t)$, select parent population $M_p(t)$ for GA operations;
- 8: Create offspring population $Q(t)$ by implementing crossover and mutation operations from $M_p(t)$;
- 9: For every offspring in $Q(t)$, calculate fitness function;
- 10: From $P(t)$ and $Q(t)$, select succeeding generation $P(t + 1)$;
- 11: Set $t = t + 1$;
- 12: **end while**

3 Methodology

A mathematical formulation of the PFSS problem with a release date and objective to minimize makespan is presented in Sect. 3.1. This PFSS problem is an NP-hard optimization problem, which is solved using the GA approach detailed in Sect. 3.2.

3.1 Mathematical Formulation

The PFSS problem, as discussed in Sect. 1, shall be denoted as $Fm|rj, prmu|C_{max}$. MILP formulation employs numerous indices, parameters, and decision variables listed below. It is to be noted that the time unit, i.e., hours, days, etc., depends on the product that is being manufactured in the assembly line.

- Indices

i –Index of machine

j –Index of job

k –Position of jobs to be processed on machines

- Parameters

m – Number of machines

n – Number of jobs

M – $M = \{1, 2, \dots, m\}$, denoting set of machines

N – $N = \{1, 2, \dots, n\}$, denoting set of jobs to be scheduled

r_j – Release date of job j in time units

p_{ij} – Processing time of job j on machine i

- Continuous Variables

C_{ik} – Completion time of job on machine i at position k

S_{ik} – Start time of job on machine i at position k

C_{max} – Makespan

- Decision Variables

$$x_{kj} = \begin{cases} -1, & \text{if job } j \text{ is allocated on machines to position } k \\ 0, & \text{otherwise} \end{cases}$$

Below is the mathematical formulation as MILP for $Fm \mid r_j, pmu \mid C_{max}$. Objective (1) depicts the minimization of C_{max} of the PFSS problem. For each position k , a job j is assigned only once as indicated by Constraints (2). Similarly, a position k is assigned to each job j , as depicted in Constraints (3).

Constraints (4) assist in assigning a job at position $k = 1$ on machine $i = 1$. Likewise, Constraint (5) assigns remaining jobs for positions $k \geq 2$ on machine $i = 1$. Constraints (6) ensures that the start time S_{ik} of job j should be greater than the job availability r_j at position k on machine $i = 1$.

Constraints (7) is employed to allocate jobs for k positions on machines $i \geq 2$ considering the completion time of a job at position k on the previous machine. The allocation of jobs at position k on machine i such that the C_{ik} position is larger than the previous position $k - 1$ is ensured by Constraints (8). Constraint (9) ensures that the start time of jobs for position k on machine $i \geq 2$ is determined by the job completion time at position k on the previous machine. Constraints (10) assist in determining C_{max} , i.e., the last job completes its processing on machine m . Finally, Constraints (11)–(12) depict the non-negative constraints.

$$\text{Minimize } C_{max} \tag{1}$$

subjected to

$$\sum_{j \in N} x_{kj} = 1, \forall k \in N \tag{2}$$

$$\sum_{k \in N} x_{kj} = 1, \forall j \in N \quad (3)$$

$$S_{11} = C_{11} - \sum_{j \in N} x_{1j} \cdot p_{1j} \quad (4)$$

$$S_{1k} = C_{1k} - \sum_{j \in N} x_{kj} \cdot p_{1j}, k = 2 \dots n \quad (5)$$

$$S_{1k} \geq \sum_{j \in N} x_{kj} \cdot r_j, \forall k \in N \quad (6)$$

$$C_{ik} \geq C_{(i-1)k} + \sum_{j \in N} x_{kj} \cdot p_{ij}, i = 2 \dots m, \forall k \in N \quad (7)$$

$$C_{ik} \geq C_{i(k-1)} + \sum_{j \in N} x_{kj} \cdot p_{ij}, i \in M, k = 2 \dots n \quad (8)$$

$$S_{ik} = C_{ik} - \sum_{j \in N} x_{kj} \cdot p_{ij}, i = 2 \dots m, \forall k \in N \quad (9)$$

$$C_{max} = C_{mn} \quad (10)$$

$$C_{ik} \geq 0, \forall i \in m, \forall k \in N \quad (11)$$

$$x_{kj} \in \{0, 1\}, \forall k, j \in N \quad (12)$$

The scheduling problem $F_m \mid r_j, pmu \mid C_{max}$ is an NP-hard optimization problem [16, 27]. Subsequently, a (near) optimal fitness value to the above-formulated PFSS problem is computed using the GA approach.

3.2 GA Approach

The following paragraphs elaborate on the different steps of GA to compute the near-optimal fitness function to the previously proposed mathematical problem of the PFSS problem. The corresponding literature on these individual steps is also provided to understand the elaborated GA approach better.

1. **Solution Representation:** It is crucial to map the real-world problem (i.e., phenotype) to the GA world (i.e., genotype) by defining the possible representation of solutions or individuals [4]. Subsequently, representing a solution/individual as a chromosome is critical, impacting the encoding, decoding, and other operations associated with GA. In the current context, the length of the chromosome depends on the number of jobs, and gene values indicate the job number [11, 16, 18, 28], as illustrated in Fig. 2.
2. **Population and Initial Population Generation:** A parent population $P(t)$ contains N' valid individuals, which are generated randomly. Randomness enables diversity in the population [5]. However, a few individuals are generated based on the available heuristic techniques that are employed to rapidly converge to optimal or near-optimal fitness value [5]. In the current context, an individual is generated based on the NEH

heuristic [12, 13, 15, 21], $m - 1$ individuals are generated using the CDS heuristic [1, 2, 5, 19], and an individual is generated using the Slope heuristic [19, 22]. The remaining individuals, i.e., $N' - I[\text{NEH}] - (m - 1)[\text{CDS}] - I[\text{Slope}]$, are randomly generated by allocating a job j randomly from jobs available and to randomly available genes.

1	2	3	4	5	6	7	8
3	6	4	7	2	8	1	5

$n = 8$, i.e., indices = $\{1, 2, \dots, 8\}$

Fig. 2. Representation of solution/individual.

3. **Fitness Function:** The objective function of the PFSS problem is to minimize C_{max} indicated by C_{mn} , which can be considered as a fitness function. The fitness function must also be modified to include penalty terms to address the scheduling constraints [4]. In the current context, release dates r_j constraints do not require penalty terms as the individual will calculate the completion time C_{ik} considering the release dates r_j on the first machine.
4. **Parent Selection Strategy:** It is necessary to distinguish among individuals, preferably based on fitness value, to identify better individuals as parents to generate offspring for the succeeding generation. The selection of individuals is highly probabilistic. High-quality individuals have a high chance of becoming parents. Still, at the same time, there is a slight chance that low-quality individuals get an opportunity to become parents to avoid local optimum. The mating pool population $M_p(t)$ must be selected from $P(t)$ for every generation. The offspring population $Q(t)$ is produced after $M_p(t)$ undergoes crossover and mutation operations. There are different parent selection algorithms available to choose from, namely, roulette wheel [4, 11], elite selection [11], and k -way tournament selection [4, 8, 10, 11], which is employed with $k = 2$ in the current context.
5. **Crossover Operation:** New individuals are created by performing specific operations on the selected individuals [4]. The crossover (or recombination) operation ensures the creation of one or two offspring individuals by exchanging information between two parents chosen from $M_p(t)$. Multiple ways exist to accomplish a crossover operation, namely, one-point crossover [14, 20], two-point crossover [11, 14, 20, 28, 30], Linear Order Crossover (LOX) [2, 5, 16], and position-based crossover [16, 20, 28]. In the current context, two offspring are produced using the LOX operation, as depicted in Fig. 3. Randomness is created by selecting the parents randomly, identifying crossover points, and generating probability r_c randomly in $[0, 1]$ for comparison with predefined crossover probability P_c . If the $r_c \leq P_c$, a crossover operation is performed; otherwise, parents are exactly replicated as offspring.
6. **Mutation Operation:** The population diversity can be increased through mutation operations that result in minor modifications in offspring. There are various methods to carry out mutation, namely, reverse order mutation [12, 16, 28], insert mutation [12, 16, 28], swap mutation [5, 11, 12, 16, 18, 27], and shift mutation [5, 14, 20,

30], and so forth. A probability r_m is randomly generated in $[0,1]$ for comparison with predefined mutation probability P_m . If the $r_m \leq P_m$, a mutation operation is performed as depicted in Fig. 4; otherwise, there is no change in offspring.

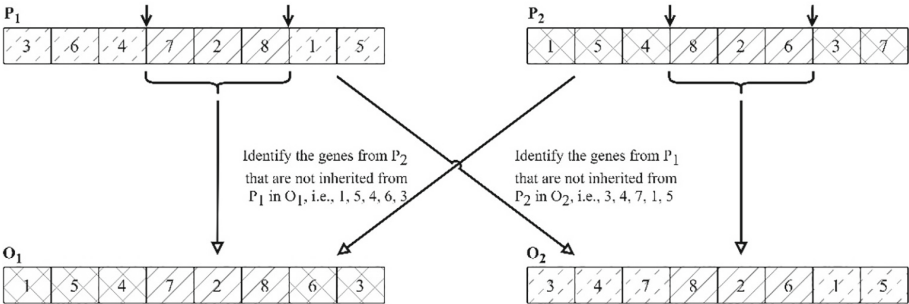


Fig. 3. Linear Order Crossover (LOX).

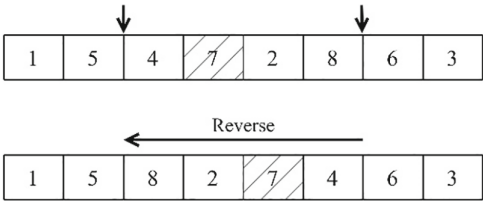


Fig. 4. Reverse mutation.

7. **Survivor Selection Strategy:** A survivor selection strategy is defined as choosing $P(t + 1)$ from $P(t)$ and $Q(t)$ [4]. Here, $(\mu + \lambda)$ selection strategy is used to choose the best μ individuals from $P(t)$ and λ individuals from $Q(t)$.
8. **Termination:** The algorithm needs to be terminated, and there are different options to terminate the algorithm. The algorithm can be terminated after a predefined T generation [4, 5, 7, 17]. The algorithm can also be terminated when the change in fitness value is within a threshold for a given number of generations. Also, the algorithm can be terminated after the predefined CPU time elapses. Finally, the algorithm can be terminated when the diversity in the population drops below a given threshold. Here, the algorithm terminates if the fitness value remains unchanged for a specified number of generations or after T generations.

4 Computational Experiments

MILP and GA approach towards solving PFSS problem is implemented on Intel Intel Xeon CPU E5-2420 V2 @ 2.20 GHz with 64 GB RAM on Windows 10 64bit OS using Python 3.9.16 64-bit programming language. The PFSS problem scenario discussed in the current article is taken from local industry. However, the corresponding data cannot be published. Subsequently, the Taillard data sets are taken from available sources to validate the mathematical formulation and GA approach [25, 26]. The data set includes processing times p_{ij} and C_{max} as an upper bound, as shown in Fig. 5. The release dates r_j are randomly generated as they are missing in the Taillard data sets. The results obtained from the GA approach were compared with those obtained from the IP solver, namely Google OR Tools, which internally employs the SCIP (Solving Constraint Integer Programs) framework [6].

```

number of jobs, number of machines, initial seed, upper bound and lower bound:
           20                5          73654221          1278          1232
processing times :
54 83 15 71 77 36 53 38 27 87 76 91 14 29 12 77 32 87 68 94
79  3 11 99 56 70 99 60  5 56  3 61 73 75 47 14 21 86  5 77
16 89 49 15 89 45 60 23 57 64  7  1 63 41 63 47 26 75 77 40
66 58 31 68 78 91 13 59 49 85 85  9 39 41 56 40 54 77 51 31
58 56 20 85 53 35 53 41 69 13 86 72  8 49 47 87 58 18 68 28

```

Fig. 5. Screenshot of Taillard's data set for $n = 20$ and $m = 5$.

The experiments are carried out for numerous Taillard data sets. Table 1 presents GA parameters for solving different data sets. Two groups of experiments were carried out. The mathematical formulation and corresponding GA were solved assuming $r_j = 0$, i.e., an attempt was made to solve the $F_m | pmul C_{max}$ in the first group of experiments. These experiments were carried out to validate the mathematical formulation and GA approach, and the corresponding results are presented in Table 2. It was noticed that the C_{max} obtained by the IP solver was in line with the upper bound listed in Taillard data sets but computationally very expensive. The NEH heuristic can be relied on to provide the initial solution for the GA approach. Finally, the C_{max} obtained from the GA approach has very minimum gaps, i.e., (GA – MILP)/MILP, obtained after a few iterations, as illustrated in Fig. 6.

Table 1. Parameter settings employed during GA computation.

Parameter	Common Settings			
P_c	0.8			
P_m	0.1			
μ	0.1			
λ	0.9			
	$n \leq 10$	$10 \leq n < 20$	$20 \leq n < 50$	$n \geq 50$
N	50	100	150	200
#Similar Fitness	10	20	30	40
	$n * m < 100$	$100 \leq n * m < 500$	$500 \leq n * m < 1000$	$n * m \geq 1000$
T	50	100	150	200

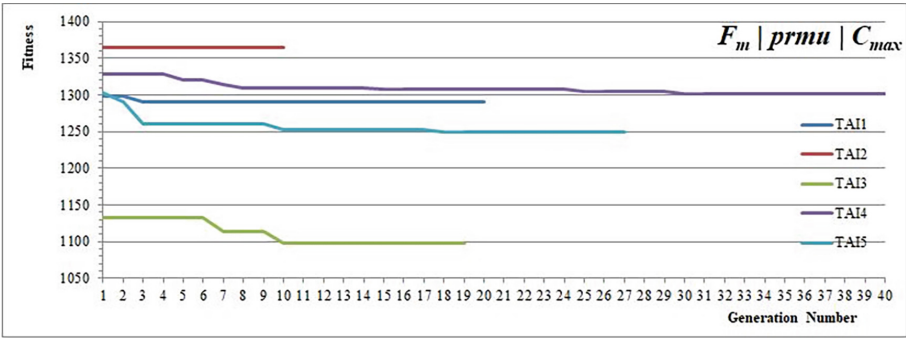


Fig. 6. Progress curves of GA experiments for $F_m | prmu | C_{max}$.

The mathematical formulation for $F_m | r_j, prmu | C_{max}$, and the implemented GA was validated as part of the second group of experiments, with results presented in Table 3. It was observed that the C_{max} obtained by solving MILP was computationally very expensive, and the solution by MILP solver did not converge even after 24 h of computation time in case of larger problem size. Hence, the MILP solver is terminated prematurely after 24 h, even before the objective function converges. On the contrary, GA has reached fitness function convergence after a few iterations and in a few minutes. Also, C_{max} obtained from GA does not deviate from the (near) optimal value obtained after a few iterations, as illustrated in Fig. 7. Also, the Gantt chart has been depicted in Fig. 8. It is observed that GA converges to the optimal or near-optimal value faster than the IP solver. Thus, the GA approach to the solution for the PFSS problem is more accurate and reduces computational cost.

Table 2. Computational results for $F_m | prmu | C_{max}$.

Instance	Problem Size mxn	MILP C_{max}	NEH C_{max}	SLOPE C_{max}	CDS C_{max}	GA C_{max}	Gap %
TAI1	5×20	1278	1299	1384	1443	1290	0.939
TAI2	5×20	1359	1365	1439	1469	1365	0.442
TAI3	5×20	1081	1132	1162	1355	1098	1.573
TAI4	5×20	1293	1329	1420	1514	1301	0.619
TAI5	5×20	1235	1305	1360	1367	1250	1.215
TAI11	10×20	1582*	1681	1790	1952	1646	4.046
TAI21	20×20	2297*	2443	2818	2823	2362	2.830
TAI31	5×50	2724*	2729	2774	2909	2729	0.184
TAI41	10×50	3025*	3164	3453	3829	3164	4.595
TAI51	20×50	3875*	4006	4272	4944	4004	3.329
TAI61	5×100	5493*	5514	5749	6076	5495	0.036
TAI71	10×100	5770*	5897	6161	6799	5846	1.317
TAI81	20×100	6286*	6584	7075	7601	6573	4.566

* IP solver was terminated prematurely after 24 h of computational time before the convergence of C_{max} .

Table 3. Computational results for $F_m | r_j, prmu | C_{max}$.

Instance	Problem Size mxn	MILP C_{max}	NEH C_{max}	SLOPE C_{max}	CDS C_{max}	GA C_{max}	Gap %
TAI1	5×20	1457	1658	2109	2351	1599	9.746
TAI2	5×20	1593	1720	2258	2288	1716	7.721
AI3	5×20	1420	1496	1994	2174	1465	3.169
TAI4	5×20	1622	1777	2183	2277	1723	6.227
TAI5	5×20	1501	1559	2120	2275	1548	3.131
TAI11	10×20	1916*	2078	2502	2804	2023	5.585
TAI21	20×20	2744*	2698	3446	3670	2659	– 3.098 ⁺
TAI31	5×50	4729*	3586	4786	5103	3492	– 26.158 ⁺
TAI41	10×50	5321*	3927	5652	6125	3895	– 26.799 ⁺
TAI51	20×50	5800*	4749	6497	7354	4741	– 18.259 ⁺
TAI61	5×100	9715*	7292	10463	10871	7155	– 26.351 ⁺
TAI71	10×100	10418*	7579	10917	11658	7459	– 28.403 ⁺
TAI81	20×100	11914*	8279	12233	12280	8240	– 30.838 ⁺

* IP solver was terminated prematurely after 24 h of computational time before the convergence of C_{max} .

⁺MILP C_{max} is not optimally computed, leading to a negative gap due to premature termination of the IP solver

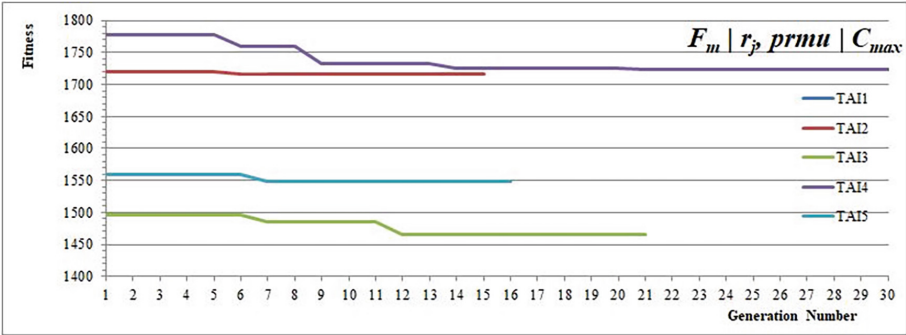


Fig. 7. Progress curves of GA experiments for $F_m | r_j, pmu | C_{max}$.

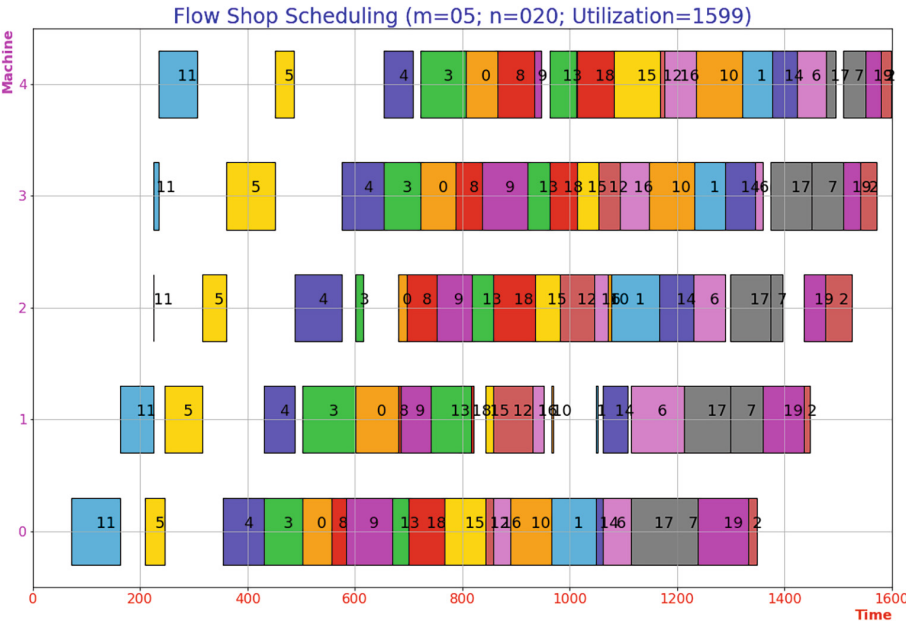


Fig. 8. Gantt chart with $C_{max} = 1599$ for $F_m | r_j, pmu | C_{max}$ obtained by GA and corresponding job sequence is 11, 5, 4, 3, 0, 8, 9, 13, 18, 15, 12, 16, 10, 1, 14, 6, 17, 7, 19, and 2, with job index starting at 0.

5 Conclusions

A mathematical formulation using MILP for PFSS problem subjected to release date constraints, i.e., $F_m | r_j, pmu | C_{max}$, was presented in the current article. As the problem is an NP-hard optimization problem, computing a feasible or optimal objective value is unlikely within a given time. Subsequently, the GA approach has been designed to solve the MILP formulation. The computational steps involved in the GA approach

are elaborated: choosing a solution representation, approach to initial population generation, determining fitness values, strategy to select parent population, performing crossover operations and mutation operations, strategy to select survivor population, and determining termination condition.

The Taillard data sets are inputs during simulation experiments with and without release dates r_j . Google OR Tools are employed to solve the mathematical formulation. Likewise, the GA approach determines the fitness value and is compared with C_{max} . The comparison is done for the scenarios $F_m | prmu | C_{max}$ to validate both the approaches and $F_m | r_j, prmu | C_{max}$ for the actual scenario. The GA approach results in both scenarios agree well with the IP solver. Moreover, the GA approach outruns the IP solver, as indicated by the performance measured in CPU time for execution.

The proposed GA approach is significantly superior to a typical IP solver in terms of computational cost (time), especially when many jobs and machines are involved, i.e., big data contextual scenarios often found in industries.

References

1. Campbell, H.G., Dudek, R.A., Smith, M.L.: A heuristic algorithm for the n job, m machine sequencing problem. *Manag. Sci.* **16**(10), B-630–B-637 (1970). <https://doi.org/10.1287/mnsc.16.10.B630>
2. Chen, C.L., Vempati, V.S., Aljaber, N.: An application of genetic algorithms for flow shop problems. *Eur. J. Oper. Res.* **80**(2), 389–396 (1995). [https://doi.org/10.1016/0377-2217\(93\)E0228-P](https://doi.org/10.1016/0377-2217(93)E0228-P)
3. Della Croce, F., Grosso, A., Salassa, F.: A matheuristic approach for the two-machine total completion time flow shop problem. *Ann. Oper. Res.* **213**(1), 67–78 (2011). <https://doi.org/10.1007/s10479-011-0928-x>
4. Eiben, A., Smith, J.: *Introduction to Evolutionary Computing*. Natural Computing Series, Springer Berlin Heidelberg, Berlin, Heidelberg (2015). <https://doi.org/10.1007/978-3-662-44874-8>
5. Etiler, O., Toklu, B., Atak, M., Wilson, J.: A genetic algorithm for flow shop scheduling problems. *J. Oper. Res. Soc.* **55**, 830–835 (2004). <https://doi.org/10.1057/palgrave.jors.2601766>
6. Google: Or tools (2023). <https://developers.google.com/optimization/>. Accessed 19 Sept 2023
7. Hasani, K., Kravchenko, S.A., Werner, F.: Simulated annealing and genetic algorithms for the two-machine scheduling problem with a single server. *Int. J. Prod. Res.* **52**(13), 3778–3792 (2014). <https://doi.org/10.1080/00207543.2013.874607>
8. Karadgi, S., Hiremath, P.S.: Job scheduling on parallel machines with precedence constraints using mathematical formulation and genetic algorithm. In: Sharma, S., Subudhi, B., Sahu, U.K. (eds.) *Intelligent Control, Robotics, and Industrial Automation*, pp. 835–847. Springer Nature Singapore, Singapore (2023). https://doi.org/10.1007/978-981-99-4634-1_65
9. Khatami, M., Salehipour, A., Hwang, F.: Makespan minimization for the m machine ordered flow shop scheduling problem. *Comput. Oper. Res.* **111**, 400–414 (2019). <https://doi.org/10.1016/j.cor.2019.06.012>
10. Kim, H., Jeong, H.I., Park, J.: Integrated model for production planning and scheduling in a supply chain using benchmarked genetic algorithm. *Int. J. Adv. Manuf. Technol.* **39**, 1207–1226 (2008). <https://doi.org/10.1007/s00170-007-1298-z>
11. Leyman, P., Vanhoucke, M.: A new scheduling technique for the resource-constrained project scheduling problem with discounted cash flows. *Int. J. Prod. Res.* **53**(9), 2771–2786 (2015). <https://doi.org/10.1080/00207543.2014.980463>

12. Li, X., Yin, M.: A discrete artificial bee colony algorithm with composite mutation strategies for permutation flow shop scheduling problem. *Scientia Iranica* **19**(6), 1921–1935 (2012). <https://doi.org/10.1016/j.scient.2012.10.034>
13. Li, X., Yin, M.: A hybrid cuckoo search via lévy flights for the permutation flow shop scheduling problem. *Int. J. Prod. Res.* **51**(16), 4732–4754 (2013). <https://doi.org/10.1080/00207543.2013.767988>
14. Lian, Z., Gu, X., Jiao, B.: A novel particle swarm optimization algorithm for permutation flow-shop scheduling to minimize makespan. *Chaos Solitons Fractals* **35**(5), 851–861 (2008). <https://doi.org/10.1016/j.chaos.2006.05.082>
15. Liang, J., Pan, Q.K., Tiejun, C., Wang, L.: Solving the blocking flow shop scheduling problem by a dynamic multi-swarm particle swarm optimizer. *Int. J. Adv. Manuf. Technol.* **55**, 755–762 (2011). <https://doi.org/10.1007/s00170-010-3111-7>
16. Liang, Z., Zhong, P., Liu, M., Zhang, C., Zhang, Z.: A computational efficient optimization of flow shop scheduling problems. *Sci. Rep.* **12**(1) (2022). <https://doi.org/10.1038/s41598-022-04887-8>
17. Liu, C.: A hybrid genetic algorithm to minimize total tardiness for unrelated parallel machine scheduling with precedence constraints. *Math. Probl. Eng.* **2013**, 1–11 (2013). <https://doi.org/10.1155/2013/537127>
18. Maciel, I., Prata, B., Nagano, M., Abreu, L.: A hybrid genetic algorithm for the hybrid flow shop scheduling problem with machine blocking and sequence dependent setup times. *J. Project Manag.* **7**(4), 201–216 (2022). <https://doi.org/10.5267/j.jpm.2022.5.002>
19. Modrák, V., Pandian, R.S.: Flow shop scheduling algorithm to minimize completion time for n-jobs m-machines problem. *Tehnički vjesnik* **17**(3), 273–278 (2010)
20. Murata, T., Ishibuchi, H., Tanaka, H.: Genetic algorithms for flowshop scheduling problems. *Comput. Ind. Eng.* **30**(4), 1061–1071 (1996). [https://doi.org/10.1016/0360-8352\(96\)00053-8](https://doi.org/10.1016/0360-8352(96)00053-8)
21. Nawaz, M., Ensore, E.E., Ham, I.: A heuristic algorithm for the m machine, n-job flow-shop sequencing problem. *Omega* **11**(1), 91–95 (1983). [https://doi.org/10.1016/0305-0483\(83\)90088-9](https://doi.org/10.1016/0305-0483(83)90088-9)
22. Pinedo, M.: *Scheduling Theory, Algorithms, and Systems*. Springer (2016)
23. Ruiz, R., Vázquez-Rodríguez, J.A.: The hybrid flow shop scheduling problem. *Eur. J. Oper. Res.* **205**(1), 1–18 (2010). <https://doi.org/10.1016/j.ejor.2009.09.024>
24. Sayadi, M., Ramezani, R., Ghaffari-Nasab, N.: A discrete firefly meta-heuristic with local search for makespan minimization in permutation flow shop scheduling problems. *Int. J. Ind. Eng. Comput.* **1**(1), 1 (2010). <https://doi.org/10.5267/j.ijiec.2010.01.001>
25. Taillard, E.: Benchmarks for basic scheduling problems. *Euro. J. Oper. Res.* **64**(2), 278–285 (1993). [https://doi.org/10.1016/03772217\(93\)90182-M](https://doi.org/10.1016/03772217(93)90182-M), project Management and Scheduling
26. Taillard, E.: Scheduling instances (2023). <http://mistic.heigvd.ch/taillard/problemes.dir/ordonnancement.dir/ordonnancement.html>. Accessed 19 Sept 2023
27. Umam, M.S., Mustafid, M., Suryono, S.: A hybrid genetic algorithm and tabu search for minimizing makespan in flow shop scheduling problem. *J. King Saud Univ. Comput. Inf. Sci.* **34**(9), 7459–7467 (2022). <https://doi.org/10.1016/j.jksuci.2021.08.025>
28. Wei, H., Li, S., Jiang, H., Hu, J., Hu, J.: Hybrid genetic simulated annealing algorithm for improved flow shop scheduling with makespan criterion. *Appl. Sci.* **8**(12) (2018). <https://doi.org/10.3390/app8122621>
29. Zaied, A.N.H., Ismail, M.M., Mohamed, S.S.: Permutation flow shop scheduling problem with makespan criterion: literature review. *J. Theor. Appl. Inf. Technol.* **99**(4), 830–848 (2021)
30. Zobel, G., Tarantilis, C., Ioannou, G.: Minimizing makespan in permutation flow shop scheduling problems using a hybrid metaheuristic algorithm. *Comput. Oper. Res. Oper. Res.* **36**(4), 1249–1267 (2009). <https://doi.org/10.1016/j.cor.2008.01.007>



Estimate Initial and Final Data Bounding for Big Data Progressive Sampling in Parallel Computing Environment

B. C. Yathish Aradhya^{1(✉)}, H. A. Dinesh², and M. R. Lokesh³

¹ Visvesvaraya Technological University-Research Resource Center, Jnana Sangama, VTU Main Rd, Visvesvaraya Technological University, Machhe, Belagavi, Karnataka 590018, India
docs.kit@gmail.com

² Shridevi Institute of Engineering and Technology (SIET), Tumakuru, Karnataka is affiliated to Visvesvaraya Technological University, Belagavi, Karnataka 572106, India

³ Department of Information Science and Engineering, A J Institute of Engineering and Technology, Mangalore, India

Abstract. Big Data Progressive Sampling requires initial and final data bounding values to generate optimal number of samples in order to train any learning algorithm. Any learning algorithm can be trained for minimal hypothesis space. Approximating data bounding to Rademacher Averages of a given hypothesis space and learning algorithm ensure final data bounds with ϵ -Approximation. This final Data bounding value serves as stopping criterion for Progressive Sampling. Statistically optimal sample size estimation technique ensure minimal initial sample size. In this work, we propose a methodology to approximate initial and final Data bounding to Progressive Sampling using local Rademacher in big data Hadoop environment with experimental values along with computational parameters. Also the performance of learning algorithm to various highly complex big datasets for their corresponding initial minimal value and final Data bound to Rademacher Averages are experimentally shown in Hadoop Distributed environment.

1 Introduction

Learning a large set of hypothesis to classify a given dataset requires a right approximation of samples. Approximating quantity of samples needed for learning is a tedious task. If the size of the data highly voluminous and complex in nature it further becomes a statistically dependent problem and also NP- hard. To learn from such a big data, it may incur huge cost and may have to trade off with computational ways and parameters. Progressive Sampling is a mechanism to generate samples from highly complex datasets such as big data. Since big data is vast and voluminous, it is very tedious to make multiple pass through databases and extract samples having qualities to represent whole dataset. Statistical learning theory offers wide range of ways estimate and generate size of datasets adequate enough to represent the entire dataset in single scan pass of the database.

Progressive Sampling is an iterative algorithm which runs using initial minimal number of samples and final cardinality of sample set. PSA is designed to operate with minimal sample size and to increment the size of sample set with fractional geometric ratio and checked for accuracy of classifier whether or not, its classification is within Probably Acceptably correct (PAC). The accuracy of classifier depends on the training data set used. Samples generated using PSA decides the result of classification. However, with the number of samples used for training and learning algorithm selected play overall accuracy and computational efficiency. There are a good numbers of approaches proved in past time regarding sample generation.

The Estimating an Optimal initial size and final bound to sampling process avoids under sampling and over sampling in training phase. Thus lesser the difference between Initial sample size and final bound more optimal will be the cardinality of training sets and also less number of iteration in PSA. To estimate initial Sample Size Statistically optimal sample Size (SOISSize) is reasonably better and to approximate Local Rademacher Averages is proven to be good in all scenarios than when compared to all recent advances. Hence Local Rademacher Averages which happens to be a statistical tool, is used to derive bounds for learning problem.

The Existing work carried out based on statistical learning theory, there was no such combination of statistical divergence and Local Rademacher averages used efficiently to approximate bounds for big datasets in Hadoop Environment. Thus in this work we propose a methodology which incorporates statistical divergence and Rademacher averages in a massively parallel computing environment in order to derive initial and final bound to generate samples for progressive sampling approach. To the best of knowledge, the approach work guarantees to achieve better accuracy of classification while also retaining the performance of classification to be within PAC framework than to all concerned exiting works. Since this work confines to determine efficient samples over all computational time for convergence of classifier reduces as the PSA would possibly less computational time.

The present work on progressive sampling technique will have an overall runtime that is proportional to the number of iterations it performs, which includes time spent increasing the sample size at each iteration and time spent training the classifier until the stopping requirement is satisfied [9, 10]. In this case, the classifier's convergence to ensure that it is within the PAC framework is referred to as the stopping criterion. Big-Oh (Sample size + Sample Schedule computation time + Training Classifier and Testing run times) can be used to express the overall computing time for progressive sampling.

In the Section one defines the motivation of PS in Big data, initial sample size sampling problems in Progressive Sampling in big data, problem formulation, and outcome of PS in Big data. Section two studies the background. Section three covers preliminaries of statistical divergence and Rademacher averages. Section Four provides the proposed method statistically optimal sample size (SOISSIZE). Section FIVE provides the proposed method Final Bound estimation. Section six provides validation of the dataset with experimental setup and compares results obtained with the existing state. Section seven gives comparative studies.

2 Background

In this section, the essentials of sampling in big data and data mining with the need for progressive sampling, optimized Convergence and divergence of learning algorithms for optimal training datasets, apply of discussed issues in big data parallel computing environment-related issues are discussed as follows:

The sampling of large data has proved versatile in solving many data mining tasks in addition to its primary usage in training classifiers. For example, in the paper [10] proved that sample set cardinality derived using Chernoff bounds can reduce the time consumed to mine association rules. However, such size bounds were found to be too conservative in practice. In the paper [8, 12] provided a VC- dimension-based sample bound on Big Data sets for the randomized algorithm PARMA which extracts frequent item sets and associations. In the paper [12] provided Rademacher Averages-based sample bound on Big Data sets for the randomized algorithm PARMA which extracts frequent item sets and association rules.

In the Sampling data Methodologies on progressive sampling discussed by Bradley et al. proposed and proved that existing clustering algorithms can be scaled up to large databases by a repeated updating of the current model with randomly drawn samples [13]. Sampling is also widely adopted in database retrieval methodologies. For instance, according to the self-tuning samples presented by Ganti et al., the probability that a tuple will be chosen is proportional to how frequently it must respond to inquiries. (exactly) [14]. However, in both [12] and [13], It is not indicated how to select the appropriate sample size. Although Ganti et al. also offered a metric to quantify the difference between two data sets in terms of the models constructed by a certain data mining technique [11], which might address the issue of generating models using random samples, it did not address how to choose the appropriate sample size.

The Sampling of data size were discussed by Baohua Gu and Bing Liu [8] proposed a similar work to that of Ganti et al. [11, 13, 15] in providing a metric to put a number on the variation between two data sets. However, the work discussed in [8] is based on statistical information divergence of data sets mostly from the UCI repository, and Sample sizes derived were independent of any data mining algorithm unlike in [3]. Alfonso Estrada and Eduardo F. Morales [27] proposed a Novel Progressive sampling technique aiming to reduce the overall run time to estimate the samples required for the convergence of any learning algorithm. However, the selection of the initial sample size by the work discussed in [28] is dependent on both types of data sets in use and the algorithm was chosen to act upon that data. Both [28] and [8] have not addressed the problem of huge data sets such as big data in Distributed Computational environments. For the concept of the most number of samples required for the convergence of learning algorithms use of statistically derived bounds by Matteo Riondato outdates all previous works [12]. However [12] [8, 13, 15] does not focus on improvising contemporary strategies followed in progressive sampling design. The fact that the initial sample size chosen crucially determines the overall run time of any progressive sampling algorithm.

In the Rademacher Approximation is statistical tool to derive bounds on datasets required for the convergence of learning problem [18–21]. To determine the sample numbers required to obtain the necessary accuracy for adaptive learning, localized Rademacher complexities are made use of. Such probabilistic bounds have been proved

on the number of active examples and several applications to binary classification problems are considered [18]. Function classes can be created as combinations of functions from basis classes, and they can be shown to have Rademacher and Gaussian complexities that are constrained by the complexity of the underlying basis classes. [19, 22]. Biomedical data sets were efficiently used to estimate total sampling size using progressive sampling without deal with its initial sample size [23].

The research work proposed in [7, 12] demonstrate the use statistical bound such as VC-Dimension and Rademacher Averages to derive ϵ -Approximation of given datasets and thus contributes to generate training samples in a one-shot sampling scheme. However, this may lead under sampling or oversampling depending on the type of dataset and classifiers used in learning. Even though [12] achieve parallel processing and one-time database scanning, it has not considered overall sampling time and heterogeneity scenarios of datasets and learning algorithm. On the other hand, work proposed in [8] proposes methodologies to compute minimal initial sample size for simple data sets which can fit into primary memory. In the phase of improvisation of iterative nature of progressive sampling scheme for big data, work contribution from all previous works aims little towards optimization of computation. In this work, an enhanced methodology named Statistically Optimal Initial Sample size (SOISSize) for deriving to derive the initial minimal sample set size is provided. This work can be applied to carry out Progressive sampling on Big Data with size beyond the capacity of main memory, in a distributed computational environment such as HDFS. Since all computations in the proposed work are optimized in a parallel computation using Hadoop, the overall performance comparatively out rates all concerned previous works.

3 Progressive Sampling of Big Data

In this section architecture of computation and data handling in a big-data Hadoop environment is modeled and discussed. The multi-Vectored and highly dimensional big data set can be loaded to Secondary memory and preprocessed to apply Map Reduce function to generate key-value pairs. In the following Sect. 3 Progressive Sampling of Big Data. The Fig. 1 shown below clearly depicts the architecture of big data in a distributed file system architecture.

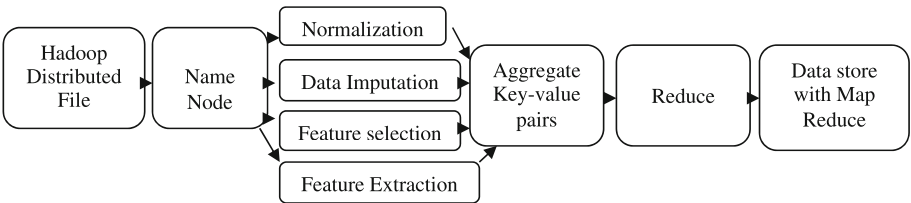


Fig. 1. Architecture big data sets in the Hadoop environment

In the Fig. 1 give conceptual modelling of usage of big data in parallel computation environment. Data sets considered for the processing and analytics will be loaded into

the HDFS system which names node. Entire data sets which require Giga bytes of space in memory will be loaded to n data nodes while all metadata required for controlling the data node will be kept created in the data node. HDFS analyses the data pattern and reports errors in the consistency of data. All reported error data will be resolved with basic preprocessing techniques [38]. Preprocessing techniques such as normalization, filling missing values, feature verification and selection, etc. are incorporated here. Post to preprocessing Map Reduce framework is applied to reduce the data to key-value pairs. Further preprocessed data will be subjected to parallel computation to estimate the initial minimal sample size. Pre-Processing stages applied to Biomedical and UCI big data incorporate the following standard procedures. Data chosen for the estimation of the initial minimal sample size and sample set is an acquisition from various data sources.

3.1 Normalization

This pre-processing stage ensures that each row in the data set has a unit norm. Any row in the data set chosen, if found to mismatch in the expression of the unit for any quantity, such rows are normalized to maintain integrity.

3.2 Data Imputation

Data imputation is a technique where the missing value row will be retained by substituting it with a suitable value. The substitution of value depends on the probable and classified reason for missing data. For the biomedical dataset chosen for estimating the initial sample size, the most occurring value is replaced.

3.3 Feature Selection and Extraction

The selection of a subset of features concerned with the setting of the experiment refers here to as feature selection. Most prominent features are just sufficient to make analytics hints feature selection, eliminating the most redundant ones. Extraction is a technique that can be applied to a subset of features to estimate a new feature in the case of biomedical data set incorporated for progressive sampling, any additional features related to visibility or structure are introduced.

3.4 Hadoop Distributed File System

Hadoop Distributed File system implements a multi-node cluster mechanism to store data in a data nodes system called racks. Each data node has processing and storage capabilities. All data nodes are configured and controlled by a special node called the Name node. All data nodes under a single name node are enabled to run distributed applications including mining and analytics. In this context after preprocessing of big data sets, generated key-value pairs will be used to estimate and determine a minimal initial sample set for a progressive sampling of big datasets using Rademacher Averages [39].

3.5 Map-Reduce Framework

Map-Reduce Algorithm generates key-value pairs out of the data provided to it as input. Mapping algorithms map data in different blocks and generate intermediate key-value pairs which will be provided as input to Reducer which reduces key-value pairs which will be then saved in HDFS. The Map Reduce algorithm is constructed based on two functions such as a map and reduce where the input is obtained from the key-value pairs which are denoted as (k,v) . The map function obtains a pair of inputs in a single iteration and provides a multiset of pairs i.e. $\{(k_1,v_1),(k_2,v_2),\dots,(k_n,v_n)\}$. The multiset union is denoted as U which is comprised of the multisets of map function when it is applied to every pair of inputs. The set U is partitioned into U_k , where the specified key is denoted as k - and the U_k - is comprised of the pair of values denoted as (k,v) . Similarly, the reduce function takes the input key as k - and provides multiple sets of values [38, 39, 40].The results of pre-processing of big datasets are tabulated as follows

Table 1. Analysis of preprocessing on datasets.

Data set	Features selection	Features Extracted	No missing value rows and imputed	Size in GBs,	Framework
ECBLD	985	3	35	23	HDFS
Medicare	123	3	56	25	HDFS
Melanoma	117513	0	23	103	HDFS
Splice	100000	3	789	98	HDFS
adult	14	2	6	0.2	HDFS
Census	14	2	5	0.2	HDFS
LED	7	2	7	0.1	HDFS
COVType	54	2	7	0.5	HDFS

In the Table 1 provides a complete analysis of each dataset loaded to Hadoop and the methodology applied on each data node to reduce the loaded data into key-value pairs

4 Statistical Divergence and Rademacher Averages

If a given dataset D , R is called to be Range space where $R \subset D$. Range space $R_i \in R$ will be used to locally estimate sample size and later all of them are aggregated to form a Statistical Optimal Sample size. The primary criteria to estimate the sample size locally at each data node is to consider sample quality. The metric adopted here to measure the quality of a sample is Divergence or Deviation contributed by Kullback's Information Measure [12]. Although Divergence Concept is used in their work by Baohua Gu and Bing Liu [8] to estimate it did not experiment with Big data sets in a distributed environment.

4.1 Divergence

If z is a particular value of any generic variable Z and H_i is the concept that Z is from a statistical population with generalized probability densities (under a probability measure λ) $f_i(x)$, $i = 1, 2$, then the information divergence [8] is defined by

$$J(1, 2) = \int (f_1(x) - f_2(x))(\log f_1(x) - \log f_2(x)) d\lambda(x).$$

According to [8][8] $J(1,2)$ is referred to as a measure of the divergence between hypotheses H_1 and H_2 which indicates the measure of the difficulty of discrimination between them. Given two multinomial populations, if P_{ij} is the probability of occurrence of a j th value in the i th population then the information divergence is

$$J(1, 2) = \sum_j = 1 (p_{1j} - p_{2j}) (\log p_{1j} - \log p_{2j})$$

The information divergence has a limiting property which is described in theorem 1 whose proof can be obtained from [8]. Theorem 1 serves as a fundamental basis for the estimation of Sois.

Theorem 1: If $f(x)$ is probability density function and $\{f_n(x)\}$, is a series of probability density functions where $n \rightarrow +\infty$, denote the information divergence from $f_n(x)$ to $f(x)$ as $J(f_n(x), f(x))$, then we have, if $J(f_n(x), f(x)) \rightarrow 0$, so that $f_n(x)/f(x) \rightarrow 1[\lambda]$, uniformly. Here $[\lambda]$ refers to the limitation and the fraction held in a probability measure λ . [8].

Definition 1: For any Range Space R (with n attributes) subset of a big data set i.e. $R \subset D$ being processed in distributed environment such as HDFS and its sample S , then the quality of sample $Q(S) = \exp(-J(S, R))$ where $JK(S, R)$ denotes information divergence between S and R on attribute K and it is given by $J = (1/n) \sum_{k=1}^n JK(S, R)$ [8].

To calculate $JK(S, R)$ at each data node of the Hadoop system all categorical attributes are treated as multinomial and continuous attributes such as numerical attributes are also treated as multinomial by considering respective bin classes and bin class frequency through the construction of histograms. As per [8] the information divergence $J > 0$, hence $0 < Q \leq 1$, where $Q = 1$ means that no information divergence exists between S and R . Larger the information divergence, the smaller will be the sample quality; and vice versa [8, 24].

Upon a single scan of a specific Range Space R at each data node, both categorical values and frequencies of numerical values that fall in the bins can be incrementally gathered. Since scan and quality measure to estimate minimal sample size, runs parallel, the time complexity of calculating sample quality at any data node will be $O(N_d)$ where N_d is the total number of instances at each data node. Since all data nodes in HDFS works the parallel same amount of time would require to process the entire data in HDFS and measure sample quality and hence to have an approximation of the quality of all samples in the entire data as a whole.

At each data node of Hadoop, occurrences of categorical values and Frequencies of numerical values can be incrementally collected. Hence Time complexity of calculating sample quality is $O(N_d)$ (N_d is the total number of instances at any one data node of HDFS) [25, 26]. Since computation runs massively parallel time taken by one data node to compute the sample quality will be the time for the entire big data set. However, data processing speed depends on the nature of the data under processing. Space complexity

issues are trivial in impact as HDFS manages the entire data set on a secondary disk only. One scan of data at each node is sufficient enough to make processing and analytics.

4.2 Rademacher Averages

Within a PAC framework and statistical learning setting, it can be assumed that a learning algorithm chooses its hypothesis from some fixed hypothesis class H . Generalization of error analysis provides theoretical results bounding generalization of error of hypotheses $h \in H$ which will be based on the sample and properties of hypothesis class. For any given hypothesis h , its generalization error is the probability that a randomly drawn example is misclassified;

$$\varepsilon_p(h) = P(h(x) \neq y)$$

Although goal of any learning algorithm is to find a hypothesis with a small generalization error, it cannot be computed on sample per se, as it would also depend on probability distribution P . However, an attempt can be made to approximate generalization error of hypothesis h , by its training error on n examples: where L is 0/1 loss function

$$\varepsilon_n(h) = 1/n \sum_{i=1}^n L(h(x_i), y_i)$$

$$L(z, z') = \begin{cases} 1, & \text{if } z \neq z' \\ 0, & \text{otherwise} \end{cases}$$

The Empirical Risk minimization (ERM) is a principle according to which only such hypothesis whose training error is minimal is to be chosen. To ensure that ERM yields hypothesis with small generalization error the bound $\sup_{h \in H} |\varepsilon_p(h) - \varepsilon_n(h)|$ can be used. The difference of training error of hypothesis h on n examples and true generalization error converge to 0 in probability as n tends to infinity, provided examples are independent and identical in distribution while the hypothesis class H is not complex. A Rademacher Random Variable (Koltchinskii 2001) takes values $+1$ and -1 with probability $1/2$ each. Let Rademacher random variables $r_1, r_2, r_3, \dots, r_m$ be i.i.d. Rademacher variables independent of the data $(x_1, y_1), \dots, (x_n, y_n)$. The Rademacher Penalty of hypothesis class H is defined as follows

$$R_n(H) = \sup_{h \in H} \left| \frac{1}{n} \sum_{i=1}^n r_i L(h(x_i), y_i) \right|$$

By symmetrisation inequality of the theory of empirical processes

$$E\{\sup_{h \in H} |\varepsilon_p(h) - \varepsilon_n(h)|\} \leq 2E\{R_n(H)\} \quad (1)$$

where expectations are taken over the choices of examples on the left and over the choice of examples and Rademacher random variables on the right hand side. Using standard concentration of inequalities that with probability at least $1-\delta$

$$\varepsilon_p(h) \leq \varepsilon_n(h) + 2 R_n(H) + \varepsilon(\delta, n) \quad (2)$$

where $\eta(\delta, n) = 5\sqrt{(\ln(2/\delta)/2n)}$ is a small error term that takes care of fluctuations of analyzed random variable around their expectations. The usefulness of the inequality (2) is raised from the fact that its right hand side depends only on the training sample and not on p directly. This means that $R_n(H)$ can be computed with algorithm used for ERM.

Rademacher Penalties can be applied to provide approximate solution to progressive sampling algorithm, particularly in estimation of stopping time in terms data dependent upper bound ($\mathcal{E}$ -Approximation) of sample set used for training the learning algorithm (Koltchinskii et al. 2000). The minimal number of samples required to guarantee ERM is within a distance of $\mathcal{E}$ from generalization error of h for every $h \in H$:

$$\text{Argmin}\{\Pr\{\sup_{h \in H} |p(h) - \mathbb{E}n(h)| \geq \varepsilon\} \leq \delta\} \quad (3)$$

The rademacher stopping time $v(\mathcal{E}, \delta)$ with parameters $(\mathcal{E}, \delta)$ for hypothesis of class H is

$$v(\varepsilon, \delta) = \min\{n \mid 2n \mathbb{E}n(\varepsilon, \delta) | R_n(H) < \varepsilon\} \quad (4)$$

Koltchinskii et al. (2000) derived data dependent results that hold for any distribution that could have produced a sample S .

4.3 Algorithm for Starting Sample Size Estimation (SOISsize) and Approximating Final Bound for PSA

Determining initial sample size for iterative progressive sampling algorithm critically plays a role in overall convergence time and input data sufficient enough to be within PAC framework.

a. Program Code Starting sample size Estimation (SOISsize).

```

Input: a large Dataset (D) of Size N {  $S_i \mid i=1, 2, 3, \dots, n$  }
Output: Statistically Optimal Initial Sample Size (SOISsize)
Step1: Load  $\forall r_i \in R$  where  $R = \{r_1, r_2, \dots, r_n\}$  and
        $R \subset D$  to specific node of HDFS
Step2: Estimate the Probability Distribution Function of
       each  $r_i \in R$  at all nodes
       And select some samples from each bin Compute  $N_{min}$ 
       for each Range with a Predefined confidence level
Step 3: Compute Sample Quality for each sample  $Q(S_i)$ 
Step 4: Plot the intermediate regression curve and
       Estimate its find mid-point called SOISsize
Step 5: Find the Mean of individual data nodes SOISsize
Step 6: Find the Mean of  $N_{min}$  at each specific Node. And
       finally, find the average of SOISsize and  $N_{min}$  to
       yield the most Expected optimal SOISS
Step 7: Apply Random sampling at each node Select a
       sample set whose cardinality is SOISsize,
Step 8: Random sample the sample set resulted in Step 7
       again to gather accurate SOSS

```


b. Program Code approximating final bound for PSA.

Input: a large Dataset (D) of Size N $\{S_i | i=1, 2, 3, \dots, n\}$
Output: Final Bound for PSA
Step1: Load $\forall r_i \in R$ where $R = \{r_1, r_2, \dots, r_n\}$ and
 $R \subset D$ to specific node of HDFS
Step2: Estimate the Probability Distribution Function of
each $r_i \in R$ at all nodes
And select some samples from each bin
Step 3: Apply Rademacher Averages equation

$$R_n(H) = \sup_{h \in H} \left| \frac{1}{n} \sum_{i=1}^n r_i L(h(x_i), y_i) \right|$$
 at Data
node.
Step 4: Find the Mean of Final bound individual data node
Step 6: Random selection of data sample at each node to a
cardinality equal to that of final bound
Step 7: Random sample the sample set resulted in
Step 6 again to gather good approximation of final bound
samples

5 Results and Discussion

Experiment setting for the objective of this work considers I.I.D (Identical and independent distribution data) and structured big data whose memory requirement are beyond primary and very high dimensional. A brief about the datasets used and pre-processing of the same are tabulated in the tables

5.1 Experimental Set up

The proposed algorithm for computing final bounds and initial bounds for Progressive sampling is implemented using python programming language. A python implementation PSA available at used <https://github.com/yuexy/PS-ViT.git> For testing behaviour of classifier implementation available at <https://github.com/topics/naive-bayes-classifier> is used and accuracy computed. The evaluation is carried out in a system with an Intel i5 2.7 GHz processor, 6GB Random Access Memory (RAM), and windows 10 operating system. All computations were carried upon UCI data sets. Determining initial.

5.2 Result Analysis

The result shown below the number of instances and type of dataset. Datasets listed in above table are chosen from biomedical domain because of volume and veracity. The proposed algorithm for computing final bounds and initial bounds for Progressive sampling is implemented

Table 2. Font sizes of headings. Table captions should always be positioned *above* the tables.

Domain	ECBDL	Insurance Claims	Clinical records	Genomics
Big data datasets	Proteomics	Medicare	Melanoma	Splice
Native instances	7998231	3692555	9531408	4627840
Positive Instances	171,933	1409	17246	14549
Class ratio	2.15%	0.03815%	0.181%	0.341%
Features	985	123	117513	1000000

The results shows in the Table 2, the number of instances and type of dataset. Datasets listed in above table are chosen from UCI website for machine learning repository because of velocity and veracity.

Table 3. Font sizes of headings. Table captions should always be positioned *above* the tables.

Domain	ECBDL	Insurance Claims	Clinical records	Genomics
Big data datasets	Adults	Led	census	covtype
Native instances	48000	100K	199.5k	581012
Positive Instances	36000	NA	48842	360000
Class ratio	51.23%	45.76%	2.84%	5.98%
Features	14	7	14	54

In the Table 3 provides a complete analysis of each dataset loaded to Hadoop and the methodology applied on each data node to reduce the loaded data into key-value pairs

Table 4. Font sizes of headings. Table captions should always be positioned *above* the tables.

Datasets	Total Data	Initial Bound(Sois)	Final Bound
ECBDL	7998231	1279716	5118864
Medicare	3692555	590809	1772427
Melanoma	9531408	1,525,025	6100100
Splice	4627840	740454	2221362
adult	48000	8000	16000
led	100K	10000	20000
census	199.5k	35000	70000
covtype	581012	70000	140000

Table 5. Font sizes of headings. Table captions should always be positioned *above* the tables.

Data set	Features selection	Features Extracted	No missing value rows and imputed	Size in GBs,	Framework
ECBLD	985	3	35	23	HDFS
Medicare	123	3	56	25	HDFS
Melanoma	117513	0	23	103	HDFS
Splice	100000	3	789	98	HDFS
adult	14	2	6	0.2	HDFS
Census	14	2	5	0.2	HDFS
LED	7	2	7	0.1	HDFS
COVType	54	2	7	0.5	HDFS

The results shown in the Table 4 and Table 5 the values of initial bound and final bound for PSA big data set on which progressive sampling will be performed (Table 6)

Table 6. Accuracy of classification comparison with exiting methodology

P-ID	Type of Big data	methodology	Rademacher Bound records	Avg Tpsa	Accuracy of classification
[35]	Volume and veracity	Monte-Carlo Empirical Rademacher	census	covtype	91.3%
[29]	Volume and veracity	Local Rademacher Averages	Not Discussed	Not Discussed	95.4%
[2] [3]	Volume and veracity	Global RC bounds	Not Discussed	Not Discussed	89.2%
[7]	Volume and veracity	Rademacher Averages	Not Discussed	Not Discussed	91.8%
[33]	Volume and veracity	Global RC and XRC for binary classification	Not Discussed	Not Discussed	91.8%
[34]	Volume and veracity	Rademacher Averages	Not Discussed	12.3	91.8%
Proposed work	Velocity, Volume and veracity	Local Rademacher Averages in parallel computation	Not Discussed	8.5	95.3%

Figure 2 Plot of results obtained are shows the plot and comparison of total instances in the dataset and their bounds

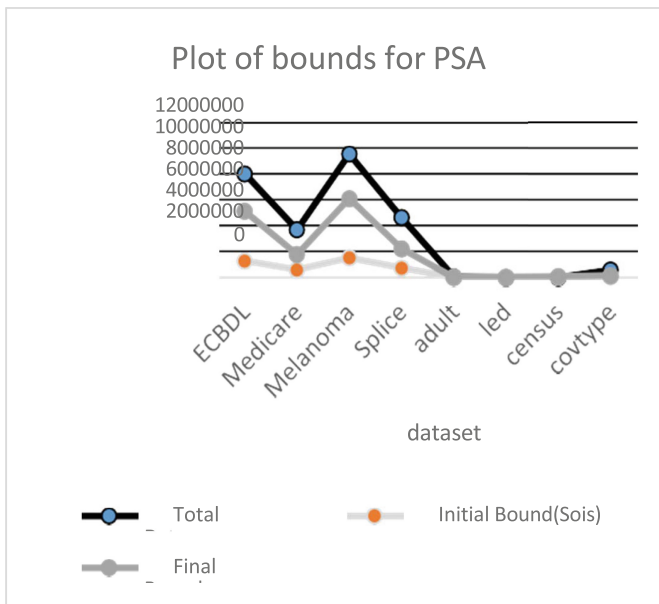


Fig. 2. Total instances in the dataset and their bounds

The Fig. 3 results obtained are shows the plot of accuracy achieved by training of Naïve Bayes classifier for both biomedical and UCI datasets. Using Local Rademacher Averages to derive bound for PSA a classification accuracy of 95.3 can be achieved

6 Comparative Study

The comparative Table 5 shown below are the Mote-Carlo Empirical Rademacher Averages adopted to derive new bounds on UCI data sets to approximate pattern mining and achieved an accuracy 91.3. Matteo Riondato and Eli Upfal [29] have presented an estimation of ϵ -Approximation of frequent item set via progressive sampling approach with Rademacher averages with an assumed random initial sample set size [29] has used local Rademacher averages to derive bounds and achieved an accuracy 95.4% in classification [2, 3] adopted Global RC Bounds to achived an accuracy of 91.8 [7] uses Rademacher Averages and arrived at an accuracy of 91.8%. Proposed work makes use of Rademacher Averages to find final bound in HDFS environment and achived an accuracy of 95.3% with reduced computation time for convergence of classifier.

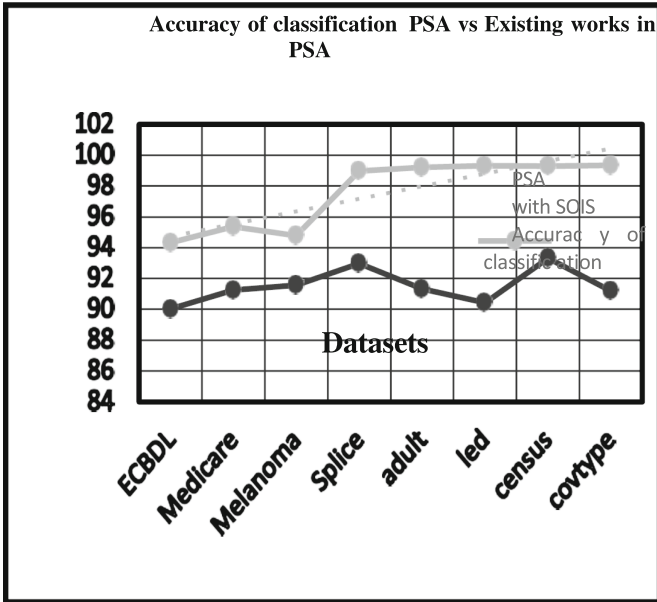


Fig. 3. Accuracy achieved by training of Naïve Bayes classifier for both biomedical and UCI datasets.

7 Conclusion

In this research work proposes approximated bounds to local Rademacher averages of a function learning algorithm to ensure minimal training and reasonably accurate classifying rate. Recent advances in Rademacher Averages were evaluated to ensure that Local Rademacher Averages provide best approximation to all scenarios. Leveraging, potential of massive Parallel Computation technology like Hadoop to approximate Rademacher Averages for complex big data sets is achieved which results substantial improvements in overall time and space complexity of Progressive sampling algorithm. The proposed methodology provides best approximation of Rademacher bounds to ensuring that PSA will be reasonably efficient. The Methodology SOISsize (statistically optimal sample size) adopted here to estimate Initial sample size also ensures to maintain possibly less difference between final and initial value for PSA. Thus, this works significantly contributes to reducing training time and cost of learning algorithm while also keeping the result of a classifier to be in the probably approximately correct framework as the classification results is within PAC (Probably Approximately Correct Framework)

References

1. Zhao, J., et al.: A novel clustering-based sampling approach for a minimum sample set in the big data environment. *Int. J. Pattern Recogn. Artif. Intell.* **32**(02), 1850003 (2018)
2. Christensen, P.: Pixar Progressive Sampling Strategies for Disk Light Sources (2018)

3. Provost, F., Jensen, D., Oates, T.: Progressive sampling. In: The Springer International Series in Engineering and Computer Science, vol. 608. Springer, Boston, MA (2001)
4. BC, Y.A., Gowramma, Y.P.: Progressive sampling algorithm with rademacher averages for optimized learning of big data: a novel approach. *Int. J. Comput. Appl.* **175**(15), 37–40 (2020)
5. BC, Y.A., Gowramma, Y.P.: Rademacher Averages bounded progressive sampling algorithm for big data analytics: a novel approach. *Int. J. Sci. Eng. Res.* **11**(7), 1374 (2020). ISSN 2229–5518
6. Elomaa, T., Kääriäinen, M.: Progressive rademacher sampling. In: AAAI/IAAI (2002)
7. Gu, B., Liu, B., Hu, F., Liu, H.: Efficiently determining the starting sample size for progressive sampling. In: De Raedt, L., Flach, P. (eds.) *Machine Learning: ECML 2001*. ECML 2001. Lecture Notes in Computer Science(), vol. 2167, pp. 192–202. Springer, Berlin, Heidelberg (2001). https://doi.org/10.1007/3-540-44795-4_17
8. Static and Dynamic sampling by GH John - 1996 Aug 2, 1996 - Publication: KDD 1996: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (1996)
9. Zaki, M.J., et al.: Evaluation of sampling for data mining of association rules. In: Proceedings Seventh International Workshop on Research Issues in Data Engineering. High Performance Database Management for Large-Scale Applications. IEEE (1997)
10. Sampling-based Randomized Algorithms for Big Data Analytics by Matteo Riondato, Ph.D., Brown University (2014)
11. Ganti, V., Gehrke, J., Ramakrishnan, R., Loh, W.Y.: A framework for measuring changes in data characteristics. In: Proceedings of PODS'99 (1999)
12. An Overview of Statistical Learning Theory. Vladimir N. Vapnik. Abstract—Statistical learning theory was introduced in the late. 1960's. Until the 1990's it was a by VN Vapnik - - 1999 Cited by 5582 articles
13. Mining frequent item through progressive sampling with rademacher averages. In: Publication: KDD '15: Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1005–1014 (2015). <https://doi.org/10.1145/2783258.2783265>
14. Zeng, X., Luo, G.: Progressive sampling-based Bayesian optimization for efficient and automatic machine learning model selection. *Health Inf. Sci. Syst.* **5**, 1–21 (2017)
15. Lozano, F.: Model selection using Rademacher penalization. In: Proceedings of the 2nd ICSC Symposium on Neural Computation (NC2000). Berlin, Germany. ICSC Academic Press (2000)
16. Koltchinskii, V.: Rademacher complexities and bounding the excess risk in active learning. *J. Mach. Learn. Res.* **11**, 2457–2485 (2010)
17. Koltchinskii, V., Panchenko, D.: Rademacher processes and bounding the risk of function learning. In: *High Dimensional Probability II*. Birkhäuser Boston (2000)
18. Pellegrini, L., et al.: MCRapper: Monte-Carlo Rademacher averages for poset families and approximate pattern mining. *ACM Trans. Knowl. Disc. Data (TKDD)* **16**(6), 1–29 (2022)
19. Banger, N., Kayarvizhya N.: A progressive sampling-based approach to reduce sampling time. In: 2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT), pp. 74–78 (2019)
20. Pellegrina, L., et al.: MCRapper: Monte-Carlo Rademacher averages for poset families and approximate pattern mining. *ACM Trans. Knowl. Disc. Data (TKDD)* **16**, 1–29 (2022)
21. De Stefani, L., Upfal, E.: A rademacher complexity based method for controlling power and confidence level in adaptive statistical analysis. In: 2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 71–80 (2019)
22. Oneto, L., et al.: Global rademacher complexity bounds: from slow to fast convergence rates. *Neural. Process. Lett.* **43**, 567–602 (2016)

23. Oneto, L., et al.: Fast convergence of extended Rademacher Complexity bounds. In: 2015 International Joint Conference on Neural Networks (IJCNN), pp. 1–10 (2015)
24. Cortes, C., et al.: Learning Kernels Using Local Rademacher Complexity. NIPS (2013)
25. Anguita, D., et al.: A deep connection between the Vapnik-Chervonenkis entropy and the rademacher complexity. IEEE Trans. Neural Netw. Learn. Syst. **25**, 2202–2211 (2014)
26. Advani, M.S., Saxe, A.M., Sompolinsky, H.: High-dimensional dynamics of generalization error in neural networks. Neural Netw. **132**, 428–446 (2017)
27. Liang, T., et al.: Learning with square loss: localization through offset Rademacher complexity. In: Annual Conference Computational Learning Theory (2015)
28. Bartlett, P.L., Mendelson, S.: Local Rademacher complexities and oracle inequalities in risk minimization. Ann. Stat. (2006)
29. Oneto, L., et al.: Learning with kernels: a local Rademacher complexity-based analysis with application to graph kernels. IEEE Trans. Neural Netw. Learn. Syst. **29**, 4660–4671 (2018)
30. Bartlett, P.L., et al.: Local rademacher complexities. Ann. Stat. **33**, 1497–1537 (2005)
31. Santoro, D., et al.: Mining sequential patterns with VC-dimension and rademacher complexity. Algorithms **13**, 123 (2020)
32. Bartlett, P.L., Mendelson, S.: Rademacher and gaussian complexities: risk bounds and structural results. J. Mach. Learn. Res. **3**, 463–482 (2003)
33. Ying, Y., Campbell, C.: Rademacher chaos complexities for learning the kernel problem. Neural Comput. **22**, 2858–2886 (2010)
34. Anguita, D., et al.: Some results about the Vapnik-Chervonenkis entropy and the rademacher complexity. In: The 2013 International Joint Conference on Neural Networks (IJCNN), pp. 1–8 (2013)
35. El-Yaniv, R., Pechyony, D.: Transductive Rademacher Complexity and Its Applications. ArXiv abs/1401.3441 (2007)
36. Smith, T.F., Waterman, M.S.: Identification of common molecular subsequences. J. Mol. Biol. **147**, 195–197 (1981)



Digital Glamour: Exploring the Intersection of Facial Makeup and Augmented Reality

Vinayambika S. Bhat^(✉), Deepthi Shetty, and Deepthi Kotian

Mangalore Institute of Technology and Engineering, Moodabidri, India
vinayambika09@gmail.com

Abstract. The COVID-19 global pandemic has affected the world in such a way that it has forced new systems to inhibit Augmented Reality (AR) technologies so as to avoid physical contact and also the business have turned to remote work mode. The Augmented Reality solution to help us through pandemic challenges and it may just represent the future of world business. Virtual try-on is an advanced way of trying a product. In this project, we examined and reviewed the existing works where they augment the face data and based on training models employed output is displayed as a blend of virtual and real object. Among many approaches, our emphasis is on models which inhibit deep learning and machine learning models with generative adversarial networks which are faster and more accurate. We are comparing various technologies for e-business that are trying to create a virtual image using web camera. The proposed model will work in the following way: When a person enters beside camera's vision, Camera will start capturing the live video stream of him which is nothing but a short video of specified frames. The deep learning algorithm works in background will capture the facial encodings of the person based on product which he wishes to try. Each time a new person enters into our portal (when he come's infront of camera), dynamic input of his face video stream is taken and based on the product which he/she wishes to buy facial encodings of that part is extracted and virtual product will be blended along with specified facial part. Final product will be a virtual try-on experience where in a person could actually visualize how his product would look when he actually wears it. The enhanced product is created by integrating deep learning and classical machine learning approaches, with the input given to the convolutional neural network being processed. For better interface with the user a website built using HTML and CSS is used to develop a front-end framework.

Keywords: Augmented Reality · Deep Learning · Data Augmentation · Deep Neural Network

1 Introduction

Augmented reality (AR), which is a subset of Virtual Reality (VR) is a rising technology that provides a visual experience of object that reside in physical world which are enhanced by the machine object (virtual object) across multiple sensory such as face, voice, dynamic image etc. COVID-19 has turned to be a global pandemic and it

has affected world in severe terms. It has caused a severe amount of deaths and it has put many people's life in darkness, hence to overcome from this pandemic many countries announced lockdown which intern shifted the office work to remote work. Since lockdown made physical marketing impossible the focus shifted to virtual/augmented reality-based approach. Augmented Reality is one of the new innovations that deliver immersive solutions for companies through digital marketing platforms [1]. The outbreak of COVID-19 has pushed the adoption of Virtual Reality. This is increasingly considered essential for ensuring business continuity. VR adoption is rapidly becoming mainstream for a variety of industrial and commercial applications. The future of virtual reality is bright, and it is anticipated that those who do not adopt this technology now will be left behind in the future [2]. Virtual try-on is a new approach to shop online. Since e-commerce sector is leading the global economy virtual try-on-based approach can surely turn to be a catch as it enhances marketing at remote areas with less cost. It also saves the time of traditional shopping and opens up many more opportunities for a customer. In a virtual try-on-based shopping approach, in internet purchasing, customers can select their favorite makeup/beauty products by comparing the actual wearing effects of various samples on lips and eyes [3]. When a person walks into the field of view of the camera, the camera begins recording a live video stream of him, which is nothing more than a brief video of predetermined frames. In the background, a deep learning algorithm will collect the person's facial encodings based on the product he/she wants to sample. When a new person opens a website a dynamic input of his face video stream is recorded, and based on the product that he wants to buy, facial encodings of that portion are extracted, and a virtual product is blended with that part. The final product will be a virtual try-on experience that allows users to see how their purchase will look when worn [4]. Let us assume that the original face data set as S then face data augmentation can be calculated by the following mapping function: $\emptyset:S \rightarrow \tau$.

Where τ is the virtually augmented set of S which is the plain face datapoints. Hence now the original dataset has been extended by combining the original and augmented sets.: $s^{\wedge} = s \cup \tau$.

This Augmented Reality Virtual Try-On is an online shopping website. Which involves augmenting the face related fashion products for lips, face and for eyes. Thus, the system provides user a virtual vision on how the product could actually look on them when they physically apply it. This is achieved by accessing webcam of the user. The user when enters the field of the view of the camera, there the camera captures the live video stream of user of specified frames which is input to our system [5]. When the user selects a particular item to try out, based on selected item the module extracts the target parts encodings and superimposes the virtual product on the target part and hence the user can try out various brands and decide whether that particular item suits him/her or not by them experience like manual shopping. There are many products which we apply on face which lack in displaying how it looks without actually applying it. The proposed project only covers face related accessories. The system consists of a camera and a inbuilt module (includes face detection and augmentation algorithms, user web interface, database which has collection of images of various virtual objects) [6]. Website actually provides listing of various products and their virtual try-on. Firstly, user has to login to our portal and should provide required credentials, later on he could access

various facilities provided by our website. To employ these functionalities, we use ‘Deep Neural Network’ for face data augmentation, ‘MongoDB database’ to store images and HTML, CSS to clone it to a website.

1.1 Full Stack Development

Full stack development is the creation of both the front end (client side) and backend (server side) components of a web application. Full stack web developers can create full-fledged web apps and websites. It will perform on the frontend, backend, database and debugging of web applications or websites. The front end is responsible for the visual look and feel of the website, while back end is responsible for the behind-the-scenes logic and infrastructure of the site [7].

1.2 Front-End Web Development

It is the visible portion of a website or online application that is responsible for the user’s experience. The front-end part of the web application or website is directly interacted with by the user. Client-side development, also known as front-end web development, is the process of creating an user-friendly and interactive interface that allows users to interact with the platform so as to navigate it or enter user data as required. It entails the use of HTML, CSS, and JavaScript, among other web technologies and tools. HTML is a mark-up language that determines a web page’s structure. It defines the appearance of a web page and can be thought of as the backbone of any online application. CSS is a language for style sheets that enhances HTML texts with additional style and design. Among these technologies, The latest language is called JavaScript. JavaScript modifies the Document Object Model (DOM) in HTML to give users a dynamic interface [8]. Furthermore, it provides users with an interactive interface by generating pop-up messages, form validation, and layout changes based on mouse or user input clicks. To create a front-end web interface, the browser controls all of these technologies.

1.3 Back-End Web Development

It is the server-side connection of a web application that informs the server upon which webpage to serve depending upon the user’s request. It is in charge of maintaining the database up to date using queries and APIs via client-side instructions. The implementation of computer programs and databases that will be utilized to service the client is known as server-side development, also referred to as back-end web development. Both front-end as well as back-end services are required for a web application to work. Back-end technology frequently uses a variety of programming languages, such as PHP, Ruby, Python, Java, Node.js, and numerous frameworks [9].

1.4 Database

Many websites store data whose data is saved on a database server with SQL. When retrieving quarry data from database servers, a full stack developer utilizes a software

program such as MySQL. Because many people use the SQL language, a well-liked program substitute for MySQL is MongoDB and different database servers are available. Databases are similar to collections of tables or spreadsheets that are connected by fields. IDs and keys enable fields to function so that database tables are correctly structured.

2 Methodology

A Data present in real world entity seems to be superimposed with virtual object (computer generated object) using AR technology. It is accomplished by blending a machine generated object (virtual object) with the real-world object. There are four forms of transforming plain data into virtual reality - namely marker dependent, marker independent, projection dependent and superimposition dependent augmented reality. Here we employ superimposition-dependent approach. There are many AR systems which employ superimposition-dependent approach. AR has a good scope in the improvement of medical results, manufacturing speed, education quality, entertainment field and robotic operations. Augmented reality and Virtual Reality are complementary [10]. The camera captures the input video and geometric adjustments were done so as to fit it with virtual image. The computer-generated object is superimposed on real world object using AR module on real time. This superimposition creates an augmented product. Virtual Try-On facility allows user to try different product they wish to purchase in online shopping. Products which we focused to implement here are those that are applied on face. Web camera captures train of images called frames from user as input. After the face recognition, virtual try on technique is applied for the user and user satisfies with that product then he/she can buy from the website.

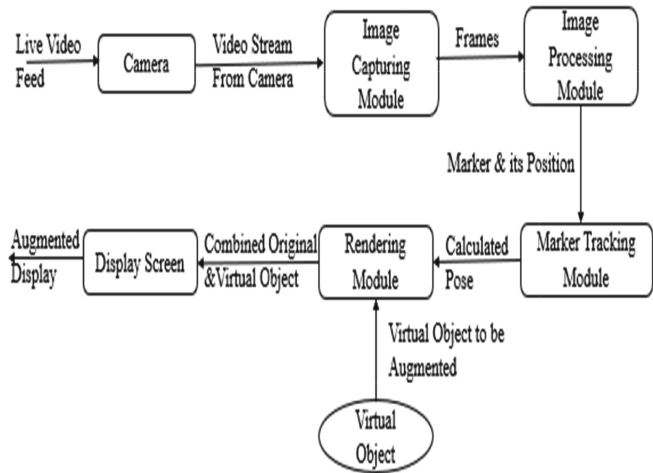


Fig. 1. Block Diagram of Augmented Reality virtual try-on

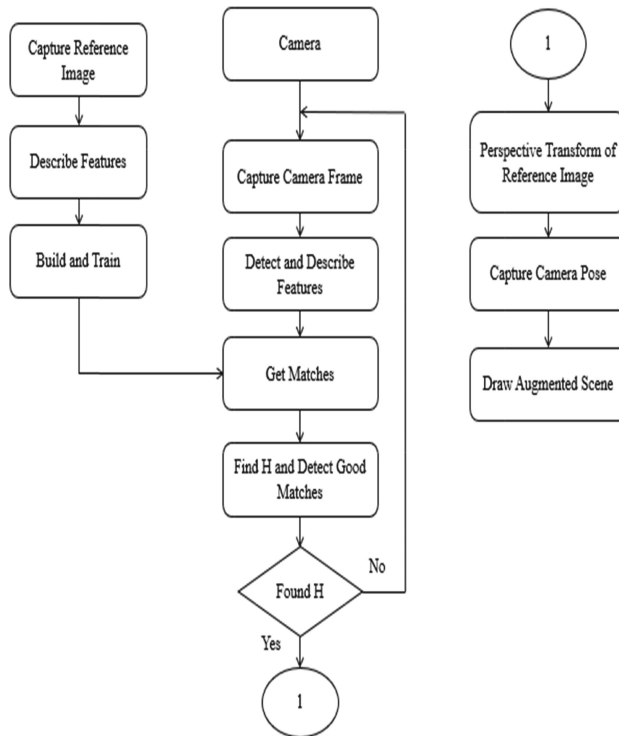


Fig. 2. Process design flow of virtual try-on system

The Fig. 1 represents the block diagram of our overall system. For the use of Augmented Reality technology in various virtual try-on systems, it is important to merge virtual things into the actual world. Face recognition is the primary process of system implementation. Face recognition has many processes involved within itself namely face tracking, feature extraction, face detection, and normalization, reduction in size (for optimization) and matching recognitions. From the input captured the actual range of face should be available if not then optimization is required. Face detection can take image or a video as an input. If the input is still image then face data extraction is done instantaneously; if input is a video stream then facial data extraction of each frame is performed and values are normalized to obtain final results. Frame the input face image feature extraction is done for augmentation. The extracted feature is stored in a high-dimensional matrix vector, where the values correspond to the pixel values. Since facial features data set is huge processing of it increases time and space complexity. The geometric features of face are adjusted before training which enhanced the model's efficiency in training input face. From input face image the target part is labelled with feature points which are nothing but dots which computer can analyze. After many processes of rotation, translation and scaling, the selected image now is said as primary sample. The local plain parameters are extracted from the facial images after the feature reduction. In order to increase the positioning accuracy of the model, the robust SIFT (Sender Initiated File Transfer) algorithms used for extracting the local or primary features to generate

the first sample [11]. Augmented reality involves blending the user's real-time image with digital overlays to form a better interface where user can try many products, before selecting their final product to buy. Augmented Reality enabled virtual try-on system allows buyers to experience the merchandise look on them with just screens before of them. The foremost technologies involve identification, feature tracking and computer vision.

The Fig. 2 represents process involved in virtual try-on. Based on camera input the position of targeted object is visualized based on its geometric orientation and position with respect to camera. The poses are adjusted either by changing the camera angle or by moving the object. The face recognition model determines maps the 3-dimensional geometric relationship between a point on the face surface and a point on the picture or video frame. It is that the defined geometry of our head model. The parameters are extracted by experiments and training the model by giving numerous inputs. This process is named labelling. Camera labelling is necessary as it calculates and analyses the geometric and optical properties of input and also helps in adjusting camera's vision relative to world frame of reference. The face pose estimation is done in following way. In the first step we map 2 dimensional co-ordinates on face image with the 3-dimensional co-ordinates of corresponding points on 3D face model. Using this co-ordinate, the head pose is calculated [12]. Supported posture of virtual object is mixed with real object to provide augmentation output.

2.1 Model Construction Phase

The Fig. 3 explains how the color prediction model works and process involved in virtual coloring of lips. We obtain the input as the user's lips color in terms of RED GREEN BLUE and corresponding RGB values are calculated, we might wish to predict the RGB of lip colors after wearing the lipstick. Numerical RGB values are calculated. This way of representing color pattern is most commonly adapted in most of the computers.

Then considered 3 lipstick shades to map our prediction model. The data values are scanned continuously and supervised learning algorithms which has property of regression are used for predicting the lip color. For every lipstick variant, we employ a mathematical approach which incorporates deep learning to find three basic functions, one for each RGB component.

$$\gamma_R: RO, GO, BO \rightarrow RN$$

$$\gamma_G: RO, GO, BO \rightarrow GN$$

$$\gamma_B: RO, GO, BO \rightarrow BN$$

where GN, BN, and RN are the required parameters (N represents the user's worn color) and BO, RO, and GO are the predictors (O means the first undertone color). When the functions have been understood, they will be placed to a test lip color [13]. As a result, for three lipstick shades, a maximum of nine functions can be mapped.

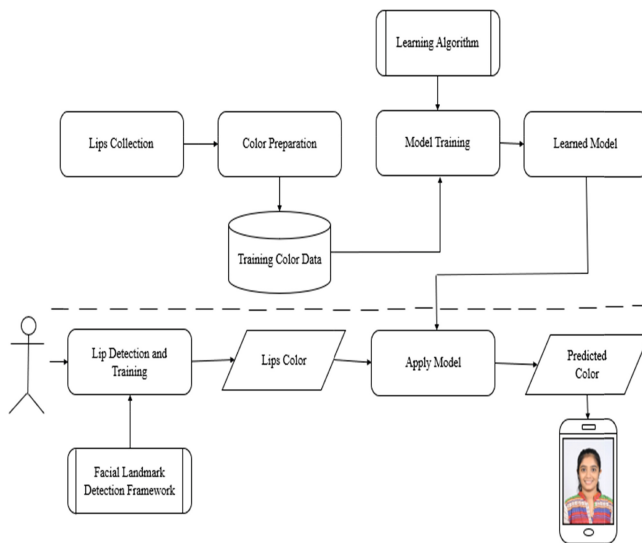


Fig. 3. Lips Color Prediction Approach

2.2 Color Prediction Model

Based on the obtained data values by training the input, a linear model that suit these data values can be predicted using the (single or multiple) Linear Regression. Here we match the RGB component with linear model to get the target object, blue for blue, red for red, and green for green. For other color combination it combines red, green and blue light to obtain desired color. RGB is incorporated in most of the on-screen applications like graphic design, television and computer. Every color channel is represented from 0 (least saturated) to 255 (most saturated). Hence a total of 16,777,216 colors can be represented using RGB color space.

2.3 Dlib Library

It's an open source software library written in C++ for Machine learning applications. One of the most important uses is face detection. This library uses a pre-trained model that identifies 68 data points on the user's face depending upon the position of the eyes, nose and mouth. We are using this library to detect the eyes and mouth of the user and to apply appropriate try on [14]. The Fig. 4 illustrates the 68 landmarks numbering on the user's face when using this library and Fig. 5 represents 68 landmarks when the algorithm is run on the user's face.



Fig. 4. Sample of 68 datapoints on the user’s face

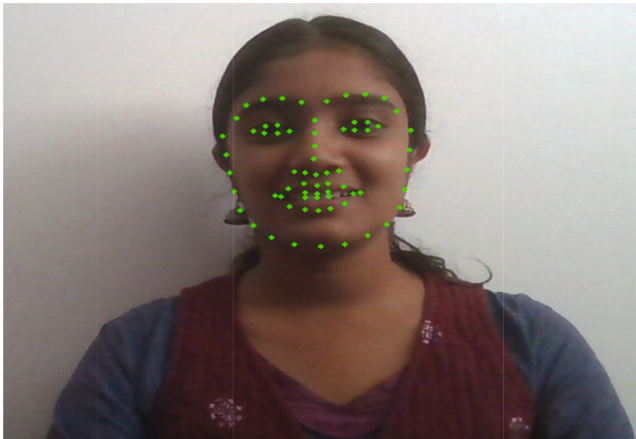


Fig. 5. 68 landmarks when the algorithm is run on the user’s face

2.4 Eye Detection

Facial features have a predetermined relationship. For example, the mouth is at the lowest point on the torso, while the eyes are at the highest part of the face.. This approach detects the targets on the Region of Interest (ROI) region after determining the ROI of the mouth and eyes, in order to increase detection accuracy and speed. The top one-eighth of the image is removed, the bottom seven-eighths of the image are set aside for the eye ROI, and the upper half of the image is filtered and saved as image1. This is demonstrated in Fig. 6(a). The eye area is located in this ROI using the EyeMap algorithm. This method is really effective. In the space, this method creates two EyeMaps, EyeMappingC and EyeMappingL, as shown in Fig. 6(b) and Fig. 6(c) respectively. Which are then integrated into a single map as shown in Fig. 7(a). The Fig. 7(b) represents the connected pair of

components which are the used to characterize the eye region in rectangular box as in Fig. 7(c). Experiments reveal that the eyes have high Cb and low Cr components where Cb and Cr are difference in red a blue chroma component respectively. X and Y represents the two coordinates and EyeMappingC is estimated as:

$$\text{EyeMappingC} = [[[C_b]^2 + [(255 - C_r)]^2 + C_b / C_r] / 3$$

EyeMappingL is calculated as shown below:

$$\text{EyeMappingL} = (Y * (X, Y) \oplus g(x, y)) / (Y * (X, Y) \ominus g(x, y))$$

Then, EyeMappingC is multiplied by EyeMappingL to obtain EyeMaps.

$$\text{EyeMaps} = \text{EyeMappingC} * \text{EyeMappingL}$$

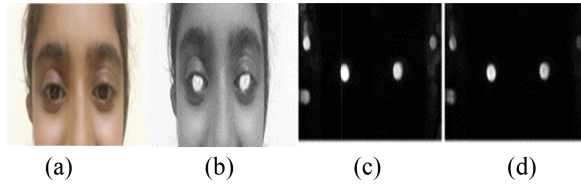


Fig. 6. EyeMaps construction illustration: (a) the original eye ROI (b)EyeMappingC.(c) EyeMappingL.(d) EyeMaps.

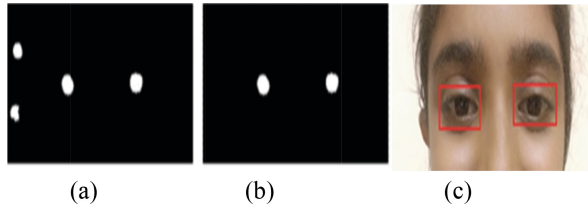


Fig. 7. Various stages of the eye tracking method: (a) Image of EyeMaps after color binarization. (b) A connected pair of binarized components. (c) Use a rectangular box to characterise the eye region.

2.5 Mouth Detection

The efficiency and time taken to detect the mouth portion of the image is directly decided by the ROI characteristics of the data points distributed across the mouth placed on the face region. The lower part of the targeted face region contained the mouth region.. The bottom third of the face image is cropped out and stored as image2, as described in the green box in Fig. 8(a), and the central portion of image2 is collected and stored as the image of the mouth ROI. However, when the mouth opens wide (yawning), as seen in Fig. 8(b), it is impossible to get a whole mouth region. The whole mouth region is obtained by expanding the facial points' height by one fifth downward, as shown in Fig. 8(c).From this pixels will be selected and presented in Fig. 8(d).When the mouth opens across a wider area, the ROI is too large, affecting the selective extraction of the internal geometry/ROI within the mouth; consequently, essential to exactly targeted

the mouth. The area around the mouth is exactly aligned using segmentation of lips regarding the difference in color between the lips and the skin, and the lips are divided based regarding the color intensity / chromaticity ‘S’ in the space of RGB [15]. The chromaticity S value is calculated as follows:

$$S = 2 \times \tan^{-1}(\frac{R - G}{R})/\pi$$

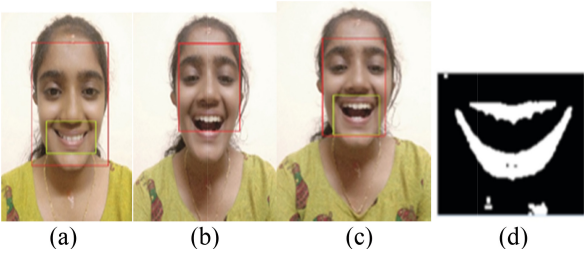


Fig. 8. Several phases are involved in method of detecting the mouth: (a) Initial mouth ROI. (b) The mouth expands widely. (c) The final ROI of the mouth. (d) Pixels chosen

The Fig. 9 and Fig. 10 illustrates the different virtual try-on for the eye such as eye liners, mascara, eye-shadow, kajal and eyebrow. When the user opens this page, it uses a web camera for face detection and whenever user select a particular product it will be augmented on the user’s face. It has various makeup options for eyes and lips. It has different sub section also with different colors. User can try any of these virtually.

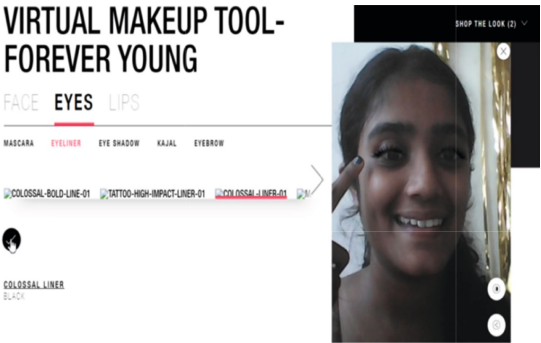


Fig. 9. Virtual eye-liner try-on

Figure 11 illustrates the different virtual try-ons for lip makeup such as lipstick and lip balm. A palette of colors is also provided for the user to choose from.

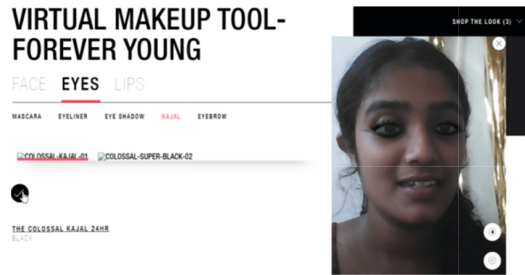


Fig. 10. Virtual Kajal try-on

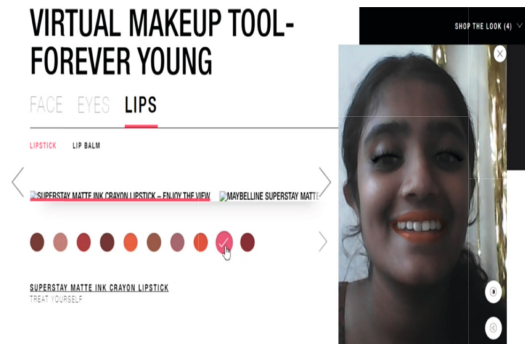


Fig. 11. Virtual lipstick try-on

3 Conclusion

A common problem faced by customers while shopping is the need to spend hours trying out a variety of lipsticks and other makeup products physically. This can be a difficult for user and the time available might be short. The proposed solution to overcome this problem is the use of a Virtual Try-On that acts as a virtual trial room. It uses a facial recognition technic and there by extracting the features of eyes and mouth. Then different makeup colors for lips, eyes are augmented on it. Thereby eliminating the need to physically try on and hence help to save the time. The project proposed here allows the customers to choose the makeup of their choice This project can be improvised by developing more features such as virtual clothing, glasses try on, and jewelry try on.

References

1. Asadzadeh, A., Samad-Soltani, T., Rezaei-Hachesu, P.: Applications of virtual and augmented reality in infectious disease epidemics with a focus on the COVID-19 outbreak. Inform. Med. Unlocked **24**. Elsevier Ltd (2021). <https://doi.org/10.1016/j.imu.2021.100579>
2. Laghari, A.A., Juman, A.K., Kumar, K., Chhajro, M.A.: Systematic analysis of virtual reality & augmented reality. Int. J. Inf. Eng. Electron. Bus. **13**(1), 36–43 (2021). <https://doi.org/10.5815/ijieeb.2021.01.04>

3. Gao, W., Liu, Y., Liu, Z., Li, J.: How does presence influence purchase intention in online shopping markets? an explanation based on self-determination theory. *Behav. Inf. Technol.* **37**(8), 786–799 (2018). <https://doi.org/10.1080/0144929X.2018.1484514>
4. Cheng, Q., Zhang, S., Bo, S., Chen, D., Zhang, H.: Augmented reality dynamic image recognition technology based on deep learning algorithm. *IEEE Access* **8**, 137370–137384 (2020). <https://doi.org/10.1109/ACCESS.2020.3012130>
5. Zhang, B.: Augmented reality virtual glasses try-on technology based on iOS platform. *Eurasip J. Image Video Process.* (2018). <https://doi.org/10.1186/s13640-018-0373-8>
6. Alfian, G., Ijaz, M.F., Syafrudin, M., Syaekhoni, M.A., Fitriyani, N.L., Rhee, J.: Customer behavior analysis using real-time data processing: a case study of digital signage-based online stores. *Asia Pacific J. Mark. Logist.* **31**(1), 265–290 (2019). <https://doi.org/10.1108/APJML-03-2018-0088>
7. Botwe, D.A., Davis, J.G.: International journal of computer science and mobile computing a comparative study of web development technologies using open source and proprietary software (2015). www.ijcsmc.com
8. Maratkar, P.S., Adkar, P.: React JS-An Emerging Frontend Javascript Library (2021)
9. Abdullah, H.M., Zeki, A.M.: Frontend and backend web technologies in social networking sites: facebook as an example. In: *Proceedings - 3rd International Conference on Advanced Computer Science Applications and Technologies, ACSAT 2014*, pp. 85–89 (2014). <https://doi.org/10.1109/ACSAT.2014.22>
10. Yaoyuneyong, G., Foster, J.K., Flynn, L.R.: Factors impacting the efficacy of augmented reality virtual dressing room technology as a tool for online visual merchandising. *J. Glob. Fash. Mark.* **5**(4), 283–296 (2014). <https://doi.org/10.1080/20932685.2014.926129>
11. Wang, X., Wang, K., Lian, S.: A Survey on Face Data Augmentation (2019). <https://doi.org/10.1007/s00521-020-04748-3>
12. Feng, Z., Jiang, F., Shen, R.: Virtual glasses try-on based on large pose estimation. *Procedia Comput. Sci.* **131**, 226–233 (2018). <https://doi.org/10.1016/j.procs.2018.04.207>
13. Wiwatwattana, N., Chareonvassakul, S., Maleerutmongkol, N., Charoenvitvorakul, C.: Enhancing lipstick try-on with augmented reality and color prediction model. *Adv. Intell. Syst. Comput.* **738**, 359–367 (2018). https://doi.org/10.1007/978-3-319-77028-4_48
14. Feng, Y., Xie, Z., Li, F.: Intelligent virtual lipstick trial makeup based on OpenCV and Dlib. In: *Advances in Intelligent Systems and Computing, AISC*, vol. 1274, pp. 121–127 (2021). https://doi.org/10.1007/978-981-15-8462-6_14
15. Ji, Y., Wang, S., Lu, Y., Wei, J., Zhao, Y.: Eye and mouth state detection algorithm based on contour feature extraction. *J. Electron. Imaging* **27**(05), 1 (2018). <https://doi.org/10.1117/1.jei.27.5.051205>



Local Stereo Matching Technique Based on Collaborative Cost Aggregation and Improved Disparity Refinement

Deepa^{1,2(✉)}, K. Jyothi^{2,3}, and Abhishek A. Udupa⁴

¹ NMAM Institute of Technology-Affiliated to Nitte (Deemed to be University), Nitte, Karnataka, India

deepashetty17@nitte.edu.in

² Visvesvaraya Technological University, Belagavi, India

³ J.N.N.C.E, Shimoga, Karnataka, India

⁴ Belagavi, India

Abstract. Stereo matching is used in several applications. Due to occluded and mismatched regions, it is still considered as a difficult problem. An efficient cost for matching and an adaptive support weight technique for cost aggregation to obtain raw disparity is proposed in this paper. Finally, the raw disparity is refined to improve the stereo matching performance in occluded and mismatched areas. Initially, an effective matching cost technique based on census transform is applied. This matching cost can withstand any radiometric variations. Next, an optimized Adaptive support weights framework for cost aggregation is used. In this framework, a joint bilateral filter and a guided filter are combined to form collaborative cost volume filtering. Finally, a novel multistep refinement technique, including outlier detection, interpolation and inpainting is proposed. The experimental outcomes of the proposed method outperforms currently existing techniques on Middlebury datasets.

Keywords: Adaptive support · Bilateral · Census · Disparity · Inpainting · Occluded

1 Introduction

Stereo matching focuses on finding a dense disparity map by comparing the corresponding similarity between stereo images that are taken from separate angles. The depth details of the scene is further calculated based on the disparity [1, 2]. The correctness of the disparity map generated from stereo matching will have a significant control on the resultant 3D reconstruction. Due to the existing noise, error in the matching, or any kind of distortion available, stereo matching is a difficult task and hence estimating an efficient disparity map continues to be a challenging issue. Stereo matching helps to find the 3D details in an inexpensive way as compared to other methods which includes active 3D sensors like LiDAR, ToF, and structured light. The significance of stereo matching is increasing lately, because 3D information finds its use in several emerging areas such

as virtual reality, augmented reality and robot vision. The two main properties in stereo matching are accuracy and speed of disparity map. In the past decade, an extensive research is carried out to increase the precision and reduce the computational time of these algorithms. But, the stereo matching algorithms that are developed to obtain good quality disparity map generally require higher computational time. The speed and precision are the preferable but conflicting characteristics. Therefore it is highly difficult to realize both the characteristics at the same time. Hence, we need to focus on developing an efficient stereo matching algorithm.

In this paper, an efficient cost for matching using census transform is applied. This cost is robust to any radiometric deviations. We propose a combined ASW method by performing cost aggregation using joint bilateral filter (JBF) and guided filter. Further a new multistep refinement technique, including outlier detection, interpolation and inpainting is proposed. The outcomes on Middlebury benchmark show the effectiveness of the suggested approach.

2 Literature Review

A study on various stereo matching algorithms is performed in [3]. Depending on this review, the stereo matching methods are categorized into local and global methods. Global methods generally determine all the disparities by reducing the energy function that is based on global optimization algorithm. The commonly used global methods are graph cut method [4], belief propagation [5] (BP) and dynamic programming [6]. Li et al. [7] used a unique technique to enhance both the efficiency and correctness of matching in the global methods. Wang et al. [8] implemented a technique for stereo matching that relies on global correlation. This method has a good matching integrity and produces good quality disparity maps. Wang et al. [9] proposed the technique of using both local details and global details. In this method, a combined information for gradient and intensity is used as cost function in BP method. This method performs quite well specially in the texture-less and discontinuous areas. Pan and Liu [10] made use of an auto-encoder in order to obtain the features which are then used for matching cost. Despite of the high accuracy, computational time and the memory requirements of these global methods are high.

Local methods use the similarities between neighbouring pixels in the given support region. Initially, the cost for each individual pixel at given disparity value is computed using certain measure for pixel correspondence. Next, to minimize the uncertainty in the matching cost, the aggregated costs of each of the pixels in support region is computed at all disparity. Later, the best disparity value that yields the least aggregated cost is adopted using a suitable optimization method. The authors in [11, 12] applied the Census transformation for the matching cost calculation. This cost is robust to radiometric variations cost and produces good matching results. It overcomes the amplitude distortion caused by traditional similarity measure function based on SAD. A local algorithm is proposed by Chai et al. [13] that combines various measures of similarity cost. This algorithm is effective and shows good results to lighting distortions. Local methods assume the implicit smoothness of pixels in a support region. But, this do not hold good at discontinuous regions and this may cause an edge fattening effect. Thus, in order to get good

results both at the homogeneous regions as well as at depth discontinuities, a suitable support window has to be chosen adaptively for every pixel.

The existing techniques for cost aggregation achieved limited accuracy, especially in the mis-matched and occluded regions. The vital improvement in these algorithms is the use of adaptive support weights (ASW) methods. These methods attain accuracy that is similar to the global methods. The main goal of ASW methods is allocating a feasible weight to each pixel in the given support region. These weight assigned determines the likelihood that the centre and a neighbouring pixel in the support region belong to the same region. The concept of ASW strategy was first introduced in [14]. In this approach, the pixel weight within a given support region is calculated depending on the resemblance of the color and spatial displacement. The same idea is used in bilateral filter (BF). ASW for facial images is proposed in [15]. The authors in [16] assigned the weights in a support region by calculating the geodesic displacement to the centre pixel. The guided filter (GF) is proposed in [17], showed better results over BF both in terms of quality and accuracy. GF is used in [18] for filtering the cost volume and achieved good quality results as compared to most of the local methods. They took one image as the guidance image and used GF to conduct the filtering of cost volume. An improvised stereo matching technique is presented in [19] which relies on the adaptive guided filtering (AGF). They used adaptive window that is rectangular in shape, rather than the square window of fixed size. An efficient matching cost criteria is proposed by Liu et al. [20] that used an adaptive shape-guided filter-based technique for aggregating the cost to enhance the performance of the in texture-less regions. Baek et al. [21] used the diffusion of colour image and guided filter for cost aggregation but in the cost computation step the absolute difference and gradient details are used. This takes into consideration only the pixel information to be matched. Since the information of individual pixels might be prone to noise, the accuracy may be less. The authors in [22] used the Sobel method where the image is split into an edged and non-edged areas, later the non-edged areas are divided into smooth and non-smooth areas using cross window method. But they do not perform well in the outlier region. Information like gradient and colour difference to allot appropriate weights to matching techniques was introduced in [23], which enhanced the accuracy of matching in the weak-textured areas. But to calculate the matching cost they used AD and gradient details which do not have sufficient reliable information. It did not consider the information from the neighbouring pixels, that reduced the accuracy of the algorithm. A segment-based ASW in cost aggregation is proposed in [24] which do not produce good results in the outlier regions.

Most of the existing stereo matching algorithms tried to improve the accuracy but there are still certain artifacts in the occluded and mismatched regions. These methods have not focused much on the improvement of cost aggregation and disparity refinement stage. The local stereo matching technique based on collaborative cost aggregation and improved disparity refinement strategy is proposed. The experimental outcomes show the proposed method performs well in the occluded and mismatched regions and generates good quality disparity map.

3 Methodology

The steps followed in the proposed stereo method is shown in Fig. 1. In the sections that follow, the steps are explained in detail.

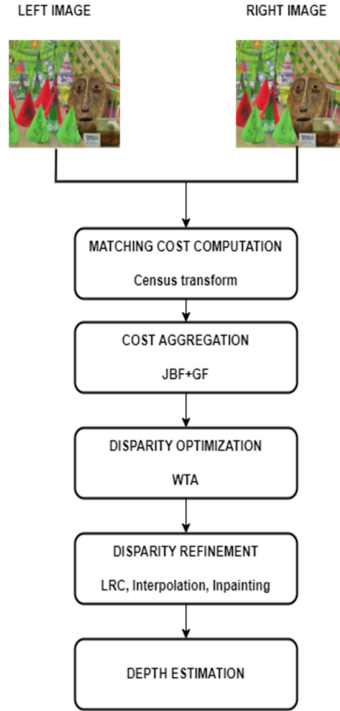


Fig. 1. Flow diagram of the methodology

3.1 Matching Cost Computation

Census transformation (CT) is used for matching cost computation [25]. It mainly relies on the value of the neighboring pixels. Hence, the local matching algorithm that relies on Census transformation has good resistance against noise in images that has large illumination variations and hence produce a good quality disparity map. CT first considers a window having pixel p as the center and compares the intensity of all the neighboring pixels within the window. If the value of pixel p is more than the neighboring pixels, then the value 1 is assigned to the resultant bit string, else, it is assigned as 0. Consequently, the resultant pixel transformation is computed by combining them in the form of bit strings. This can be formulated as follows:

$$C_T(p) = \otimes_{q \in N} C(\ln(p), \ln(q)) \quad (1)$$

Where $C(ln(p), ln(q))$ the function for comparison is, p is center pixel, and q represent all the neighboring pixel in the window centered on p . $ln(p)$, $ln(q)$ are the intensity values of pixel p and q respectively.

$$C(ln(p), ln(q)) = \begin{cases} 1 & ln(p) \leq ln(q) \\ 0 & ln(p) > ln(q) \end{cases} \quad (2)$$

The bit string values of pixels are compared and the cost of matching is found using the similarity measure called Hamming distance, which is as shown below

$$C(p, d) = Hamming(C_{TL}(p), C_{TR}(p, d)) \quad (3)$$

Here we take an example of window size size 3×3 . The process followed in Census transformation is shown in Fig. 2. The transformation codes are computed as 00011010 and 10010010. These transformed codes are XORed, and the matching cost is obtained as 2.

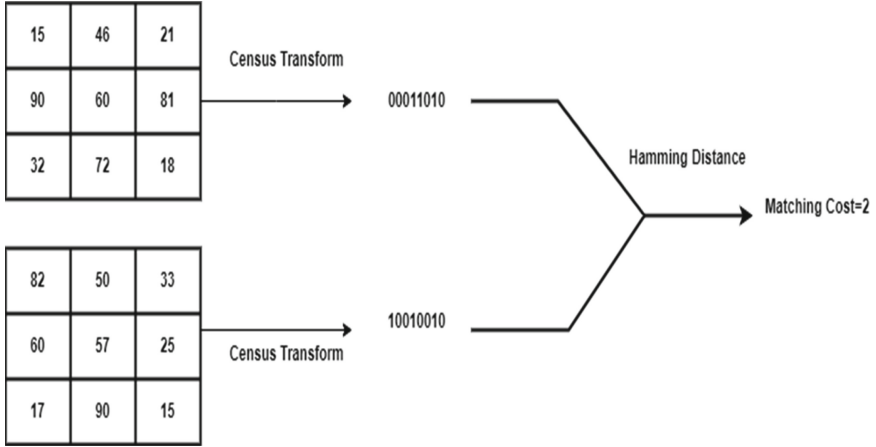


Fig. 2. Census Transformation

3.2 Cost Aggregation

Aggregation of cost is crucial in minimizing the noise and errors in matching in the initial cost volume. ASW strategy is used for cost aggregation. ASW methods assigns a suitable weight to pixels in the support region. The weight is calculated using the similarity of color to the center pixel and spatial displacement from the center pixel. In these stereo methods, the aggregation is performed by filtering the cost volume. The aggregated cost of pixel p at disparity d is computed as follows:

$$C^A(p, d) = \sum_{q \in w_p} W(p, q) C(q, d) \quad (4)$$

Here w_p represent a support region with pixel p as center, and $W(p, q)$ represent the weight function.

To deal with regions of low texture as well as regions of high texture, we propose a hybrid cost aggregation method, using a joint bilateral filter and a GF to collaboratively perform the cost volume filtering. The final weight function is the weighted average value of the JBF and the GF. The JBF determines weighted average of the pixels in the support region by considering the spatial information and the difference in intensity between the pixel at the center and neighboring pixels. The effect of a neighboring pixel does not depend only on the spatial displacement but also on pixel value similarity so as to decrease the effect of pixels that are separated by an edge. GF produce higher quality results as compared to several other local edge-aware filters and has less computational complexity, so we also use GF to perform cost aggregation.

Let $W_{JBF}(p, q)$ be the weight function of the JBF and $W_{GF}(p, q)$ is the weight function of GF.

The weight function for JBF is expressed as

$$W_{JBF}(p, q) = \sum_{k:(p,q) \in \omega_k} \exp\left(\frac{|p - q|^2}{s^2}\right) \exp\left(\frac{|I_p - I_q|^2}{c^2}\right) \quad (5)$$

where p and q represent pixel indexes. The s denotes a spatial adjustment factor and c denotes similarity factor. The $|p - q|$ represent the spatial Euclidean distance and $|I_p - I_q|$ denotes the intensity difference.

GF is an efficient edge-preserving filter. It is used in several local stereo matching algorithms, and has achieved a good quality disparity maps. The weight function $W_{GF}(p, q)$ for GF is expressed as:

$$W_{GF}(p, q) = \frac{1}{|\omega|^2} \sum_{k:(p,q) \in \omega_k} \left[1 + \frac{(G_p - \mu_k)(G_q - \mu_k)}{\sigma_k^2 + \epsilon} \right] \quad (6)$$

where p and q represent pixel indexes. G_p and G_q are values of pixels for the guidance image at the location p and q . μ_k and σ_k^2 denote the mean value and variance value of window ω_k in guided image G . $|\omega|$ is the total pixels in the window ω_k . ϵ represent a parameter for smoothness.

The weight function $W(p, q)$ of the proposed approach is the average of $W_{JBF}(p, q)$ and $W_{GF}(p, q)$ which is represented as

$$W(p, q) = \frac{1}{2} [W_{JBF}(p, q) + W_{GF}(p, q)] \quad (7)$$

The final aggregated cost is denoted as

$$C^A(p, d) = \sum_{q \in w_p} \frac{1}{2} [W_{JBF}(p, q) + W_{GF}(p, q)] C(q, d) \quad (8)$$

$$C^A(p, d) = \frac{1}{2} [C_{JBF}^A(p, d) + C_{GF}^A(p, d)] \quad (9)$$

Where C_{JBF}^A represent the cost volume obtained using JBF and C_{GF}^A is the cost volume of the GF.

Choosing proper threshold values are very crucial in local stereo matching techniques. Here, we discuss the effect of various threshold values. The influence of the threshold values for σ_s , σ_c for JBF and ϵ for GF for Teddy image are shown in Fig. 3, 4, and 5 respectively. The value of σ_s is selected as 9, σ_c is selected as 6 and ϵ to 0.25 as it yields the least percentage of bad matching pixels.

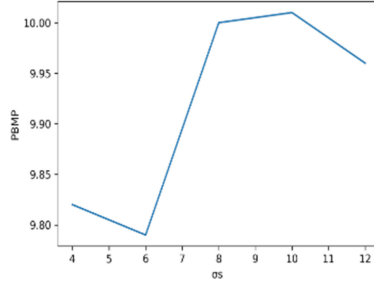


Fig. 3. PBMP for various threshold values of σ_s for Teddy image

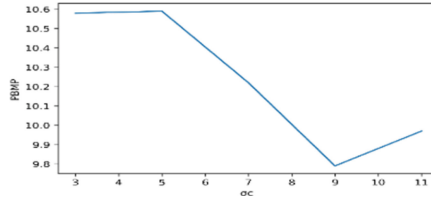


Fig. 4. PBMP for various threshold values of σ_c for Teddy image

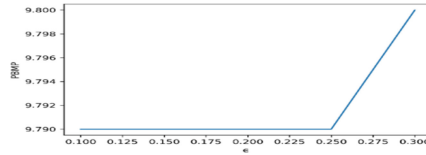


Fig. 5. PBMP for various threshold values of ϵ for Teddy image

3.3 Disparity Optimization

To get an accurate disparity mapping, we estimate the disparity map by applying the Winner Take All (WTA) method. This method chooses the minimum value of the aggregated cost obtained at each pixel location. The formula for WTA is given by

$$d_p = \operatorname{argmin}_{d \in D} C^A(p, d) \quad (10)$$

3.4 Disparity Refinement

The disparity maps obtained may have several occluded and mismatched pixels. The occluded and mismatched regions are detected using outlier detection. A multi-step disparity refinement technique that includes interpolation and inpainting are used to refine the disparity errors.

3.4.1 Outlier Detection

The outliers in the raw disparity map are detecting by applying a left-right consistency check. Let $d_{LR}(x)$ be the left disparity map and $d_{RL}(x)$ be the right disparity map. In this strategy, the pixels are considered as outliers if it satisfies the following inequality

$$|d_{LR}(x) - d_{RL}(x - d_{LR}(x))| \geq T_{LR} \quad (11)$$

The T_{LR} is the threshold for outlier detection. The detected outliers are classified as mismatched regions if $1 < T_{LR} \leq 2$. The outliers are classified as occluded regions if $T_{LR} > 2$. In the Fig. 3 the occluded regions are marked in green color and mismatched regions are in blue color for Teddy image (Fig. 6).



Fig. 6. Outlier detection: (a) Left Teddy image (b) Outlier map

3.4.2 Interpolation

In the interpolation process, for each pixel marked as occluded we fill the pixel by the taking the information from the reliable pixels. Let I_l be the disparity map computed by replacing the occluded pixel by the closest reliable pixel to the left. Let I_r be the disparity map computed by replacing the occluded pixel by the closed reliable pixel to the right. The disparity map after interpolation is obtained by the following equation.

$$D_p = \min(I_l, I_r) \quad (12)$$

3.4.3 Inpainting

Inpainting is the method of filling any missing areas of image by taking the details from the neighboring region so that the resultant image looks seemingly natural. In this paper, we used inpainting technique based in “Navier-Stokes” [26] to fill the mismatched

regions. Let the disparity map obtained by the above step represent the image domain. The mismatched region (unknown region) is taken as the area for inpainting. The inpainting algorithm travels along the edges of the image moving from the known regions to the unknown regions, transferring the details from the known to the inpainting region. The filling process is performed such that the isophotes line coming at the boundary of the unknown region are filled from inside making a smooth transition of information to the center part of the unknown region. Let D be the disparity map and $\Omega \in D$, where Ω is the region in D to be inpainted as represented in Fig. 7, $\delta\Omega$ is the boundary of Ω . The aim of inpainting is to interpolate the information from the boundary $\delta\Omega$ and fill the inpainting region Ω with suitable pixel value.



Fig. 7. Inpainting process

3.5 Depth Estimation

Once the disparity map for a pair is generated, the depth Z can be estimated using the following formula

$$Z = \frac{fXB}{D} \quad (13)$$

Here f is the focal length and B represent the baseline of stereo camera. The depth for the Teddy image is shown in Fig. 8.



Fig. 8. Depth of Teddy image

4 Results and Discussions

The experiments were performed on a PC having a 2.11 GHZ Intel Core i5-10210U CPU with memory of 8 GB. Windows 10 operating system and Python 3.10.7 language are used to implement the algorithm. To verify the accuracy, the results are evaluated using Middlebury datasets [27–30]. This dataset includes rectified stereo images.

4.1 Evaluation Metrics

The accuracy of the method proposed is quantified using peak signal to noise ratio (PSNR) and PBMP between the ground truth and estimated disparities. Higher value of PSNR determined better efficiency. The efficiency is better when PBMP values are lower. Let the total number of pixels be N . d_t and d_g be the disparity map estimated and ground truth disparity maps respectively.

PSNR is computed as:

$$PSNR = 10 \log_{10} \frac{255^2 * N}{\sum (d_t(x, y) - d_g(x, y))^2} \quad (14)$$

PBMP is calculated as follows:

$$PBMP = \left[\frac{1}{N} \sum |d_t(x, y) - d_g(x, y)| > T \right] * 100 \quad (15)$$

4.2 Performance Evaluation of the Method Proposed

We used the 2001, 2003 and 2014 datasets in the Middlebury benchmark for evaluation of the stereo matching technique proposed in the paper. The images such as Cones, Teddy and Venus having dimensions 450×375 , 450×375 , and 434×383 with the disparity level of 60, 60, and 20 respectively are considered from Middlebury 2001 and 2003. The images such as Adirondack, JadePlant, Motorcycle, Piano, Pipes, and Recycle were considered from Middlebury 2014. The proposed method is applied step by step to compute the disparity map. The qualitative results achieved by the proposed method for Cones, Teddy, and Venus images are presented in Fig. 9. The qualitative results achieved by the proposed method for Adirondack, JadePlant, Motorcycle, Piano, Pipes, Recycle images are presented in Fig. 10. By visually evaluating the disparity map, we find that the suggested method produce accurate disparity map.

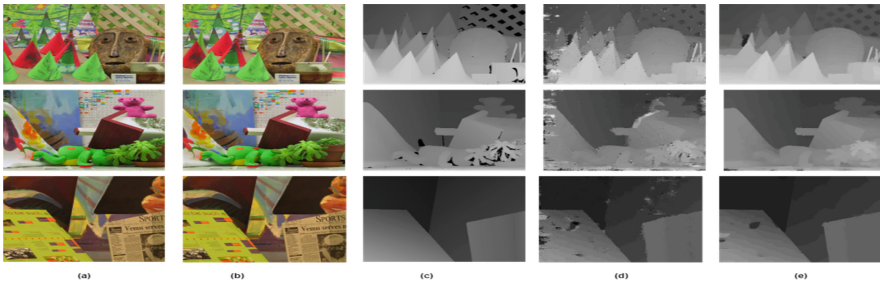


Fig. 9. Qualitative results for images (a) Left image (b) Right image (c) Ground truth (d) Initial Disparity map (e) Final disparity map

The effectiveness of the various steps followed in the proposed method on Cones, Teddy and Venus images in terms of PBMP is shown in the Fig. 11. We find the PBMP gradually decreases at each step.

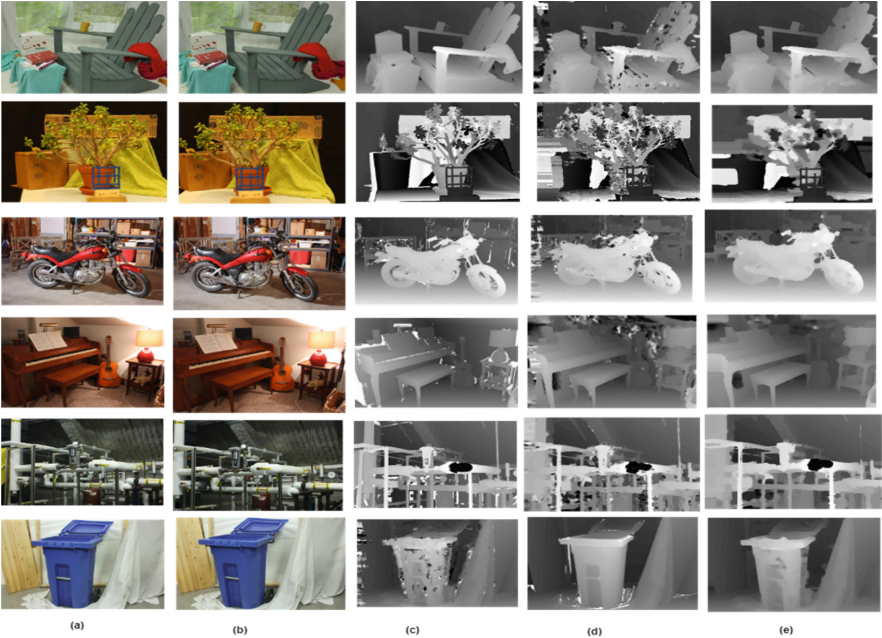


Fig. 10. Qualitative results for images (a) Left image (b) Right image (c) Ground truth (d) Initial Disparity map (e) Final disparity map

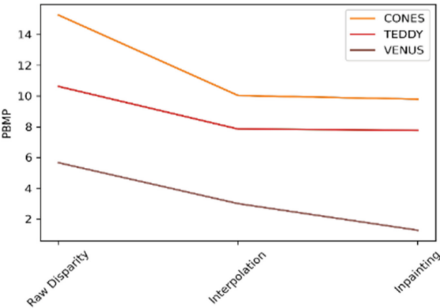


Fig. 11. Performance of the proposed method for images Cones, Teddy, Venus

The suggested method is compared with existing algorithms [21–23] to show the performance. Baek et al. [21] consider only the pixel information to be matched which may not generate good quality disparity map.. The methods proposed in [22] do not perform well in the outlier region. Cheng et al. [23] introduced combined costs, which enhanced the accuracy of matching in the weakly-textured regions, but information from the neighboring pixels is not taken into consideration, which reduced the accuracy of the method. The PBMP results of all the images obtained by our method and compared method is shown in Table 1. We find that, suggested method works better than the compared methods in terms of PBMP because it's PBMP is lower than those of the compared

methods. The PSNR results of the images obtained by our method and compared method is shown in Table 2. From the table we find that the PSNR of our method is higher than the compared method indicating that our method works better than the compared methods in terms of PSNR.

Table 1. Comparison of PBMP results

	Cones	Teddy	Venus
Baek et al. [21]	12.88	13.50	1.90
Huang et al. [22]	12.06	15.50	0.64
Cheng et al. [23]	10.72	13.06	0.69
Proposed	7.77	9.79	1.27

Table 2. Comparison of PSNR (db) results

Images	Hanie et al. [31]	Proposed
Adirondack	29.76	34.59
Jade Plant	25.98	27.97
Motorcycle	29.35	37.85
Piano	28.15	36.16
Pipes	25.74	36.32
Recycle	31.63	39.99

5 Conclusion

The paper proposes local stereo matching technique based on effective cost measurement, collaborative cost aggregation and improved disparity refinement techniques to obtain good quality disparity map in the occluded and mismatched regions. A matching cost that relies on census transform is proposed. Besides, a collaborative cost aggregation method using joint bilateral filter and guided filter is proposed to enhance the accuracy of disparity estimation. Further multistep refinement method using interpolation, in painting is proposed to refine the raw disparity map. To show the efficacy of the suggested approach, the experimental results of the method were assessed on the Middlebury dataset. The results show that the suggested stereo matching technique produces precise disparity maps.

References

1. Su, C., Tan, G., Luo, Y.: Research on stereo matching technology based on binocular vision. OALib **06**(09), 1 (2019). <https://doi.org/10.4236/oalib.1105755>

2. Binh Do, P.N., Chi Nguyen, Q.: A review of stereo-photogrammetry method for 3-D reconstruction in computer vision. In: Proceedings of the 2019 19th International Symposium on Communications and Information Technologies, ISCIT 2019, pp. 138–143 (2019). <https://doi.org/10.1109/ISCIT.2019.8905144>
3. Scharstein, D., Szeliski, R., Zabih, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vis.* **1**, 7–42 (2002). <https://doi.org/10.1109/SMBV.2001.988771>
4. Kolmogorov, V., Monasse, P., Tan, P.: Kolmogorov and Zabih's graph cuts stereo matching algorithm. *Image Process. Line* **4**, 220–251 (2014). <https://doi.org/10.5201/ipol.2014.97>
5. Sun, J., Shum, H.Y., Zheng, N.N.: Stereo matching using belief propagation. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2351, pp. 510–524. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-47967-8_34
6. Salehian, B., Fotouhi, A.M., Raie, A.A.: Dynamic programming-based dense stereo matching improvement using an efficient search space reduction technique. *Optik (Stuttg)* **160**, 1–12 (2018). <https://doi.org/10.1016/j.ijleo.2018.01.021>
7. Li, J., et al.: A long baseline global stereo matching based upon short baseline estimation. *Meas. Sci. Technol.* **29**(5) (2018). <https://doi.org/10.1088/1361-6501/aab079>
8. Wang, Z., Yue, J., Han, J., Jin, Y., Li, B.: Regional fuzzy binocular stereo matching algorithm based on global correlation coding for 3D measurement of rail surface. *Optik (Stuttg)* **207**, 164488 (2020). <https://doi.org/10.1016/j.ijleo.2020.164488>
9. Wang, X., Su, Y., Tang, L., Tan, J.: A combined back and foreground-based stereo matching algorithm using belief propagation and self-adapting dissimilarity measure. *Int. J. Pattern Recognit. Artif. Intell.* **32**(6), 1–18 (2018). <https://doi.org/10.1142/S0218001418500192>
10. Pan, C., Liu, Y., Huang, D.: Novel belief propagation algorithm for stereo matching with a robust cost computation. *IEEE Access* **7**, 29699–29708 (2019). <https://doi.org/10.1109/ACCESS.2019.2902249>
11. Peña, D., Sutherland, A.: Disparity estimation by simultaneous edge drawing. In: Chen, C.S., Lu, J., Ma, K.K. (eds.) ACCV 2016. LNCS, vol. 10117, pp. 124–135. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-54427-4_10
12. Katrolia, J.S., Krämer, L., Rambach, J., Mirbach, B., Stricker, D.: Semantic segmentation in depth data: a comparative evaluation of image and point cloud based methods. In: Proceedings of the International Conference on Image Processing, ICIP, September 2021, pp. 649–653 (2021). <https://doi.org/10.1109/ICIP42928.2021.9506334>
13. Chai, Y., Cao, X.: Stereo matching algorithm based on joint matching cost and adaptive window. In: Proceedings of the 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 442–446 (2018). <https://doi.org/10.1109/IAEAC.2018.8577495>
14. Yoon, K.J., Kweon, I.S.: Adaptive support-weight approach for correspondence search. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(4), 650–656 (2006). <https://doi.org/10.1109/TPAMI.2006.70>
15. Jyothi, K., Prabhakar, C.J.: Multi modal face recognition using block based curvelet features. *Int. J. Comput. Graph. Animat.* **4**(2), 21–37 (2014). <https://doi.org/10.5121/ijcga.2014.4203>
16. Hosni, A., Bleyer, M., Gelautz, M., Rhemann, C.: Local stereo matching using geodesic support weights. *Institute for Software Technology and Interactive Systems, Vienna University of Technology, Science*, vol. 80, no. 1 (2009)
17. He, K., Sun, J., Tang, X.: Guided image filtering. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(6), 1397–1409 (2013). <https://doi.org/10.1109/TPAMI.2012.213>
18. Hosni, A., Rhemann, C., Bleyer, M., Rother, C., Gelautz, M.: Fast cost-volume filtering for visual correspondence and beyond. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(2), 504–511 (2013). <https://doi.org/10.1109/TPAMI.2012.156>

19. Yang, Q., Ji, P., Li, D., Yao, S., Zhang, M.: Fast stereo matching using adaptive guided filtering. *Image Vis. Comput.* **32**(3), 202–211 (2014). <https://doi.org/10.1016/j.imavis.2014.01.001>
20. Liu, H., Wang, R., Xia, Y., Zhang, X.: Improved cost computation and adaptive shape guided filter for local stereo matching of low texture stereo images. *Appl. Sci.* **10**(5) (2020). <https://doi.org/10.3390/app10051869>
21. Baek, E.T., Yang, H.J.: Cost aggregation for stereo matching using total generalized variation with fusion tensor. *IEEE Access* **7**, 134505–134513 (2019). <https://doi.org/10.1109/ACCESS.2019.2941215>
22. Huang, C.S., Huang, Y.H., Chan, D.Y., Yang, J.F.: Shape-reserved stereo matching with segment-based cost aggregation and dual-path refinement. *Eurasip J. Image Video Process.* **1**, 2020 (2020). <https://doi.org/10.1186/s13640-020-00525-3>
23. Han, C., Lu, S., Jin, L., Jiang, S., Li, H.: Segment-tree stereo matching algorithm based on improved matching costs. *IAENG Int. J. Comput. Sci.* **47**(3), 550–556 (2020)
24. Qi, J., Liu, L.: The stereo matching algorithm based on an improved adaptive support window. *IET Image Process.* **16**(10), 2803–2816 (2022). <https://doi.org/10.1049/ipr2.12527>
25. Zabih, R., Woodfill, J.: Non-parametric local transforms for computing visual correspondence. In: Eklundh, J.O. (eds.) *ECCV 1994*. LNCS, vol. 801, pp. 151–158. Springer, Heidelberg (1994). <https://doi.org/10.1007/BFb0028345>
26. Bertalmío, M., Bertozzi, A.L., Sapiro, G.: Navier-Stokes, fluid dynamics, and image and video inpainting. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1 (2001). <https://doi.org/10.1109/cvpr.2001.990497>
27. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 195–202 (2003). <https://doi.org/10.1109/cvpr.2003.1211354>
28. Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2007). <https://doi.org/10.1109/CVPR.2007.383191>
29. Hirschmüller, H., Scharstein, D.: Evaluation of cost functions for stereo matching. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2007). <https://doi.org/10.1109/CVPR.2007.383248>
30. Scharstein, D., et al.: High-resolution stereo datasets with subpixel-accurate ground truth. In: Jiang, X., Hornegger, J., Koch, R. (eds.) *GCPR 2014*. LNCS, vol. 8753, pp. 31–42. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11752-2_3
31. Shabanian, H., Balasubramanian, M.: A new hybrid stereo disparity estimation algorithm with guided image filtering-based cost aggregation. In: *IS&T International Symposium on Electronic Imaging Science and Technology 2021*, no. 2 (2021). <https://doi.org/10.2352/ISSN.2470-1173.2021.2.SDA-059>



Self-Healing Networks Leverage Intelligent Protocols Enable Autonomous Fault Detection and Recovery

V. Lakshmi^(✉) and S. M. A. K. Azad^(✉)

School of Electronics Engineering, VIT AP University, Amaravati, Andhra Pradesh, India
kalluru.vl@gmail.com, freedom_smak@rediffmail.com

Abstract. In an era defined by our increasing dependence on networked systems, the resilience and reliability of these networks have emerged as critical considerations. Network outages and disruptions can severely impact businesses, critical infrastructure, and communication. The concept of self-healing networks, underpinned by intelligent protocols, has risen to the forefront of networking technologies. This paper introduces an in-depth exploration of self-healing networks, their capacity to detect and recover from faults autonomously, and the integral role of intelligent protocols. This paper delves into the mechanisms through which intelligent protocols enable autonomous fault detection. It explores real-time monitoring, wherein network conditions are continuously assessed, and anomalies are identified promptly. These anomalies, whether caused by hardware failures, cyberattacks, or unexpected surges in traffic, serve as early warning signals.

Keywords: Self-Healing Networks · Intelligent Protocols · Autonomous Fault Detection · Fault Recovery · Network Resilience · Real-Time Monitoring · Predictive Network Management

1 Introduction

In the digital age, our world is more interconnected than ever before. From critical business operations and healthcare services to personal communications and financial transactions, our lives are increasingly reliant on the seamless and uninterrupted functioning of networked systems. The objective of the paper is to network infrastructure across diverse industries and the increasing challenges posed by network faults and downtime. Recognizing the critical role of self-healing networks in addressing these challenges, the motivation is to explore how intelligent protocols contribute to the autonomy of fault detection and recovery, ultimately enhancing the resilience of networked systems. Recognizing the need for network resilience and fault tolerance, the concept of self-healing networks has emerged as a transformative and indispensable solution. Self-healing networks represent a paradigm shift in the field of networking. They are designed to autonomously detect [1] and recover from faults and disruptions, ensuring the seamless and uninterrupted flow of data and services. In essence, they embody the

concept of networks capable of “healing” themselves, akin to the natural resilience of a living organism. At the core of self-healing networks [2] lies the application of intelligent protocols, and advanced technologies that equip network infrastructure with the capacity to analyze, adapt, and respond to network conditions in real-time. These intelligent protocols serve as the neural pathways of self-healing networks, enabling them to not only identify anomalies and vulnerabilities but also to initiate autonomous recovery processes, thus minimizing the impact of disruptions.

The importance of self-healing networks cannot be overstated in our interconnected world. Their capacity to detect and recover from faults autonomously [3] is not merely a convenience but a strategic imperative, particularly in critical sectors. As industries across the spectrum continue to rely on the operational continuity of network infrastructure, self-healing networks, underpinned by intelligent protocols, become the linchpin of network resilience. They are the key to reducing downtime, mitigating financial losses, maintaining public safety, and ensuring that vital services continue to function seamlessly, even in the face of unforeseen challenges. This paper delves into the intricacies of self-healing networks and the central role played by intelligent protocols in enabling autonomous fault detection and recovery. It explores the fundamental principles, mechanisms, benefits, challenges, and real-world applications of self-healing networks. The subsequent sections will unravel the transformative potential of self-healing networks, the hurdles they face, and the innovative solutions that are paving the way toward a more resilient and reliable networked world.

The pervasive integration of networked systems has been fueled by the ceaseless march of technological advancement, resulting in the creation of a global web of digital interdependence. Whether it's the financial sector conducting high-frequency trading, healthcare professionals relying on telemedicine, manufacturers optimizing their production lines, or individuals navigating the intricacies of daily life through a myriad of online services, the constant and uninterrupted operation of network infrastructure [4] is imperative. Several interconnected forces drive this phenomenon. The globalization of business has rendered efficient and dependable networks a cornerstone of international trade and collaboration, necessitating the establishment of seamless communication channels that transcend geographical boundaries. The omnipresence of data-driven decision-making compels organizations to foster secure and expeditious data exchange, enhancing their capacity to harness the power of information. The ongoing digital transformation, characterized by increased automation and the convergence of physical and digital realms, has precipitated a seismic shift in the way industries operate. The rise of remote work, [5] an enduring legacy of recent global events, and the necessity for robust networks that facilitate remote access to corporate resources support video conferencing, and ensure the continuity of business operations. The proliferation of Internet of Things (IoT) devices has ushered in an era of real-time data collection and analysis from a multitude of interconnected devices, further intensifying the demand for resilient networks capable of supporting these intricate data flows. In the realm of e-commerce and online services, these networks have become the conduits through which the modern consumer engages with the digital world, encompassing everything from online shopping and banking to streaming entertainment. It is against this backdrop of increasing

interconnectedness and reliance on network infrastructure that the imperative for self-healing networks emerges as a strategic necessity. These networks, characterized by their ability to autonomously detect and recover from faults and disruptions, are not just a convenience but a foundational element of network resilience in border efficiency.

2 Need for Self-healing Networks

The need of self-healing networks is interconnected, as industries and sectors across the spectrum increasingly rely on network infrastructure, the consequences of network faults and downtimes extend well beyond mere inconvenience. In our digital age, where data is the lifeblood of business, critical services, and personal communications, network interruptions can have severe financial repercussions, threaten public safety, and disrupt the delivery of essential services. Whether it's the healthcare sector [6] relying on networked systems for patient care or financial institutions executing millions of transactions daily, the cost of network downtime is significant, measured not only in financial terms but also in terms of reputation and customer trust.

The need for autonomous fault detection and recovery in self-healing network mechanisms is driven by the understanding that traditional approaches, often reactive or reliant on manual intervention, are no longer sufficient to ensure the uninterrupted flow of data and services. Self-healing networks, with their inherent ability to autonomously identify and rectify faults, represent a transformative solution to this challenge, assuring the resilience and reliability of our interconnected world.

2.1 Network Infrastructure in Various Industries

The use of network infrastructure has become more and more important across various industries. This is due to the increasing demand for fast and secure communication of data between different devices and systems. As a result, businesses and organizations are relying on network infrastructure to support their operations, boost productivity, and enhance customer experience. Additionally, the rise of the Internet of Things (IoT) has further accelerated the adoption of network infrastructure, as more devices are being connected to the Internet and generating vast amounts of data that need to be transmitted and analyzed in real-time. The need for self-healing network infrastructure is rapidly increasing across various industries as technology continues to evolve. The growing reliance on networked systems is driven by multiple factors discussed in below, which together make these systems pivotal in supporting and transforming a self-healing network.

- (i) **Globalization and International Trade:** The forces of globalization have propelled businesses into the global arena. Companies rely on intricate network systems to establish seamless communication channels across geographical boundaries. These networks facilitate [14] international trade, collaboration, and the exchange of information, allowing organizations to conduct operations on a global scale. In sectors such as international logistics, supply chain management, and global financial markets, the necessity for real-time data exchange is paramount.

- (ii) **Data-Driven Decision-Making:** Organizations across industries have embraced data-driven decision-making as a core strategy for success. Data, [15] often generated and disseminated through networked systems, is the lifeblood of these decisions. From market analysis and customer insights to predictive analytics and resource allocation, the speed and reliability of data transmission through networks is indispensable. In finance, algorithmic trading systems depend on ultra-fast networks to make split-second investment decisions.
- (iii) **Digital Transformation:** The ongoing digital transformation of industries has been a driving force behind the increasing reliance on network infrastructure. Manufacturing, for instance, has embraced the principles of Industry 4.0, where interconnected sensors and devices communicate data in real-time to optimize production processes. Healthcare has witnessed a shift toward [7–10] digital health records and telemedicine, requiring robust network systems to support these innovations. Public services, too, have harnessed the power of networks to improve efficiency and provide better services to citizens.
- (iv) **Telecommuting and Remote Work:** The advent of remote work and telecommuting, further accelerated by recent global events, has magnified the importance of network infrastructure. Organizations rely on networks to maintain a virtual workforce, ensuring employees have secure and efficient access to corporate resources, video conferencing, and collaboration tools. This shift has blurred the lines between the traditional office and remote workspaces, fundamentally altering the way we conduct business and interact.
- (v) **Internet of Things (IoT):** The proliferation of IoT devices has generated an exponential surge in data traffic. These devices, ranging from smart appliances and wearable technology to industrial sensors [11–13] and autonomous vehicles, generate an immense volume of data. Network infrastructure is essential to transmit this data in real-time, enabling monitoring, analysis, and control. In agriculture, for example, IoT-enabled sensors collect data on soil conditions and weather, optimizing crop yields and resource usage.

2.2 Network Faults and Downtime

Network faults and downtime refer to any issues that may lead to a disruption or complete shutdown of a network. These issues can be caused by internal faults, external factors, or a combination of both. As shown in Fig. 1 when a network experiences downtime, it means that users are unable to access network resources, which can result in lost productivity and revenue. Therefore, it is important to take measures to prevent network faults and minimize downtime.

- (i) **Financial Losses:** Network outages and disruptions carry significant financial implications. For businesses, each minute of downtime translates into potential revenue loss. This is especially pertinent in sectors such as e-commerce, where even brief interruptions can result in substantial monetary losses. In the financial industry, [17] where high-frequency trading relies on millisecond-level transactions, network downtime can lead to significant financial setbacks.
- (ii) **Operational Disruption:** Network faults and downtimes can disrupt the operational workflows of businesses and organizations. Manufacturing plants, for instance, rely

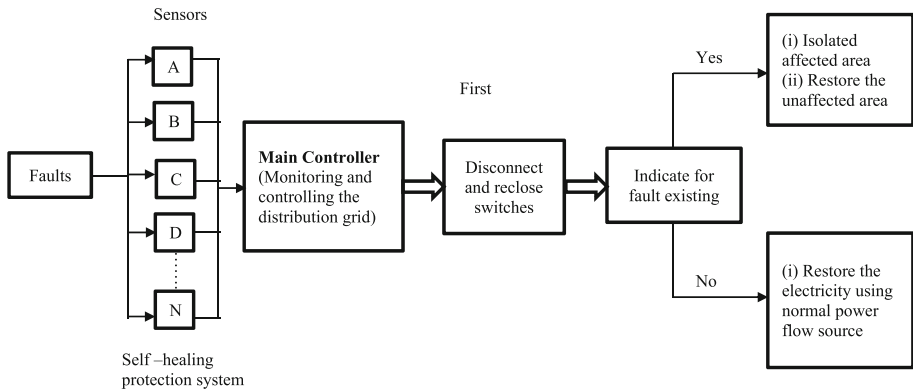


Fig. 1. Block Diagram of Self-Healing Protection System

on networked systems for supply chain management and process automation. When networks fail, it can lead to production stoppages, equipment malfunctions, and delayed deliveries. In healthcare, network disruptions can jeopardize patient care by impeding access to medical records and critical diagnostic information.

- (iii) **Reputation Damage:** Repeated network disruptions can tarnish an organization's reputation. In the era of instantaneous communication and social media, news of network outages can spread rapidly, impacting public perception and trust. Customers and stakeholders may lose confidence in the reliability of a business or service provider. This can result in loss of market share and a damaged brand image.
- (iv) **Data Loss and Security Risks:** Network faults can lead to data loss, which can be costly and have security implications. In scenarios where data is not properly backed up or replicated, network outage may result in permanent data loss. Additionally, network vulnerabilities may be exacerbated during downtime, [14] making systems more susceptible to cyberattacks, data breaches, and unauthorized access. The risk of confidential information exposure becomes particularly acute in industries such as finance, healthcare, and government.
- (v) **Customer Dissatisfaction:** Network disruptions, especially inspectors where customer interactions are vital, can lead to customer dissatisfaction. In e-commerce, for example, customers expect seamless online shopping experiences. Network interruptions can disrupt transactions, leading to abandoned shopping carts and lost sales. In customer service and call centers, network downtime hinders communication and support, leading to frustrated customers and potentially damaged client relationships.
- (vi) **Public Safety Implications:** In sectors like healthcare and emergency services, network disruptions can have direct implications for public safety. Networked systems are crucial for coordinating emergency responses, accessing medical records, and ensuring the rapid delivery of critical services. Any interruption in these networks can compromise public safety and well-being.

3 Intelligent Protocols and Self-healing Networks

Self-healing networks rely on a combination of network protocols and technologies to enable autonomous fault detection and recovery. These protocols and technologies work together to ensure the network's resiliency and continuity. Here are some key types of protocols and technologies commonly used in self-healing networks.

3.1 Routing Protocols

Dynamic Routing Protocols: Protocols like OSPF (Open Shortest Path First) and EIGRP (Enhanced Interior Gateway Routing Protocol) dynamically [15] adjust routing tables to find the most efficient path for data transmission. In the event of a network fault, these protocols can quickly reroute traffic to avoid the affected area.

BGP (Border Gateway Protocol): BGP is essential for routing in large-scale networks. It can detect and adapt to changes in network topology, making it a valuable component of self-healing networks, particularly in scenarios where network segments may go offline or experience disruptions.

3.2 Network Monitoring and Management Protocols

SNMP (Simple Network Management Protocol): SNMP is used for monitoring and managing network devices. It allows for real-time monitoring [16] of device status, performance, and resource utilization. When anomalies are detected, SNMP can trigger alerts or actions for self-healing processes as shown in Fig. 2.

NetFlow: Net Flow is a network protocol used for collecting and monitoring network traffic. It enables the analysis of network data to detect unusual patterns or spikes in traffic, which can be indicative of faults or attacks.

3.3 Security Protocols

Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS): In Fig. 2 shows the security protocols continuously monitor network traffic for suspicious activity. In the case of an intrusion or security breach, IDS and IPS can take action to mitigate the threat and protect the network.

3.4 Quality of Service (QoS) Protocols

Differentiated Services (DSCP): DSCP is a QoS protocol that allows network devices to prioritize traffic based on its importance. In self-healing networks, DSCP can be used to ensure that critical applications and services continue to function even during network faults.

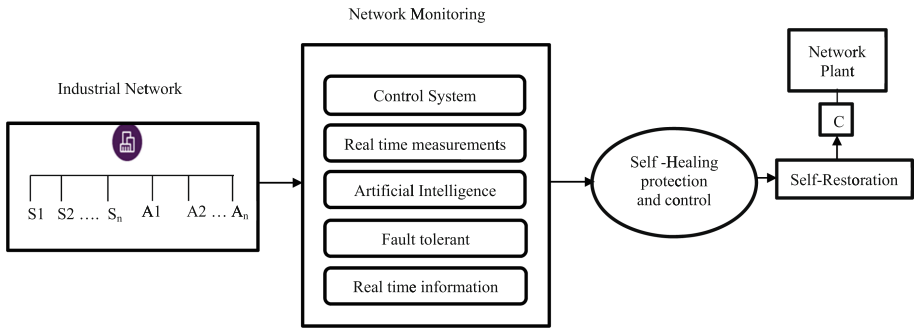


Fig. 2. Self-Healing Network Monitoring

3.5 Virtualization and SDN (Software-Defined Networking)

SDN Controllers: SDN technology allows for centralized network control. SDN controllers can reconfigure network paths and resources in real time, making them a valuable component in self-healing networks. When faults occur, the controller can adapt the network's behavior to maintain service continuity.

Network Function Virtualization (NFV): NFV enables the virtualization of network functions, allowing for rapid provisioning and reconfiguration of network services. This flexibility is crucial for self-healing networks as it enables quick adaptation to changing network conditions. These protocols are designed to enhance network performance, reliability, and security by incorporating elements of artificial intelligence (AI) and advanced algorithms.

Their primary role in self-healing networks is to enable autonomous fault detection and recovery, ensuring network continuity even in the face of disruptions. Intelligent protocols are the neural pathways of self-healing networks, allowing them to analyze network data, adapt to changing conditions, and respond to faults, all in real-time.

3.6 Features and Capabilities of Intelligent Protocols

Intelligent protocols in self-healing networks possess several key features and capabilities that are essential for their effectiveness. Intelligent protocols continuously monitor network conditions, collecting data on performance, traffic patterns, and resource utilization. This real-time monitoring allows for immediate awareness of any anomalies or deviations from normal network behavior.

- a) **Anomaly Detection:** These protocols excel at detecting anomalies within the network. Using advanced analytics and AI algorithms, they identify irregularities or unusual patterns in network data, serving as early warning indicators of potential fault.
- b) **Predictive Analysis:** Intelligent protocols employ predictive analysis to forecast potential network issues. By identifying early warning signs in the form of anomalies, these protocols can proactively address them before they escalate into full-scale network faults.

- c) **Dynamic Routing and Resource Allocation:** In the presence of anomalies or faults, intelligent protocols can autonomously determine the optimal path for network traffic and allocate resources efficiently. This dynamic routing ensures data can circumvent problematic areas, minimizing the impact of the fault.
- d) **Redundancy Management:** Self-healing networks incorporate redundancy mechanisms, and intelligent protocols oversee their management. These protocols ensure that backup resources are seamlessly integrated when primary resources experience faults, guaranteeing the continuity of services.
- e) **Automated Recovery Processes:** Intelligent protocols are responsible for triggering automated recovery processes upon detecting a fault or anomaly. These processes, such as switching to redundant resources, [17, 18] isolating compromised segments, or reconfiguring network paths, often occur with minimal to no manual intervention rapid recovery.
- f) **Machine Learning and Adaptation:** Intelligent protocols leverage machine learning to adapt and improve their fault detection and recovery capabilities over time. They learn from past incidents [19] and feedback, continuously enhancing their responses to network issues.
- g) **Security Enhancement:** Beyond fault detection and recovery, intelligent protocols incorporate robust security measures to identify and respond to security breaches, enhancing network security and resilience.

3.7 How Intelligent Protocols Contribute to Fault Detection and Recovery

Intelligent protocols play a central role in the fault detection and recovery processes in self-healing networks. They are designed to be smart and efficient, which helps in identifying anomalies and implementing corrective actions. By using these protocols, businesses can reduce downtime and maintain optimal performance levels.

- Through continuous real-time monitoring, they collect data on network conditions, performance, and anomalies.
- By analyzing the data, they can detect and flag anomalies and deviations from normal network behavior.
- Intelligent protocols leverage predictive analysis to identify potential network issues and take proactive measures.
- They ensure dynamic routing and resource allocation, optimizing the flow of network traffic during faults.
- Redundancy mechanisms, overseen by these protocols, are automatically activated when faults are detected, ensuring service continuity. Automated recovery processes are triggered by intelligent

They adapt and learn from past incidents, continuously improving their fault detection and recovery capabilities. Protocols, allowing for swift resolution of network issues. Intelligent protocols also enhance network security by identifying and mitigating security threats.

4 Autonomous Fault Detection

Autonomous fault detection is a critical component of self-healing networks, wherein network systems and intelligent protocols are designed to identify and respond to faults and anomalies without requiring manual intervention. It plays a pivotal role in ensuring the uninterrupted operation of networked systems. As shown in Fig. 3 an overview of autonomous fault detection in self-healing networks.

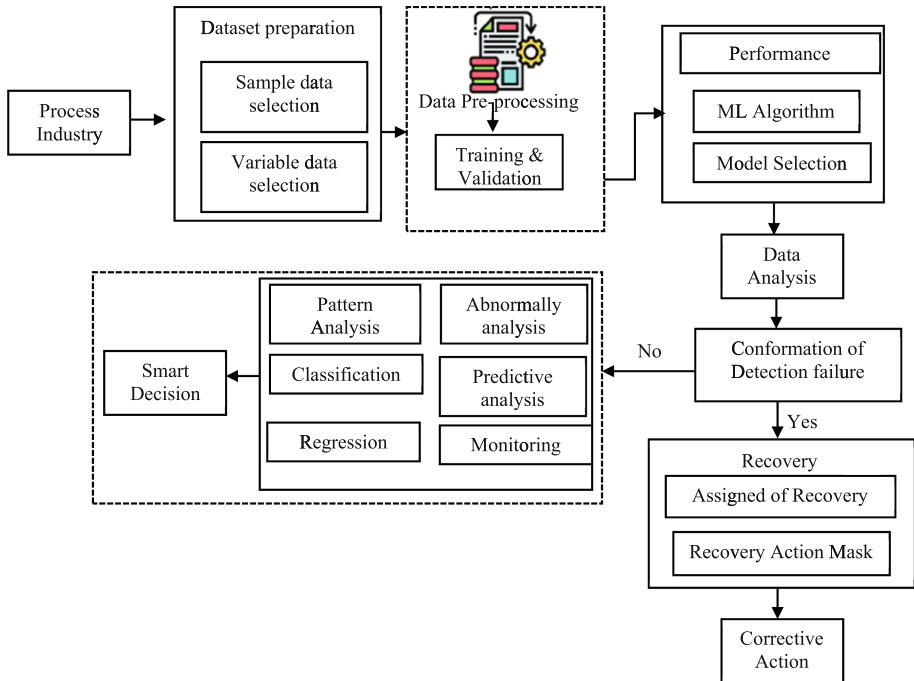


Fig. 3. Industrial Process of Autonomous Fault Detection and Recovery

4.1 Mechanisms of Autonomous Fault Detection

- (i) **Real-Time Monitoring::** Autonomous fault detection begins with real-time monitoring of network conditions. Advanced sensors, monitoring tools, and intelligent protocols continuously collect data on network performance, traffic patterns, and resource utilization. This real-time data forms the foundation for detecting anomalies.
- (ii) **Anomaly Detection:** The collected data is subjected to advanced analytics and machine learning algorithms that can identify anomalies. These anomalies can stem from various sources, including hardware failures, software glitches, cyberattacks, or unexpected surges in network traffic. The algorithms compare current data with historical patterns to detect deviations.

- (iii) **Predictive Analysis:** Autonomous fault detection goes beyond identifying existing anomalies; it involves predictive analysis. Machine learning models and intelligent protocols forecast potential issues by identifying early warning signs in the form of anomalies. This proactive approach allows for pre-emptive measures to be taken before anomalies escalate into network faults.

4.2 Significance of Autonomous Fault Detection

- (i) **Minimizing Downtime:** One of the primary benefits of autonomous fault detection is the minimization of downtime. By identifying anomalies and potential faults in their early stages, self-healing networks can take swift actions to mitigate issues before they disrupt network operations. This proactive approach ensures that network services remain accessible and functional.
- (ii) **Reducing Operational Costs:** Autonomous fault detection reduces operational costs associated with manual [20] fault identification and resolution. Human intervention is no longer required for routine fault detection, allowing IT personnel to focus on strategic tasks and more complex issues.
- (iii) **Enhancing Network Resilience:** Self-healing networks are inherently more resilient due to autonomous fault detection. They can adapt to changing conditions and take corrective actions in real-time, ensuring that network services are maintained even in challenging scenarios.
- (iv) **Improving Security:** Autonomous fault detection isn't limited to technical faults; it also includes the identification of security breaches and cyber threats. By recognizing unusual patterns of network traffic or unauthorized access attempts, self-healing networks can enhance their overall security posture.
- (v) **Optimizing Resource Utilization:** Early fault detection allows self-healing networks to optimize resource utilization. They can allocate resources efficiently, balance network traffic, and reroute data to prevent bottlenecks, ensuring that the network operates at its best capacity.
- (vi) **Boosting User Satisfaction:** For businesses and services that rely on network connectivity, autonomous fault detection contributes to improved user satisfaction. Network disruptions are kept to a minimum, ensuring that customers, clients, and end-users can access services without interruptions. Autonomous fault detection is a fundamental component of self-healing networks, enabling them to swiftly identify, respond to, and mitigate potential faults and anomalies. This capability not only reduces downtime and operational costs but also enhances network resilience, security, and the overall user experience. Autonomous fault detection empowers self-healing [25, 26] networks to proactively safeguard the continuity of operations, services, and data, making them.

5 Autonomous Fault Recovery

Autonomous Fault Recovery refers to the ability of a system to detect and recover from faults or errors without human intervention. It is a self-healing process that allows a system to continue functioning even in the event of a failure. A fault detection and recovery system involves the process of identifying and diagnosing faults or errors in a system

and taking appropriate actions to restore the system [3] to its normal operation. Fault detection and recovery mechanisms are designed to minimize the impact of faults on the overall system performance and ensure the system's reliability and availability. These mechanisms can include various strategies such as redundancy, error correction, and fault tolerance. Self-healing networks represent a paradigm shift in network management, offering the capability to autonomously detect and recover from faults, anomalies, and disruptions. In [21] at the core of their functionality are intelligent protocols and sophisticated mechanisms that transform traditional network infrastructure into adaptive, resilient, and self-aware entities. Here's a closer look at how self-healing networks contribute to fault detection and recovery.

As shown in Fig. 3 Self-healing networks leverage predictive analysis to forecast potential issues before they lead to full-scale faults or disruptions. If confirmation of detection failure is yes, the data comes into the recovery process, here what fault occurs in the event of detected anomalies or faults, self-healing networks employ dynamic routing to redirect network traffic and circumvent problematic areas. Here Redundancy mechanisms are automatically activated when faults are detected, maintaining service continuity and taking corrective action.

5.1 Automated Recovery Processes

Self-healing networks go beyond fault detection to provide automated recovery processes. Upon detecting a fault or anomaly, [22] intelligent protocols trigger predefined actions, such as switching to redundant resources, isolating compromised segments, or reconfiguring network paths. These processes often occur with minimal to no manual intervention, ensuring rapid recovery.

6 Challenges and Limitations

6.1 False Positives

One common challenge is the occurrence of false positives. Autonomous fault detection mechanisms may occasionally misidentify normal network behavior as anomalies, triggering unnecessary recovery actions [23, 24].

Limitation: False positives can lead to network instability and resource wastage as the network takes action based on incorrect alerts.

6.2 Scalability

As networks grow in size and complexity, ensuring that self-healing mechanisms remain effective at scale becomes challenging. Scalability issues can result in slower fault detection and recovery times in large and distributed networks.

Limitation: Self-healing networks may not always scale seamlessly to accommodate extensive infrastructure.

6.3 Security Concerns

While self-healing networks enhance security by detecting and mitigating threats, they can also be vulnerable to attacks. An attacker who compromises the autonomous recovery process can cause disruptions or evade detection.

Limitation: Self-healing networks must strike a balance between automation and security to prevent misuse or exploitation by malicious actors.

6.4 Complexity and Management

The design, implementation, and ongoing management of self-healing networks can be complex and require specialized expertise.

Limitation: Smaller organizations or those with limited resources may find it challenging to fully adopt and manage self-healing network technologies.

6.5 Adaptation to New Threats

Self-healing networks must continually adapt to new and evolving threats and vulnerabilities. Staying ahead of emerging risks and vulnerabilities requires constant updates and fine-tuning.

Limitation: The network's ability to autonomously adapt to unforeseen threats may be limited by the availability of updated threat intelligence and response mechanisms.

7 Conclusion

The transformative realm of self-healing networks underscored the indispensable role of intelligent protocols in shaping their capabilities. Self-healing networks, driven by real-time monitoring, predictive analysis, and adaptive responses, stand as a beacon of resilience, proactively safeguarding network continuity.

The significance of intelligent protocols cannot be overstated, as they empower networks to anticipate, detect, and mitigate faults, thereby minimizing downtime and enhancing security. As we look to the future, the call for further research and development in this field resonates loudly, encouraging the pursuit of advanced solutions to tackle false positives, scalability concerns, and security.

vulnerabilities. The self-healing networks are ongoing, promising a future where network faults are not setbacks but mere ripples in the seamless flow of data and services.

References

1. Shahnia, F., et al.: Primary control level of parallel distributed energy resources converters in system of multiple interconnected autonomous microgrids within self-healing networks. *IET Gener. Transm. Distrib.* **8**(2), 203–222 (2014)

2. Trehan, A.: Algorithms for self-healing networks. Dissertation. The University of New Mexico (2010)
3. Quattrociochi, W., Caldarelli, G., Scala, A.: Self-healing networks: redundancy and structure. *PLoS ONE* **9**(2), e87986 (2014)
4. Yang, Y., Urban, M.W.: Self-healing polymeric materials. *Chem. Soc. Rev.* **42**(17), 7446–7467 (2013)
5. Trask, R.S., Williams, H.R., Bond, I.P.: Self-healing polymer composites: mimicking nature to enhance performance. *Bioinspiration Biomimetics* **2**(1), P1 (2007)
6. Ma, Y., et al.: A dynamic affinity propagation clustering algorithm for cell outage detection in self-healing networks. In: *Proceedings of the 2013 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE (2013)
7. Liu, T., et al.: Extremely strengthening fatigue resistance, elastic restorability and thermodynamic stability of a soft transparent self-healing network based on a dynamic molecular confinement-induced bioinspired nanostructure. *Mater. Horiz.* (2023)
8. Wei, Z., et al.: Dual cross-linked biofunctional and self-healing networks to generate user-defined modular gradient hydrogel constructs. *Adv. Healthc. Mater.* **6**(16), 1700523 (2017)
9. Minnebo, P., et al.: A novel design of autonomously healed concrete: towards a vascular healing network. *Materials* **10**(1), 49 (2017)
10. Palacios, D., Khatib, E.J., Barco, R.: Combination of multiple diagnosis systems in self-healing networks. *Expert Syst. Appl.* **64**, 56–68 (2016)
11. Strachota, B., et al.: Control of gelation and properties of reversible Diels–Alder networks: design of a self-healing network. *Polymers* **11**(6), 930 (2019)
12. Barco, R., Lazaro, P., Munoz, P.: A unified framework for self-healing in wireless networks. *IEEE Commun. Mag.* **50**(12), 134–142 (2012)
13. Shields, Y., et al.: A review of vascular networks for self-healing applications. *Smart Mater. Struct.* **30**(6), 063001 (2021)
14. Arunbabu, D., et al.: Thermoreversible self-healing networks based on a tunable poly-methacrylate crosslinker having pendant maleimide groups. *Macromol. Chem. Phys.* **217**(19), 2191–2198 (2016)
15. Zhang, H., et al.: Low-maintenance superamphiphobic coating based on a smart two-layer self-healing network. *Surf. Coat. Technol.* **331**, 97–106 (2017)
16. Moazzen, K., et al.: Toward poly (furfuryl alcohol) applications diversification: Novel self-healing network and toughening epoxy–novolac resin. *J. Appl. Polym. Sci.* **135**(12), 45921 (2018)
17. Gómez-Andrades, A., et al.: Methodology for the design and evaluation of self-healing LTE networks. *IEEE Trans. Veh. Technol.* **65**(8), 6468–6486 (2015)
18. Dhinnesh ADC, N., Sundareswaran, N.: An efficient self-healing network through quadratic probing optimization mechanism. *Int. J. Commun. Syst.* **35**(7), e5098 (2022)
19. Zhang, D.-G., Zhou, S., Tang, Y.-M.: A low duty cycle efficient MAC protocol based on self-adaption and predictive strategy. *Mob. Netw. Appl.* **23**, 828–839 (2018)
20. Varalakshmi, K., Azad, S.M.A.K.: A study on reinforced support vector machine classifier for periodic and aperiodic scheduling in networked control systems. In: *Proceedings of the 2023 International Conference on Smart Systems for Applications in Electrical Sciences (ICSSES)*. IEEE (2023)
21. Mottaghinia, Z., Ghaffari, A.: Fuzzy logic based distance and energy-aware routing protocol in delay-tolerant mobile sensor networks. *Wireless Pers. Commun.* **100**, 957–976 (2018)
22. Li, F., et al.: FarpScusn: Fully anonymous routing protocol with self-healing capability in unstable sensor networks. *Sensors* **20**(22), 6683 (2020)
23. Zhang, D., et al.: An energy-balanced routing method based on forward-aware factor for wireless sensor networks. *IEEE Trans. Ind. Inf.* **10**(1), 766–773 (2013)

24. Zhang, D., et al.: Novel unequal clustering routing protocol considering energy balancing based on network partition & distance for mobile education. *J. Netw. Comput. Appl.* **88**, 1–9 (2017)
25. Zhang, T., et al.: A new method of data missing estimation with FNN-based tensor heterogeneous ensemble learning for internet of vehicle. *Neurocomputing* **420**, 98–110 (2021)
26. Zhang, D.-G., et al.: A multi-path routing protocol based on link lifetime and energy consumption prediction for mobile edge computing. *IEEE Access* **8**, 69058–69071 (2020)



A Secure System for Decentralized Preservation of Digital Library Collections Using Private Blockchain and Interplanetary File System (IPFS)

Patrick Kipkorir Laboso^{1(✉)}, A. Martin¹, and P. Thiagarajan²

¹ Department of Computer Science, Central University of Tamil Nadu, Thiruvavur 610005, India
labosopatrick95@gmail.com, martin@cutn.ac.in

² Department of Computer Science, Rajiv Gandhi National Institute of Youth Development,
Nemili 602105, India
thiyagu.phd@gmail.com

Abstract. Digital libraries are critical in preserving and providing access to vast cultural and academic information. However, traditional preservation methods face challenges to data integrity, authenticity, and storage access. This research explores blockchain technology and the Interplanetary File System's potential to address security and privacy issues of digital library preservation. By leveraging blockchain's decentralized, immutable nature and smart contracts, digital libraries can achieve improved security, transparency, and accountability in their preservation. Furthermore, it presents a platform built on an elliptic curve digital signature algorithm (ECDSA) and zero-knowledge proofs cryptographic to boost the security system communication. Lastly, this framework will support a scalable double-blockchain by using flexible consensus algorithms to prevent attacks.

Keywords: Digital libraries · Preservation · Private Blockchain · Smart contracts · Interplanetary File System · Privacy

1 Introduction

Digital libraries have revolutionized how information is accessed, shared, and stored in the digital age. These cultural and academic heritage repositories provide unprecedented global knowledge dissemination and access opportunities [1]. However, the preservation of digital library collections presents numerous challenges that require innovative solutions to ensure the integrity, authenticity, and long-term accessibility of stored information. Traditional digital library preservation methods heavily rely on centralized systems, posing inherent data security and integrity vulnerabilities [2]. Factors such as data tampering, the potential for single points of failure, and the need for continuous monitoring and maintenance have raised concerns regarding the long-term preservation of digital content [3]. In light of these challenges, there is a growing need for transformative technologies that can enhance the preservation practices of digital libraries.

One such transformative technology is blockchain, which has gained significant attention due to its decentralized and immutable nature. Originally introduced as the underlying cryptocurrency technology, blockchain has demonstrated its potential to revolutionize various industries by providing secure, transparent, and tamper-resistant systems [4]. The core features of blockchain, such as decentralization, immutability, and transparency, make it an attractive candidate for addressing the challenges digital libraries face in preservation [5]. In conjunction with blockchain, smart contracts offer automation and programmability, further enhancing the potential of this technology for digital library preservation. Smart contracts are self-executing agreements with predefined conditions, which can automate various aspects of preservation processes [6]. They provide a means to enforce trust and accountability among the stakeholders involved in the preservation ecosystem.

This research paper aims to explore how to build a strong, secure system for decentralized preservation of digital library collections. By leveraging the decentralized and immutable nature of blockchain and the automation capabilities of smart contracts, digital libraries can strengthen data integrity, improve authenticity verification, and ensure long-term access to their collections. Through an analysis of case studies, an evaluation of benefits, limitations, and an examination of future directions of prior works, this research seeks to contribute to understanding how blockchain and smart contracts can transform digital library preservation practices. In the subsequent sections, we will delve into the foundational concepts of blockchain technology and interplanetary file system and how to authenticate, secure, and preserve data and materials for long-time access [7]. Lastly, interoperability is currently a key challenge for digital preservation in libraries, as digital assets are stored and managed across different platforms and systems. Blockchain technology facilitates interoperability by creating a common platform for sharing and accessing digital assets and providing a verifiable record of ownership and provenance that can be shared across different systems.

Main Contributions: This paper incorporates the following contributions-

1. We present an approach to secure the storage of files in the digital library storage system
2. This work incorporates the interoperability of the different e-library systems
3. We present an experimental evaluation of two widely used technologies (Blockchain and IPFS)
4. This work critically claims the scope built on an elliptic curve digital signature algorithm to improve speed and protection

The paper is organized as follows: Sect. 2 illustrates the related work, Sect. 3 addresses the proposed security framework, Sect. 4, Illustrates the experimental setup implementation, Sect. 5 conclusion.

2 Related Work

In modern times as libraries across the world move away from the traditional system to digital systems, much research work has been conducted to foster these changes. [2] proposed a system called LibBlock for integrating smart contracts and IPFS to provide

a flexible, decentralized, and adaptive e-Library, which enables the ease of scalability and record keeping. This technology addresses the challenges faced by many libraries across the world in the digital age but fails to secure the system with robust security mechanisms to protect against malicious attacks and authorized access. Our research helps to progress the migration to the new digital age by proposing a secure system for the decentralized preservation of digital library collections using private blockchain and interplanetary file system (IPFS) with polkadot, and ECDSA, a complex public key cryptography encryption algorithm, and zero-knowledge proof to provide security to malicious attacks.

2.1 Polkadot

Polkadot is a scalable multi-chain platform that aims to enable interoperability and connectivity between different blockchain systems in a secure, decentralized, and governable manner. Its unique architecture that separates the consensus mechanism (relay chain) from the execution environment (para chains) addresses the limitations and challenges faced by existing blockchain networks, such as scalability and security. The relay chain is the core of the Polkadot network that is responsible for network consensus and security. It uses a unique mechanism called Nominated Proof-of-Stake (NPoS) for block production and finality. Para-chains are independent set inter-chain that works in the Polkadot network. They possess their own consensus mechanisms, governance models, and token economies. These enhance security and interoperability in the entire system. Finally, bridges part of the main Polkadot allow communication and transaction between the Polkadot network and other external systems.

2.2 Elliptic Curve Digital Signature Algorithm (ECDSA)

ECDSA is a complex public key cryptography encryption algorithm. It is based on the mathematics of elliptic curves and provides a secure way to authenticate and verify the integrity of digital data. It generates two sets of cryptographic keys: a private key and a corresponding public key. The public key is derived from the private key using mathematical operations on an elliptic curve, while the private key is a randomly chosen secret value that should be kept secret [8]. For verification and authenticity, the ECDSA signs a message using public-key infrastructure systems to create the digital signature certificate for the data with non-repudiation property [9]. This provides a strong level of security while using shorter key lengths with more resistance to attacks from quantum computers compared to other traditional cryptographic algorithms like RSA, making it more efficient for resource-constrained environments.

2.3 Zero-Knowledge Proof (ZKP)

In a zero-knowledge proof, the prover aims to convince the verifier of the truthfulness of a statement without revealing any specific information about how they arrived at that knowledge. Where by the verifier gains no knowledge about the secret itself or any other useful information related to it. A decentralized preservation system for digital

library collections, leveraging a private blockchain and the Interplanetary File System (IPFS), can be enhanced with the integration of zero-knowledge proofs for robust security and privacy. Zero-knowledge proofs enable users to interact with the system without disclosing sensitive information while ensuring the verifiability of various operations. Through these proofs, users can prove their identities and ownership of digital assets without revealing personal details, ensuring privacy and preventing unauthorized access [10]. Additionally, zero-knowledge proofs facilitate content authentication, data integrity verification, and secure smart contract execution, bolstering the reliability and trustworthiness of the decentralized library. By incorporating these proofs, the system can also enable selective data sharing, confidential document verification, and anonymous content distribution, allowing users to control the information they reveal and enhancing the privacy of interactions within the network. As a result, the combined use of private blockchain, IPFS, and zero-knowledge proofs establishes a secure and efficient system for decentralized digital library preservation, fostering a trusted environment for preserving and accessing digital collections while safeguarding user privacy.

3 Security Framework

In this paper, we proposed a multi-chain secured resolution based on blockchain technology and Polkadot. The blockchain-based digital preservation system is a sophisticated solution designed to store and maintain digital content for the long term securely. The system employs blockchain technology, encryption, decentralization, access control, and consensus mechanisms to ensure the integrity, immutability, and accessibility of preserved data [11]. At its core, the system allows users to submit digital content, such as files, documents, or media, to the preservation network. Before storing the content, it undergoes encryption using robust elliptic curve digital signature algorithms, guaranteeing its security and integrity. The encrypted content is then divided into smaller data units called blocks. These blocks and their corresponding cryptographic hashes are stored on a blockchain network. The blockchain acts as an immutable ledger, with each block containing the previous block's hash, thus creating a chain of interconnected data.

3.1 Multi-Factor Authentication (MFA)

In our secure system for the decentralized preservation of digital library collections, which utilizes private blockchain and Interplanetary File System (IPFS), we have implemented Multi-Factor Authentication (MFA) to enhance security. When users register and create accounts, they provide basic information such as username, email, and password. The traditional password serves as the first authentication factor. Once the password is successfully entered, users are prompted to provide a second factor for authentication. This second factor can be a one-time password (OTP) sent to their registered mobile device or email, biometric authentication, push notifications, or smart cards as shown in Fig. 1 [12]. To prevent lockouts, we offer recovery mechanisms, allowing users to set up backup methods such as backup codes or security questions. Session management ensures active user sessions have an expiry time and requires periodic re-authentication. All authentication attempts are logged for audit purposes, and secure communication

channels are employed to transmit sensitive data. Regular monitoring and updates keep the MFA implementation robust and resilient against potential threats. With MFA in place, we safeguard user accounts and valuable data, ensuring a high level of security for our decentralized preservation system.



Fig. 1. Multi-Factor Authentication

3.2 Access Control

Access control is a security mechanism that regulates and manages who can access specific resources, data, or services. We implement Role-Based Access Control (RBAC) which serves as a vital component of our security strategy. RBAC is a systematic approach to access control that revolves around defining roles based on users’ responsibilities and associating permissions with those roles rather than assigning permissions directly to individual users. First, we identify distinct roles, such as administrators, librarians, researchers, and visitors, each tailored to specific tasks and functions within the system [13]. Users are then assigned to appropriate roles based on their job requirements. Next, we define a set of permissions, encompassing actions like read, write, execute, and more, essential for various tasks related to the digital library collections. By mapping roles to their corresponding permissions, we establish a structured access control framework. When users attempt to access resources, the system enforces access control based on their assigned roles, granting or denying access accordingly. Regular auditing and monitoring ensure that access activities are scrutinized for unauthorized attempts or potential security breaches. We maintain the RBAC model over time, adapting role assignments to match evolving system requirements. Our comprehensive RBAC implementation, accompanied by documentation and user training, ensures precise and efficient access control, bolstering the security of our decentralized preservation system.

3.3 Distributed Denial of Service (DDoS)

Our implementation of DDoS Protection involves a comprehensive set of strategies and tools to mitigate the impact of disruptive DDoS attacks. We employ traffic monitoring

and analysis tools to detect abnormal traffic patterns indicative of DDoS activity. To control incoming traffic, rate limiting and traffic shaping mechanisms are implemented to manage the flow of legitimate requests. A Web Application Firewall (WAF) helps filter out malicious traffic and protect against application-layer attacks [14]. IP filtering and blacklisting are used to block traffic from known malicious sources. Additionally, we leverage Content Delivery Networks (CDNs) to distribute content across multiple servers, reducing the impact of DDoS attacks by distributing the traffic load [15]. Load balancing ensures that traffic is evenly distributed across servers, maintaining system availability during attacks. For larger-scale attacks, we collaborate with ISPs to implement BGP rate limiting or any-cast routing, dispersing traffic across multiple data centers. We may also introduce CAPTCHA or challenge-response mechanisms to differentiate between human users and bots. Redundancy and scalability are inherent in our system design, ensuring the system can withstand DDoS attacks without compromising availability. An incident response plan outlines protocols and procedures for responding to DDoS incidents swiftly and effectively. By implementing these robust DDoS Protection measures, our Secure System for Decentralized Preservation can maintain uninterrupted availability, optimal performance, and resilience against DDoS attacks, safeguarding the valuable digital library collections and ensuring a secure and reliable user experience.

4 Implementation

The process begins in the blockchain by hashing the data or any other digital library materials to create a unique identifier for the content. This hash is then used as the basis for interacting with the content throughout the process. The hashed content is uploaded to the IPFS network, which breaks it into chunks, assigns hashes to each chunk, and distributes them across a decentralized network. Then on the blockchain, a record is created to store the IPFS hash of the uploaded content. This record is typically stored as a transaction. And then, before the record is added to the blockchain, the creator of the record (perhaps the digital library or content owner) signs the transaction with their private ECDSA key. This creates a digital signature that attests to the authenticity and integrity of the record. The signature proves the creator authorized this record's addition and associated content hash. Furthermore, the content creator or librarian must complete a multi-factor Authentication process. This could involve providing a password along with an OTP code from a mobile app, adding an extra layer of verification. After all the required steps are approved the content is stored on the IPFS.

The proposed system's access to initiate such transactions could be limited to specific roles or users within the organization, such as administrators or designated content creators. This way, only authorized personnel can initiate preservation transactions. The blockchain's immutability ensures that the content's IPFS hash record and the associated digital signature remain secure. Once added to the blockchain, the record and signature cannot be altered without invalidating the signature, which is cryptographically linked to the original content.

Library users who want to access the preserved content can query the blockchain for the IPFS hash. Upon obtaining the hash, they can retrieve the content directly from the IPFS network. Then using selective disclosure of access rights in ZKPs, users prove that

they have access rights to certain content without revealing the actual content, the access permissions, or their identity. This way, users can validate their eligibility to access the content without exposing unnecessary information. Depending on the roles defined within the RBAC system, certain users might have read-only access to the content, while others might have the ability to modify metadata or perform other actions. RBAC ensures that only authorized users can interact with the preserved materials.

4.1 Polkadot System

Polkadot, a revolutionary multi-chain blockchain platform, redefines the landscape of decentralized systems by enabling seamless interoperability and communication among different blockchain networks [16]. The Polkadot ecosystem revolves around the Relay Chain, which coordinates the entire network's operation. The Relay Chain assumes the vital role of securing the system through the Nominated Proof-of-Stake (NPoS) consensus algorithm, where validators are elected by DOT token holders to produce and validate blocks. Surrounding the Relay Chain are individual blockchains called para-chains. Each para-chain operates independently and can employ its consensus mechanism, providing an opportunity for specialization to cater to various use cases and applications. This design allows para-chains to be highly customizable.

The seamless communication and data transfer between para-chains is facilitated by Cross-Chain Message Passing (XCMP). XCMP allows para-chains to interact with each other in a secure and trustless manner, enabling a seamless exchange of assets and information across the network as shown in Fig. 2. This inter-chain interoperability is a game-changer, fostering collaboration and harmony among various Private blockchains. Polkadot extends beyond its network through the use of bridges, which external IPFS networks will be connected through it. Bridges act as connectors, enabling assets and data to flow between blockchain and IPFS.

Blockchains or networks can be connected to the Polkadot ecosystem. Bridges act as connectors, enabling assets and data to flow between different blockchain ecosystems. This connectivity enhances Polkadot's potential and expands its influence to include external blockchain communities. A decentralized governance framework governs the evolution of the Polkadot network. Stakeholders can propose and vote on upgrades, parameter changes, and the addition or removal of para-chains. This democratic governance ensures that the network evolves based on the consensus of its community, empowering participants to shape the security of the system.

4.2 The IPFS Integration with the Blockchain

Integrating IPFS handles the actual storage and distribution of the content chunks across its network of nodes. The IPFS hashes serve as unique identifiers for these chunks, allowing them to be efficiently retrieved and assembled. The IPFS network ensures that the content is distributed across multiple nodes, promoting data redundancy and availability—Polkadot as a multi-chain network that enables different blockchains to interoperate and share information as shown in Fig. 2. To integrate Blockchain and IPFS, a para-chain on the Polkadot network is dedicated to managing and facilitating the communication between the blockchain and IPFS for data storage and access. The

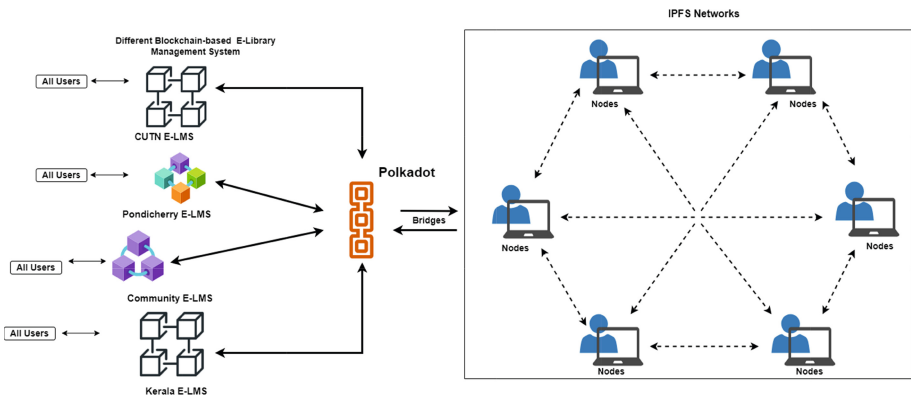


Fig. 2. Integration blockchain and IPFS.

Polkadot para-chain hosts smart contracts or logic that manage metadata related to the preserved content, such as ownership, timestamps, and access permissions. The IPFS reference (hash) for each preserved content can also be stored in the para-chain’s storage.

Integrating the Inter-Planetary File System (IPFS) with a blockchain-based digital preservation system offers significant advantages for storing and retrieving preserved content. IPFS is a decentralized file storage protocol in this system that enables efficient and secure content distribution across multiple nodes [17]. When constructing a blockchain-based digital preservation system, content is not directly stored on the blockchain itself. Instead, the system generates hash pointers that represent the content on IPFS. These hash pointers and relevant metadata are recorded on the blockchain, establishing permanent and immutable references to the preserved data as shown in Fig. 2.

IPFS breaks down the content into smaller chunks and distributes them across a network of IPFS nodes as shown in Fig. 3 [18]. Each node stores a copy of some or all of the content chunks, ensuring redundancy and availability [19]. The content is identified by unique cryptographic hashes, making it highly secure and tamper-resistant. When a user requests preserved content, the system retrieves the corresponding hash pointer from the blockchain. With this hash pointer, the IPFS network locates the distributed content chunks and reconstructs the original data. This process ensures the integrity and authenticity of the content, as it can be verified by comparing its hash with the retrieved data.

IPFS’s decentralized nature contributes to the system’s robustness. If specific nodes become unavailable, other nodes in the network can still provide access to the content, guaranteeing its availability and reliability. This redundancy is significant for long-term preservation, as it minimizes the risk of data loss or unavailability. Moreover, IPFS employs content-based addressing, where content is retrieved based on its unique hash rather than its location. This eliminates the need for centralized servers or specific file paths, making content retrieval more efficient and independent of traditional hierarchical file systems. Integrating IPFS with a blockchain-based digital preservation system leverages both technologies’ strengths. The blockchain ensures the immutability and

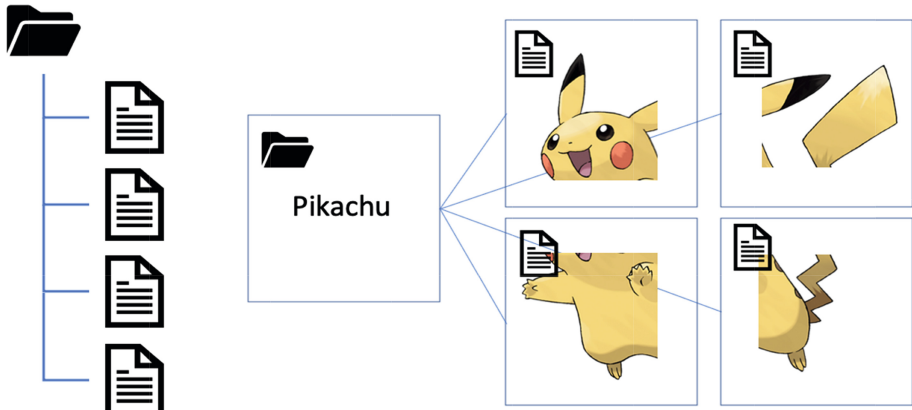


Fig. 3. Files breakdown into small chunks

integrity of the references to preserved content, while IPFS provides a decentralized and redundant storage infrastructure [17]. Together, they create a powerful framework for preserving and accessing digital materials securely, reliably, and scalable.

5 Conclusion

In conclusion, the successful implementation of the Secure System for Decentralized Preservation of Digital Library Collections is a result of the seamless integration of cutting-edge technologies, including Private Blockchain, Interplanetary File System (IPFS), Polkadot (ECDSA), and Zero-knowledge proof (ZKP). This sophisticated system represents a significant advancement in the domain of decentralized data preservation, effectively addressing critical security challenges and ensuring the confidentiality, integrity, and availability of valuable digital library collections. The incorporation of Private Blockchain brings forth the benefits of a robust and immutable ledger, securing the metadata and transaction records associated with the preserved digital assets. Additionally, the integration of IPFS introduces a decentralized and distributed storage solution, guaranteeing seamless access to the actual content of the digital library collections. This not only enhances data redundancy and availability but also reinforces the system's resilience against data loss. The inclusion of Polkadot has further elevated the security of the preservation system by enabling cross-chain interoperability and shared security among diverse para-chains. This innovative approach fosters collaboration and teamwork across different blockchain ecosystems, positioning the system as an integral part of a broader, interconnected network.

References

1. Bhatia, S., De Hernandez, A.D.W.: Blockchain is already here. what does that mean for records management and archives? *J. Arch. Organ.* **16**(1), 75–84 (2019). <https://doi.org/10.1080/15332748.2019.1655614>

2. Chiu, W.Y., Meng, W., Li, W.: LibBlock - towards decentralized library system based on blockchain and IPFS. In: Proceedings of the 2021 18th International Conference on Privacy, Security and Trust (PST) 2021, pp. 1–9 (2021). <https://doi.org/10.1109/PST52912.2021.9647821>
3. Bashir, F., Warraich, N.F.: Future libraries' blockchain opportunities and challenges: a systematic literature review and research agenda. *Digit. Libr. Perspect.* (2023). <https://doi.org/10.1108/DLP-08-2022-0065>
4. Nakamoto, S.: Bitcoin : A Peer-to-Peer Electronic Cash System (2009). Cryptography Mailing list at <https://metzdowd.com>
5. Bralić, V., Stančić, H., Stengård, M.: A blockchain approach to digital archiving: digital signature certification chain preservation. *Rec. Manag. J.* **30**(3), 345–362 (2020). <https://doi.org/10.1108/RMJ-08-2019-0043>
6. Bodorik, P., Liu, C.G., Jutla, D.: TABS: Transforming automatically BPMN models into blockchain smart contracts. *Blockchain Res. Appl.* **4**(1), 100115 (2023). <https://doi.org/10.1016/j.bcr.2022.100115>
7. Masenya, T.M.: Application of modern technologies in the management of records in public libraries. *J. South African Soc. Arch.* **53**, 65–79 (2020). <https://doi.org/10.4314/jsasa.v53i1.5>
8. Ali, S., Wang, G., White, B., Cottrell, R.L.: A blockchain-based decentralized data storage and access framework for PingER. In: Proceedings of the 2018 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE), pp. 1303–1308 (2018). <https://doi.org/10.1109/TrustCom/BigDataSE.2018.00179>
9. Oliveira, M.T., Carrara, G.R., Fernandes, N.C.: Towards a performance evaluation of private blockchain frameworks using a realistic workload. In: ICIN, pp. 180–187 (2019)
10. Wylde, V., et al.: Cybersecurity, data privacy and blockchain: a review. *SN Comput. Sci.* **3**(2), 1–12 (2022). <https://doi.org/10.1007/s42979-022-01020-4>
11. Bhutta, M.N.M., et al.: A survey on blockchain technology: evolution, architecture and security. *IEEE Access* **9**, 61048–61073 (2021). <https://doi.org/10.1109/ACCESS.2021.3072849>
12. Esolutions: Multi-factor authentication (MFA). <https://www.monash.edu/esolutions/accounts-passwords/multi-factor-authentication>
13. Gao, H., Ma, Z., Luo, S., Xu, Y., Wu, Z.: BSSPD: a blockchain-based security sharing scheme for personal data with fine-grained access control. *Wirel. Commun. Mob. Comput.* **2021** (2021). <https://doi.org/10.1155/2021/6658920>
14. Kumar, S., Bharti, A.K., Amin, R.: Decentralized secure storage of medical records using blockchain and IPFS : a comparative analysis with future directions. *Secur. Priv.* **4**(5), 1–16 (2021). <https://doi.org/10.1002/spy2.162>
15. Vishwakarma, R., Kumar, A.: A survey of DDoS attacking techniques and defence mechanisms in the IoT network. *Telecommun. Syst.* **73**(1), 3–25 (2020). <https://doi.org/10.1007/s11235-019-00599-z>
16. Companion, S.: A multiple blockchains architecture on inter-blockchain communication (2018). <https://doi.org/10.1109/QRS-C.2018.00037>
17. Sun, J., Yao, X., Wang, S., Wu, Y.: Non-repudiation storage and access control scheme of insurance data based on blockchain in IPFS. *IEEE Access* **8**, 155145–155155 (2020). <https://doi.org/10.1109/ACCESS.2020.3018816>

18. Wu, E.: Understanding IPFS/FileCoin. https://www.google.com/search?sca_esv=554284329&sxsrf=AB5stBhzj2ShfOnl4nLMBY6-NwRJbfaakg:1691351739030&q=how+ipfs+breaks+down+large+files+into+small+chunks&tbm=isch&source=lnms&sa=X&ved=2ahUKEwiXiuGc6MiAAxWxR2wGHbZbApQQ0pQJegQIChAB&biw=1366&bih=611&dpr=1
19. Athanere, S., Thakur, R.: Blockchain based hierarchical semi-decentralized approach using IPFS for secure and efficient data sharing. *J. King Saud Univ. Comput. Inf. Sci.* **34**(4), 1523–1534 (2022). <https://doi.org/10.1016/j.jksuci.2022.01.019>



Learning Systems in Forgery Image Detection

Sudheer Shetty¹(✉), Devadiga Likhit kumar Ganesh², Rahul R. Poojary², Raviraj²,
R. Tejas², Nagaratna P. Hegde³, and Sireesha Vikkurty³

¹ Department of ISE, Sahyadri College of Engineering & Management, Mangaluru Affiliated to
Visvesvaraya Technological University, Belagavi, Karnataka, India
sudheershetty06@gmail.com

² Department of ISE, Alvas Institute of Engineering and Technology, Moodbidri Affiliated to
Visvesvaraya Technological University, Belagavi, Karnataka, India

³ Department of CSE, Vasavi College of Engineering, Hyderabad, Telangana, India

Abstract. In an era dominated by the ubiquitous sharing of visual content on social media, the proliferation of fake photographs and digital fakes has emerged as a critical issue. This paper explores how fake images are detected with fusion of by various techniques of image processing, CNN, and machine learning, presenting an innovative approach to address the intricate challenge of identifying manipulated images. This paper gives insight on various approaches of image processing techniques and state-of-the-art machine learning algorithms used in a robust forgery image detection system. A comprehensive framework is proposed on various image forensic methods, ensuring the precise detection of fake images across diverse contexts. The paper also emphasizes the impact of countering the dissemination of fake photographs to preserve the credibility of digital media. This combination of image processing, CNN and machine learning propels the ongoing quest to uncover the truth behind the surge of digital fakes in today's digital landscape.

Keywords: Machine learning · fake detection · image processing · deep fake · social media

1 Introduction

In today's digital landscape, the fusion of social media and advanced technology has revolutionized communication and information sharing. However, this digital era has also ushered in a pressing concern – the proliferation of manipulated or counterfeit images, commonly known as digital fakes. The widespread dissemination of such misleading visual content across social media platforms has given rise to significant issues, including misinformation, propaganda, and threats to personal and professional reputations. Consequently, the development of robust techniques for detecting forged images has become an urgent necessity.

This paper explores the convergence of multiple crucial domains, including image processing, machine learning, CNN and image forensics, with a specific focus on the

detection of fake images. By harnessing state-of-the-art technology, we present an innovative approach to address the intricate challenge of identifying manipulated images within the vast realm of social media. Our study employs advanced image processing methods to meticulously examine subtle inconsistencies and artifacts, while machine learning algorithms automate the detection process.

Our comprehensive framework incorporates diverse image forensic techniques, ensuring the precise identification of fake images across various contexts. Through the integration of these interdisciplinary approaches, our paper aims to offer an effective solution to counter the proliferation of digital fakes. Beyond fulfilling the immediate need for detecting image forgery, our work makes a substantial contribution to the broader field of image forensics. Here, the synergistic application of image processing and machine learning continues to drive advancements in uncovering the truth behind digital manipulations.

Taiping Yao [1] highlights the traditional face forgery detection methods rely on standard convolutional neural networks (CNNs) to classify facial images as real or fake. However, these approaches often focus on limited facial regions, limiting their ability to detect complex forgeries. Recent methods have attempted to target specific forgery patterns, such as noise characteristics and local textures, but they remain vulnerable to unknown manipulation techniques and variations.

Amit Lokre [2] explains neural networks play a pivotal role in detecting tampered images and documents. The application excels at discerning original documents from digitally manipulated ones, addressing the surge in cybercrimes facilitated by sophisticated photo editing tools. Leveraging advanced techniques like Discrete Wavelet Transform (DWT), Scale Invariant Feature Transform (SIFT), and Local Binary Pattern (LBP), the study delves into passive forgery detection methods, enhancing resilience and efficiency. Comparative analysis empowers users to select the optimal forgery detection method, ensuring the integrity of digital media in an era where manipulation has become indistinguishable to the human eye. On the other hand Priyanka [3] explains JPEG is a format that loses some data during compression, but the degree of error in each resave varies. Any changes made to the image will affect stable regions.

Figure 1 explains an image underwent alterations using Photoshop, primarily relying on the initial 75% resave. This process involved duplicating books on a shelf and introducing a toy dinosaur. When subjected to 95% Error Level Analysis (ELA), changes became evident as areas no longer maintained their minimal error levels.

Error level analysis operates by deliberately resaving the image at a specified error rate, such as 95%, and then gauging the disparity between the images. Minimal alterations indicate that the cell has attained its local error minimum at that quality level, while substantial changes suggest that the pixels are not at their local minimum and are considered “original”.

The result is shown in the above Fig. 2 of machine learning implemented using the Neuropath library in Java, was employed to detect discrepancies in the error Level-Analyzed images. This encompassed converting images into ELA representations, standardizing them to 100×100 pixels, and converting them into arrays. These arrays contained 30,000 integer values signifying the image's pixels, facilitating neural network analysis to pinpoint manipulated areas.

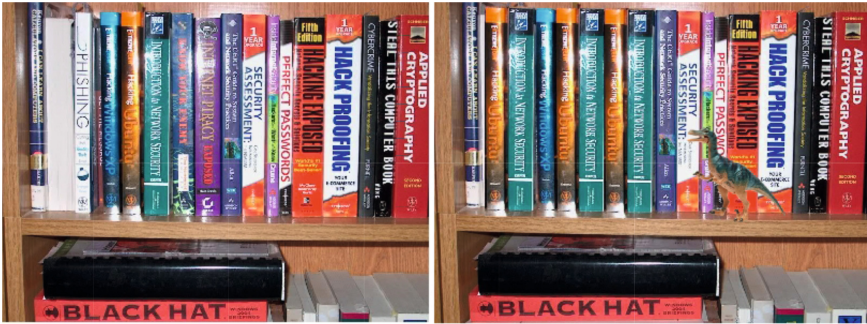


Fig. 1. Error Level Analysis (Source: Hacker Factor)

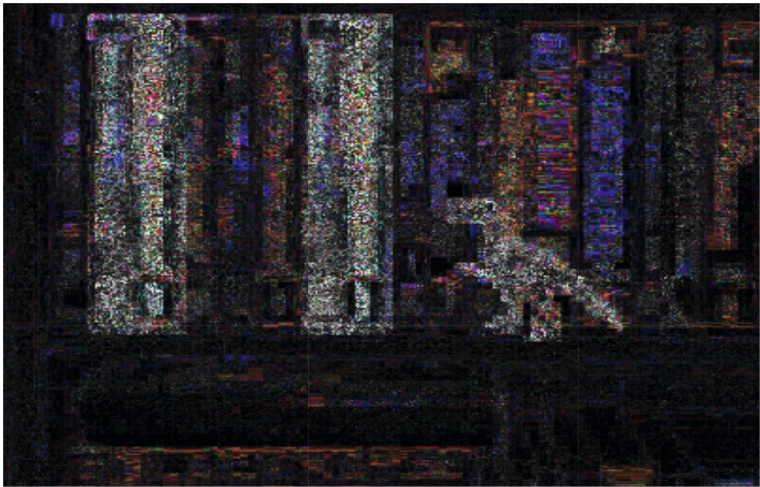


Fig. 2. Error Level Analysis after Employing Machine Learning (Source: Hacker Factor)

2 Related Work

Many endeavours have been made to improve the efficiency of facial forgery detection. However, the convolutional neural network (CNN) structures, borrowed from image classification models, primarily emphasize distinguishing among diverse categories rather than concentrating on the nuances that set apart real and fabricated images. Recently, as manipulated facial images have become more visually convincing, several research works have advocated a more in-depth exploration of particular forgery traits such as noise statistics, local textures, and frequency information to discern fake faces from genuine ones. Where in the case of using the machine learning techniques there are several steps in identifications as well as operating on a manipulated image.

Muhammed Afsal Villan [4] explains it as; image files contain not only visual content but also invaluable metadata, providing insights into the image’s origin. This metadata includes details about the camera used, colour space information, and application notes.

Different image formats store varying amounts of metadata. For instance, formats like BMP, PPM, and PBM typically hold minimal information, mainly focusing on image dimensions and colour space. In contrast, JPEG files from cameras tend to contain extensive metadata, encompassing camera make and model, aperture settings, and timestamps. PNG files usually have limited metadata, unless they have been converted from JPEG or edited using software like Photoshop, in which case some metadata from the source file format may be retained.

Arun Sekar Rajasekaran [5] explains it as, Image Processing involves applying operations to images to enhance them or extract valuable information. It is a form of signal processing where an input image is processed to generate another image or derive properties from it. Image processing is particularly relevant for analyzing discontinuous objects or patterns within an image. Machine Learning (ML) and Artificial Intelligence (AI) have greatly impacted the field of image processing. Google Lens is an example of how deep ML and AI techniques can process complex images. For instance, it can recognize and describe objects seen through a camera, making it useful for language translation and identification which includes interpolation, filtering, gray level manipulation, geometric processing, image subtraction, segmentation.

Njood Mohammed AlShariah [6] adds by explaining it by utilizing MATLAB software alongside the deep learning toolbox offers the flexibility to either train a Convolutional Neural Network (CNN) from scratch or employ a pretrained model for transfer learning. The choice depends on resource availability, application type, and goals.

Training a CNN from the ground up entails defining layer and filter numbers, along with other adjustments, demanding substantial data and time resources. Alternatively, transfer learning leverages pretrained models to extract features from new datasets, streamlining the process, especially when a large dataset and extended training time are not feasible. Miki Tanaka [7] explains it as, Image manipulation has a history predating deep learning. Techniques like fragile watermarking, statistical methods for double JPEG compression detection, and camera-specific PRNU patterns have been used to spot tampering. However, these methods often fail to detect fake images generated by GANs and struggle to differentiate fake images from simple manipulations like resizing, which aren't necessarily fake images.

Abhilash Kumar Saxena [8] explains it by illustrating, LSTM (Long Short-Term Memory) is a recurrent neural network designed for handling long-term tasks. It features input, forget, and output gates to manage information flow. LSTM is widely used for time series data analysis and forecasting, with its cell state serving as long-term memory. The input gate selects important attributes, the forget gate discards irrelevant information using a sigmoid function, and the output gate influences future steps by deciding which current-time data is essential.

Muhammed Afsal Villan [4] explains it as; analyzing metadata has proven effective for detecting anomalies in unshared images. It successfully identifies alterations in “photoshopped” or “gimped” images with minimal processing. However, it encounters challenges with images shared through platforms like WhatsApp and Google +. Additionally, its accuracy diminishes when dealing with images that have manipulated metadata.

Paolo Bestagini [9] explains, DeepFake Detection Challenge (DFDC) is the training dataset for the Kaggle challenge, consisting of 119,000 + video sequences, both real and fake. Real videos showcase diverse actors with varying characteristics, while fake videos are created using DeepFake techniques applied to real footage. The exact algorithms for generating fakes remain undisclosed. Sequence length is about 300 frames, with a substantial majority of fake videos, approximately 100,000 compared to 19,000 real ones.

Chengrui Wang [10] explains that Forgery Attention Map (FAM) is introduced for accurate identification of the detector's sensitive area, enabling guided erasing and visualization of forgeries. Specifically, it identifies the region where perturbations have the most significant impact on the detection outcome. During forward propagation, the detector processes a facial image (I) and produces two logits (Oreal and Ofake) to assess the authenticity of I.

Zhiqing Guo [11] explained with respect to CNN technique that is, CNNs are widely used in image forensics due to their superior performance and fall into three categories: (1) Custom CNN modules for specific tasks, (2) Using hand-crafted residual features combined with CNNs, (3) Improving the convolution layer, e.g., Constrained-Conv, for direct feature learning from tampering evidence.

Nidhal El Abbadi [12] explains it as, Image authenticity, in essence, involves the capacity to discern genuine images from counterfeits, encompassing both 2D composites and 3D computer-generated graphics. However, effectively detecting the authenticity of digital images presents an ongoing challenge. This challenge has been compounded by the widespread use and increasing sophistication of desktop photo editing software, which has effectively eroded the once-clear boundary between genuine and manipulated images, rendering the task of differentiation increasingly complex.

Chih-Chung Hsu [13] explains by using GANs technique where training a deep neural network to distinguish fake and real images involves assembling a diverse dataset. However, the classifier may struggle to detect images from new GANs or advanced generators, lacking knowledge of discriminative features. Gathering training images from all possible GANs is impractical, requiring frequent retraining with each new GAN proposed and with the help of GANs it is much easier to identify the altered images.

Usha Kosarkar [14] explains the steps to detecting deepfakes by relying on algorithms akin to those used in their creation. These technologies identify discrepancies absent in authentic photos or videos. Detection signs include irregular eye blinking, unusual skin tones, lip sync issues, additional pixels in frames, inconsistent lighting, audio mismatches, and persistent blur in low-resolution videos. Various techniques aid detection, such as background comparison, temporal pattern analysis, facial attribute consideration, mesoscopic analysis, and head pose estimation. A notable indicator is abnormal blinking patterns.

Jessica Fridrich [15] explains in Copy-Move forgery, a portion of an image is duplicated and pasted elsewhere within the same image, often to conceal an object. This technique is commonly applied to textured areas like grass or fabric, making the copied segments blends seamlessly. As the copied parts share the image's characteristics, such

as noise and color palette, conventional methods struggle to detect inconsistencies. Additional tools like feathered crops or retouching further obscure traces of the manipulated segments, enhancing the forgery's concealment.

Hemanth C [16] explains the concept of Copy-Move Forgery, a prevalent method where a portion of an image is duplicated and discreetly inserted into another location within the same image, often to conceal specific information. This technique, extensively utilized for image manipulation, introduces a formidable detection challenge due to the manipulated segment seamlessly residing within the original image. The primary intent is to obscure details intentionally, rendering the identification of the forged component a complex task. This nuanced form of manipulation underscores the need for advanced detection methods capable of discerning such intricacies within digital content.

Misbah Shaikh [17] explains the detection of fake image by Laser-Induced Break-down Spectroscopy is employed to analyze spectrums from both authentic and questioned documents. The extracted spectrums are structured into a dataset with nodes and edges, forming a network representing the documents. This network is visualized and grouped. The detection method relies on network cluster data, such as the number of clusters, demonstrating the effectiveness of the proposed approach in distinguishing between documents. This approach by utilizing the LIBS technique gives a solid result and also it is one of the fastest deep learning techniques. The summary of the above literature is discussed in the below shown Table 1.

Table 1. The Summary of Literature

Author	Contribution	Future enhancements
Cao et al. [1]	This paper introduces the RECCE framework. The framework comprises three main components: • Reconstruction Learning • Multi-Scale Graph Reasoning • Reconstruction Guided Attention	• Real-time Detection • Enhanced Training Data • Robustness to Post-processing • Privacy Protection
Lokre et al. [2]	The paper highlights the growing prevalence of cybercrimes involving fake documents and images. It suggests using neural networks for detecting digitally manipulated documents, emphasising the need to combat document-based cybercrimes	• Cross-Domain Detection • Neural network's accuracy

(continued)

Table 1. *(continued)*

Author	Contribution	Future enhancements
Priyanka [3]	This study focuses on detecting and unveiling fake images in the digital era, crucial for addressing image-based cybercrimes. It combines neural networks, Error Level Analysis, and image metadata for effective detection	• Developing user-friendly mobile applications for broader use • Exploring potential applications in digital authentication and legal evidence evaluation
Muhammed Afsal Villan [4]	This paper focuses on detecting fake images, crucial for combating image-based cybercrimes. It employs neural networks to identify tampered regions, Error Level Analysis, and image metadata to enhance detection reliability and plans for an Android implementation	• Refining the Error Level Analysis (ELA) technique for fake • Image detection. Exploring advanced software tools, and incorporating machine understandable algorithms to analyze, visual content within images
Arun Sekar Rajaekaran [5]	Image manipulation involves altering digital images using image processing techniques. Machine learning and AI enhance image processing, aiding industries like healthcare and defense	• Neural network's accuracy in cross domain • Exploring potential applications in digital authentication and legal evidence evaluation
Njood Mohammed AlShariah [6]	The literature discusses efforts to detect fake images using various methods, including deep learning CNNs, ReLU activation functions, and SoftMax classifiers. MATLAB implementation for image classification is also detailed, with AlexNet and transfer learning discussed	• Aim to improve the accuracy of identifying manipulated images • Integration of real-time AR overlays in image processing apps for context aware information

(continued)

Table 1. (*continued*)

Author	Contribution	Future enhancements
Miki Tanaka [7]	The proposed method involves robust hash computation for real vs. fake image detection. A robust hashing technique is used, compared with Wang's method, with better accuracy and robustness, especially against additional image manipulations	• Develop advanced neural network architectures and training techniques for detecting sophisticated fake images, including deepfakes • Incorporate blockchain technology for image authentication and tracking to improve verification processes
Abhilash Kumar Saxena [8]	Deepfake technology, driven by deep learning, produces deceptive content that poses serious threats, from misinformation to cybercrimes. Detection methods and collaboration between researchers and tech companies are essential for mitigation	• Enhancing robustness. Expanding datasets, advanced hashing, real-time processing, fusion techniques • Explainability, generalization, scalability, countermeasure resistance, and user-friendly interfaces • Cross-platform as well as domain flexibility
Paolo Bestagini [9]	FF++ and DFDC datasets, used for testing deepfake detection methods. FF++ comprises 1.8 million images from manipulated videos, while DFDC contains 119,000 sequences of real and fake videos	• Providing more detailed information about the results and findings of the experiments conducted on the FF++ and DFDC
Chengrui Wang [10]	The proposed method, Representative Forgery Mining (RFM), enhances fake face detection through attention-based data augmentation. It uses Forgery Attention Maps (FAM) to reveal detector sensitivity and selectively erases suspicious forgeries during training	• Exploring advanced techniques in forgery attention map generation • Enhancing the augmentation process for even more accurate fake face detection

(continued)

Table 1. *(continued)*

Author	Contribution	Future enhancements
Zhiqing Guo [11]	Content discusses experimental settings for AMTENnet in detecting face image manipulations. It explores dataset creation, model design modifications, and comparisons with state-of-the-art models, focusing on improving detection robustness through post-processing operations	• Investigating additional image manipulation techniques • Exploring real-world applications beyond forensics • Evaluating model scalability and efficiency
Nidhal El Abbadi [12]	The paper presents an improved SVD based fake image detection method with a 0.0% false positive rate, eliminating the need for original images, and enhancing detection efficiency	• Detect all types of manipulation traces in every image format • Eliminating the need for original images, and enhancing detection efficiency
Chih-Chung Hsu[13]	Discusses how to differentiate between fake and real images using GANs requires a diverse dataset. However, detecting images from new GANs is challenging due to a lack of discriminative features. Gathering training data from all GANs is impractical, necessitating frequent retraining. GANs facilitate the identification of altered images	• Develop an adaptive learning approach that dynamically updates the image detection • Implement a collaborative framework between various generative models, enabling the image detection system to learn from a collective dataset generated by different GANs
Usha Kosarkar[14]	Paper proposess steps to detect any fake image and based on the steps carry out the necessary detection algorithm to analyze where the image has been manipulated	• Additional steps should be implemented • Every step action should be carry out to improve efficiency

(continued)

Table 1. (continued)

Author	Contribution	Future enhancements
Jessica Fridrich[15]	The paper discuss the Copy-Move Forgery involves duplicating and subtly placing an image segment within itself, typically to hide specific details	• Improve detection in all image formats • Improve the accuracy
Mr.Hemanth C[16]	Copy-Move forgery duplicates a part of an image within itself, concealing objects. Often applied to textured areas, it seamlessly blends copied segments. Detection is challenging as the copied parts share image characteristics, confounding conventional methods	• Cannot be used in large scale system • Can be made more reliable in medium level sub system or domains
Misbah Shaikh[17]	The paper discuss Laser-Induced Breakdown Spectroscopy (LIBS) is used to examine spectra from genuine and questioned documents. The obtained spectra are organized into a dataset with nodes and edges, constructing a network that visually groups the documents	• Cannot be used in small system as technology is costly • Should be made portable in system if made for smaller system

3 Need and Issues in Forgery Image Detection

Forgery image detection is essential in today's digital landscape due to various challenges and opportunities. It involves real-time identification of manipulated images shared on different platforms, necessitating improved training data for accurate detection across domains. Ensuring resilience to post-processing is crucial to counter sophisticated forgery techniques while safeguarding privacy. The Ability to Detect Manipulation Traces Across All Image Formats is Vital for Comprehensive Detection.

- **User-Friendly Interfaces.** To address these needs, user-friendly mobile applications can widen the usage of forgery image detection tools. Exploring applications in digital authentication and legal evidence evaluation can enhance its value. The refinement of the Error Level Analysis (ELA) technique and the exploration of advanced software tools are necessary steps.
- **Trained model.** Advanced neural network architectures and training methods are necessary to detect increasingly sophisticated fake images like deepfakes. Integrating blockchain technology for image authentication and tracking can enhance verification

processes. Robustness improvements involve expanding datasets, advanced hashing, real-time processing, and fusion techniques.

- **Model Scalability and Efficiency.** Considering factors like explainability, generalization, scalability, countermeasure resistance, and user-friendly interfaces is crucial. Ensuring cross-platform and cross-domain flexibility, along with widespread availability on mobile platforms, is key for broad adoption.
- **Ethical and Societal Implications.** Additionally, providing comprehensive experiment result details, addressing ethical and societal implications, exploring advanced forgery attention map generation, enhancing the augmentation process, and adapting to modern tools and technology are essential steps in meeting the challenges of forgery image detection. Overall, the demand for forgery image detection arises from the evolving digital landscape and the need for reliable authentication methods.
- **Cross-Domain Detection.** Detecting manipulation traces across various image formats and domains requires adaptable algorithms.
- **Privacy Protection.** Balancing the need for image analysis with privacy concerns is a challenge, as sensitive information may be present in images.
- **Real-Time Detection.** Achieving real-time identification of manipulated images, especially in dynamic online environments, is challenging.

4 Conclusion

Forgery image detection is crucial in our rapidly evolving digital landscape. The demand for real-time detection has surged as images circulate swiftly across diverse platforms. Robust detection hinges on high-quality and diverse training data, necessitating comprehensive datasets. Addressing robustness against post-processing techniques and preserving privacy remain ongoing challenges. Cross-Domain detection, mobile application accessibility, and exploring applications beyond image forensics highlight the versatility and potential of forgery image detection tools.

Accuracy, model scalability, explainability, and countermeasure resistance are pivotal for success. User-friendly interfaces and ethical considerations add depth to the discourse, emphasizing responsible technology development. In the face of multifaceted challenges, the future of forgery image detection holds promise. Advancements in technology, research, and collaboration will continue to drive innovation in this critical field. With a commitment to addressing emerging techniques, forgery image detection will remain a potent weapon against image manipulation and forgery in the digital age.

References

1. Cao, J., Ma, C., Yao, T., Chen, S., Ding, S., Yang, X.: End-to-end reconstruction-classification learning for face forgery detection. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, pp. 4103–4112 (2022). <https://doi.org/10.1109/CVPR52688.2022.00408>
2. Lokre, A., Thorat, S., Patil, P., Gadekar, C., Mali, Y.: Fake image and document detection using machine learning. *Int. J. Sci. Res. Sci. Technol. (IJSRST)*, **5**(8), 104–109 (2020). Print ISSN: 2395-6011. Online ISSN: 2395-602X

3. Priyanka: Review Paper on Identifying Fake Images by Digital Image Processing. *Int. J. Sci. Eng. Res. (IJSER)* **5**(5) (2017)
4. Villan, M.A., Kuruvilla, K., Paul, J., Elias, E.P.: Fake image detection using machine learning. *IRACST – Int. J. Comput. Sci. Inform. Technol. Secur. (IJSITS)* **7**(2) (2017)
5. Pavan, A., Rajasekaran, A., Geetha, B., Subramanian, M., Perti, A.: Machine learning systems in image manipulation and fake detection. *Turkish J. Physiother. Rehabil.* **32**(2) (2022)
6. AlShariah, N.M., Saudagar, A.K.J.: Detecting fake images on social media using machine learning. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* **10**(12), 170 (2019)
7. Tanaka, M., Kiya, H.: Fake-image detection with Robust Hashing. [arXiv:2102.01313v1](https://arxiv.org/abs/2102.01313v1) (2021)
8. Saxena, A.K., Raja, P.K. N.: An examination on implementation of deep fake in images through deep learning. In: 2022 Fourth International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT) (2022)
9. Bonettini, N., Cannas, E.D., Mandelli, S., Bondi, L., Bestagini, P., Tubaro, S.: Video face manipulation detection through ensemble of CNNs. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 5012–5019. IEEE (2021)
10. Wang, C., Deng, W.: Representative Forgery Mining for Fake Face Detection. Beijing University of Posts and Telecommunications [arXiv:2104.06609](https://arxiv.org/abs/2104.06609) (2021)
11. Guo, Z., Yang, G., Chen, J., Sun, X.: Fake Face Detection via Adaptive Manipulation Traces Extraction Network. Affiliated with Hunan University and Nanjing University of Information Science and Technology. [arXiv:2005.04945v2](https://arxiv.org/abs/2005.04945v2) (2020)
12. El Abbadi, N., Hassan, A.M., AL-Nwany, M.M.: Blind fake image detection. *Int. J. Comput. Sci. Issues (IJCSI)*, **10**(4), 180 (2013)
13. Hsu, C.C., Lee, C.Y., Zhuang, Y.X.: Learning to detect fake face images in the wild. In: 2018 International Symposium on Computer, Consumer and Control (IS3C), pp. 388–391. IEEE (2018)
14. Kosarkar, U., Sakarkar, G., Gedam, S.: An analytical perspective on various deep learning techniques for deep fake detection. *Int. J. Innov. Eng. Sci.* **7**(8), 25–30 (2022)
15. Fridrich, J., Soukal, D., Lukas, J.: Detection of copy-move forgery in digital images. In: *Proceedings of Digital Forensic Research Workshop*, vol. 3, no. 2, pp. 652–63, August 2003
16. Hemanth, C., Srivathsa, D.A., Gouthami, M., Monica, S., Sarika, B.V.: Digital image sham detection using deep learning. *Int. Res. J. Eng. Technol. (IRJET)* **10**(04), 1816 (2023)
17. Shaikh, M., Patil, D.: Image forgery/tampering detection using deep learning and cloud. *Int. Res. J. Eng. Technol. (IRJET)* **09**(06), 2105 (2022)



Correction to: A Comprehensive Review of Optimization Techniques for Automatic Voltage Regulator Controllers

Rumana Ali and Vinayambika S. Bhat

Correction to:

**Chapter 15 in: T. Shreekumar et al. (Eds.): *Intelligent Systems
in Computing and Communication*, CCIS 2232,
https://doi.org/10.1007/978-3-031-75608-5_15**

In the originally published version of chapter 15, author corrections were not updated.
They have now been updated.

The updated version of this chapter can be found at
https://doi.org/10.1007/978-3-031-75608-5_15

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
T. Shreekumar et al. (Eds.): ISCComm 2023, CCIS 2232, p. C1, 2025.
https://doi.org/10.1007/978-3-031-75608-5_28

Author Index

A

Acharya, Ravish II-235
Adapa, Rithika I-14
Alaswad, Feisal I-86
Ali, Rumana II-168
Aljaddouh, Batoul I-86
Anh, Nguyen Duc II-130
Anjutagi, Akshata B. II-89
Aradhya, B. C. Yathish II-275
Archana, A. R. I-221
Arun, S. II-181
Asmitha, K. A. I-221
Azad, S. M. A. K. II-317

B

Babu, Ch Ravindra I-342
Balachandran, Rohith I-221
Baligar, Vishwanath P. I-309
Bharamagoudar, Geeta I-270, I-285
Bhaskar, K. II-181
Bhat, Amulya S. I-72
Bhat, S. Arundhathi II-235
Bhat, Vinayambika S. I-188, I-207, II-81, II-168, II-291
Bhushan, Bharat I-254
Bindu Madhavi, J. II-111
Bommegowda, K. B. II-21

C

Chand, Mutyala Venkata Gopi I-145
Chand, S. Veera Venkata Gopi I-342
Chandana, C. S. II-139
Chander, Satish I-102
Cherian, Renil M. I-221
Chinthakrinda, Vennela Preethi I-318
Churantal, Mondal II-139

D

Damarla, Yaswanth Kumar II-3, II-32
Deenadayal, Bhagya Rekha II-3, II-32
Deepa II-303

Deepika, K. M. II-89
Devi, T. II-248
Dhananjaya, B. I-3
Dhanesha, R. I-158
Dinesh, H. A. II-275
Dinesha, A. S. II-195
Doddannavar, Pooja I-309

F

Farhanaaz II-139
Fathima Afrah, K. N. I-221
Fernandes, Roshan I-102

G

G. Totad, Shashikumar I-270
G.Totad, Shashikumar I-285
Ganesh, Devadiga Likhit kumar II-342
Geetha Mary, A. II-150
Girish, G. N. I-158
Gouli, Prajwalkumar A. I-309
Gowda, A. Menaka I-26
Gurumoorthi, E. II-72

H

Hanji, Sanjay I-364
Hegde, G. P. I-318
Hegde, Nagaratna P. II-342
Hemalatha, N. I-14
Hiremath, Gurudeva Shastri I-158
Hiremath, P. S. II-217, II-262

I

Illur, Naresh I-309

J

Jaglan, Gaurav II-100
Jayanth, T. Siva I-342
Jayaprakash, M. C. II-81, II-121
Jolly, Aman II-100

Jose, Manju I-326, I-354
 Joshita, V. II-248
 Jyothi, E. V. N. I-145, I-342
 Jyothi, K. II-303

K

K., Manohara K. I-285
 Karadagi, Renuka II-89
 Karadgi, Sachin II-262
 Katiravan, Jeevaa II-45
 Kavya, V. I-207
 Kirankumar, M. V. I-128
 Koithyar, Aravinda II-181
 Kotian, Deepthi I-207, II-291
 Krishna, M. I-26
 Kulamarva, Ravishankara II-130
 Kumar, A. Aneesh I-334
 Kumari, Pavitra I-128
 kumari, Pavitra II-111
 Kurian, Sainamole I-14

L

Laboso, Patrick Kipkorir II-331
 Lakshmi, B. N. I-26
 Lakshmi, V. II-317
 Lathesh II-235
 Lavanya, K. II-89
 Lokesh, M. R. II-275

M

Malagund, Pooja M. I-309
 Malathi, S. Y. I-270
 Martin, A. II-331
 Mody, Parth I-50
 Mohan Kumar Naik, B. II-111
 Mohan, Divya I-72, II-150
 Murali, P. S. II-21
 Muthudhivakar, M. II-45

N

Nagaratna Hegde, P. I-318
 Naik, Sandeep Seetaram I-334
 Nainwal, Ankita I-3
 Nallagachu, Swathi I-171
 Nanade, Archana I-50
 Narayankar, Prashant I-309
 Naveen, Rashmi II-150
 Nayana, M. P. II-57

Nichenametla, Hemanth Kumar I-232,
 II-235
 Nishmitha I-128
 Nithin, Ramakrishnan II-139

P

Pai, Prathamesh I-14
 Pandey, Vikas II-100
 Patil, Prakashgoud II-217
 Peal, K. P. Sonia I-334
 Peddoju, Suresh Kumar I-232, II-235
 Pooja, M. R. I-119
 Poojary, Dhanush I-207
 Poojary, Rahul R. II-342
 Poovammal, E. I-86
 Pradyumna, G. R. II-21
 Prasad, Durga II-21
 Prassana, K. Naga I-342
 Praveenkumar, S. M. II-217
 Priyadharshini, K. I-72, II-150
 Priyanka, R. I-119

R

Rafidha Rehiman, K. A. I-221
 Raghavendra, B. K. I-128
 Rajesh, Peram I-145
 Rajeshwari I-128
 Raksha I-14
 Rakshitha, R. II-235
 Ravi, M. S. II-21
 Ravindra Babu, Ch. I-145
 Raviraj II-342
 Reddy, Annem Tharun Kumar I-145
 Revathy, G. II-72
 Rodrigues, Anisha P. I-102
 Rohith Raj, R. I-221

S

Sabarinath, S. II-45
 Sah, Astha I-254
 Sajeesh, P. K. I-14
 Salian, Chaithra II-21
 Sandanalakshmi, R. I-171
 Sandani, Sk. I-342
 Sanju, V. II-139
 Santhosh, S. II-111
 Sathyabama, A. R. II-45
 Seeri, Shivananda V. I-334
 Sejjpal, Param I-50

Senthilkumar, J. [II-72](#)
 Shama, M. P. [II-81](#)
 Sharma, Garima [I-3](#)
 Shashidhara, Shrividya [I-26](#)
 Shashikant [II-100](#)
 Shetty, Charishma [II-21](#)
 Shetty, Chinmai [I-254](#)
 Shetty, Deepthi [II-291](#)
 Shetty, Prakyath [II-21](#)
 Shetty, Sudheer [I-232, I-318, II-235, II-342](#)
 Shiragudikar, Sharada K. [I-270, I-285](#)
 Shivaprasad [II-195](#)
 Shrinivasa Naika, C. L. [I-158](#)
 Shwetha, R. J. [I-145, I-342](#)
 Shyamalagowri, M. [II-72](#)
 Siddarth, Sangavi [II-139](#)
 Singh, Ankur [I-50](#)
 Singh, Anurag [I-50](#)
 Singh, Indrasen [II-100](#)
 Singhal, Dhruv [I-3](#)
 Soumya, S. [II-57](#)
 Sreekumar, K. M. [I-14](#)
 Sri Krishna Shastri, C. [II-121](#)
 Sudheendra, P. [I-188](#)
 Suganya, P. V. Raja [II-248](#)
 Supriya, B. [I-86, I-102](#)
 Suresha, D. [II-130](#)

T

Teja, Kunchala [I-145](#)
 Tejas, R. [II-342](#)
 Thiagarajan, P. [II-331](#)
 Titus, Bins [I-221](#)
 Tiwari, Shrikant [I-364](#)

U

Udupa, Abhishek A. [II-303](#)
 Udupi, Prakash Kumar [I-326, I-354](#)
 Umesha, D. K. [I-158](#)

V

Verma, Deepanshu [I-3](#)
 Vijaya, P. [I-102](#)
 Vikkurty, Sireesha [I-318, II-342](#)
 Vinh, Nguyen Quang [II-130](#)
 Vinod, P. [I-221](#)
 Vyankatesh, Kulkarni Sourabh [I-128](#)

W

Wadawadagi, Ramesh [I-364](#)

Y

Y., Malathi S. [I-285](#)
 Yashodha, M. N. [II-89](#)