

Rabindra Nath Shaw  
Sanjoy Das  
Vincenzo Piuri  
Monica Bianchini *Editors*

# Advanced Computing and Intelligent Technologies

Proceedings of ICACIT 2022

# Lecture Notes in Electrical Engineering

## Volume 914

### Series Editors

Leopoldo Angrisani, Department of Electrical and Information Technologies Engineering, University of Napoli Federico II, Naples, Italy

Marco Arteaga, Departament de Control y Robótica, Universidad Nacional Autónoma de México, Coyoacán, Mexico

Bijaya Ketan Panigrahi, Electrical Engineering, Indian Institute of Technology Delhi, New Delhi, Delhi, India

Samarjit Chakraborty, Fakultät für Elektrotechnik und Informationstechnik, TU München, Munich, Germany

Jiming Chen, Zhejiang University, Hangzhou, Zhejiang, China

Shanben Chen, Materials Science and Engineering, Shanghai Jiao Tong University, Shanghai, China

Tan Kay Chen, Department of Electrical and Computer Engineering, National University of Singapore, Singapore, Singapore

Rüdiger Dillmann, Humanoids and Intelligent Systems Laboratory, Karlsruhe Institute for Technology, Karlsruhe, Germany

Haibin Duan, Beijing University of Aeronautics and Astronautics, Beijing, China

Gianluigi Ferrari, Università di Parma, Parma, Italy

Manuel Ferre, Centre for Automation and Robotics CAR (UPM-CSIC), Universidad Politécnica de Madrid, Madrid, Spain

Sandra Hirche, Department of Electrical Engineering and Information Science, Technische Universität München, Munich, Germany

Faryar Jabbari, Department of Mechanical and Aerospace Engineering, University of California, Irvine, CA, USA

Limin Jia, State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China

Janusz Kacprzyk, Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Alaa Khamis, German University in Egypt El Tagamoa El Khames, New Cairo City, Egypt

Torsten Kroeger, Stanford University, Stanford, CA, USA

Yong Li, Hunan University, Changsha, Hunan, China

Qilian Liang, Department of Electrical Engineering, University of Texas at Arlington, Arlington, TX, USA

Ferran Martín, Departament d'Enginyeria Electrònica, Universitat Autònoma de Barcelona, Bellaterra, Barcelona, Spain

Tan Cher Ming, College of Engineering, Nanyang Technological University, Singapore, Singapore

Wolfgang Minker, Institute of Information Technology, University of Ulm, Ulm, Germany

Pradeep Misra, Department of Electrical Engineering, Wright State University, Dayton, OH, USA

Sebastian Möller, Quality and Usability Laboratory, TU Berlin, Berlin, Germany

Subhas Mukhopadhyay, School of Engineering & Advanced Technology, Massey University, Palmerston North, Manawatu-Wanganui, New Zealand

Cun-Zheng Ning, Electrical Engineering, Arizona State University, Tempe, AZ, USA

Toyoaki Nishida, Graduate School of Informatics, Kyoto University, Kyoto, Japan

Luca Oneto, Department of Informatics, Bioengineering., Robotics, University of Genova, Genova, Genova, Genova, Italy

Federica Pascucci, Dipartimento di Ingegneria, Università degli Studi "Roma Tre", Rome, Italy

Yong Qin, State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China

Gan Woon Seng, School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore, Singapore

Joachim Speidel, Institute of Telecommunications, Universität Stuttgart, Stuttgart, Germany

Germano Veiga, Campus da FEUP, INESC Porto, Porto, Portugal

Haitao Wu, Academy of Opto-electronics, Chinese Academy of Sciences, Beijing, China

Walter Zamboni, DIEM - Università degli studi di Salerno, Fisciano, Salerno, Italy

Junjie James Zhang, Charlotte, NC, USA

The book series *Lecture Notes in Electrical Engineering* (LNEE) publishes the latest developments in Electrical Engineering—quickly, informally and in high quality. While original research reported in proceedings and monographs has traditionally formed the core of LNEE, we also encourage authors to submit books devoted to supporting student education and professional training in the various fields and applications areas of electrical engineering. The series cover classical and emerging topics concerning:

- Communication Engineering, Information Theory and Networks
- Electronics Engineering and Microelectronics
- Signal, Image and Speech Processing
- Wireless and Mobile Communication
- Circuits and Systems
- Energy Systems, Power Electronics and Electrical Machines
- Electro-optical Engineering
- Instrumentation Engineering
- Avionics Engineering
- Control Systems
- Internet-of-Things and Cybersecurity
- Biomedical Devices, MEMS and NEMS

For general information about this book series, comments or suggestions, please contact [leontina.dicecco@springer.com](mailto:leontina.dicecco@springer.com).

To submit a proposal or request further information, please contact the Publishing Editor in your country:

#### **China**

Jasmine Dou, Editor ([jasmine.dou@springer.com](mailto:jasmine.dou@springer.com))

#### **India, Japan, Rest of Asia**

Swati Meherishi, Editorial Director ([Swati.Meherishi@springer.com](mailto:Swati.Meherishi@springer.com))

#### **Southeast Asia, Australia, New Zealand**

Ramesh Nath Premnath, Editor ([ramesh.premnath@springernature.com](mailto:ramesh.premnath@springernature.com))

#### **USA, Canada**

Michael Luby, Senior Editor ([michael.luby@springer.com](mailto:michael.luby@springer.com))

#### **All other Countries**

Leontina Di Cecco, Senior Editor ([leontina.dicecco@springer.com](mailto:leontina.dicecco@springer.com))

**\*\* This series is indexed by EI Compendex and Scopus databases. \*\***

Rabindra Nath Shaw · Sanjoy Das ·  
Vincenzo Piuri · Monica Bianchini  
Editors

# Advanced Computing and Intelligent Technologies

Proceedings of ICACIT 2022



Springer

*Editors*

Rabindra Nath Shaw  
Bharath Institute of Higher Education  
and Research  
Chennai, Tamil Nadu, India

Vincenzo Piuri  
Department of Computer Science  
University of Milan  
Milan, Italy

Sanjoy Das  
Regional Campus Manipur  
Indira Gandhi National Tribal University  
Imphal, Manipur, India

Monica Bianchini  
Department of Information Engineering  
and Mathematics  
University of Siena  
Siena, Italy

ISSN 1876-1100

Lecture Notes in Electrical Engineering

ISBN 978-981-19-2979-3

<https://doi.org/10.1007/978-981-19-2980-9>

ISSN 1876-1119 (electronic)

ISBN 978-981-19-2980-9 (eBook)

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022

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 Singapore Pte Ltd.  
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721,  
Singapore

# **ICACIT 2021 Organizing Committee**

## **General Chairs**

Monica Bianchini, Department of Information Engineering and Mathematics,  
University of Siena, Italy  
Sanjoy Das, IGNTU, India

## **Conference Chair and Chairman, Oversight Committee**

Rabindra Nath Shaw, Bharath Institute of Higher Education and Research, India

## **Conference Secretaries**

Ankush Ghosh, TNU, India  
Saravanan D., Galgotias University, India

## **Technical Chair**

Nishad Mendis, Det Norske Veritas, Australia

## **Publication Chairs**

Vincenzo Piuri, FIEEE, Professor, University of Milan, Italy  
Valentina E. Balas, Aurel Vlaicu University of Arad, Romania

## **Publicity Chair**

Prashant R. Nair, Amrita Vishwa Vidyapeetham, India

## **Springer/ICACIT Liaison**

Aninda Bose, Senior Editor, Springer Nature

## **International Advisory Board**

Ajay Gupta, IEEE Computer Society, USA  
Mohammad S. Obaidat, Fellow of IEEE University of Sharjah, UAE  
Fawnizu Azmadi Hussin, Chair, IEEE Malaysia Section  
Dharmendra Sharma, Chairman, University Academic Board, University of Canberra  
Maria Virvou, University of Piraeus, Greece  
Marcin Paprzycki, Polish Academy of Sciences, Poland  
Laxmi C. Jain, Co-founder, KES International, UK  
Celia Shahnaz, Chairperson, IEEE Bangladesh Section  
Anna Esposito, Seconda Università di Napoli, Italy  
Sanjeevi Kumar Padmanaban, Aalborg University Esbjerg, Denmark  
Atiqur Rahman Ahad, University of Osaka, Japan  
George T., University of Piraeus, Greece  
Margarita N. Favorskaya, Reshetnev Siberian State University  
Urszula Stańczyk, Silesian University of Technology (SUT), Gliwice, Poland  
Valentina E. Balas, University of Arad, Romania  
N. R. Pal, President, IEEE CIS  
Yen-Wei Chen, Ritsumeikan University, Japan  
Milan Simic, RMIT University, Australia

## Technical Program Committee and Reviewers

Brijesh Iyer  
Meenakshi Bhardwaj  
Anshu Parashar  
Maya L. Pai  
Vinod Kumar  
Vijayakumar Ponnusamy  
Sreeja P. S.  
Rajashree Narendra  
Saru Kumari  
Mallikharjuna Rao K.  
R. Ramanathan  
Rishibind Kumar Upadhyay  
Jasminder Sandhu  
A. S. M. Touhidul Hasan  
Arvind Dagur  
Hussain Mahdi  
Suneet Kumar  
S. Sriniwasulu Raju  
Pinaki Chakraborty  
Preeti Rai  
Sandeep Mathur  
Sachin Goyal  
Amit Jain  
Rajesh Prasad  
Kapil Gupta  
Nuparam Chauhan  
Jitali Patel  
Ankit Saxena  
Prashant Nair  
Vishwesh Laxmikant Akre  
Kamlesh Lakhwani  
Irfan Siddavatam  
Malaya Nath  
Era Johri  
Abhishek Dubey  
Arun K. Singh  
Karthikeyan B.  
Rohit Tripathi  
Krishnananda Shet  
Indrani Das  
Shrawan Kumar  
Jayshree Pande

Aditi Paul  
Mahesh Pawar  
Priyanka Singh  
Shrikant Sonekar  
Nuzhat Shaikh  
Drmandeep Kaur  
Keshav Gupta  
Sanjeev Rana  
Pranav Kumar Singh  
Joe Louis Paul I.  
Venugopala P. S.  
Ashwini B.  
Ebha Koley  
Mohd Sadiq  
Suraiya Jabin  
Pawan Kumar  
Dalia Nandi  
S. Vijay Kumar  
Sameena Naaz  
Kedar Nath Sahu  
Mohit Tiwari  
Ashwini Kumar  
Harshal Shah  
Sunil Singh  
Surbhi Gupta  
Jay Kumar Jain  
Vijayalakshmi Kakulapati  
Kirthiga S.  
Alka Leekha  
Sanjeev Kumar  
Jawahar P. K.  
Shyamal Mondal  
Prachi Garg  
Krishna Kumar Singh  
Sartajvir Singh  
M. T. L. Gayatri  
Md. Forhad Rabbi  
Rohit Raja  
Vinay Goyal  
Prashantkumar Vats  
Mansaf Alam  
Parul Dawar  
Rajendra Patil  
Prem Chand  
Jinesh M. K.

Hardeo Kumar Thakur  
Abhijit Lahiri  
Shailendra Dwivedi  
Vandana Niranjan  
Akshat Agrawal  
Anup Kumarbarman  
Sarita Kumari  
Ahmed Ali Shah  
Anjana Pandey

# Preface

This book gathers selected high-quality research papers presented at the 2022 International conference on Advanced Computing and Intelligent Technologies (ICACIT), held at NCR New Delhi, India, on March 20–21, 2021, jointly organized by Bharath Institute of Higher Education and Research, India, and Department of Information Engineering and Mathematics, Università Di Siena, Italy. It discusses emerging topics pertaining to advanced computing, intelligent technologies and networks including AI and machine learning, data mining, big data analytics, high-performance computing network performance analysis, Internet of Things networks, wireless sensor networks and others. Written by respected experts and researchers working on computing and intelligent technologies, the book offers a valuable asset for researchers both from academia and industries involved in advanced studies.

We are thankful to all the authors that have submitted papers for keeping the quality of ICACIT 2022 at high levels. The editors of this book would like to acknowledge all the authors for their contributions and the reviewers. We have received invaluable help from the members of the International Program Committee and the chairs responsible for different aspects of the workshop. We also appreciate the role of Special Sessions Organizers. Thanks to all of them, we had been able to collect many papers on interesting topics, and during the conference, we had very interesting presentations and stimulating discussions.

We hope that the volume will provide useful information to professors, researchers and graduated students in the area of soft computing techniques and applications and all will find this collection of papers inspiring, informative and useful. We also hope to see you at a future ICACIT event.

Siena, Italy  
Milan, Italy  
Imphal, India  
Chennai, India

Monica Bianchini  
Vincenzo Piuri  
Sanjoy Das  
Rabindra Nath Shaw

# Contents

<b>Evolution of MANO Towards the Cloud-Native Paradigm for the Edge Computing .....</b>	1
Alejandro Fornés-Leal, Ignacio Lacalle, Rafael Vaño, Carlos E. Palau, Fernando Boronat, María Ganzha, and Marcin Paprzycki	
<b>Prognosis of Parkinson's Malady—A Multimodal Approach .....</b>	17
Shanvi Sharma and Pratima Singh	
<b>A Study on Conversion of Image Caption to Speech Using Neural Network Models .....</b>	33
Swati Shinde and Prachi Waghmare	
<b>Automated Detection of Hypertension Disease Using Machine Learning and Signal Processing-Based Methods .....</b>	41
Jaypal Singh Rajput and Manish Sharma	
<b>Novel Score Function and Accuracy Function for Spherical Linguistic Fuzzy Numbers and Their Application in Multi-criteria Decision-Making Problems .....</b>	55
Laxmi Rajput and Sanjay Kumar	
<b>Application of Matrix Tri-Factorization for Predicting miRNA-Disease Associations .....</b>	67
J. R. Rashmi and Lalitha Rangarajan	
<b>A Novel Framework for Malpractice Detection in Online Proctoring ...</b>	81
Shaik. Mohissin Sultana and M. Kameswara Rao	
<b>Classification of Land Cover Hyperspectral Images Using Deep Convolutional Neural Network .....</b>	89
J. Arun Pandian, Saurav Kr. Gupta, Rohit Kumar, Shourjya Hazra, and K. Kanchanadevi	

<b>A Comprehensive Comparative Study Between LBP and LBP Variants in Face Recognition .....</b>	99
Shekhar Karanwal	
<b>Ten-Layered Deep Convolution Neural Network-Based Tea Leaf Disease Prediction .....</b>	117
J. Arun Pandian, M. Shyamala Devi, K. Kanchanadevi, R. Aruna, Mupparaju Kavya Sree, Tella Malathi, and Kanagala Saideepak	
<b>Maximum Relevancy and Minimum Redundancy Based Ensemble Feature Selection Model for Effective Classification .....</b>	131
A. Saravanan, C. Stanly Felix, and M. Umarani	
<b>Comparative Study of Pre-trained Models on Cotton Plant Disease Detection Using Transfer Learning .....</b>	147
Satyam Kumar, Dayima Musharaf, and Anil Kumar Sagar	
<b>A Comparative Study of Time Series, Machine Learning, and Ensemble Models for Crude Oil Price Prediction .....</b>	157
Ankit Prakash and Sunil Kumar Singh	
<b>Performance Analysis of Improved Mobility Models to Check Their Impact on QoS in MANET .....</b>	173
Munsifa Firdaus Khan and Indrani Das	
<b>A Comprehensive Analysis of Application-Based MAC Protocol for Wireless Sensor Network .....</b>	183
Mamta Mann and Rishipal Singh	
<b>A Framework for Lung Cancer Detection Using Machine Learning .....</b>	199
Aakash Nakarmi, Anil Kumar Sagar, Seerat Musharaf, and Hadya Jahangir	
<b>Efficient Diagnosis of Alzheimer's Disease Using EfficientNet in Neuroimaging .....</b>	211
H. Shareen, B. Dhanush, P. Rukmani, and D. Dhanya	
<b>LiDAR Integration with ROS for SLAM Mediated Autonomous Path Exploration .....</b>	225
Rapti Chaudhuri and Suman Deb	
<b>Textual Inference Identification in the Malayalam Language Using Convolutional Neural Network .....</b>	237
Sara Renjit and Sumam Mary Idicula	
<b>Automated Crack Severity Level Detection and Classification for Surface Crack Using Deep Convolutional Neural Networks .....</b>	251
Harshad B. Nehate, Parth K. Kharkar, Pratiksha S. Bhat, Varad S. Rane, and Kavita Tewari	

<b>Haze Elimination Strategy for Indoor Robot Vision with Dark Channel Prior and Encoded Blur Kernel Space .....</b>	<b>267</b>
Shubham and Suman Deb	
<b>Financial Modeling Using Deep Learning .....</b>	<b>281</b>
Gunpreet Singh Walia, Nikita Sinha, Nalin Kashyap, Divakar Kumar, Subrata Sahana, and Sanjoy Das	
<b>Application of Hybrid ANFIS-CSA Model in Suspended Sediment Load Prediction .....</b>	<b>295</b>
Abinash Sahoo, Nihar R. Mohanta, Sandeep Samantaray, and Deba Prakash Satapathy	
<b>An Enhanced Technique to Improve the Performance of Multi-image Compression Technique .....</b>	<b>307</b>
Dibyendu Barman, Abul Hasnat, and Bandana Barman	
<b>Diabetes Prediction Using Machine Learning Techniques .....</b>	<b>317</b>
Sourav Simanto, Krishna Mridha, Runa Saha, Milan Limbu, Ankush Ghosh, and Rabindra Nath Shaw	
<b>Machine Learning in Bioinformatics: New Technique for DNA Sequencing Classification .....</b>	<b>335</b>
Shakil Sarkar, Krishna Mridha, Ankush Ghosh, and Rabindra Nath Shaw	
<b>Hybrid Exponential Smoothing-LSTM-Based Univariate Stock Market Prediction for Financial Sectors in NIFTY50 .....</b>	<b>357</b>
C. Koushik, M. V. Pranav, R. K. Arjun, and S. Shridevi	
<b>Smart Contract for Academic Certificate Verification Using Ethereum .....</b>	<b>369</b>
Shivani Pathak, Vimal Gupta, Nitima Malsa, Ankush Ghosh, and R. N. Shaw	
<b>Comprehension of Climate Change with IoT-Enabled CNN .....</b>	<b>385</b>
Priyanka Singh, Debaroti Sammanit, Rabindra Nath Shaw, and Ankush Ghosh	
<b>Employing Adaptive Learning and Intelligent Tutoring Robots for Virtual Classrooms and Smart Campuses: Reforming Education in the Age of Artificial Intelligence .....</b>	<b>395</b>
Ashraf Alam	
<b>A Comprehensive System for Detecting Profound Tiredness for Automobile Drivers Using a CNN .....</b>	<b>407</b>
Jitendra Singh Kushwah, Dhyanendra Jain, Prashant Singh, Amit Kumar Pandey, Sanjoy Das, and Prashant Vats	

<b>Satellite Data Investigation for Change Estimation During COVID Era by Fusing Pixel and Object-Based Technique .....</b>	417
Amit Kumar Shakya, Ayushman Ramola, and Anurag Vidyarthi	
<b>The Soiling Impact Effects on the Performance of the Solar Panels in Rabigh, Saudi Arabia .....</b>	429
Abdullah Mohammed Zaki Khayyat, Abdulrahman Hamad Alharthi, Faisal Lafi Almohammadi, Ziyad Saad Almarwani, Nithiyanthan Kannan, and Youssef Mobarak	
<b>Machine Learning Models for the Electrical Power Generation by Savonius Vertical Axis Wind Turbine .....</b>	441
Youssef Kassem, Hüseyin Çamur, Ahmed Hamid Mohamed Abdalla Zakwan, and Nkanga Nkanga Amanam	
<b>Performance Evaluation of SIMON and SPECK Block Ciphers to Secure IoT-Enabled Smart Cities .....</b>	451
Monika Jangra and Buddha Singh	
<b>Human Activity Recognition for Disease Detection Using Machine Learning Techniques—A Comparative Study .....</b>	463
Arpan Adhikary, Koushik Majumder, Santanu Chatterjee, Rabindra Nath Shaw, and Ankush Ghosh	
<b>Use of Various Optimization Algorithms in the Energy Minimization Problem Domain of WSN: A Survey .....</b>	477
Sudip Kumar De, Avishek Banerjee, Koushik Majumder, Rabindra Nath Shaw, and Ankush Ghosh	
<b>Prediction of Glaucoma Using Machine Learning-Based Approaches—A Comparative Study .....</b>	489
Tiyasha Dhara, Arpan Adhikary, Koushik Majumder, Santanu Chatterjee, Rabindra Nath Shaw, and Ankush Ghosh	
<b>Implementation of IoT-Based Health Care and Saline Monitoring System Using Arduino UNO .....</b>	513
Rinkesh Pantawane, Amlan Jyoti, Sangeeta Sheoran, Yogeshwar Uikey, Shivali Sonarkar, and Rajesh Thakare	
<b>Blockchain-Based Academic Certificate Verification System—A Review .....</b>	527
Shivani Pathak, Vimal Gupta, Nitima Malsa, Ankush Ghosh, and R. N. Shaw	

Contents	xvii
<b>Performance Analysis of VANETs Using Trusted Location and Trustworthiness of Nodes .....</b>	<b>541</b>
Indrani Das, Sanjoy Das, Rishi Pal Singh, Ashwini Kumar, and Subrata Sahana	
<b>Author Index .....</b>	<b>557</b>

## About the Editors

**Dr. Rabindra Nath Shaw** is currently working as Director, International Relations, Bharath Institute of Higher Education and Research (Deemed to be University), Chennai, India. Before joining BIHER he has served also Galgotias University as Director, IR&C, He is an alumnus of the Applied Physics department, University of Calcutta, India. He is a Senior Member of IEEE Industry Application Society, USA and Fellow of Nikhil Bharat Shiksha Parishad, India. Dr. Shaw is a global leader in organizing International conferences. His brand of world leading conference series includes IEEE International Conference on Computing, Power and Communication Technologies (GUCON), IEEE International Conference on Computing, Communication and Automation (ICCCA), IEEE IAS Global Conference on Emerging Technologies (GlobConET), International Conference on Electronics and Electrical Engineering (ICEEE), International Conference on Advances in Computing and Information Technology (ICACIT) etc. He holds the position of Conference Chair, Publication Chair, and Editor for these conferences. These Conferences are held in collaboration with various international universities like Aurel Vlaicu University of Arad, University of Malaya, University of Siena. Many world leaders are working with Dr. Shaw in these conferences. Most of these conferences are fully sponsored by IEEE Industry Applications Society, USA. He is also an expert in organizing International Seminars/Webinars/ Faculty Development Programme in collaboration with leading institutes across the world. His research interests include optimization algorithms and machine learning techniques for power systems, IoT applications, Renewable Energy, and Power electronics converters. He has published more than hundred twenty five papers in scopus/web of Science/SCI indexed international journal/proceedings including high impact factor journals in the field of renewable energy. He has also published several edited books from Springer and Elsevier publishing houses. Dr. Shaw has worked on many national/ international patents. Dr. Shaw has successfully executed his duty in various positions like, Governing body member, Centre in Charge, NBA Coordinator, University Examination Coordinator, University MOOC's Coordinator, University Conference Coordinator, and Faculty In Charge, Centre of Excellence for Power Engineering and Clean Energy Integration. He has more than eleven years of teaching, research and administrative experience

in leading institutes like Motilal Nehru National Institute of Technology Allahabad, Jadavpur University, Gargi Memorial Institute of Technology, Brainware Group of Institution, Dream Institute of Technology.

**Dr. Sanjoy Das** is currently working as Associate Professor, Department of Computer Science, Indira Gandhi National Tribal University (A Central Government University), Amarkantak, M.P. (Manipur Campus), India. He received his Ph.D. in Computer Science from Jawaharlal Nehru University, New Delhi, India. Before joining IGNTU, he has worked as Associate Professor, School of Computing Science and Engineering, Galgotias University, India, July 2012 to September 2017, and also, as Assistant Professor G. B. Pant Engineering College, Uttarakhand, and Assam University, Silchar, from 2001 to 2008. His current research interest includes Mobile Ad hoc Networks and Vehicular Ad hoc Networks, Distributed Systems, Data Mining. He has published numerous papers in international journals and conferences including IEEE and Springer.

**Dr. Vincenzo Piuri** is Professor at the University of Milan, Italy (since 2000). He was Associate Professor at Politecnico di Milano, Italy, Visiting Professor at the University of Texas at Austin, USA, and Visiting Researcher at George Mason University, USA. He has founded a start-up company in the area of intelligent systems for industrial applications and is active in industrial research projects. He received his M.S. and Ph.D. in Computer Engineering from Politecnico di Milano, Italy. His main research and industrial application interests are artificial intelligence, intelligent systems, computational intelligence, pattern analysis and recognition, machine learning, signal and image processing, biometrics, intelligent measurement systems, industrial applications, distributed processing systems, Internet of things, cloud computing, fault tolerance, application-specific digital processing architectures, and arithmetic architectures. He published over 400 papers in international journals, international conference proceedings, and books. He is Fellow of the IEEE and Distinguished Scientist of ACM. He has been IEEE Vice President for Technical Activities (2015), Member of the IEEE Board of Directors (2010–2012, 2015), and President of the IEEE Computational Intelligence Society (2006–2007). He is Editor-in-Chief of the *IEEE Systems Journal* (2013–2019).

**Dr. Monica Bianchini** received the Laurea cum laude in Mathematics and the Ph.D. degree in Computer Science from the University of Florence, Italy, in 1989 and 1995, respectively. After receiving the Laurea, for two years, she was involved in a joint project of Bull HN Italia and the Department of Mathematics (University of Florence), aimed at designing parallel software for solving differential equations. From 1992 to 1998, she was a Ph.D. student and a Postdoc Fellow with the Computer Science Department of the University of Florence. Since 1999, she has been with the University of Siena, where she is currently Associate Professor at the Information Engineering and Mathematics Department. Her main research interests are in the field of artificial intelligence and applications, machine learning, with emphasis on neural networks for structured data and deep learning, approximation theory, information

retrieval, bioinformatics, and image processing. She has authored more than seventy papers and has been Editor of books and special issues on international journals in her research field. She has been a participant in many research projects focused on machine learning and pattern recognition, founded by both Italian Ministry of Education (MIUR) and University of Siena (PAR scheme), and she has been involved in the organization of several scientific events, including the NATO Advanced Workshop on Limitations and Future Trends in Neural Computation (2001), the 8th AI\*IA Conference (2002), GIRPR 2012, the 25th International Symposium on Logic-Based Program Synthesis and Transformation, and the ACM International Conference on Computing Frontiers 2017. She served as Associate Editor for *IEEE Transactions on Neural Networks* (2003–09), *Neurocomputing* (from 2002), and *International Journal of Computers in Healthcare* (from 2010). She is Permanent Member of the Editorial Board of IJCNN, ICANN, ICPR, ICPRAM, ESANN, ANNPR, and KES.

# Evolution of MANO Towards the Cloud-Native Paradigm for the Edge Computing



Alejandro Fornés-Leal , Ignacio Lacalle , Rafael Vaño , Carlos E. Palau , Fernando Boronat , Maria Ganzha , and Marcin Paprzycki

**Abstract** The Management and Orchestration framework (MANO) is the main element of the Network Function Virtualization paradigm. It is in charge of managing the life cycle of virtualized functions, from instantiation to manageability, live configuration, and termination. This kind of framework was originally designed to orchestrate network functions over virtual machines. However, the Cloud-Native approach, based on containers and microservices, has emerged and needs to be included as a part of MANO, to leverage all the inherent benefits that it brings. This contribution identifies the key enablers that have to be addressed, from the MANO perspective, to fully exploit the capabilities and to obtain real added value from implementing this novel approach, focusing mainly on resource-constrained environments. Besides, an analysis of current status of open-source frameworks aiming at the Cloud-Native adaptation is presented, showing that while Cloud-Native approaches vis-a-vis network functions are widely accepted (at least, by the research community), there is still room for further research and integration.

**Keywords** MANO · Cloud-Native · NFV · Edge computing

## 1 Introduction

Edge computing and Network Function Virtualization (NFV) have been two of the main technological paradigms of the IT environments that emerged during past years. The former is focused on bringing computation capabilities as close to the source of data (and actuation) as possible, hence optimizing network bandwidth usage, supporting low-latency applications, and reducing privacy and security breaches in

---

A. Fornés-Leal · I. Lacalle ( ) · R. Vaño · C. E. Palau · F. Boronat  
Communications Department, Universitat Politècnica de València, Valencia, Spain  
e-mail: [iglaub@upv.es](mailto:iglaub@upv.es)

A. Fornés-Leal  
e-mail: [alforlea@upv.es](mailto:alforlea@upv.es)

M. Ganzha · M. Paprzycki  
Systems Research Institute Polish Academy of Sciences, Warsaw, Poland

contrast to traditional cloud computing models [1]. The latter (NFV) aims at virtualizing network functions, facilitating their instantiation on general-purpose equipment, thus decoupling the provided services from the hardware that deliver them [2].

The European Telecommunications Standards Institute (ETSI) defined an architectural framework for NFV Management and Orchestration (MANO) [3], with the objective of facilitating the management of the life cycle of the Virtualized Network Functions (VNFs), from their instantiation to their configuration and termination. This framework has had a great acceptance not only by researchers but also within the business scope, due to great possibilities that it brings for developing new applications and business models.

However, MANO initially relied on virtual machines (VMs) for delivering the intended functionalities. Therefore, it was focused mostly on cloud infrastructures, which do not integrate well with the current move towards the Cloud-Native approach. Rather than referring to the place where applications are instantiated, Cloud-Native is about the way in which they are created and deployed [4], aiming at increasing their speed (of development and deployment), offering better scaling, and leveraging only the required hardware resources.

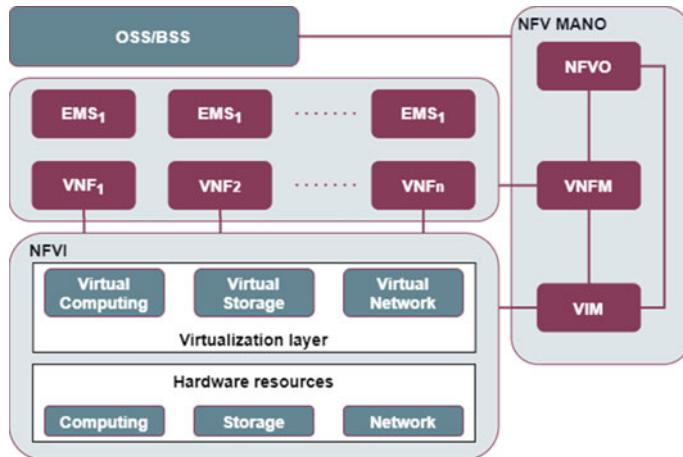
Current research has shown that offering the VNFs with containers rather than with VMs can provide great benefits, while opening the execution environment possibilities. Specifically, it can enable managing and orchestrating network (and non-network) virtualized functions in hardware with less resources. This, in turn, can promote emergence of new (or improved) use cases and business models. However, Cloud-Native solutions have to be integrated within MANO frameworks to be useful.

This contribution focuses on the current status of the integration of NFV and the Cloud-Native approach for edge distributed deployments, analysing the key enablers and potential future of this technological symbiosis. The remainder of this paper is organized as follows: Sect. 2 presents a review of MANO, edge computing, and the Cloud-Native model; Sect. 3 focuses on the key enablers and technologies for unlocking the adoption of Cloud-Native Virtualized Network Functions (CNFs); afterwards, in Sect. 4 an analysis of the current MANO solutions is presented, whereas in Sect. 5 an analysis of barriers and potential future of this paradigm transformation is considered. Finally, conclusions are drawn in Sect. 6.

## 2 MANO Framework and Edge Within IoT

### 2.1 ETSI MANO for NFV

ETSI was selected to host the Industry Specification Group for Network Function Virtualization (ETSI ISG NFV). Apart from the MANO architectural framework, the initial documents include an overview of the infrastructure, descriptions of the network, hypervisor, computing domains, and additional aspects such as security



**Fig. 1** MANO architecture

and trust, resilience, and quality of service. Based on these documents, the MANO architecture is presented in Fig. 1.

NFV requires access to hardware computing, storage, and network resources, which are provided by the NFV Infrastructure (NFVI), and assigned to the VNFs, depending on the specific demands. VNFs can be managed locally by the element management system (EMS).

All the hardware/virtual systems and the virtualized functions are managed by the NFV Management and Orchestration (MANO) that is composed of three elements: the NFV Orchestrator (NFVO), the VNF Manager (VNFM), and the Virtualized Infrastructure Manager (VIM).

- The NFVI is managed by the VIM. Here, computation, storage, and network-related resources are assigned to the virtual resources needed by specific functions.
- A VNF accesses its respective resources, globally configured, and supervised by the VNFM component. The VNFM also performs the respective coordination and adaptation role for configuration and event reporting between the VIM and the EMS.
- The NFVO is responsible of connecting or combining NFVs as building blocks, managing orchestration of NFVI resources across multiple VIMs and life cycle management of network services (NSs).

## 2.2 Edge Computing and NFV

Since the beginning, edge computing has been divided into three main categories of implementation: multi-access edge computing (MEC), cloudlets, and fog computing [5]. All of them share the vision of edge computing and strongly rely on mechanisms

such as virtualization, safety resources management, and metering. However, there are clear differences in configuration, characteristics, and scope. In summary:

- Multi-access (formerly, mobile) edge computing is associated with radio access network (RAN), where the edge capabilities are located at the base stations. This implementation is oriented towards ISP providers and is focused on achieving edge computing benefits for 4/5G use cases. Cloudlets can be understood as “replicas” of the cloud capabilities but closer to the edge of the network, thus reducing latency, round-trip time, and backhaul bandwidth consumption. They are conceived as “cloud in a box”, acting as cloud running over one, or a cluster of, resource-rich server(s).
- Fog computing (FC), instead, aims at leveraging the flexibility of IoT, to perform edge computing functions. Using “fog nodes”, that can be spanned through the edge-to-cloud continuum creating 1 to N “near-end” layers or tiers, FC orchestrates their functioning to take advantage of wide range of devices in the continuum of the spectrum.

The combination of NFV and edge implementations provides great benefits, especially when considering distributed environments such as industrial IoT. First of all, NFV allows instantiating and modifying existing services much faster, leveraging general-purpose equipment and avoiding the need of deploying dedicated hardware. Moreover, NFV enables the possibility of deploying and configuring services automatically, which can be further enhanced via intent-based methods [6], or self-organizing techniques [7]. Besides, NFV enables network slicing, so both network bandwidth and latency can be optimized, boosting the inherent latency reduction provided by the edge computing paradigm. Hence, it is clearly visible how NFV came into play with the objective of reducing both the time and cost related to initial deployments and operations, i.e. for reducing CAPEX and OPEX.

### ***2.3 Benefits of the Cloud-Native Approach for NFV***

As aforementioned, edge deployments can benefit from NFV, when compared to the use of dedicated hardware equipment. Still, as will be argued in what follows, the NFV model can be further improved if extended with the Cloud-Native approach, based on microservices and containers rather than virtual machines and traditional software architectures:

1. *Reducing (further) development and operational costs:* Cloud-Native applications are based on a set of granular, small microservices, which can be developed, deployed, and optimized independently, improving software DevOps cycles (avoiding having to package a complete, monolithic solution). At the edge, where hardware resources can be limited, containers reduce the amount of overhead required by the VMs, resulting in an effective reduction of costs.

2. *Improving the agility of a system:* In contrast to VMs, containers (that host the microservices) are much faster to deploy, substitute, and scale. The latter is a key feature in an edge environment since containers are flexible, using only the required resources and leaving space for other applications.
3. *Novel business paradigm:* The paradigm shifts towards renting additional servers on-demand, rather than acquiring a fixed set of VMs in advance. This model entails reduction of costs to both infrastructure operators and to end-users. In addition, less storage is needed, as Cloud-Native paradigm pushes towards stateless microservices.

### 3 Cloud-Native MANO Enablers

Evolving from VNFs to CNFs is not just as trivial as changing the VIM from Open-Stack (or a similar infrastructure management technology) to a container orchestrator platform, such as Kubernetes (k8s). This is particularly the case if this evolution is intended to extract all the potential brought by the Cloud-Native approach. The key enablers that MANO and CNFs need to support are presented in this section.

#### 3.1 *Microservices*

The Cloud-Native approach follows an architecture design paradigm based on microservices. It provides a robust solution as a set of small, loosely coupled, independent services, which are isolated in small coherent and autonomous units, to solve the problem of complex architectures and code redundancies. Microservices architecture allows scaling, or updating, each service without affecting the rest of the services of the system.

Although service-oriented architecture could be adopted, this paradigm is not as flexible as microservices, especially in those cases where the development team is spread out, the components have clear functionality boundaries, and if components can be potentially reused for other applications, microservices extract the full potential of DevOps cycles.

#### 3.2 *Containerization*

It is a lightweight, agile virtualization alternative to VMs. A container packages all the software needed to run a single application or microservice, including all code, libraries, and required dependencies. They are smaller, faster, and more portable than VMs, since they do not require including guest operative system (OS) in

each instance, leveraging the host kernel (OS virtualization), instead of the virtualized hardware infrastructure (as the VMs do). Here, despite the fact that *Docker* is currently the most popular container engine, there are also alternatives, such as *CRI-O*, *Containerd*, or *runc*.

Apart from containers, another lightweight virtualization technology, that is worth to mention, is *unikernels*. Unikernels are similar to virtual machines, but without many of the inherent services from the OS, leaving just the ones that are actually needed for executing their application [8]. In principle, they are more difficult to design than VMs or containers, but they have at least the same potential of performance as containers plus improved security and isolation features, which can be a valuable aspect in distributed environments (lesser attack surfaces).

### 3.3 Container Orchestration Technologies

VIMs, like OpenStack, OpenVIM, or other available vendor alternatives, have been designed for managing virtualized hardware resources to deploy VMs, not for managing virtualized OS spaces; therefore, they are not valid for orchestrating containers: specific container orchestration technologies are needed, among which one can find *k8s*, the *de facto* standard, *Docker Swarm*, or *Apache Mesos*.

Apart from *k8s*, lightweight alternatives based on it such as *k3s* or *MicroK8s* are also very interesting for distributed, resource-constrained environments, like industrial IoT.

### 3.4 Packaging and Management of Functions: Helm Charts and Juju Charts

Both *Helm* and *Juju* are package and operations managers for *k8s*. A *Helm* chart is a collection of files that describes a set of *k8s* resources and that can be used for deploying either an application or a component of a larger application. It provides *templating*, which allows users to declare variables and use functions to modify parameters of the applications (in this case, of CNFs).

Despite *Helm* charts being more widespread, *Juju* charts, based on hooks (typically written as shell scripts), are claimed to be a more scalable tool and more effective for complex container life cycle operations.

### 3.5 Networking

Typical networking schemas that apply for VMs are not valid for containers. This is due to the fact that containers are processes that share the kernel of the host system. Therefore, from the outside world perspective, they have a common IP address.

Currently, there are two solutions for addressing this issue: Container Network Model (CNM) and Container Network Interface (CNI). Here, the latter is close to becoming the *de facto* standard, despite the fact the former is being supported by Docker.

A CNI is a plug-in that implements a network interface within the container namespace, assigning an IP address to it and setting the required bridges with the host. There are different technologies for supporting networking, like *Flannel*, *Canal*, and *Weave*. Networking technologies can address different connectivity aspects, from L2/L3 networking, VXLAN, Overlay, and BGP, which can be of great interest for improving the NSs deployed, and enable more efficient schemas of multi-cluster networking.

### 3.6 Service Mesh and Service Discovery

Service mesh is a software infrastructure layer for controlling the communication between services. In a similar way to SDN, it decouples the control plane from the data plane. The latter is implemented as proxies on top of microservices, transparent to the business functionality, whereas the former interacts with proxies to provide different functionalities related to connectivity (service discovery, load balancing, dynamic routing control), security (encryption, policy enforcement), and observability (alerting based on traffic alerts).

In contrast to the networking section, service mesh provides application level features. Here, among the most popular tools, one can find *Istio*, *Linkerd*, and *Consul*.

Besides, service discovery mechanisms are required when a large number of microservices are available. To that end, a dedicated database of services (i.e. service registry) should be in place and expose the location of services (IPs addresses, ports) when queried by rightful users/services. *Consul* and *CoreDNS* are examples of service discovery engines.

### 3.7 Communication Bus

Microservices require communication and exchange of information for delivering a stand-alone service or application. In general, REST APIs are enough to handle it. Still, some network services may require larger information and telemetry data

exchange. Here, a dedicated bus (e.g. *Apache Kafka*) may be needed for providing scalable, reliable, and resilient communication.

### **3.8 Bare-Metal Deployment**

Another deployment paradigm that is recently evolving consists in running container orchestration platforms in bare metal, instead of a virtualized infrastructure. It involves executing Cloud-Native applications in containers running directly on the hardware, which results in a great simplification of network set-up. Getting rid of the virtualized infrastructure layer comes with many benefits among which one can find: increase of the available resources, since less overhead for virtualization is needed (can be key for resource-constrained environments); improvement of the operational performance; and reduction of costs related to licenses.

Nevertheless, MANO systems are designed to be communicated to NFVI, so integration effort is required in most cases.

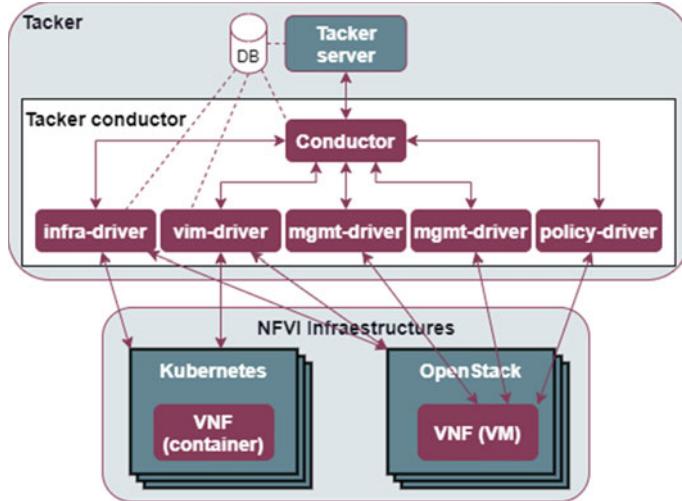
### **3.9 Hybrid Network Services Support**

The emergence of CNFs does not involve elimination of VNFs or Physical Network Functions (PNFs, provided by specialized hardware) from the MANO ecosystem, at least not yet.

Great effort has been put into designing and implementing VNFs. However, migrating them to CNFs is not a trivial task, especially for big appliances (for instance, a firewall with all the functionalities it provides). Hence, MANO should have the capability of managing all network function types, either physical, virtual (on top of a VIM) and Cloud-Native ones (leveraging a container orchestrator platform). Nevertheless, over time, full migration to Cloud-Native should occur, as “edge world” cannot afford to have two computing environments.

## **4 Current Status of MANO Solutions**

In this section, a summary of the current status of different open-source MANO frameworks, for supporting CNFs, is provided. Although in different maturity levels, all of them support CNF creation and management, illustrating the already-achieved pervasiveness of Cloud-Native approaches (at least, as considered from the research sector), despite the fact that further effort is needed, especially as what concerns standardization and integration activities.



**Fig. 2** Tacker architecture

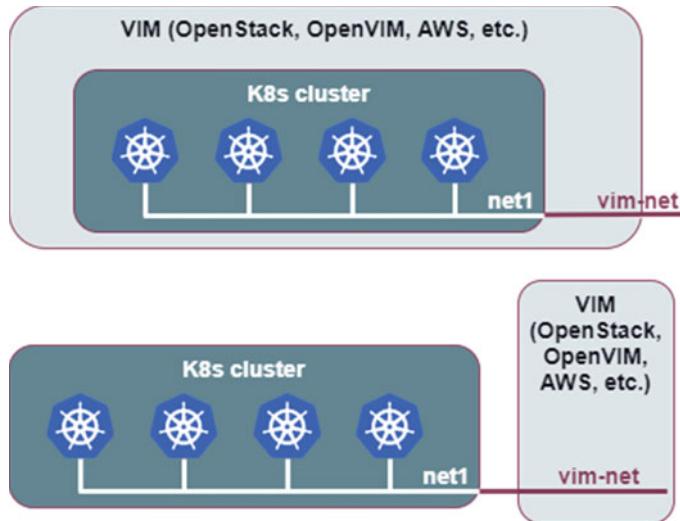
#### 4.1 Tacker

Tacker is a generic VNFM and an NFVO from OpenStack, based on ETSI MANO, developed to operate VNFs and compose NSs on an infrastructure platform like OpenStack or Kubernetes. Tacker proposes an architecture, in which Kubernetes acts as a VIM, parallel to OpenStack (see Fig. 2). Tacker is composed by a set of drivers to act over both VIMs: (1) *infra*, responsible of the operations to operate the VIMs; (2) *vim*, in charge of their registration; (3) *mgmt*, which facilitates the configuration of VNFs; (4) *monitor*, responsible for executing actions towards that end; and (5) *policy*, for VNF operations based on policies. Support for k8s has only been provided for the first two plug-ins.

However, the information regarding different Cloud-Native aspects is quite low, from both official documentation and existing literature. No reference to packaging and management functions, networking or service mesh is provided, so any of these functionalities have to be provided outside of the scope of MANO.

#### 4.2 OSM

OSM is the MANO framework developed under the ETSI umbrella. To manage CNFs, OSM is able to register *k8s* clusters as long as they are connected to an OpenStack-like VIM, as depicted in Fig. 3, allowing hybrid deployments (they can be connected to a “dummy” VIM, in which case those hybrid deployments are not possible).



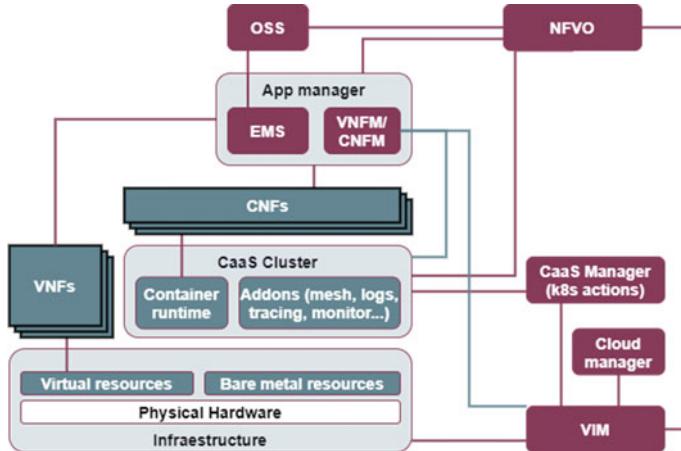
**Fig. 3** Interaction between VIMs and Kubernetes in OSM

OSM provides support for both *Helm* and *Juju*, which allow not only deploying but also configuring CNFs via primitives using charts or charms. Some features related to *k8s* networking, and service mesh, are not implemented yet, as a part of the framework (these features can be managed externally). This is hindering the orchestration for managing different clusters in multi-domain environments. This problem can be addressed by connecting each of them to VIMs to manage inter-cluster networking.

#### 4.3 Anuket

Anuket is based on the merge between OPNFV and the CNTT and developed under the umbrella of the Linux Foundation. It is one of the actions that push towards fully exploiting all Cloud-Native capabilities. It defines a Reference Architecture (RA, see Fig. 4) and a set of requirements to be fulfilled during its development and implementation. Up to this moment, it supports only the first of the two defined RAs, with basic features regarding CNF instantiation (e.g. Helm chart packaging is not yet supported).

The project is currently working on fully integrating Cloud-Native approach, from the networking perspective to *k8s* add-ons management (service mesh, monitoring, logging, tracing, etc.). It shows great potential and is worthy further monitoring.



**Fig. 4** Anuket architecture

#### 4.4 ONAP

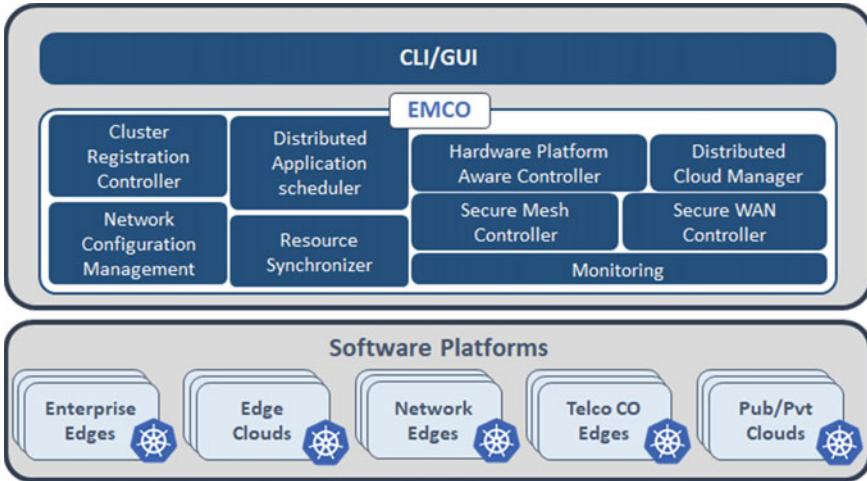
ONAP is one of the largest automation frameworks, composed of different subsystems, covering much more aspects than the rest to create an end-to-end platform. ETSI MANO architecture blocks can be mapped to ONAP ones, although alignment work is still being done to make it fully compliant.

ONAP allows packaging, deploying, and configuring CNFs with Helm charts. Current version contains dedicated APIs for creating and modifying k8s resource templates, checking services' health status, and communicating with telemetry tools, while also supporting hybrid and multi-cluster environments. Its future role regarding CNF networking, inventory, and service mesh enablers still needs to be evaluated.

#### 4.5 Other MANO Frameworks

Let us now briefly summarize two other MANO frameworks. SONATA was one of the first MANO frameworks providing support to CNFs on top of k8s. It includes CNF life cycle support and descriptor validation, allows defining and collecting custom metrics from both CNFs and k8s infrastructure, and implements hybrid network schemas. Unlike OSM, SONATA has a plug-in to connect directly to k8s VIM (as well as ONAP and Tacker's vim driver).

Besides, Cloudify is an open-source solution that goes beyond MANO purposes, similarly to ONAP. It has evolved differently to other alternatives to integrate multiple cloud environments (many platforms supported, e.g. k8s, Azure, AWS, etc.), and Cloud-Native features into the network orchestration model, aligned but not fully



**Fig. 5** EMCO architecture

embracing MANO specifications. Among its features, Cloudify includes instantiation and configuration of CNFs, intent placement based on policies, networking among clusters and hybrid NSs support, being one of the most advanced open-source solution.

## 4.6 EMCO

The Edge Multi-Cluster Orchestrator (EMCO), previously known as ONAP4K8S, is a different framework compared to the previous ones, as it is not following the MANO specifications (see Fig. 5). It is designed for deploying and orchestrating only Cloud-Native applications over a set of  $k8s$  clusters, from cloud to edge (hence multi-cloud but not supporting hybrid network schemas). This framework has been leveraged by two vendor open-source projects: Intel *Openness* and Aarna Networks *AMCOP*.

Being focused purely on Cloud-Native approach, the functionalities provided for CNFs are much more advanced in comparison with other frameworks (e.g. mesh network supported with Istio, better analytics gathering and service discovery features), while being more adapted to the edge environments. They allow hybrid deployments (VNF and PNF support), although without following MANO specifications, which in the long term may cause some interoperability issues.

Summarizing what has been described thus far, in Table 1, main features of noted frameworks have been presented.

**Table 1** Comparative table between MANO frameworks

Feature	MANO frameworks							
	Tacker	OSM	Anuket	ONAP	SONATA	Cloudify	EMCO	AMCOP
MANO compliance	Yes	Yes	Yes	Partially	Yes	Partially	No	No
CNF on-boarding	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
CNF validation	Proposed	Yes	Yes	Yes	Yes	Yes	No	Yes
CNF Helm/Juju support	No	Yes	No	Yes	No	Yes	Yes	Yes
CNF monitoring	No	No	Proposed	Yes	Yes	Yes	Yes	Yes
CNI-based networking	No	No	Proposed	No	No	Yes	Yes	Yes
Service mesh	No	No	Proposed	No	No	No	Yes	Yes
Hybrid deployment support	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Multi-cluster k8s support	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

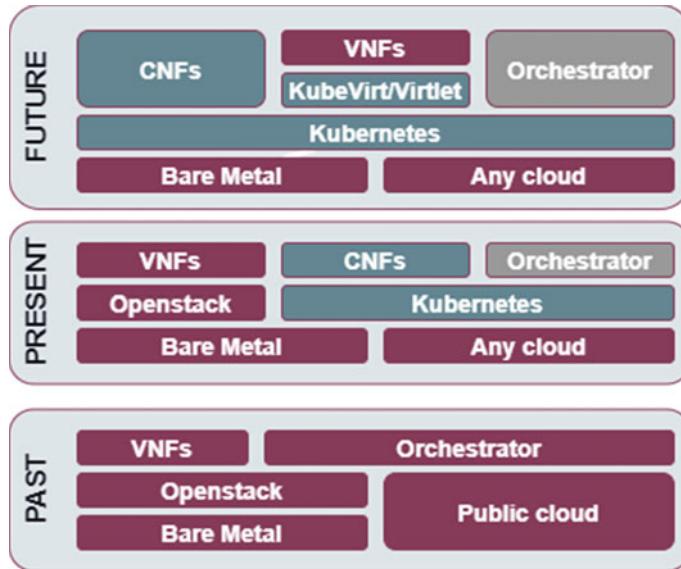
## 5 Discussion and Expected Outcomes

This section aims at outlining the set of existing barriers, as well as the expected evolution of MANO frameworks, towards achieving full Cloud-Native support. Here, note that the state of evolution among the different frameworks depends on different aspects. First of all, those solutions that have been present for some years need more adaptation effort to optimally accommodate CNFs. This becomes more challenging if they strictly follow the MANO specifications, since specifications towards the embracement of Cloud-Native additional features are not yet finalized (including the role of some of the enablers identified in Sect. 3).

On the contrary, novel solutions such as EMCO (and other non-open-source solutions) have greater potential as they are very agile embracing novel Cloud-Native solutions. Still, specifications are needed so the NFV ecosystem keeps the high interoperability level that was brought by ETSI MANO.

Among the current initiatives (and besides ETSI MANO), Anuket is the one promoting harder a standard RA that includes both VNFs and CNFs, deployed either over k8s and OpenStack-like VIMs.

To the authors' expectations, hybrid deployments will be the norm during current decade, since there has been a great effort put into developing VNFs and deploying complete virtualized infrastructures, which will not be immediately substituted by the



**Fig. 6** Expected evolution of MANO towards the cloud-native approach

Cloud-Native paradigm (besides, some VNFs are very challenging to be containerized into CNFs). Further explanation of the expected evolution has been presented in [4] and is summarized in Fig. 6.

In future, a model purely based on kubernetes, for both VNFs and CNFs, leveraging technologies like *KubeVirt* or *Virtlet* to deploy those network functions that could not be deployed as containers is foreseen.

Cloud-Native approach promises great benefits, like an automated installation and configuration of CNFs; dynamic scaling according to workload; self-healing and fault-tolerant reliable mechanisms; automated performance monitoring; high reusability and portability, etc. [9]. However, as any novel paradigm, it faces challenges and barriers to overcome, such as trust issues over administrative domains; isolation and security of CNFs (containers are less secure than VMs); network function chaining; adaptability of existing VNFs; change from the software development point of view, moving towards microservices approach; business adaptation delay needed for transforming current model towards Cloud-Native; and necessity needs for further research and, most importantly, of standardization actions [4, 9].

In some implementations, although CNF on-boarding, configurations and metrics retrieval can be performed via MANO, Cloud-Native capabilities are not managed by the framework. Features such as container networking, service mesh (layer 7 or even novel layer 3, IP models) and service discovery are left outside the scope of it and are expected to be managed externally via auxiliary tools. This diminishes the potential that can be obtained from integrating those features within the scope of

MANO, especially in the networking area, in the same manner than SDN and WAN schemas were introduced, in the past, for managing NFV networks.

## 6 Concluding Remarks

This paper provides a comprehensive evaluation of the key enablers for bringing the Cloud-Native benefits to the NFV ecosystem, especially for edge deployments, along with the current evolution of existing MANO frameworks and expected barriers and future developments related to this approach.

There are different aspects related to container orchestration, such as service mesh, networking, and service discovery that have not yet been addressed directly by the existing frameworks, especially by those that follow the MANO specifications.

This shows the necessity of advancing current NFV standardization activities towards Cloud-Native, effort that, for instance, the Anuket project is pushing forward with its Reference Architecture. In this context, there are already solutions that fully embrace the Cloud-Native model, integrating many of its features while keeping support for traditional VNFs and for composing hybrid network services. On the one hand, they show the benefits that can be extracted from it, but on the other hand, the lack of standards is very likely to cause interoperability problems in the long term.

Hybrid services, combining VNFs, CNFs, and PNFs, are expected to remain for many years. Cloud-Native brings many benefits, but it is still a novel, and hence an immature concept.

Further work is needed not just for integrating existing MANO frameworks with Cloud-Native technologies, but for delivering specifications agreed by all the actors involved in the NFV ecosystem, as well as for attracting both developers and market towards this new paradigm.

**Acknowledgements** This work is part of ASSIST-IoT project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement 957258.

## References

1. Cao, K., Liu, Y., Meng, G., Sun, Q.: An overview on edge computing research. *IEEE Access* **8**, 85714–85728 (2020)
2. Tipantuna, C., Yanchapaxi, P.: Network functions virtualization: an overview and open-source projects. In: 2017 IEEE 2nd Ecuador Technical Chapters Meeting ETCM 2017, vol. 2017-January, pp. 1–6 (2018)
3. ETSI: GS NFV-MAN 001 Network Functions Virtualisation (NFV); Management and Orchestration (2014)
4. 5G-PPP Software Network Working Group: Cloud-Native and Verticals' services (2019)

5. Dolui, K., Datta, S.K.: Comparison of edge computing implementations: fog computing, cloudlet and mobile edge computing. In: GIoTS 2017—Global Internet Things Summit, Proc., LNCS (2017)
6. Zeydan, E., Turk, Y.: Recent advances in intent-based networking: a survey. In: 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), pp. 1–5. IEEE (2020)
7. Klaine, P.V., Imran, M.A., Onireti, O., Souza, R.D.: A survey of machine learning techniques applied to self-organizing cellular networks. *IEEE Commun. Surv. Tutorials* (19), 2392–2431 (2017)
8. 5G-PPP Technology Board Working Group and 5G-IA’s Trials Working Group: Edge Computing for 5G Networks (2021)
9. Shah, S.D.A., Gregory, M.A., Li, S.: Cloud-Native network slicing using software defined networking based multi-access edge computing: a survey. *IEEE Access* (9), 10903–10924 (2021)

# Prognosis of Parkinson's Malady—A Multimodal Approach



Shanvi Sharma and Pratima Singh

**Abstract** The increment in elderliness is directly proportional to the enhancement in Parkinson's disease patients. Lamentably, the authentic and prompt prognosis of PD plays a big challenge in emergent nations on the grounds of lack of resources and alertness. Moreover, the symptoms of PD patients are not identical nor do they all occur at the same stage of the disease. Thus, in our study, we have suggested a joint prototype by analyzing and conducting tests of three main symptoms, namely tremor analysis, dysphonia analysis and writing analysis test of PD occurring in the initiatory phase of the disease to predict the disease in the early stages and control it accordingly. This model can also prove to be a boon to patients living in areas, where qualified neurologists are not immediately accessible.

**Keywords** Parkinson's disease · Tremor · Dysphonia · Handwriting analysis · Machine learning · Telemonitoring

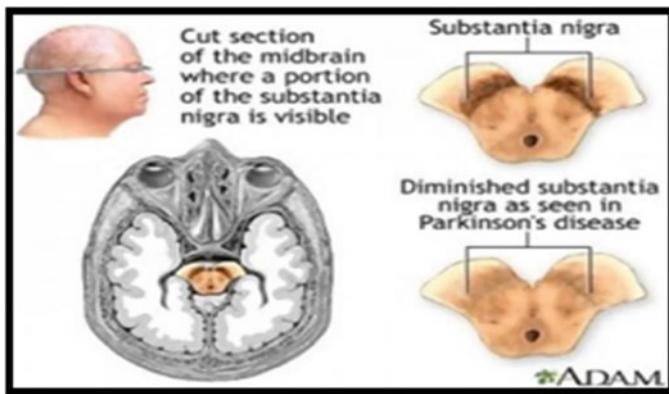
## 1 Introduction

Parkinson's disease (PD) is a neurodegenerative disease, a repercuSSION of necrosis of dopaminergic agents, cells of the stroma, which are a configuration of the basal ganglia positioned in the midbrain. Such neurology sickness profoundly influences the excellent of lifestyles of sufferers and their families [1]. Age is taken into consideration to be one of the maximum essential chance elements that specify why PD is frequently visible in human beings over the age of 50. The prognosis system of PD makes use of neurological assessments and mind scans to diagnose. These traditional approaches are out of reach of local people and are highly costly, thus requiring high qualifications.

---

S. Sharma · P. Singh  
AKGEC, Ghaziabad, Uttar Pradesh, India  
e-mail: [shanvi2010007m@akgec.ac.in](mailto:shanvi2010007m@akgec.ac.in)

P. Singh  
e-mail: [singhpratima@akgec.ac.in](mailto:singhpratima@akgec.ac.in)



**Fig. 1** Causes of Parkinson's malady [2]

Currently, there are only symptomatic treatments for PD, and starting treatment at a later stage will not help much when the disease is advanced. Therefore, timely detection of PD is critical for early management and allows the implementation of neuroprotective strategies earlier in the pathological processes, when available (Fig. 1).

It is observed that most people with PD are physically and emotionally exhausted. They even experience boredom, problem concentrating, spasms, etc. PD has a huge variety of scientific functions starting from motion to non-motor signs [3]. Some motor signs are reduced voice, rigidity, and tremor at rest. Non-motor signs are hallucinations, depression, constipation, sleep disturbances, impaired cognition, and impulse manipulate troubles. In maximum cases, medical doctors have problem thinking about whether or not a given affected person has been tormented by ailment or is possibly to broaden Parkinson's ailment [4]. To triumph over this, the improvement of certain computer fashions needs to be finished to assess and summarize the records of an affected person and are expected to give sufficient accuracy that he/she can be able to broaden PD (Fig. 2).

This paper proposes a multimodal scheme, where for the primary time, we are developing a model with clinical accuracy that can analyze tremor, speech impairment, and handwriting detection in PD patients in few minutes and helps for the early detection of the disease and further predicting its stages (Table 1).

## 2 Literature Survey

Predicting Parkinson's malady is one of the important issues that must be detected at the early phases of the disease for the purpose of reducing the rate of disease progression in the individual. Much research has been done to find the root cause,

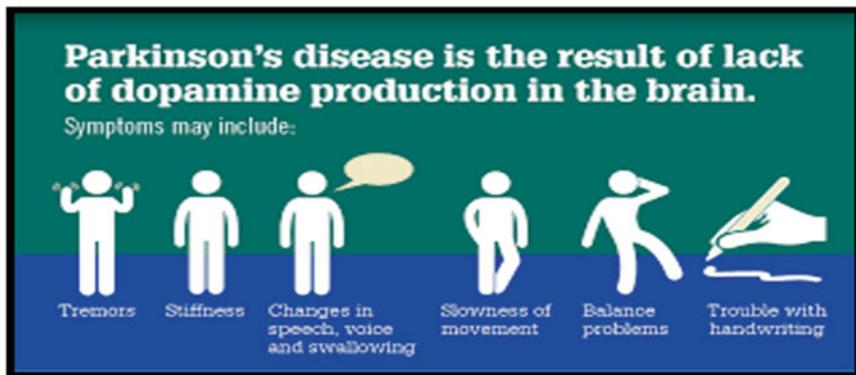


Fig. 2 Symptoms of PD [5]

**Table 1** Stage assignment to PD patients after prognosis

Updrs	Tremor	Speech	Handwriting
0	Absent	No problem	Absent
1	Slightly present	Dampen tone yet understandable	HC subject
2	Mild	Dampen tone with shadowy words	Low state
3	Moderate	Speech is poorly inferred	intermediate state
4	Severe	Most speech is unfathomable	Severe state

and some have reached new heights in recommending a system that helps distinguish healthy people from those with NDS using other machine learning techniques together. Many preprocessing, feature selection, and classification techniques have been implemented and developed over the past decades. Here is the donation work done in predicting Parkinson's disorders (Table 2).

### 3 Proposed Methodology

The proposed method is a combination of a resting tremor test, a speech disorder analysis test, and a handwriting analysis test performed on PD patients as well as on healthy people to train and reinforce the machine learning model. The most accurate models were then applied to predict the disease stage of patients with suspected PD. Hence, a qualified neurologist is to be consulted accordingly.

**Table 2** Summary of the survey of various methodologies and their performances

S. No.	Author	Year	Input	Methodology	Accuracy (%)
1	Leandro A. Passos	2018	Handwriting, speech	OPF classifier, RESNET- 50	96
2	Thomas J. Hirschauer	2015	Speech	Enhanced PNN	98.6
3	Deepak Gupta	2018	Handwriting, Speech	Optimized Cuttlefish algorithm	94
4	Hui-Ling Chen et al	2016	Speech	Extreme ML, Kernel ELM	96.47
5	Bo Penga et al	2015	Speech	t-test, mRMR	95
6	DeryaAvci and Akif Dogantekin et al	2016	Speech	Wavelet kernel, ELM	96.81
7	Ligia Sousa et al	2019	Speech	DNN, PCA, KNN	93.4
8	Othman Ibrahim	2016	Speech	PCA	96.23
9	Oana German et al	2015	Speech	SVM, DNN	90
10	Tarigoppula VS Sriram	2013	Speech	SVM, RF, NB	97
11	Indrajit Mandal	2017	Speech	Multinomial LR, ANN, boosting methods	100
12	Ipsita Bhattacharya	2010	Speech	SVM, Kernel	96
13	Mohammad S Islam	2014	Speech	Feed-forward back propagation built ANN	90
14	Hui-Ling Chen et al	2013	Speech	FKNN, SVM	96.07
15	Athanasiou Tsanas	2012	Speech	Speech signal processing	99
16	Resul Das et al	2010	Speech	Neural network, DT	92.9
17	Max A. Little et al	2009	Speech	SVM	91.4
18	Zachary C. Lipton	2016	Speech	LSTM-RNN, MLP	Comparison of several accuracies
19	Freddie Astrom	2011	Speech	NN	88.4
20	C. Okan Sakar and Olcay Kursun	2010	Speech	SVM	92.75
21	Marius Ene et al	2008	Speech	PNN	Range: 79–81

(continued)

**Table 2** (continued)

S. No.	Author	Year	Input	Methodology	Accuracy (%)
22	Ali H AL Fatlawi	2016	Speech	DBN, ANN, restricted Boltzmann machine	94
23	Chien Wen Cho	2003	Speech	PCA	95.49
24	Magnus Johnson B, David Gil A	2009	Speech	ANN, SVM	90

### 3.1 Tremor Detection Test

Tremor is also known as shaking. At the beginning of the disease, tremor starts on one side of the body and then gradually increases to all over the body as the disease worsens. Tremor test is done to analyze the intensity of tremor in the body of PD patient and is classified into high-amplitude tremor (HAT) and low-amplitude tremor (LAT).

#### Data Collection

The data is collected from the database of PhysioBank [6]. The database embodies recordings having files containing vibrational readings of about 1 s. All the readings are taken in millimeters per second.

#### Data Preprocessing

The data is still not fit to be applied on the model directly. It requires some preprocessing. Thus, to normalize the data, we have applied min-max feature scaling technique which rescaled the data into millimeter per second [7].

#### Methodology

Fast Fourier transform is implemented to derive the power spectrum. The power spectrum (PS) of a biological time series (such as an electroencephalogram) describes the directly proportional relationship of power with frequency ( $f$ ) resulting in decrement in power leading to decrease in frequency [8].

The general equation can be given as:

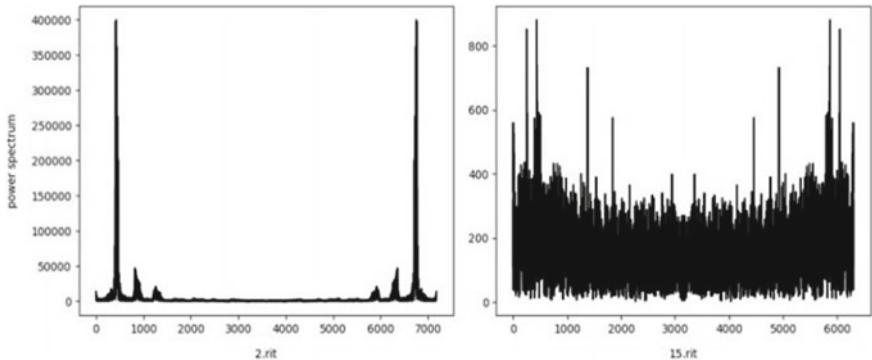
$$PS(f) = \psi \times f^{-\alpha} \quad (1)$$

where

PS Power Spectrum

$f$  Frequency

$\alpha$  Exponent



**Fig. 3** Power spectrum graphics of subjects sorted into HAT and LAT

The FFT works by disintegrating an N-point temporal aspect signal into N-temporal aspect signals, each consisting of a unique point [9]. After decomposition, the N frequency spectra is calculated complementary to these N time-domain signals. Finally, the N spectra are blended into a unique frequency spectrum. To perform disintegration, we have used bit reversal algorithm in place of interlaced decay as it transforms signals in n time as compared to interlaced decomposition which takes  $n \log n$  time complexity [10].

### Feature Extraction

The features found out which highly discerns between HAT and LAT are:

- Signal reckon
- Mean power spectral density
- Error rate of power spectrum (Fig. 3).

## 3.2 *Speech analysis test*

### Data Collection

Two approaches are followed for collecting the data

- The voice data has been taken from open-source platform Kaggle [11].
- Another approach followed is where PD patients are asked to vocalize the vowels for 15 s in a single breath by placing their mobile's sound recorder 8 cm away from mouth (Fig. 4)..

### Data Preprocessing

The initial data is not fit to be applied on the model. Thus, some preprocessing is required. Spontaneous phonation results in high frequency at the beginning and

A	B	C	D	E	F	G	H	I	J	K	L	
1	MDVP:Fo MDVP:Fhi MDVP:Flo MDVP:Jitt MDVP:Jitt MDVP:RA MDVP:PPQ Jitter:DDP MDVP:Shi MDVP:Shi Shimmer: Shimmer:											
2	119.992	157.302	74.997	0.00784	0.00007	0.0037	0.00554	0.01109	0.04374	0.426	0.02182	0.0313
3	122.4	148.65	113.819	0.00968	0.00008	0.00465	0.00696	0.01394	0.06134	0.626	0.03134	0.04518
4	116.682	131.111	111.555	0.0105	0.00009	0.00544	0.00781	0.01633	0.05233	0.482	0.02757	0.03858
5	116.676	137.871	111.366	0.00997	0.00009	0.00502	0.00698	0.01505	0.05492	0.517	0.02924	0.04005
6	116.014	141.781	110.655	0.01284	0.00011	0.00655	0.00908	0.01966	0.06425	0.584	0.0349	0.04825
7	120.552	131.162	113.787	0.00968	0.00008	0.00463	0.0075	0.01388	0.04701	0.456	0.02328	0.03526
8	120.267	137.244	114.82	0.00333	0.00003	0.00155	0.00202	0.00466	0.01608	0.14	0.00779	0.00937
9	107.332	113.84	104.315	0.0029	0.00003	0.00144	0.00182	0.00431	0.01567	0.134	0.00829	0.00946
10	95.73	132.068	91.754	0.00551	0.00006	0.00293	0.00332	0.0088	0.02093	0.191	0.01073	0.01277
11	95.056	120.103	91.226	0.00532	0.00006	0.00268	0.00332	0.00803	0.02838	0.255	0.01441	0.01725

**Fig. 4** Sample dataset of audio recording

as the lungs pressure decreases, the frequency also becomes low. Thus, to cover authenticated part of voice, we have removed the first 3 s and last 3 s from the signal [12].

To avoid the aliasing effect which makes signal indistinguishable, we need to pass the signal from a bandpass filter with 3 DB cutoff points at 50 Hz and 8 kHz, respectively as the frequency range of human voice lies between 50 Hz and 8 kHz and any value outside this threshold value is eliminated. To reduce the computational cost, the signal is further passed through the sub-sampler to reduce the sampling effect as the sampling frequency of audio recorder is 44.1 kHz which is reduced to 22.05 kHz, higher than the Nyquist requirement for voice signal admittance [13].

## Feature Extraction

The features extracted are based on six factors:

- Amplitude
- Pitch
- Pulse
- Frequency
- Harmonicity
- Voicing (Fig. 5).

## Feature Selection and Validation

The technique followed is Boruta feature selection method [14]. In this technique, random shadow copies of original features are created, and then, these shadow copies are tested against the original features to check whether the original features are worth preserving.

Feature Number	Features	Extracted
1.	Jitter (%)	Frequency components
2.	Jitter (Absolute)	
3.	Jitter: DDP	
4.	Number of voice breaks	Voicing Parameters
5.	Degree of voice breaks	
6.	Fraction of locally unvoiced Frames	
7.	Jitter (rap)	
8.	Jitter (local, absolute)	
9.	Jitter(ppq5)	
10.	Shimmer (local)	Several measures of variation in amplitude
11.	Shimmer (dB)	
12.	Shimmer: APQ3	
13.	Shimmer: APQ5	
14.	Shimmer: DDA	
15.	Noise to harmonic Ratio	
16.	Harmonic to noise Ratio	
17.	Recurrence Period Density Entropy	Nonlinear dynamical complexity measures
18.	Detrended Fluctuation Analysis	
19.	Perceived Phonatory Effort	Nonlinear measure of fundamental frequency variation
20.	Number of Pulses	
21.	Number of Period	
22.	Mean Period	

**Fig. 5** Features extracted from voice samples

Then, the importance of each original feature is taken and compared with a threshold value. Threshold here is defined as the highest feature value observed among shadow features. If the value of original feature is beyond this threshold value, then it is a hit or otherwise it is amiss.

### 3.3 Handwriting Detection

Handwriting detection is an important observation in early prognosis of PD. Advancement in IT has made it easier to assemble spiral drawings using digitized tablet. In this study, we have proposed analysis of static spiral test (SST) and dynamic spiral test (DST). For this purpose, we have used mathematical prototype to evaluate kinematic variants [15].

The steps involved in integration of analyzed model are:

- Feature extraction, converting spiral content into numerical vector.
- Feature combination, combining SST and DST tests.

- A mathematical prototype separating the traces of PD patients from healthy people.

## Data Collection

The dataset has been collected from the open platform repository UCI [16]. The database is comprised of joint tests conducted on patients suffering from PD as well as healthy individuals educated through the Department of Neurology at Cerrahpasa Medical School, University of Istanbul.

## Procedure

Every one of all ages draws three different pictures on a digitized tablet. These three types of spiral drawing tests are based on the scientific basis for assessing motor disorders and include:

**Static Spiral Test(SST):** Three scratches on the Archimedes spiral are displayed on the graphic tablet and the patient is asked to track it using a digit alpen.

**Dynamic Spiral Test(DST):** In contrast to the static Archimedes spiral, this test shows and hides the spiral only at specific time frames. The subject should remember the precedents in mind and track the blinking spiral. This determines the patient's pattern storage competencies and the pause time the patient needs when tracing the identical pattern.

**Wave motion test:** The focus of the screen contains a red mark, and the subject is asked to trace a wave without palpating the screen. This identifies the balance of the patient's hand.

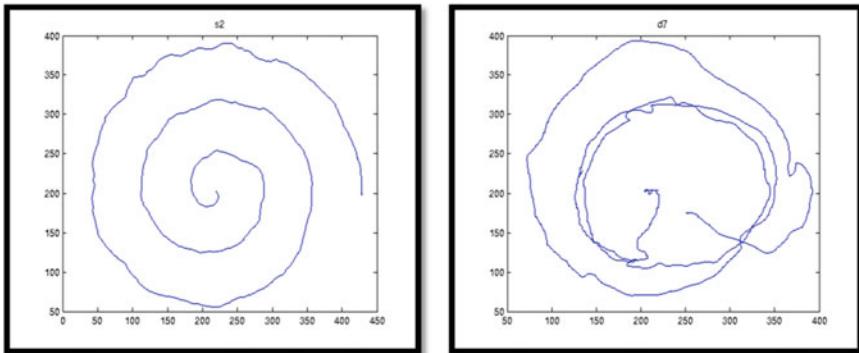
During testing, dedicated software was developed using API and C# records drawing information. Drawing is PNG images and text files (CSV format) from SST and DST is attached to the dataset. The suggested work uses the same set of text files. The handwritten dataset was created using the Wacom Cintiq 12WX graphic table [16]. It is a combination of a graphic tablet and an LCD monitor. This allows the PC screen to be displayed on the monitor and only engages with the digitized pen.

The recorded digital measurements are x and y coordinates, pressure axis in “3 axes” recording, pressure (less than unit), stylus grip angle, timestamp, test ID (SST0, DST1, WMT2). The geometric role of the pen at as elected time stamp ( $x, y$ ), the strain at the writing surface, the lean of the pen, and whether or not the motion of the pen is “with inside the air” are kinematic expressions. The time it takes to finish the drawing may be calculated by the use of the person stroke time stamps recorded w.r.t any spiral drawings (Fig. 6).

## Feature Extraction

SST and DST illustrations are analyzed in reference to speed and instantaneous acceleration when speed changes suddenly.

Instantaneous velocity,  $V$ , at time  $t$  in the process of drawing:



**Fig. 6** Static spiral test drawing and dynamic spiral test drawing

$$V_{int} = \lim \Delta t \rightarrow 0 \Delta x / \Delta t = dx/dt \quad (2)$$

Instantaneous acceleration,  $a$ , at time  $t$  is calculated as:

$$a(t) = dv(t)/dt. \quad (3)$$

The analysis on following features was done:

- Number of strokes
- Running speed
- Radial speed
- Centripetal acceleration
- Shake
- Parallel speed/acceleration
- Plumb speed/acceleration
- Change of direction of speed
- Change of direction of acceleration
- In the air time
- on facade time (Fig. 7).

### Feature Selection and Validation

Various sifting procedures such as correlation, information gain, profit ratio, mitigation, and symmetric indeterminacy principle as well as minimum redundant maximum match (mRMR) technique were used to evaluate the patient's condition [17].

B	C	D	E	F	G	H	I
no_stroke	no_stroke	speed_st	speed_dy	magnitude	magnitude	magnitude	magnitude
12	2	0.000293	0.000431	0.061342	0.038319	0.03905	0.084891
4	6	0.000286	0.000281	0.119159	0.077012	0.074216	0.160497
4	4	0.000278	0.000277	0.113889	0.071547	0.073336	0.191965
4	4	0.000286	0.000275	0.154621	0.101146	0.0963	0.152413
2	2	0.000279	0.00027	0.060058	0.039449	0.036737	0.081577
2	2	0.000251	0.000238	0.199831	0.128719	0.124814	0.275863
4	2	0.005701	0.005063	0.173365	0.138019	0.076137	0.344131
2	2	0.004495	0.004138	0.282933	0.232494	0.12439	0.233773
4	2	0.003696	0.00344	0.111742	0.090988	0.04685	0.143555
2	2	0.002756	0.003645	0.312699	0.249655	0.14067	0.424141

**Fig. 7** Features extracted from handwriting dataset

## 4 Implementation

After extracting, all features in the above performed tests and deciding on the greater capable ones, five machine learning algorithms mentioned below are applied on the selected features, respectively, and comparative analysis is made.

- Random forest
- Logistic regression
- Naïve Bayes
- KNN
- SVM

An ensemble learning technique called XGBoost is also applied on speech test to find more accurate results.

## 5 Result and Analysis

### 5.1 Speech Analysis Test Results

The combination of linear and nonlinear features has been taken in the speech dataset to analyze the intensity of impairment in voice of PD patients and further their correlational values have been found. Accordingly, the mean, standard deviation, maximum, and minimum amplitude of wave have been calculated. XGBoost algorithm has proved to give the highest accuracy and the k-nearest neighbors algorithm had the worst effect (Figs. 8 and 9; Table 3).

Out[5]:	MDVP:Fo(Hz)	MDVP:Fhi(Hz)	MDVP:Flo(Hz)	MDVP:Jitter(%)	MDVP:Jitter(Abs)	MDVP:RAP	MDVP:PPQ	Jitter:DDP	MDVP:Shimmer	MDVP:Shimmer(dB)
count	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000	195.000000
mean	154.228641	197.104918	116.324631	0.006220	0.000044	0.003306	0.003446	0.009920	0.029709	0.282251
std	41.390065	91.491548	43.521413	0.004848	0.000035	0.002968	0.002759	0.008903	0.018857	0.194877
min	88.333000	102.145000	65.476000	0.001680	0.000007	0.000580	0.000920	0.002040	0.009540	0.085000
25%	117.572000	134.862500	84.291000	0.003460	0.000002	0.001660	0.001860	0.004985	0.016505	0.148500
50%	148.790000	175.829000	104.315000	0.004940	0.000030	0.002500	0.002690	0.007490	0.022970	0.221000
75%	182.769000	224.205000	140.018500	0.007365	0.000060	0.003835	0.003955	0.011505	0.037885	0.350000
max	260.105000	592.030000	239.170000	0.03160	0.000260	0.021440	0.019580	0.064330	0.19080	1.302000

8 rows × 23 columns

**Fig. 8** Description of various speech features used for analysis of voice in PD patients

Out[8]:	status	1.000000
spread1		0.564838
PPE		0.531039
spread2		0.454842
MDVP:Fo(Hz)		0.383535
MDVP:Flo(Hz)		0.388200
MDVP:Shimmer		0.367430
MDVP:APQ		0.364316
HNR		0.361515
Shimmer:APQ5		0.351148
MDVP:Shimmer(dB)		0.350697
Shimmer:APQ3		0.347617
Shimmer:DDA		0.347608
D2		0.349232
MDVP:Jitter(Abs)		0.338653
RPDE		0.308567
MDVP:PPO		0.288698
MDVP:Jitter(%)		0.278220
MDVP:RAP		0.266668
Jitter:DDP		0.266646
DFA		0.231739
NHR		0.189429
MDVP:Fhi(Hz)		0.166136
Name: status, dtype:	float64	

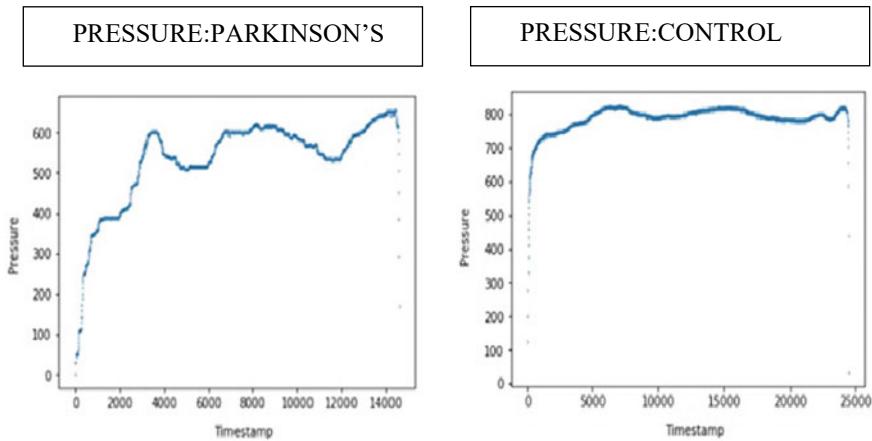
**Fig. 9** Correlational values**Table 3** Accuracies of different algorithms applied on model

Accuracy	SVM	NB	RFC	KNN	LREG	XGBOOST
%	67.7	75.7	87.02	58.5	68.2	94.91

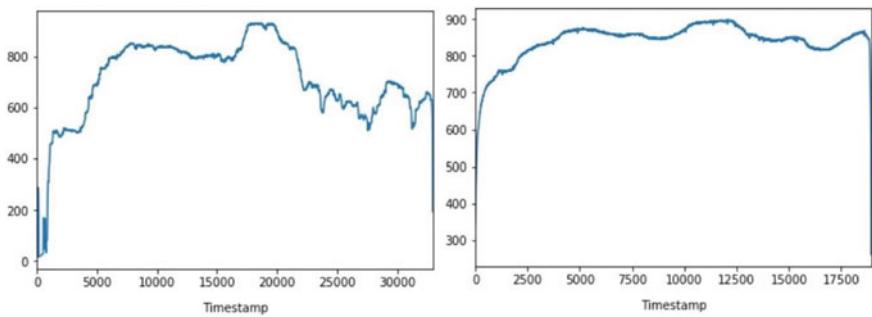
## 5.2 Handwriting Test Analysis

The results clearly differentiates between the various factors such as pressure exerted by the PD patient as compared to the healthy individual, stroke speed of PD patient compared to the healthy individual, etc. while drawing on the graphics tablet. Among the five different algorithms, applied SVM is giving best results. Thus, based on the model applied, we have analyzed both the spiral and wavy drawings of subjects and hence classified the patients as Parkinson's sufferer or healthy individual (Figs. 10 and 11; Table 4).

Thus, on comparison, SVM classifier gave the best result with ACCURACY: 100%; F1 SCORE: 0.66; PRECISION: 0.75; RECALL: 1 (Figs. 12, 13 and 14)



**Fig. 10** Sample pressure difference between affected and normal in handwriting test



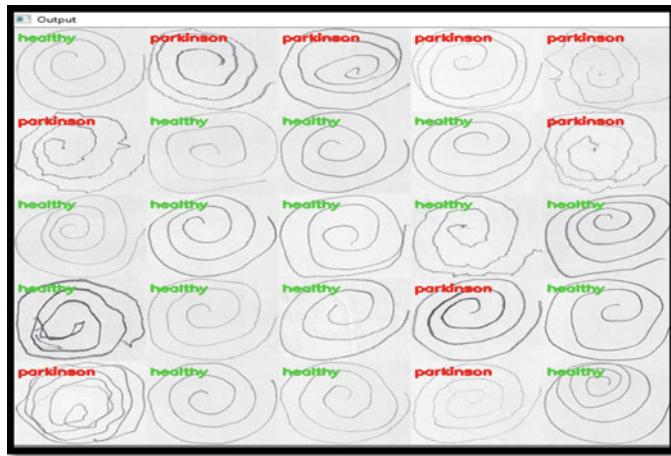
**Fig. 11** Sample stroke difference between affected and normal in handwriting test

**Table 4** Different classifiers used with their results

S. No.	Classifier	Accuracy	Recall	F1	Precision
1	Logistic regression	70.0	0.625	0.55555555555556	0.66
2	Random forest	100.0	0.9	0.66666666666666	0.5
3	Support vector machine	100.0	1.0	0.66666666666666	0.75
4	Decision tree	100.0	0.8	0.66666666666666	0.5
5	K-nearest neighbors	60	0.571425714251742	0.4705882352941	0.4



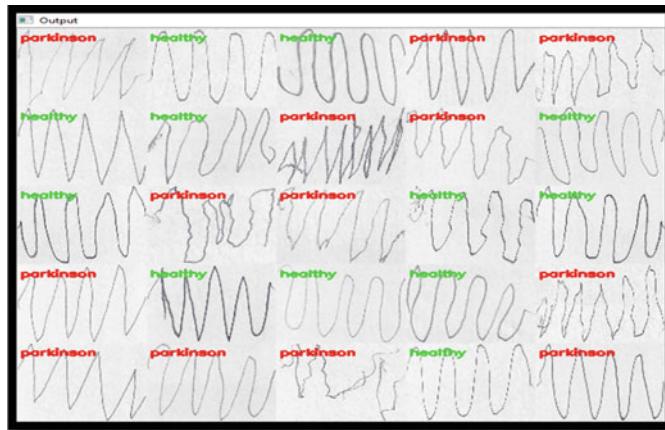
**Fig. 12** Comparative PR curve for distinct classifiers



**Fig. 13** Handwritings of PD patient and healthy individual categorized in spiral test

## 6 Conclusion

This paper is the amalgamation of research work done on early detection of Parkinson's malady by conducting tremor test, speech impairment test and handwriting test, respectively. Various ml algorithms are applied on individual models and SVM gave the highest accuracy in handwriting test, whereas XGBoost gave highest accuracy in audiometry test. The results section contains the observations of speech impairment test and handwriting analysis test, while the results of tremor test are kept for future analysis. This model can help the patients in areas, where qualified neurologists are not immediately accessible. This model will prove as a



**Fig. 14** Handwritings of PD patient and healthy individual categorized in wave test

boon to patients suffering from Parkinson's disease to control the disease in an early stage.

## References

1. Elbaz, A., Bower, J.H., Peterson, B.J., Maraganore, D.M., McDonnell, S.K., Ahlskog, J.E., Schaid, D.J., Rocca, W.A.: Survival study of Parkinson disease in Olmsted County, Minnesota. *Arch Neurol.* **60**, 91–96 (2003)
2. Tanner, C.M., Ross, G.W., Jewell, S.A.: Occupation and risk of Parkinsonism: a multicenter Claas Ahlrichs seal, Parkinson's disease motor symptoms in machine learning: a review. *Health Inf. Int. J. (HIIJ)* **2**(4), (2013)
3. Surathi, P., et al.: Research in Parkinson's disease in India: a review. *Ann Indian Acad. Neurol.* **19**(1), 9–20 (2016). <https://doi.org/10.4103/0972-2327.167713>
4. Symptoms of Parkinson's Disease and Movement disorder. <https://www.froedtert.com/sites/default/files/image/2021-04/parkinsons-disease-symptoms-infographic-1200x628.jpg>
5. Triarhou, L.C.: Dopamine and Parkinson's disease. Madame Curie Bioscience Database (2000–2013)
6. Ramananand, R.G., Sivagami, G.: Parkinson disease classification using data mining algorithms. *Int. J. Comput. Appl.* **32**(9), 17–22 (2011)
7. Hillerkuss, D., Winter, M., Teschke, M., Marculescu, A., Li, J., Sigurdsson, G., Worms, K., benEzra, S., Narkiss, N., Freude, W., Leuthold, J.: Simple all-optical FFT scheme enabling Tbit/real-time signal processing. *Opt. Express* **18**(9), 9324 (2010). <https://doi.org/10.1364/OE.18.009324>
8. Palimkar, P., et al.: Machine learning technique to prognosis diabetes disease: random forest classifier approach. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems*, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_19](https://doi.org/10.1007/978-981-16-2164-2_19)
9. Vander Plas, J.: *Python data science handbook: essential tools for working with data*. O'Reilly Media, Inc. (2016)

10. Chakraborty, A., et al.: A comparative study of myocardial infarction detection from ECG data using machine learning. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_21](https://doi.org/10.1007/978-981-16-2164-2_21)
11. Ho, A., et al.: Speech impairment in a large sample of patients with Parkinson's disease. *Behav. Neurotol.* **11**, 131–137 (1998)
12. Little, M.A., McSharry, P.E., Hunter, E.J., Spielman, J., Ramig, L.O.: Suitability of dysphonia measurements for telemonitoring of Parkinson's disease. *IEEE Trans. Biomed Eng.* **56**(4), 1015–1022 (2009)
13. Sukhanov, V.A., Ionov, I.D., Piruzyan, L.A.: Neurodegenerative disorders: the role of genetic factors in their origin and the efficiency of treatment. In: Proceedings of the Human Physiology US National Library of Medicine National Institutes of Health, vol. 31, pp. 472–482 (2005)
14. Avci, D., Dogantekin, A., et al.: An expert diagnosis system for Parkinson disease based on genetic algorithm-wavelet kernel- extreme learning machine. Hindawi Publishing Corporation Parkinson's Disease, Vol. 2016, 9p (2016). Article ID5264743. <https://doi.org/10.1155/2016/52647>
15. Eskidere, O., et al.: AComparisonofregressionmethodsforremotetrackingof-Parkinson'sdiseaseprogression. *Experts Syst. Appl.* **39**, 5523–5528 (2012)
16. Aghanavesi, S., et al.: Verification of a method for measuring Parkinson's disease related temporal irregularity in spiral drawings. *Sensors* **17**(10), E2341 (2017). <https://doi.org/10.3390/s17102341>
17. Wtal, Drotar, P.: Decision support framework for Parkinson's disease based on novel hand writing marker. *IEEE Trans. Neural Syst. Rehabil. Eng.* **23**(3), 508–516 (2015)

# A Study on Conversion of Image Caption to Speech Using Neural Network Models



Swati Shinde and Prachi Waghmare

**Abstract** In this world of artificial intelligence, the technologies are getting advanced in all fields, including captioning of images is also very trending technology it is useful in all fields. Captioning of images will be performed using neural network models based on feature extraction to get appropriate caption for given image. The paper is the study of the image caption generation and the conversion of captions into speech format. The models used for this conversion are CNN, RNN, LSTM, and how captioning can be converted into speech format is explained. The conversion of image caption is most useful for many application and blind people. The blind people does not need to depend on other person to perform their regular activities as guidance device can be used, using this conversion technology.

**Keywords** Image captioning · CNN · RNN · Long short-term memory · Feature extraction · Text-to-speech synthesis

## 1 Introduction

In field of advance technology, captioning of images is most important as human can easily explain the image just by quick vision, but the same working as human brain is difficult for machines [1]. As machine works on command, each and every task should be easily understood by machine. Image captioning is the process where image is given as input and each image undergoes different process like features are extracted from image, and considering each object from image, the detected objects are converted into sentence format using LSTM [2, 3]. Image captioning is not only generation of captions but a proper relationship among the features extracted from image should be established [4].

The image caption is performed using various models like CNN, RNN and LSTM. Once the caption is generated for given image that image is converted into speech format using “text-to-speech” conversion technique [5, 6]. This technique is mostly

---

S. Shinde · P. Waghmare (✉)

Pimpri Chinchwad College of Engineering Computer Department, Nigdi, Pune, India

e-mail: [Prachiwaghmare0109@gmail.com](mailto:Prachiwaghmare0109@gmail.com)

useful for blind peoples [7]. Many guidance devices are available for blind people which captures scenes from their surrounding and generates captions, these captions are converted into speech format, this conversion could be easily heard by people, and it becomes easy to do their work in day-to-day life [8, 9]. And they do not need to depend on other peoples and become independent to do their daily work.

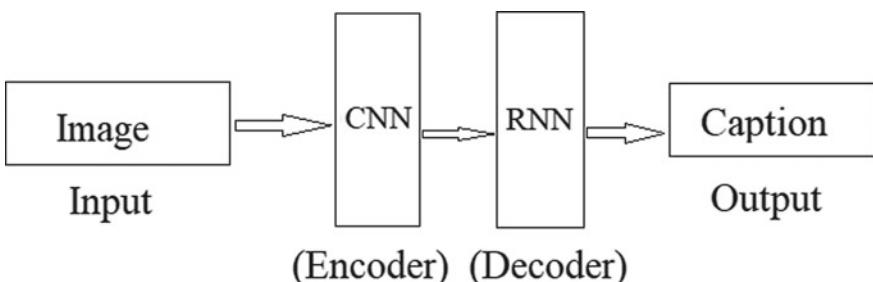
## 2 Image Caption Architecture

Figure 1 is the basic architecture of captioning [10]. Basically, image caption consists of neural network model such as convolutional neural network, recurrent neural network and LSTM. We will see each model in detail and actual work of models for image caption generation.

### 2.1 Convolution Neural Network

The CNN is a type of deep neural networks that are used to classify and recognize the features from an image. There are three main layers of CNN: They are convolutional layer, pooling layer and fully connected layer.

- (1) **Convolution layer**—This layer is used to extract the features from given input image. Here the mathematical operation is performed where a particular size of window like  $M \times M$  is considered which slides over the input image until the complete input image is scanned to get feature mapped output.
- (2) **Pooling layer**—The aim of this layer is to reduce the size of feature mapped output and reduce the computational cost [11]. Usually, this layer works as bridge between the convolution and fully connected layer.
- (3) **Fully connected layer**—The layer is a simple feed forward neural network, which forms last few layers in the network [12]. The input to this layer is output



**Fig. 1** Basic image caption architecture

from pooling or convolution layer which is flattened and then given to fully connected layer.

## 2.2 Recurrent Neural Network

Recurrent neural networks are type of neural network where the output from previous step is given as input to new step [13]. Each time process is repeated, because in order to generate a sentence it is necessary to predict next word depending on previous word to establish a proper relationship among the word and generate a meaningful caption for given input image [11].

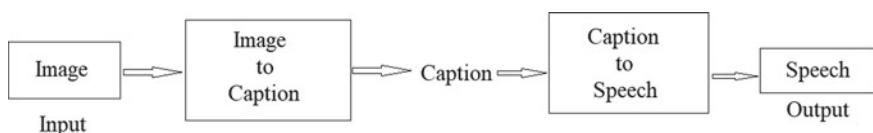
## 2.3 Long Short-Term Memory

The LSTM is extension of RNN which basically extends the memory, which enables the RNN to remember the input over long period of time because it contains information in memory similar like memory in computers [14]. It considers each word extracted as object from previous layer, arranges each word with proper sequence and generates a final caption as input [12, 13].

## 3 Image Caption to Speech Conversion Architecture

Figure 2 explains the architecture to convert caption to speech [8]. The above image is the basic diagram of converting the caption into speech using the text-to-speech synthesizer (TTS) technology, with following steps:

- (1) The first step is to give the image as input to system, then this image goes under various preprocessing steps like resizing, removing noise from image, to get accurate results for given input image [14].
- (2) Then this image is scanned from left to right by sliding a window on whole image of particular  $M \times M$  size [10], which extracts the features from image considering all the important objects in given image which is performed by CNN model [11].



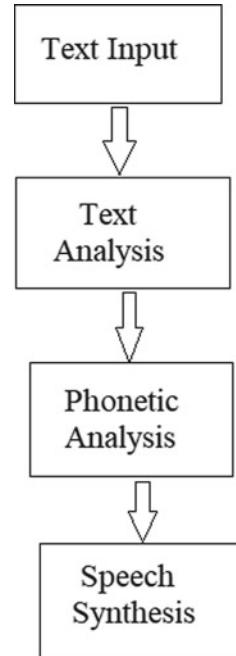
**Fig. 2** Image caption to speech conversion architecture

- (3) The RNN with LSTM considered this each feature in separate word to form a caption. The LSTM considers each previous output with current output to form a natural English sentence as discussed in the above section of image caption and generates appropriate and meaningful caption for given input image [3, 10].
- (4) Up to this, the model works for image captioning; now once the caption is generated, the caption is given as input to system, this caption undergoes various preprocesses, and the TTS system converts the caption into speech format [9, 12].
- (5) The final output of the system is in audio format (speech) [8]. So this is how the image is first converted into caption (text) format, and then this caption is converted into speech (audio) format. This system plays a vital role for blind peoples into their routine even though they cannot see the image, but by hearing the voice, they can actually understand what includes in the image.

### 3.1 Text-to-Speech Synthesis (TTS) System

Figure 3 is the basic overview of TTS system [2]. The TTS system is the technology which allows the computer system to speak like a human. The main aim of the system is to convert the text into voice format [7, 15]. The text is given as input to the system, and then by applying some algorithms of computer called as TTS engine

**Fig. 3** Basic TTS system



**Table 1** Table of TTS system

Features	Applications	Benefits	Drawbacks
(1) It should have grammatically high accuracy to read out correctly (2) System should support multiple languages (3) It should be user friendly for users (4) Fluent speaking to read out long text (5) Able to run on major platforms with support of voice XML [15]	(1) The system is used for walking device of blind peoples (2) The automatic reading of devices screen [15] (3) It is used during announcement of railways and airways information (4) The vehicles which are controlled with voice (5) It is widely used in robotics field [15]	(1) It is beneficial in education fields (2) It helps in learning unknown languages [15] (3) It is very useful for student to listen class notes or book as they can learn quickly and do not get bored of reading [15] (4) It is useful for senior age people who have vision problem due to age (5) It avoids eyestrain of reading large paragraphs from books or computer	(1) The system is time consuming as it requires large database and hard coding techniques (2) It does not give actual emotions of words like human (3) The main problem is of pronunciation from wrong written text (4) It is difficult to build accurate system (5) Filtering noise from background is difficult task [15]

which analyzes, it preprocesses the text and synthesizes the speech. The TTS engine usually generates audio data as output [6, 7].

The system basically consists of two main phases: First is text analysis where text is preprocessed and converted into phonetic representation, and the second is generation of speech waveform to get output in voice format [7]. The above image is systematic flow where the text is given as input, and then text analysis is done and then phonetic analysis which has ability to recognize sound relationship in order to identify the word and generates a final output as speech (audio) for given text [5].

#### 4 Features, Applications, Benefits and Drawbacks of Text-to-Speech Conversion

Table 1 explains some of the features and application of text-to-speech conversion and also advantages and disadvantages of this technology, the efforts taken to build this technology and how it is useful in our life [15].

#### 5 Result and Discussion

The paper is the study of image captioning and how conversion of caption is done into speech is understood using advanced technology. The result is actual understanding

of different neural models used in captioning process. And actual importance of TTS system in text-to-speech conversion process is understood. The study of general features, their advantages and disadvantages and how it is useful in our day-to-day life. In future, the system could be made more accurate, and more detail research could be done to reduce the limitations of current systems.

## 6 Conclusion

The paper is the deep study of image caption generation, the requirement of deep neural networks is explained and using this how appropriate and meaningful caption is generated by establishing a proper relationship among the extracted features from given input image. And how these captions are converted into speech format using TTS system. This technique of converting image to caption, and caption to speech is very useful for specially abled people like blind people and how it is helpful for them in their daily routines without depending on another person is explained.

## References

1. Thu Thu, C.S., Zin, T.: Implementation of text to speech conversion. (2014)
2. Sasirekha, D., Chandra, E.: Text to speech: a simple tutorial. (2012)
3. Shinde, S.V.: Mining the classification rules from database using artificial neural network. (2013)
4. Rawale, S., Ghotkar, M., Sonavane, K., Surve, P., Khonde, S., Patil, D.: IMAGE Captioning generator system with caption to speech conversion mechanism. (2021)
5. Swetha, N., Anuradha, K.: Text-To-Speech conversion. (2013)
6. Mridha, K., et al.: Deep learning algorithms are used to automatically detection invasive ducal carcinoma in whole slide images. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 123–129. <https://doi.org/10.1109/ICCCA52192.2021.9666302>
7. Isewon, I., Oyelade, J., Oladipupo, O.: Design and implementation of text to speech conversion for visually impaired people. (2014)
8. Ma, S., McDuff, D., Song, Y.: Unpaired image-to-speech synthesis with multimodal information bottleneck. SUNY Buffalo Buffalo, NY, Microsoft Redmond, WA
9. Shinde, S., Kulkarni, U.: Extended fuzzy hyperline-segment neural network with classification rule extraction. (2017)
10. Mridha, K., et al.: Plant disease detection using web application by neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 130–136 (2021). <https://doi.org/10.1109/ICCCA52192.2021.9666354>
11. Katiyar, S., Borgohain., S.K.: Comparative evaluation of CNN architectures for image caption generation. (2020)
12. Chen, X., Lawrence Zitnick, C.: Mind's eye: a recurrent visual representation for image caption generation. (2015)
13. Goyal, S.B., et al.: Multi-objective fuzzy-swarm optimizer for data partitioning. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_25](https://doi.org/10.1007/978-981-16-2164-2_25)

14. Chen, S. Jin, Q., Wang, P., Wu, Q.: Say As You Wish: fine-grained control of image caption generation with abstract scene graphs. (2020)
15. <https://www.rfwireless-world.com/Terminology/Advantages-and-Disadvantages-of-Text-to-SpeechConversion.html#:~:text=Following%20are%20the%20drawbacks%20or,less%20han%20natural%20and%20emotionless>

# Automated Detection of Hypertension Disease Using Machine Learning and Signal Processing-Based Methods



Jaypal Singh Rajput and Manish Sharma

**Abstract** Hypertension is a critical health issue and an important area of research because of its high pervasiveness and a remarkable risk factor for cardiovascular and cerebrovascular disease. However, it is a silent killer in all respects. Hardly any side effect can be found in its beginning period until an extreme Medical emergency like heart attack, stroke, or chronic kidney disease. Since individuals are unaware of hypertension, the identification is possible through measurement only. The detection of hypertension at the beginning stage can protect from serious health issues. Furthermore, the hypertension diagnosis by measuring blood pressure may not reflect any severe complication caused due to high blood pressure. Alternatively, automated machine learning and signal processing-based methods require to detect hypertension and its complication (syncope, stroke, and myocardial infarction) from the direct ECG signal. However, the ECG signal is non-stationary, and experts may commit mistakes in observation. As a result, the delay in the treatment of hypertension can be life threatening. Therefore, we have developed the automated detection algorithm for HPT-influenced electrocardiogram (ECG) signal using an optimal filter bank and machine learning. A total of six sub-bands were produced from each ECG signal using a filter bank. In addition, we have extracted the various linear and nonlinear features for all six sub-bands. Subsequently, a ten-fold cross-validation technique was employed for the k-nearest neighbor (KNN) classifier to classify the ECG signals. As a result, the proposed model has achieved a classification accuracy of 98.4%. Hence, the proposed work classifies hypertension from ECG signals in myocardial infarction, stroke, syncope, and low-risk hypertension. Moreover, we can install the proposed algorithm on a personal computer and diagnose the HPT-associated disease from an ECG signal.

**Keywords** Hypertension · ECG signal · Machine learning · Classification · Filter bank · Wavelet decomposition · Signal processing

---

J. S. Rajput (✉) · M. Sharma

Department of Electrical and Computer Science Engineering, Institute of Infrastructure Technology Research and Management, Ahmedabad 380026, India  
e-mail: [jaypal2020@gmail.com](mailto:jaypal2020@gmail.com)

## 1 Introduction

Hypertension (HPT) is a critical health issue. It can severely affect human health, and its pervasiveness increases on a global level, but the rate of HPT awareness, treatment, and control remains slow [11]. The World Health Organization (WHO) is more vigilant of HPT treatment, understanding, and diagnosis [11]. HPT is defined when the systolic and diastolic blood pressure is greater than 140/80 in more than three clinical trials.

Hypertension is a remarkable state that can indicate many severe diseases like stroke (STR), syncope (SYN), myocardial infarction (MI), and heart disease [6]. Blood pressure (BP), smoking, overweight, lack of exercise, excessive salty eating, stress, age, family ancestry, kidney illness, and thyroid disease are the few reasons that cause hypertension [24].

The functionality of the heart is recorded by an ECG signal in the form of an electrical signal. Therefore, the ECG signal is more relevant in HPT detection and the disease associated with it [10]. Simjanoska et al. [22] specify the relationship between ECG and blood pressure and how ECG changes when the blood pressure is changed.

The primary motivation of this research work is to detect hypertension-associated diseases from the ECG signal. However, early detection of hypertension can save many lives and enhance people's life quality.

Various devices, methods, and algorithms have been developed to detect hypertension. Similarly, the details of work done on HPT detection in literature are mentioned below:

Rajput et al. [8] discriminate the severity of hypertension ECG signal using hypertension diagnosis index (HDI). The developed HDI model classifies the low and high-risk hypertension ECG signals with 100% classification accuracy.

In another study [10], the authors classify the low, high-risk hypertension, and normal ECG signals using signal processing and machine learning-based methods. In addition to this, they obtained 99.95% classification accuracy using the ensemble bagged tree classifier.

Further, in the subsequent study [18], the classification accuracy of 98.05% was obtained without wavelet-based methods. The classification has been conducted on the severity of hypertension and normal ECG signals.

Moreover, in another study [9], they have classified hypertension and normal Ballistocardiogram signals using empirical mode decomposition and wavelet transform methods. As a result, the authors obtained the highest classification accuracy of 87%.

Quachtran et al. [7] extract intracranial pressure (ICP) from ECG signal and developed a deep learning model for the detection of intracranial hypertension. The deep learning model gives  $92.0 \pm 2.25\%$  accuracy.

Sau et al. [12] worked on seafarer people's depression and anxiety using machine learning. The precision and accuracy of their developed model are 82.6% and 84%, respectively.

In another study, Melillo et al. [5] designed an automated detection of high-risk hypertension algorithm from heart rate variability (HRV) signals using machine learning methods. The sensitivity and specificity of HRV-based models are 71.4 % and 87.8 %, respectively.

Ni et al. [6] employ a HRV signal-based multi-scale fine-grained model to detect the severity of hypertension. The HRV signal-based model gives 95% accuracy using a machine learning algorithm.

Simjanoska et al. [22] identified the SBP and DBP from ECG signal using machine learning methods. The SBP and DBP achieved 9.45 and 8.13 mmHg mean absolute error.

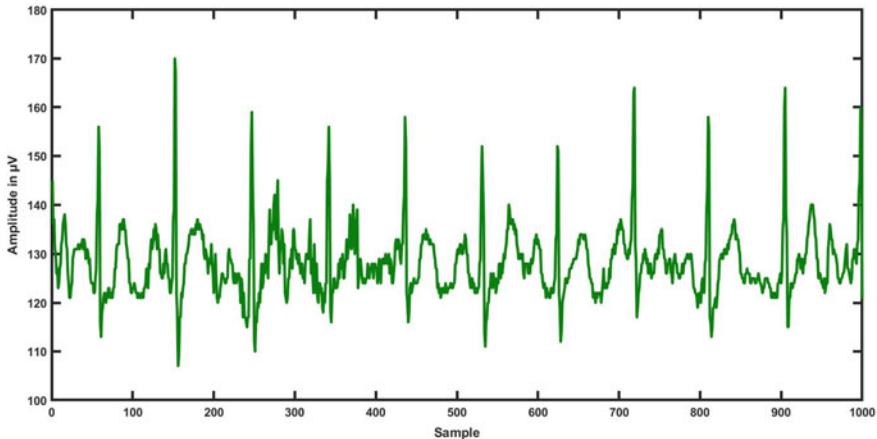
Song et al. [23] distinguish hypertension and heart disease from the HRV signal. In addition, the Naive Bayes classifier, a machine learning-based model, gives classification accuracy of 92.3%.

Hence, it is apparent from the literature, and disease (STR, SYN, MI, and LHT) associated with hypertension has not been studied yet. Therefore, in the current scenario, hypertension is attracting researchers globally. To diagnose and predict hypertension and its associated disease, a large amount of recorded data is available in hospitals and online databases. Accordingly, we have developed the hypertension diseases detection system by signal processing and machine learning-based methods. In the proposed work, we have used an orthogonal wavelet filter bank (OGWFB) to perfectly discriminate STR, SYN, MI, and LHT ECG signals. The OGWFB produces six sub-bands (SBs) for each ECG signal considering five-level wavelet decomposition. In addition, the LOGE and SLFD features were calculated for all SBs. As a result, the KNN classifier presents the highest classification accuracy of 98.4%.

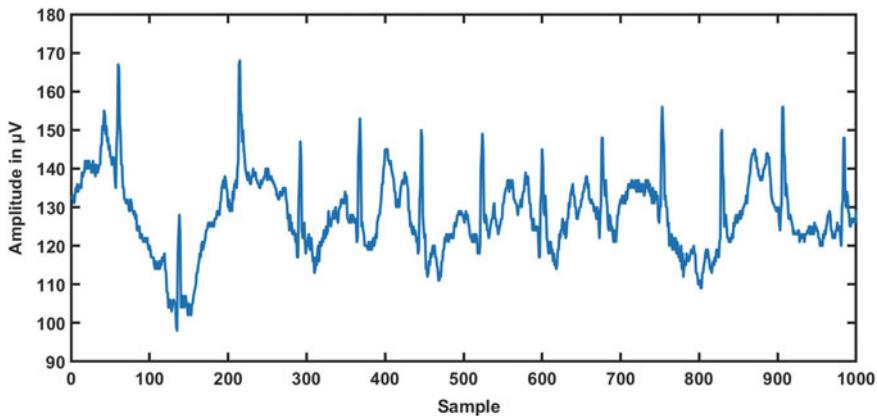
Section 2 provides the details of the dataset. Then, the methodology is explained in Sect. 3. Subsequently, the performance (result) of the developed model is discussed in Sect. 4. At last, the outcome and concluding remarks of the proposed algorithm are given in Sect. 5.

## 2 Dataset

The dataset for this research work was obtained from Physionet's online database (SHAREE database). The Ethics Committee approved the current study of Federico II University Hospital Trust. A total of 139 hypertensive recordings were used, out of which 49 are female, and 90 are male patients; the average age is 55 years. In addition to this, the length of each ECG signal is 2 h:10 min:12-s. Furthermore, each ECG signal has III, V3, V5 leads and approximately one million samples (samples/signal). The lead III, V3, and V5 are assigned as CH1 (channel1), CH2 (channel2), and CH3 (channel3). The ECG signal sampling frequency, bit resolution, and sampling intervals are 128 Hz, 8-bit, and 0.0078125 sec, respectively. Out of 139 subjects, three are SYN, three are STR, and 11 are MI, while 122 patients are low-risk hypertension LHT subjects. Further, we have segmented each ECG signal into a 5-minute duration signal. After segmentation of ECG signal, 3172 ECG signals are of LHT, 78 of stroke,



**Fig. 1** Low-risk hypertension ECG signal

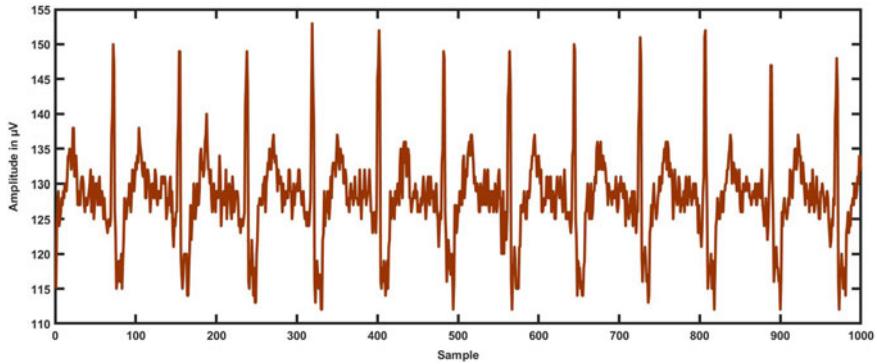


**Fig. 2** Myocardial infarction ECG signal

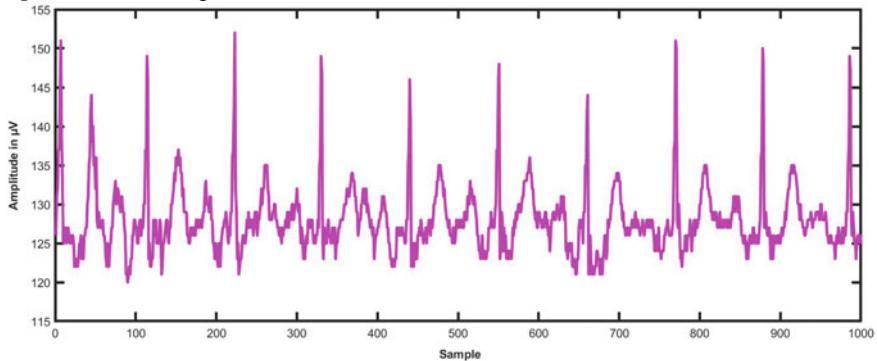
78 of syncope, and 286 of myocardial infarction. Figures 1, 2, 3, and 4 show all four classes of hypertension-associated ECG signals of 5-min duration.

### 3 Methodology

The optimally designed OGWFB discriminates LHT, MI, STR, and SYN classes of ECG signal. Each ECG signal has been decomposed into various sub-bands using a filter bank. The LOGE and SLFD features were computed for each ECG signal SBs. As a result, a total of 12 (six LOGE and six SLFD) features were obtained from each ECG signal. Subsequently, we applied various machine learning algorithms



**Fig. 3** Stroke ECG signal



**Fig. 4** Syncope ECG signal

on features calculated ECG signals. The KNN machine learning classifier gives the highest accuracy. The outline of the developed algorithm is presented in Fig. 5.

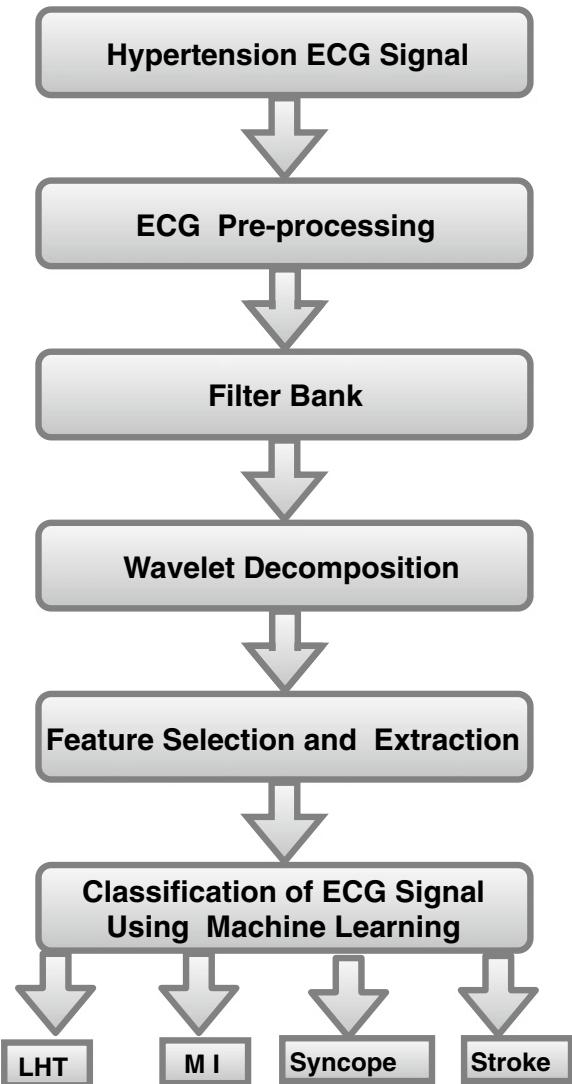
### 3.1 Preprocessing of ECG Signal

Z-score normalization is performed on each epoch of the ECG signals to eliminate the amplitude scaling problem [1, 3, 4]. Five-minute ECG signals are generated by segmenting the long-length ECG signal.

### 3.2 OGWFB Filter Bank

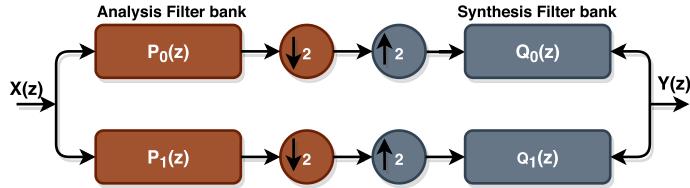
The two-band filter bank has an analysis filter bank (decomposition) and synthesis filter (reconstruction) is shown in Fig. 6. Analysis filter bank has  $P_0(z)$  low-pass

**Fig. 5** Layout of the proposed work



filter and  $P_1(z)$  high-pass filter. The high- and low-pass analysis filter bank output is down-sampled by a factor of 2, while synthesis filter bank input is up-sampled by a factor of 2. In the synthesis filter bank,  $Q_0(z)$  is low pass and  $Q_1(z)$  is high-pass filter. In the proposed work, we used an orthogonal wavelet filter bank developed by [15]. The output of the synthesis filter bank is matched with the input to the analysis filter bank to get the same result.

Perfect reconstruction is achieved using the two-channel filter bank. However, the condition of orthogonality is necessary for filters to get perfect reconstruction of signal [2, 16, 26, 28, 30, 31]. Therefore, the orthogonal filter bank can be con-



**Fig. 6** Orthogonal wavelet filter bank diagram

verted into the finite impulse response analysis low-pass filter  $p_0(n)$ , and it must fulfill the condition of orthogonality which is equivalent to the condition of perfect reconstruction and zero moments [21, 29]. Additionally, the high-pass filter  $p_1(n)$  can be produced by adjusting the sign of the coefficient of the flipped variant of the low-pass filter. Finally, the synthesis bank filters can be extracted from the time reversal of the analysis banks.

### 3.3 Wavelet Decomposition

The ECG signal is non-stationary; therefore, we cannot apply conventional (Fourier, Laplace, and short-time Fourier transform) methods [16, 19, 21]. Instead, we used an optimal wavelet filter bank (OGWFB) to decompose ECG signals in various sub-bands. In addition, a five-level wavelet decomposition was used [14, 27]- [21, 34]. As a result, it produces accurate and precise information about the ECG signals. Total  $N + 1$  sub-bands were made for  $N$  level wavelet decomposition [14, 17]. However, the SB1-SB5 are detailed, and SB6 is an approximate sub-band.

### 3.4 Features Used in Proposed Work

The important part of this work is to calculate and select the required features. Significantly, the performance of the classifier is based on the nature of the feature extracted. Moreover, it is not priorly known which feature will best discriminate each class of ECG signal. In addition, the LOGE and SLFD features were computed for all six SBs of each ECG signal [33]. Finally, the feature extracted ECG signals were fed to the machine learning classifiers. As a result, we can classify the LHT, MI, SYN, and STR ECG signals using LOGE and SLFD features with machine learning classifiers.

### 3.5 Classification and Performance Evaluation

In the proposed work, we have used various supervised machine learning algorithms for the automated classification of hypertension ECG signals. However, we use ECG signals on a variety of classifiers, including support vector machines (SVM), k-nearest neighbor (KNN), decision trees (DT), and ensemble bagged trees (EBT), to improve performance. As a result, we have obtained the highest classification performance using the KNN classifier.

**Table 1** Performance summary of filters was obtained using CH1, CH2, and CH3

Filters	CH1, ACC (%)	CH2, ACC (%)	CH3, ACC (%)
F1	98	97.4	98.4
F2	98.2	97.1	98.4
F3	98.2	97.1	98.3

**Table 2** Comparison of AUC obtained by all filters for CH1, CH2 and CH3

Filters	CH1, AUC	CH2, AUC	CH3, AUC
F1	0.95	0.96	0.98
F2	0.96	0.97	0.99
F3	0.96	0.95	0.98

**Table 3** Confusion matrix of CH1 for all classes obtained using KNN classifier

True class	Predicted class			
	Normal, 0	STR, 1	SYN, 2	MI, 3
Normal, 0	3143	6	4	19
STR, 1	7	71	0	0
SYN, 2	3	0	75	0
MI, 3	24	2	1	259

**Table 4** Confusion matrix of CH2 for all classes obtained using KNN classifier

True class	Predicted class			
	Normal, 0	STR, 1	SYN, 2	MI, 3
Normal, 0	3131	9	14	18
STR, 1	7	70	0	1
SYN, 2	21	0	57	0
MI, 3	24	0	0	262

**Table 5** Confusion matrix of CH3 for all classes obtained using KNN classifier

True class	Predicted class			
	Normal, 0	STR, 1	SYN, 2	MI, 3
Normal, 0	3149	1	5	17
STR, 1	3	74	0	1
SYN, 2	8	0	70	0
MI, 3	23	1	0	262

**Table 6** Classification accuracy of various classifiers for CH1, CH2, and CH3

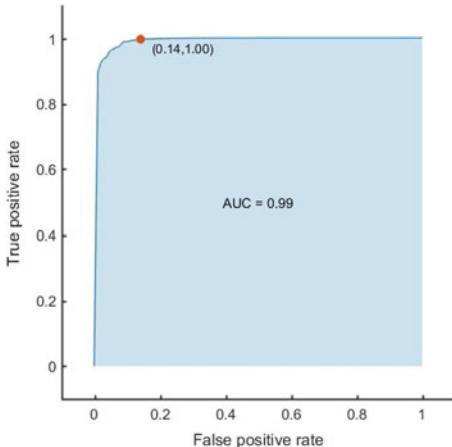
Classifier	CH1, ACC (%)	CH2, ACC (%)	CH3, ACC (%)
KNN	98.2	97.4	98.4
DT	91.0	90.1	92.4
EBT	93.5	92.6	93.4
SVM	97.3	96.0	96.7

Usually, KNN is applied for dimensionality reduction and feature selection [13, 20, 25]. In addition, the KNN is used for the k training samples, which are neighbors of the test sample, to classify it. Following this, the KNN classifier provided the lowest probability and overfitting [13, 20, 25].

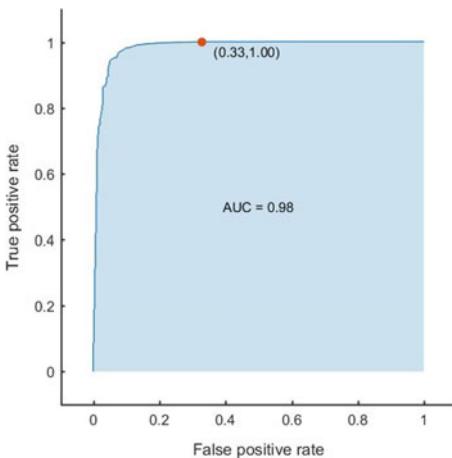
## 4 Result and Discussion

The experimental work has been performed on the MATLAB version (9.1.0), with Intel Xeon 3.5 GHz, and 16 GB RAM. The F1, F2, and F3 filters of OGFBW enhance the performance of the proposed work. In addition, filter F2 presents the highest classification accuracy compared to the other two filters. The performance of each filter in terms of classification accuracy is shown in Table 1. A filter F2 produced the highest area under the curve (AUC) of 0.99 for CH3 is mentioned in Table 2. Tables 3, 4, and 5 represent the confusion matrix of LHT, MI, STR, and SYN of CH1, CH2, and CH3 for KNN classifiers. The classification performance of each classifiers is

**Fig. 7** ROC curve obtained for CH1 using KNN classifier



**Fig. 8** ROC curve obtained for CH2 using KNN classifier

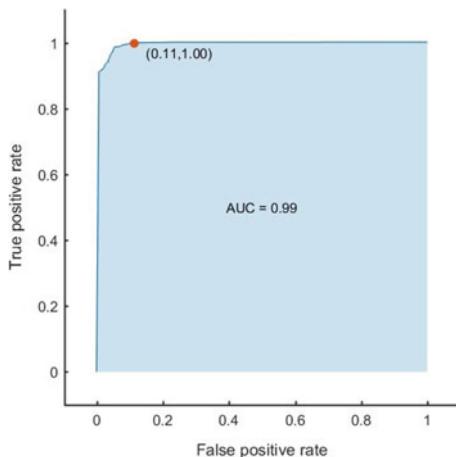


presented in the Table 6. It is evident from Table 6 that the KNN classifier gives classification accuracy of 98.4% for all classes. For testing the model performance and avoiding overfitting, we used the ten-fold cross-validation method. Table 6 shows that our model can identify 98.4 % accurately of LHT, MI, STR, and SYN classes. The best receiver operating characteristics (ROC) curve and AUC of KNN classifier are shown in Figs. 7, 8, and 9 for CH1, CH2, and CH3.

## 5 Conclusion

This study used optimal OGWFB to separate LHT, MI, STR, and SYN ECG signals. The LOGE and SLFD features were calculated for all six sub-bands. OGWFB

**Fig. 9** ROC curve obtained for CH3 using KNN classifier



can classify ECG signals accurately using LOGE and SLFD with a ten-fold cross-validation method. To check the performance of filter banks, we have applied various classifiers. KNN classifier presents an accuracy of 98.4% and an AUC of 0.99. This study can be employed for the identification of heart, brain, and kidney disease. Therefore, an adaptable, robust, reliable, and accurate model has been proposed. Furthermore, the system performance can be enhanced by extracting other features like signal sample entropy, wavelet entropy, and higher-order spectra. These findings demonstrate that our methods outperform previous models and that they can be used in large databases. Sequentially, we can test performances of the suggested technique on a large dataset for automatic detection of the severity of hypertension.

## References

- Chubb, H., Simpson, J.: The use of z-scores in paediatric cardiology. *Ann. Pediatric Cardiol.* **5**, 179–184 (2012). <https://doi.org/10.4103/0974-2069.99622>
- Daubechies, I.: Orthonormal bases of compactly supported wavelets. *Commun. Pure Appl. Math.* **41**(7), 909–996 (1988). <https://doi.org/10.1002/cpa.3160410705> <https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160410705>
- Gokhroo, R., Anantharaj, A., Bisht, D., Kishor, K., Plakkal, N., Mondal, N.: A pediatric echocardiographic z-score nomogram for a developing country: Indian pediatric echocardiography study—the z-score. *Ann. Pediatric Cardiol.* **11**, 109 (2018). [https://doi.org/10.4103/apc.APC\\_123\\_17](https://doi.org/10.4103/apc.APC_123_17)
- Long, M., Lei, W., Mou, L., Zhang, K., Liu, L., Li, Y., Liu, X., YU, W., Gao, G., Chen, X., Shen, W., Shrestha, A.: Z-score transformation of ADC values: a way to universal cut off between malignant and benign lymph nodes. *Eur. J. Radiol.* **106** (2018). <https://doi.org/10.1016/j.ejrad.2018.07.022>
- Melillo, P., Izzo, R., Orrico, A., Scala, P., Attanasio, M., Mirra, M., Luca, N., Pecchia, L.: Automatic prediction of cardiovascular and cerebrovascular events using HRV analysis. *PLOS ONE* **10**, e0118504 (2015). <https://doi.org/10.1371/journal.pone.0118504>

6. Ni, H., Wang, Y., Xu, G., Shao, Z., Zhang, W., Zhou, X.: Multiscale fine-grained heart rate variability analysis for recognizing the severity of hypertension. *Comput. Math. Methods Med.* **2019**, 1–9 (2019). <https://doi.org/10.1155/2019/4936179>
7. Quachtran, B., Hamilton, R., Scalzo, F.: Detection of intracranial hypertension using deep learning. In: Proceedings of the International Conference on Pattern Recognition (IAPR) (2016), pp. 2491–2496. <https://doi.org/10.1109/ICPR.2016.7900010>
8. Rajput, J.S., Sharma, M., Acharya, U.R.: Hypertension diagnosis index for discrimination of high-risk hypertension ECG signals using optimal orthogonal wavelet filter bank. *Int. J. Environ. Research and Public Health* **16**(21) (2019), <https://www.mdpi.com/1660-4601/16/21/4068>
9. Rajput, J.S., Sharma, M., Kumbhani, D., Acharya, U.R.: Automated detection of hypertension using wavelet transform and nonlinear techniques with ballistocardiogram signals. *Inform. Med. Unlocked* **26**, 100736 (2021)
10. Rajput, J.S., Sharma, M., Tan, R.S., Acharya, U.R.: Automated detection of severity of hypertension ecg signals using an optimal bi-orthogonal wavelet filter bank. *Computers Biol. Med.* 103924 (2020). <https://doi.org/10.1016/j.combiomed.2020.103924>. <http://www.sciencedirect.com/science/article/pii/S001048252030264X>
11. Sakr, S., Elshawi, R., Ahmed, A., Qureshi, W.T., Brawner, C., Keteyian, S., Blaha, M.J., Al-Mallah, M.H.: Using machine learning on cardiorespiratory fitness data for predicting hypertension: the Henry ford exercise testing (fit) project. *PLOS ONE* **13**(4), 1–18 (2018). <https://doi.org/10.1371/journal.pone.0195344>
12. Sau, A., Bhakta, I.: Screening of anxiety and depression among the seafarers using machine learning technology. *Inform. Med. Unlocked* (2018)
13. Shah, S., Sharma, M., Deb, D., Pachori, R.B.: 2019 International Conference on Machine Intelligence and Signal Analysis Advances in Intelligent Systems and Computing, vol. 748. Springer, Singapore, pp. 473–483 (2019). [https://doi.org/10.1007/978-981-13-0923-6\\_41](https://doi.org/10.1007/978-981-13-0923-6_41)
14. Sharma, M., Acharya, U.R.: Automated detection of schizophrenia using optimal wavelet-based  $l_1$  norm features extracted from single-channel EEG. *Cogn. Neurodyn.* 1–14 (2021). <https://doi.org/10.1007/s11571-020-09655-w>
15. Sharma, M., Bhurane, A., Acharya, U.R.: MMSFL-OWFB: a novel class of orthogonal wavelet filters for epileptic seizure detection. *Knowl. Based Syst.* (2018). <https://doi.org/10.1016/j.knosys.2018.07.019>
16. Sharma, M., Patel, V., Acharya, U.R.: Automated identification of insomnia using optimal bi-orthogonal wavelet transform technique with single-channel EEG signals. *Knowl. Based Syst.* 107078 (2021). <https://doi.org/10.1016/j.knosys.2021.107078>. <http://www.sciencedirect.com/science/article/pii/S0950705121003415>
17. Sharma, M., Pv, A., Pachori, R., Gadre, V.: A parametrization technique to design joint time-frequency optimized discrete-time biorthogonal wavelet bases. *Signal Process.* **135** (2017). <https://doi.org/10.1016/j.sigpro.2016.12.019>
18. Sharma, M., Rajput, J.S., Tan, R.S., Acharya, U.R.: Automated detection of hypertension using physiological signals: a review. *Int. J. Environ. Res. Public Health* **18**(11) (2021). <https://doi.org/10.3390/ijerph18115838>. <https://www.mdpi.com/1660-4601/18/11/5838>
19. Sharma, M., Raval, M., Acharya, U.R.: A new approach to identify obstructive sleep apnea using an optimal orthogonal wavelet filter bank with ECG signals (2019). <https://doi.org/10.1016/j.imu.2019.100170>
20. Sharma, M., Tan, R.S., Acharya, U.R.: A novel automated diagnostic system for classification of myocardial infarction ECG signals using an optimal biorthogonal filter bank. *Computers Biol. Med.* (2018)
21. Sharma, M., Tiwari, J., Acharya, U.R.: Automatic sleep-stage scoring in healthy and sleep disorder patients using optimal wavelet filter bank technique with EEG signals. *Int. J. Environ. Res. Public Health* **18**(6) (2021). <https://doi.org/10.3390/ijerph18063087>. <https://www.mdpi.com/1660-4601/18/6/3087>
22. Simjanoska, M., Gjoreski, M., Gams, M., Madevska Bogdanova, A.: Non-invasive blood pressure estimation from ECG using machine learning techniques. *Sensors* **18** (2018). <https://doi.org/10.3390/s18041160>

23. Song, Y., Ni, H., Zhou, X., Zhao, W., Wang, T.: Extracting Features for Cardiovascular Disease Classification Based on Ballistocardiography, pp. 1230–1235 (2015). <https://doi.org/10.1109/UIC-ATC-ScalCom-CBDCom-IoP.2015.223>
24. Whelton, P., Carey, M.R.: Guideline for the prevention, detection, evaluation, and management of high blood pressure in adults. *J. Am. College Cardiol.* **71** (2017). <https://doi.org/10.1016/j.jacc.2017.11.005>
25. Xing, W., Bei, Y.: Medical health big data classification based on KNN classification algorithm. *IEEE Access* **8**, 28808–28819 (2020). <https://doi.org/10.1109/ACCESS.2019.2955754>
26. Sharma, M., Darji, J., Thakrar, M., Acharya, U.R.: Automated identification of sleep disorders using wavelet-based features extracted from electrooculogram and electromyogram signals. *Computers in Biol. Med.* 105224 (2022)
27. Sharma, M., Dhiman, H.S., Acharya, U.R.: Automatic identification of insomnia using optimal antisymmetric biorthogonal wavelet filter bank with ECG signals. *Computers in Biol. Med.* 104246 (2021). <https://doi.org/10.1016/j.combiomed.2021.104246>. <https://www.sciencedirect.com/science/article/pii/S0010482521000408>
28. Sharma, M., Kumbhani, D., Yadav, A., Acharya, U.R.: Automated sleep apnea detection using optimal duration-frequency concentrated wavelet-based features of pulse oximetry signals. *Appl. Intell.* 1–13 (2021)
29. Sharma, M., Patel, S., Acharya, U.R.: Automated detection of abnormal EEG signals using localized wavelet filter banks. *Pattern Recogn. Lett.* (2020)
30. Sharma, M., Patel, S., Acharya, U.R.: Expert system for detection of congestive heart failure using optimal wavelet and heart rate variability signals for wireless cloud-based environment. *Expert Syst. e12903* (2021)
31. Sharma, M., Patel, V., Tiwari, J., Acharya, U.R.: Automated characterization of cyclic alternating pattern using wavelet-based features and ensemble learning techniques with EEG signals. *Diagnostics* **11**(8) (2021). <https://doi.org/10.3390/diagnostics11081380>. <https://www.mdpi.com/2075-4418/11/8/1380>
32. Sharma, M., Achuth, P., Deb, D., Puthankattil, S.D., Acharya, U.R.: An automated diagnosis of depression using three-channel bandwidth-duration localized wavelet filter bank with EEG signals. *Cogn. Syst. Res.* **52**, 508–520 (2018). <http://www.sciencedirect.com/science/article/pii/S1389041718302298>
33. Sharma, M., Tan, R.S., Acharya, U.R.: Detection of shockable ventricular arrhythmia using optimal orthogonal wavelet filters. *Neural Comput. Appl.* (2019). <https://doi.org/10.1007/s00521-019-04061-8>
34. Sharma, M., Tiwari, J., Patel, V., Acharya, U.R.: Automated identification of sleep disorder types using triplet half-band filter and ensemble machine learning techniques with EEG signals. *Electronics* **10**(13) (2021). <https://doi.org/10.3390/electronics10131531>

# Novel Score Function and Accuracy Function for Spherical Linguistic Fuzzy Numbers and Their Application in Multi-criteria Decision-Making Problems



Laxmi Rajput and Sanjay Kumar

**Abstract** A spherical linguistic fuzzy set is the addition of linguistic variables and a spherical fuzzy set. It includes both qualitative and quantitative estimates which can more effectively depict decision-maker's actual choices and uncertainty. In this paper, we have mentioned the limitation of the existing score and accuracy function for spherical linguistic fuzzy numbers and proposed an improved score function and accuracy function using linguistic scale function to compare spherical linguistic fuzzy numbers. We have also proposed a spherical linguistic fuzzy weighted aggregation operator ( $\check{S}LFWA$ ) and a spherical linguistic fuzzy ordered weighted aggregation operator ( $\check{S}LFOWA$ ). The proposed score, accuracy functions, and aggregation operator are applied in a numerical application.

**Keywords** Spherical linguistic fuzzy set · Score function · Accuracy function · Spherical linguistic fuzzy weighted aggregation operator · Spherical linguistic fuzzy ordered weighted aggregation operator

## 1 Introduction

In the real world, we are constantly confronted with multi-criteria problems, which we refer to as multi-criteria decision-making problems [1, 2]. So, researchers have become interested in MCDM challenges and related decision-making strategies. In a short amount of time, decision-makers are unlikely to obtain complete information on all alternatives and so to overcome this problem, DMs use fuzzy numbers in place of crisp numbers to express information. As a result, Zadeh [3] developed the fuzzy set and concept of linguistic variables. Further, intuitionistic fuzzy set, Pythagorean fuzzy set, and neutrosophic fuzzy set are also developed as fuzzy set extensions. An important extension of a fuzzy set is the intuitionistic fuzzy set developed by Atanassov [4]. For further information, we go to [5–8]. Further, the picture fuzzy set (PFS) is an extension of the intuitionistic fuzzy set that was presented by Cuong

---

L. Rajput (✉) · S. Kumar

Department of Mathematics, Statistics and Computer Science G. B. Pant, University of Agriculture and Technology, Pantnagar, Uttarakhand 263145, India

e-mail: [laxmi31rajput@gmail.com](mailto:laxmi31rajput@gmail.com)

[9, 10]. So, positive, negative, and neutral membership degrees are available in PFS. PFS includes answers yes, abstain, no, and refusal. For further information, we go to [11–17].

Inspired by the PyFS and PFS, KutluGündoğdu and Kahraman developed a spherical fuzzy set (SFS). SFSs are the extension of IFSs, PyFSs, and PFSs. In SFSs, decision-makers determine the degree of hesitancy independently so it is not dependent upon positive and negative membership degrees. So, the domain of SFS is bigger than other sets. So as compared to IFS, PyFS, and PFS, SFS is better to represent the decision-maker's opinions. For more research, we recommend [18–21].

However, IFSs, PyFSs, PFSs, and SFSs still have drawbacks because these sets only express quantitative assessments but do not consider qualitative assessments. So, scholars have to show interest to take DM's qualitative decision-making with quantitative decisions before determining the optimal result. As a result, by considering DMs qualitative evaluation and offering spherical linguistic fuzzy sets (SLFSs), this study expands the potential of SFSs theory.

The ranking is the most important part of MCDM problems under a spherical linguistic fuzzy set. For SLFNs, Ashraf et al. [22] present a score function and accuracy function. However, there are a few exceptions, where these score and accuracy functions are failed. We proposed a score and an accuracy function for SLFNs in this research. Using the linguistic scale function, we also propose spherical linguistic fuzzy weighted aggregation (SLFWA) and spherical linguistic fuzzy ordered weighted aggregation (SLFOWA) operators.

As a result, the following is the structure of this paper: In Sect. 2, we define linguistic variables, the definition of a spherical linguistic fuzzy set (SLFS), some operational laws for SLFS, and linguistic scale function. In Sect. 3, we redefine operational laws for SLFS. In Sect. 4, we introduced limitations of score and accuracy functions. Then, we propose score and accuracy functions for SLFNs. In Sect. 5, we propose ŠLFWA and ŠLFOWA operator. In Sect. 6, we present a numerical application. In Sect. 7, then finally give the conclusion.

## 2 Preliminaries

Now, we will go over the definitions of the linguistic variables, spherical linguistic fuzzy number (SLFN), and linguistic scale function (LSF).

**Definition 1** Let  $T = \{\varsigma_k : k = 0, 1, \dots, 2\tau\}$  be a finite set of linguistic variables and cardinality of  $T$  is odd, where  $\varsigma_k$  indicates a linguistic variable and meets the four conditions given below [23, 24].

- (a) if  $\varsigma_l, \varsigma_k \in T$  and  $k < l$ , then  $\varsigma_k < \varsigma_l$ ,
- (b)  $\exists$  the negation operator:  $\varsigma_k = \text{neg}(\varsigma_l)$ , where  $l = 2\tau - k$ ,
- (c)  $\max(\varsigma_k, \varsigma_l) = \varsigma_k$  if  $k \geq l$
- (d)  $\min(\varsigma_k, \varsigma_l) = \varsigma_k$  if  $k \leq l$

For  $\tau = 3$ ,  $T$  transforms into:  $\{\varsigma_0, \varsigma_1, \varsigma_2, \varsigma_3, \varsigma_4, \varsigma_5, \varsigma_6\}$ .

To maintain original information, the discrete linguistic variables set  $T = \{\varsigma_k | k \in [0, l]\}$  was expanded to a continuous linguistic variable set by Xu [25], where  $l$  is an enough large positive integer. If  $\varsigma_k \in T$ , then  $\varsigma_k$  is referred to as an original linguistic variable; otherwise, it is referred to as a virtual linguistic variable.

**Definition 2** [26] Let  $Z$  be a reference set and  $T = \{\varsigma_k : k = 0, 1, \dots, 2\tau\}$  be a continuous linguistic term set and  $\varsigma_k \in T$ . Then, a SLFS is defined as

$$E = \{(x, \varsigma_{k(x)}, \mu_E(x), \eta_E, \nu_E(x)) | x \in Z\}$$

where  $\mu_E(x)$ ,  $\eta_E(x)$ , and  $\nu_E(x)$ , respectively, represent positive, neutral, and negative membership degrees corresponding to  $\varsigma_{k(x)}$ , respectively, and meets the condition  $0 \leq \mu_E^2(x) + \nu_E^2(x) + \eta_E^2(x) \leq 1$ ,  $\forall x \in Z$ .

**Definition 3** [26] Let  $\beta_1 = (\varsigma_{\theta_1}, \mu_1, \eta_1, \nu_1)$ ,  $\beta_2 = (\varsigma_{\theta_2}, \mu_2, \eta_2, \nu_2)$  and  $\beta = (\varsigma_\theta, \mu, \eta, \nu)$  be three SLN and  $\gamma > 0$  real number, then

- (a)  $\beta_1 \oplus \beta_2 = (\varsigma_{\theta_1+\theta_2}, \sqrt{(\mu_1^2 + \mu_2^2 - \mu_1^2\mu_2^2)}, \eta_1\eta_2, \nu_1\nu_2);$
- (b)  $\beta_1 \otimes \beta_2 = (\varsigma_{\theta_1 \times \theta_2}, \mu_1\mu_2, \sqrt{(\eta_1^2 + \eta_2^2 - \eta_1^2\eta_2^2)}, \sqrt{(\nu_1^2 + \nu_2^2 - \nu_1^2\nu_2^2)});$
- (c)  $\gamma\beta = (\varsigma_{\gamma \times \theta}, \sqrt{1 - (1 - \mu^2)^\gamma}, \eta^\gamma, \nu^\gamma);$
- (d)  $\beta^\gamma = (\varsigma_{\theta^\gamma}, \mu^\gamma, \sqrt{1 - (1 - \eta^2)^\gamma}, \sqrt{1 - (1 - \nu^2)^\gamma});$

**Definition 4** [27] If  $T = \{\varsigma_k : k = 0, 1, \dots, 2\tau\}$  be a linguistic variable set and  $\theta_k \in R^+$  be positive real value, then LSF,  $g$  is mapping from  $\varsigma_k$  to  $\theta_k$ , has the following definition:

$$g : \varsigma_k \rightarrow \theta_k : (k = 0, 1, 2, \dots, 2\tau)$$

It satisfies the condition  $0 \leq \theta_0 < \theta_1 < \dots < \theta_{2\tau}$  and the function  $g$  must be a monotonic increasing function. Here, three LSFs can be defined as:

If the deviation is equal between adjacent linguistic terms, then

$$g(\varsigma_k) = \frac{k}{2\tau} \quad (1)$$

The deviations are increased between adjacent linguistic subscripts when linguistic subscripts are increased, then

$$g(\varsigma_k) = \theta_k = \begin{cases} \frac{\xi^\tau - \xi^{\tau-k}}{2\xi^\tau - 2}, & (k = 0, 1, 2, \dots, \tau) \\ \frac{\xi^\tau + \xi^{k-\tau} - 2}{2\xi^\tau - 2}, & (k = \tau + 1, \tau + 2, \dots, 2\tau) \end{cases} \quad (2)$$

Here,  $\xi$  is known as threshold and calculated according to the specific situations.

The deviations are decreased between adjacent linguistic subscripts when linguistic subscripts are increased. then,

$$g(\varsigma_k) = \theta_k = \begin{cases} \frac{\tau^\alpha - (\tau-k)^\alpha}{2\tau^\alpha}, & (k = 0, 1, 2, \dots, \tau) \\ \frac{\tau^\beta - (k-\tau)^\beta}{2\tau^\beta}, & k = \tau + 1, \tau + 2, \dots, 2\tau \end{cases} \quad (3)$$

We can calculate values experimentally. Kahneman and Tversky [28] used experiments to come up with values  $\alpha = \beta = 0.88$ , which is congruent with empirical data.

### 3 Operational Laws for SLFS

Given that the traditional SLFS has many deficiencies such as the operation result exceeding the upper bound of the linguistic variable set. So, redefine operational results based on LSFs.

**Definition 5** Let  $\beta_1 = (\varsigma_{\theta_1}, \mu_1, \eta_1, \nu_1)$ ,  $\beta_2 = (\varsigma_{\theta_2}, \mu_2, \eta_2, \nu_2)$  and  $\beta = (\varsigma_\theta, \mu, \eta, \nu)$  be three SLN and  $\gamma > 0$  real number. Then, based on LSF operational laws defined for SLFV as:

- $$\begin{aligned} \beta_1 \oplus \beta_2 &= (g^{-1}(g(\varsigma_{\theta_1}) + g(\varsigma_{\theta_2}) - g(\varsigma_{\theta_1})g(\varsigma_{\theta_2})), \\ (a) \quad &\sqrt{(\mu_1^2 + \mu_2^2 - \mu_1^2\mu_2^2)}, \eta_1\eta_2, \nu_1\nu_2); \\ \beta_1 \otimes \beta_2 &= (g^{-1}(g(\varsigma_{\theta_1}) \times g(\varsigma_{\theta_2})), \mu_1\mu_2, \sqrt{(\eta_1^2 + \eta_2^2 - \eta_1^2\eta_2^2)}), \\ (b) \quad &\sqrt{(\nu_1^2 + \nu_2^2 - \nu_1^2\nu_2^2)}; \\ (c) \quad \gamma\beta &= (g^{-1}(1 - (1 - g(\varsigma_\theta))^\gamma), \sqrt{1 - (1 - \mu^2)^\gamma}, \eta^\gamma, \nu^\gamma); \\ (d) \quad \beta^\gamma &= (g^{-1}(g(\varsigma_\theta)^\gamma), \mu^\gamma, \sqrt{1 - (1 - \eta^2)^\gamma}, \sqrt{1 - (1 - \nu^2)^\gamma}); \end{aligned}$$

**Example 1** Let  $\beta_1 = (\varsigma_5, 0.3, 0.8, 0.5)$  and  $\beta_2 = (\varsigma_4, 0.4, 0.8, 0.4)$  be two SLFVs. Suppose that  $T = \{\varsigma_k : k = 0, 1, \dots, 2\tau\}$  be a linguistic variable set and  $\tau = 3$ .  $g(\varsigma_k) = \frac{k}{2\tau}$  and  $\lambda = 2$ . The following can be obtained.

$$\begin{aligned} \beta_1 \oplus \beta_2 &= (g^{-1}(g(\varsigma_5) + g(\varsigma_4) - g(\varsigma_5)g(\varsigma_4)), \\ (a) \quad &\sqrt{(0.3^2 + 0.4^2 - 0.3^2 \times 0.4^2)}, 0.8 \times 0.8, 0.5 \times 0.4) \\ &= (\varsigma_{5.67}, 0.49, 0.64, 0.20); \\ \beta_1 \otimes \beta_2 &= (g^{-1}(g(\varsigma_5) \times g(\varsigma_4)), 0.3 \times 0.4, \\ (b) \quad &\sqrt{(0.8^2 + 0.8^2 - 0.8^2 \times 0.8^2)}, \sqrt{(0.5^2 + 0.4^2 - 0.5^2 \times 0.4^2)}) \\ &= (\varsigma_{3.34}, 0.12, 0.93, 0.61); \end{aligned}$$

$$\begin{aligned}
 (c) \quad 2\beta_1 &= (g^{-1}(1 - (1 - g(\varsigma_5))^2), \sqrt{(1 - (1 - 0.3^2)^2)}, 0.8^2, 0.5^2); \\
 &= (\varsigma_{5.83}, 0.41, 0.64, 0.25); \\
 (d) \quad \beta_1^2 &= (g^{-1}(g(\varsigma_5)^2), 0.3^2, \sqrt{(1 - (1 - 0.8^2)^2)}, \sqrt{(1 - (1 - 0.5^2)^2)}) \\
 &= (\varsigma_{4.17}, 0.09, 0.93, 0.66);
 \end{aligned}$$

## 4 Limitations of Existing Score Function and Accuracy Function for SLFS

Now, we discuss limitations of accuracy function and score function developed by Ashraf et al. [22] under an environment of the SLFS using the following examples.

**Example 2** Let SLFN for two alternatives be  $F_1 = (\varsigma_3, 0.5, 0.3, 0.7)$  and  $F_2 = (\varsigma_3, 0.9, 0.3, 0.1)$ . On applying the score function Ashraf et al. [20, 23], we calculated  $Sc(F_1) = -0.1667\varsigma_3$  and  $Sc(F_2) = -0.1667\varsigma_3$ . So,  $Sc(F_1) = Sc(F_2)$  but  $F_1 \neq F_2$ .

**Example 3** Let SLFN for two alternatives be  $F_1 = (\varsigma_4, 0.1, 0.2, 0.5)$  and  $F_2 = (\varsigma_4, 0.3, 0.3, 0.3)$ . Again, on applying the accuracy function of Ashraf et al. [22], we calculated  $Ac(F_1) = 0.3000\varsigma_4$  and  $Ac(F_2) = 0.3000\varsigma_4$ . So,  $Ac(F_1) = Ac(F_2)$  but  $F_1 \neq F_2$ .

The above two examples exhibit that score and accuracy function developed by Ashraf et al. [22] may fail for some particular SLF numbers. Motivated by this, we propose novel accuracy and score functions for SLF numbers in the next section.

### 4.1 Proposed Score Function and Accuracy Function

Now, we proposed the score function and accuracy function for an SLFS set using the linguistic scale function (LSF).

**Definition 6** Let  $\beta_k = (\mu_k, \eta_k, \nu_k)$  be aSFN. Then, score function and accuracy function for  $\beta_k$  defined as:

$$\begin{aligned}
 \text{Score function}(\beta_k) &= g(\varsigma_{\theta_k}) \frac{(1 + \mu_k - \eta_k - \nu_k)}{3} \\
 \text{Accuracy function}(\beta_k) &= g(\varsigma_{\theta_k}) \frac{(\mu_k + \eta_k + \nu_k)}{3}
 \end{aligned}$$

## 4.2 Importance of Proposed Score Function and Accuracy Function

Applying the proposed score function in Example 2, we get,  $Sc(F_1) = 0.0834$  and  $Sc(F_2) = 0.2500$ . So  $Sc(F_2) > Sc(F_1)$ . Hence,  $F_2$  is better than  $F_1$ .

On applying the proposed accuracy function in Example 3, we get,  $Ac(F_1) = 0.1778$  and  $Ac(F_2) = 0.2000$ . So  $Ac(F_2) > Ac(F_1)$ . Hence,  $F_2$  is better than  $F_1$ .

Here, we use  $g(\varsigma_k) = \frac{k}{2\tau}$  the LSF to find the value of the proposed score function and accuracy function for alternative  $F_2$  and  $F_1$ .

## 5 Now, We Propose a Definition of ŠLFWA Aggregation Operator and ŠLFOWA Aggregation Operator

**Definition 7** Let  $\beta_k = (\varsigma_{\theta(k)}, \mu_k, \eta_k, \nu_k)$  be any collection of the SLFNs and aggregation operator  $\check{\text{SLFWA}} : \check{\text{SLFN}}^n \rightarrow \check{\text{SLFN}}$ , Then,  $\check{\text{SLFWA}}$  is defined as,

$$(\check{\text{SLFWA}})(\beta_1, \beta_2, \dots, \beta_n) = \sum_{k=1}^n w_k \beta_k \quad (4)$$

where  $w = (w_1, w_2, \dots, w_n)$  is the weighting vector of  $\beta_k$ ,  $w_k \geq 0$  and  $\sum_{k=1}^n w_k = 1$ .

**Theorem 5.1** Let  $\beta_k = (\varsigma_{\theta(k)}, \mu_k, \eta_k, \nu_k)$  be any collection of the SLFNs. Then, utilizing definition 7 and operational laws for SLFNs given in Sect. 3, we get the following result.

$$\begin{aligned} \check{\text{SLFWA}}(\beta_1, \beta_2, \dots, \beta_n) &= \{g^{-1}(1 - \prod_{k=1}^n (1 - g(\varsigma_k))^{w_k}), \sqrt{(1 - \prod_{k=1}^n (1 - \mu_k^2)^k)}, \\ &\quad \prod_{k=1}^n \eta_k^{w_k}, \prod_{k=1}^n \nu_k^{w_k}\} \end{aligned} \quad (5)$$

**Definition 8** Let  $\beta_k = (\varsigma_{\theta(k)}, \mu_k, \eta_k, \nu_k)$  be any collection of the SLFNs and  $\check{\text{SLFOWA}} : \check{\text{SLFN}}^n \rightarrow \check{\text{SLFN}}$ , then  $\check{\text{SLFOWA}}$  is defined as,

$$\check{\text{SLFOWA}}(\beta_1, \beta_2, \dots, \beta_n) = \sum_{k=1}^n w_k \beta_{\sigma(k)} \quad (6)$$

where  $w = (w_1, w_2, \dots, w_n)$  is the weighting vector of  $\beta_k$ ,  $w_k \geq 0$  and  $\sum_{k=1}^n w_k = 1$ ,  $\sigma(k)$  denotes a permutation on  $\mathbb{R}$ .

**Theorem 5.2** Let any collection  $\beta_k = (\varsigma_{\theta(k)}, \mu_k, \eta_k, \nu_k)$  be the SLFNs. Then, utilizing definition 8 and operational laws for SLFNs given in Sect. 3, we get the following result.

$$\check{\text{SLFOWA}}(\beta_1, \beta_2, \dots, \beta_n) = \left\{ g^{-1} \left( 1 - \prod_{k=1}^n (1 - g(\varsigma_{\sigma(k)}))^{w_k} \right), \sqrt{\left( 1 - \prod_{k=1}^n (1 - \mu_{\sigma(k)}^2)^{w_k} \right)}, \prod_{k=1}^n \eta_{\sigma(k)}^{w_k}, \prod_{k=1}^n \nu_{\sigma(k)}^{w_k} \right\} \quad (7)$$

## 6 Algorithm for MCDM Problem Using Proposed Score Function and Aggregation Operators

Let MCDM problem with spherical linguistic information represents as follows. Let  $\{F_1, F_2, \dots, F_m\}$  and  $\{G_1, G_2, \dots, G_n\}$  be sets of m alternatives and n criteria. Let the set of weight vectors of criteria is  $w = \{w_1, w_2, \dots, w_n\}$ , satisfying  $0 \leq w_k \leq 1$  and  $\sum_{k=1}^n w_k = 1$ . For criteria  $G_k$  corresponding to alternative  $F_k$  experts need an SLN  $\beta = (\varsigma_{\theta_k}, \mu_k, \eta_k, \nu_k)$  to express information. Hence, an evaluation matrix contains SLN information indicated by  $R = (\beta_{kl})_{m \times n}$ .

**Step 1.** Normalize the original decision matrix using equation

$$\beta_{kl} = \begin{cases} (\varsigma_{\theta_k}, \mu_k, \eta_k, \nu_k) & G_k \text{ is benefit type.} \\ (\varsigma_{\theta_k}, \nu_k, \eta_k, \mu_k) & G_k \text{ is cost type.} \end{cases} \quad (8)$$

Normalizing is used only when the criteria are of an alternative type; otherwise, normalization is not required.

**Step 2.** Aggregate the decision matrix by using the ŠLFWA and the ŠLFOWA operators in Eqs. 5 and 7.

**Step 3.** Now, compute the overall assessment score values.

**Step 4.** Choose the best alternative by assigning ranking according to their score values.

### 6.1 Implementation of Proposed MCDM Method

Now, we applied the suggested method to solve the investment selection problem [26]. An investment business wants to invest in a project that would provide a

steady return. The company hires an expert to assess four potential projects from four perspectives. Let  $\{F_1, F_2, F_3, F_4\}$  and  $\{G_1, G_2, G_3, G_4\}$  be set of alternatives and criteria, where  $G_1$  is the reputation of the project,  $G_2$  indicates risk tolerance,  $G_3$  indicates the socio-economic impact, and  $G_4$  ecological friendliness, and the set  $w = \{0.32, 0.26, 0.18, 0.24\}^T$  represents weight vector of criteria and  $\rho = \{0.155, 0.345, 0.345, 0.155\}^T$  is associated weight vector [29]. Hence, spherical linguistic decision matrix is in Table 1.

**Step 1.** As all of the criteria are benefit types. So, as a result no need to normalize the matrix in Table 1.

**Step 2.** Table 1 shows the aggregates decision matrix using the ŠLFWA operator and ŠLFOWA operator. Further, calculate the score values of alternatives and rank them according to their score values in Tables 2 and 3.

As we see in Tables 2 and 3, we get ranking order  $F_2 > F_4 > F_3 > F_1$  by using ŠLFWA and ŠLFOWA operators. And the best alternative is  $F_2$  in both Tables 2 and 3.

**Table 1** Decision matrix

	$G_1$	$G_2$	$G_3$	$G_4$
$F_1$	$(\xi_5, 0.2, 0.1, 0.7)$	$(\xi_3, 0.2, 0.2, 0.5)$	$(\xi_5, 0.1, 0.3, 0.5)$	$(\xi_3, 0.1, 0.3, 0.5)$
$F_2$	$(\xi_4, 0.2, 0.2, 0.6)$	$(\xi_5, 0.4, 0.1, 0.5)$	$(\xi_4, 0.2, 0.3, 0.5)$	$(\xi_4, 0.4, 0.1, 0.5)$
$F_3$	$(\xi_3, 0.2, 0.1, 0.7)$	$(\xi_4, 0.1, 0.2, 0.5)$	$(\xi_4, 0.1, 0.2, 0.6)$	$(\xi_5, 0.3, 0.3, 0.3)$
$F_4$	$(\xi_6, 0.3, 0.2, 0.5)$	$(\xi_3, 0.1, 0.2, 0.7)$	$(\xi_4, 0.2, 0.1, 0.6)$	$(\xi_3, 0.1, 0.4, 0.5)$

**Table 2** Aggregated decision matrix using operator ŠLFWA

Alternatives	Aggregated value using ŠLFWA operator	Score value	Ranking
$F_1$	$(\xi_{4.27}, 0.186, 0.172, 0.557)$	0.11	4
$F_2$	$(\xi_{4.33}, 0.319, 0.152, 0.530)$	0.15	1
$F_3$	$(\xi_{4.07}, 0.198, 0.177, 0.509)$	0.12	3
$F_4$	$(\xi_{6.00}, 0.204, 0.208, 0.564)$	0.14	2

**Table 3** Aggregated decision matrix using operator ŠLFOWA

Alternatives	Aggregated value using ŠLFOWA operator	Score value	Ranking
$F_1$	$(\xi_{4.27}, 0.172, 0.207, 0.527)$	0.10	4
$F_2$	$(\xi_{4.20}, 0.319, 0.151, 0.532)$	0.15	1
$F_3$	$(\xi_{3.93}, 0.182, 0.168, 0.534)$	0.11	3
$F_4$	$(\xi_{6.00}, 0.182, 0.175, 0.598)$	0.14	2

**Table 4** Score values and ranking order of different MCDM methods

Operator	Score function $S(F_k)$	Ranking orders
ŠLFWA	$S(F_1) = 0.11, S(F_2) = 0.15, S(F_3) = 0.12, S(F_4) = 0.14$	$F_2 > F_4 > F_3 > F_1$
ŠLFOWA	$S(F_1) = 0.10, S(F_2) = 0.15, S(F_3) = 0.11, S(F_4) = 0.14$	$F_2 > F_4 > F_3 > F_1$
ŠLWMM [26]	$S(F_1) = 0.161, S(F_2) = 0.185, S(F_3) = 0.164, S(F_4) = 0.156$	$F_2 > F_3 > F_1 > F_4$
ŠLWDMM [26]	$S(F_1) = 1.82, S(F_2) = 2.04, S(F_3) = 1.87, S(F_4) = 1.80$	$F_2 > F_3 > F_1 > F_4$
A-PFLWAA [30]	$S(F_1) = 1.45, S(F_2) = 1.71, S(F_3) = 1.60, S(F_4) = 1.47$	$F_2 > F_3 > F_4 > F_1$

## 6.2 Comparison Analysis

We compare the suggested method with existing methods [26, 30]. Proposed MCDM method ranks  $F_2$  and  $F_1$  as the best and the worst alternative, respectively, using both ŠLFWA and ŠLFOWA operators. These ranking results are in the accordance with the methods of Liu and Zhang [30] and Cao et al. [26] (Table 4).

## 7 Conclusions

In the real world, decision-making problems consist of qualitative variables with quantitative variables. In this study, we have shown that the existing score and accuracy function for SLFNs may fail in some particular cases. To overcome those shortcomings, we have introduced novel score and accuracy functions for SLFNs. We have also proposed a novel aggregation operator for SLFNs and have proposed an algorithm for MCDM problems using the proposed score function and aggregation operator. We applied the proposed method to a decision-making situation to better understand how it works. Finally, we validate the robustness of our suggested method by comparing it to an existing method. Because different linguistic scale functions can change the ranking of alternatives against taking criteria, decision-makers must choose linguistic scale functions based on their interests and the actual needs of the problem, and the same should be integrated with score and accuracy functions for better results.

## References

1. Yu, P.L.: A class of solutions for group decision problems. *Manage. Sci.* **19**(8), 936–946 (1973)

2. Zeleny, M.: Compromise programming, multiple criteria decision-making. *Multiple criteria decisions making*, pp. 263–301 (1973)
3. Zadeh, L.A.: Fuzzy sets. *Inf. Control* **8**(3), 338–353 (1965)
4. Atanassov, K.T.: Intuitionistic fuzzy sets. *Fuzzy Sets Syst.* **20**(1), 87–96 (1986)
5. Bustince, H.: Handling multicriteria fuzzy decision-making problems based on intuitionistic fuzzy sets. *Notes on Intuitionistic Fuzzy Sets* vol. 1, no. 1, (1995)
6. Li, D.F.: Multiattribute decision making models and methods using intuitionistic fuzzy sets. *J. Comput. Syst. Sci.* **70**(1), 73–85 (2005)
7. Lin, L., Yuan, X.H., Xia, Z.Q.: Multicriteria fuzzy decision-making methods based on intuitionistic fuzzy sets. *J. Comput. Syst. Sci.* **73**(1), 84–88 (2007)
8. Wang, X.F., Wang, J.Q., Yang, W.E.: Multi-criteria group decision making method based on intuitionistic linguistic aggregation operators. *J. Intell. Fuzzy Syst.* **26**(1), 115–125 (2014)
9. Cuong, B.C.: Picture fuzzy sets-first results. Part 1, seminar Neuro-fuzzy systems with applications. Institute of Mathematics, Hanoi(2013a).
10. Cuong, B.C.: Picture fuzzy sets-first results. Part 2 seminar. Neuro-fuzzy Systems with applications. Institute of Mathematics, Hanoi, Preprint 04/2013 (2013b)
11. Garg, H.: Some picture fuzzy aggregation operators and their applications to multicriteria decision-making. *Arab. J. Sci. Eng.* **42**(12), 5275–5290 (2017)
12. Zhang, H., Zhang, R., Huang, H., Wang, J.: Some picture fuzzy Dombi Heronian mean operators with their application to multi-attribute decision-making. *Symmetry* **10**(11), 593 (2018)
13. Garg, C. et al.: Adaptive fuzzy logic models for the prediction of compressive strength of sustainable concrete. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems*, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_47](https://doi.org/10.1007/978-981-16-2164-2_47)
14. Jana, C., Senapati, T., Pal, M., Yager, R.R.: Picture fuzzy Dombi aggregation operators: application to MADM process. *Appl. Soft Comput.* **74**, 99–109 (2019)
15. Wei, G.: TODIM method for picture fuzzy multiple attribute decision making. *Informatica* **29**(3), 555–566 (2018)
16. Wei, G.: Picture fuzzy aggregation operators and their application to multiple attribute decision making. *J. Intell. Fuzzy Syst.* **33**(2), 713–724 (2017)
17. Wei, G.: Picture fuzzy Hamacher aggregation operators and their application to multiple attribute decision making. *Fundamenta Informaticae* **157**(3), 271–320 (2018)
18. Kutlu Gündoğdu, F., Kahraman, C.: Spherical fuzzy sets and spherical fuzzy TOPSIS method. *J. Intell. Fuzzy Syst.* **36**(1), 337–352 (2019b)
19. Ashraf, S., Abdullah, S.: Spherical aggregation operators and their application in multiattribute group decision-making. *Int. J. Intell. Syst.* **34**(3), 493–523 (2019)
20. Ashraf, S., Abdullah, S., Mahmood, T.: Spherical fuzzy Dombi aggregation operators and their application in group decision making problems. *J. Ambient. Intell. Humaniz. Comput.* **11**(7), 2731–2749 (2020)
21. Ghosh, M., et al.: Robust face recognition by fusing fuzzy type 2 induced multiple facial fused image. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573871>
22. Ashraf, S., Abdullah, S., Mahmood, T.: GRA method based on spherical linguistic fuzzy Choquet integral environment and its application in multi-attribute decision-making problems. *Math. Sci.* **12**(4), 263–275 (2018)
23. Herrera, F., Herrera-Viedma, E., Verdegay, J.L.: A sequential selection process in group decision making with a linguistic assessment approach. *Inf. Sci.* **85**(4), 223–239 (1995)
24. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning I. *Inf. Sci.* **8**(3), 199–249 (1975)
25. Xu, Z.: A method based on linguistic aggregation operators for group decision making with linguistic preference relations. *Inf. Sci.* **166**(1–4), 19–30 (2004)
26. Cao, H., Zhang, R., Wang, J.: Some spherical linguistic Muirhead mean operators with their application to multi-attribute decision making. *J. Intell. Fuzzy Syst.* **37**(6), 8097–8111 (2019)

27. Wang, J.Q., Wu, J.T., Wang, J., Zhang, H.Y., Chen, X.H.: Interval-valued hesitant fuzzy linguistic sets and their applications in multi-criteria decision-making problems. *Inf. Sci.* **288**, 55–72 (2014)
28. Kahneman, D., Tversky, A.: Prospect theory: an analysis of decision under risk. *Economica* **47**, 263–291 (1979)
29. Xu, Z.: An overview of methods for determining OWA weights. *Int. J. Intell. Syst.* **20**(8), 843–865 (2005)
30. Liu, P., Zhang, X.: A novel picture fuzzy linguistic aggregation operator and its application to group decision-making. *Cogn. Comput.* **10**(2), 242–259 (2018)

# Application of Matrix Tri-Factorization for Predicting miRNA-Disease Associations



J. R. Rashmi and Lalitha Rangarajan

**Abstract** Recent research on miRNAs has shown that miRNAs play an important role in disease progression, leading to the investigation of the discovery of miRNA-disease affiliation. For predicting miRNA-disease affiliation, matrix tri-factorization is used. The functional similarity of miRNA, the similarity of miRNA based solely on diseases, the similarity of miRNA based solely on environmental factors, and adjacency matrix for miRNA and disease had been calculated and considered in the proposed approach. We also took the semantic similarity of the diseases into account. We tested the proposed method on eight diseases; the method is robust, and the results of the experiments ranked 50 miRNAs for gastric cancers, 48 miRNAs for breast cancers, and 48 miRNAs for kidney cancers with in the top 50 predictions. We obtained the area under the curve of 0.90967.

**Keywords** Matrix tri-factorization · miRNA functional similarity · Disease semantic similarity

## 1 Introduction

MicroRNAs (miRNAs) are small, endogenous, non-coding RNA type which, through base pairing interactions, binds to certain target messenger RNA systems. It regulates post-transcription and RNA silencing. The discovery of the first miRNA, lin-4, by Victor Ambrose [1, 2] twenty years ago has directed to a rising number of studies demonstrating the significance of miRNAs in many biological processes [3, 4], cell formation [5, 6], proliferation [7], and virus infection [8]. In the meantime, biological experiments indicate that miRNAs are significantly allied to the formation and growth processes of different human ailments [7].

---

J. R. Rashmi · L. Rangarajan

Department of Studies in Computer Science, University of Mysore, Mysore, India  
e-mail: [rashmirsh@compsci.uni-mysore.ac.in](mailto:rashmirsh@compsci.uni-mysore.ac.in)

L. Rangarajan  
e-mail: [lalithar@compsci.uni-mysore.ac.in](mailto:lalithar@compsci.uni-mysore.ac.in)

This has necessitated the research and development of miRNA-disease prediction algorithms as conducting the biological experiments for forecasting the miRNAs affiliated with disease are expensive and time consuming.

These algorithms can be separated into two types: first one established solely on similarity and the other based solely on machine learning. By measuring the power of the node between miRNA and disease, similarity-based methods can forecast the affiliation between miRNA and ailment, while machine learning methods rely on sample data. Few of the similarity-based and “machine learning” methods are discussed below along with their advantages and disadvantages.

It was proposed by Chen and colleagues [8] to use the “random walk algorithm with restart for miRNA-disease association (RWRMDA)” to forecast the connection among disease and miRNA. “RWRMDA” will not forecast miRNAs for all ailments at a time and also isolated ailments. Chen et al. [9] also suggested “heterogeneous graph inference for the miRNA-disease association (HGIMDA).” They used the likeness of miRNA created on functionality, the semantic similarity of disease, and the Gaussian kernel similarity in this method. This algorithm can predict miRNAs for all diseases at once as well as isolated diseases, but still there is room for improvement. A methodology named ICFMDA is recommended by Jiang et al. [10] to see the anonymous miRNA-ailments affiliations. They used the similarity matrices to control the load of bipartite networks of miRNA and ailments. They have implemented a collaborative filtering technique to forecast the unknown affiliations. You et al. [11] recommended “PBMDA: A novel and effective path-based computational model for miRNA-disease association prediction” using heterogeneous graph made with the assistance of subgraph supported similarity between ailments and miRNAs. In this method, they applied the depth first search algorithm to traverse the graph’s path to seek out the possible affiliation for miRNA and ailments. The techniques mentioned above are graph based, which are simply biased toward miRNAs or ailments that have several best-known associations.

Machine learning techniques develop a model from data sample, referred to as “training data,” in order to take decisions or make predictions without being explicitly programmed to do so. Predictive models in a different class are machine learning compatible. Chen et al. [12] were given credit for a global, semi-supervised RLSMDA method. Using a continuous ordering function, they calculated the likelihood of every miRNA being allied with a particular ailment and were able to forecast the connections between ailments and miRNAs that had no previously known connection. As the consistent classification for the miRNA network and the ailments network are created distinctly, knowledge about miRNA and ailments is not fully integrated.

In the previous few years, it is seen that matrix factorization is actually used for predicting miRNA ailment association. Computational technique named “matrix completion for miRNA-ailments association prediction (MCMDA)” to forecast miRNA-disease affiliations by Li et al. [13]. In this procedure, they have used only miRNA-disease affiliations for achieving better performance. The only limitation of this technique is it cannot be useful to the ailment which does not have any affiliation.

Gu et al. [14] created a “network consistent projection algorithm” to identify missing associations by combining associated networks along with similarity

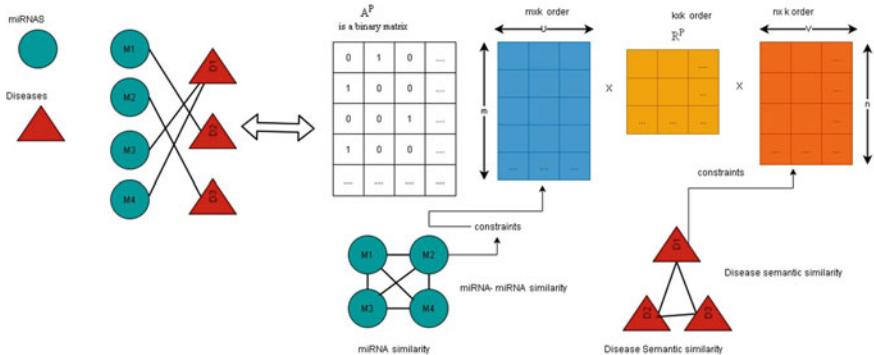
networks. The performance of this technique is not very satisfactory as it is not possible to foresee the miRNAs for ailments which did not have any previous affiliation. Shen et al. [15] proposed “collaborative matrix factorization for miRNA-disease association prediction (CMFMDA)” to identify possible miRNA-disease affiliations by combining miRNA data based on functional similarity, data based on semantic similarity between diseases, and miRNA-disease affiliations which are experimentally confirmed. Even though “CMFMDA” has better forecast performance, the limitation still exists in it and still there is room for improvement. Firstly, CMFMDA may be prejudiced to miRNAs with more known affiliated ailments. Secondly, the known miRNA-disease affiliations which are experimentally verified are insufficient. Gao et al. [16] introduced a way of “nearest profile-based collaborative matrix factorization (NPCMF)” to forecast miRNA-disease affiliations. One among the foremost observable hindrances of NPCMF is that it presents an excessive amount of NP information, which can cut back the forecast accuracy whereas adding additional noise.

Xu et al. [17] proposed “probability matrix factorization (PMFMDA).” Here we have a tendency to initially combine miRNA and illness similarity. Afterward, the identified affiliation matrix and combined similarity matrix are employed to construct likelihood matrix factorization algorithmic rule to spot probably significant miRNAs for ailments. Though fairly acceptable results are got from “PMFMDA,” there are still few shortcomings with present approach. Firstly, we have a tendency to solely use semantic similar data and also Gaussian kernel similarity to develop illness similarity network. It is going to be useful to increase the prognostic performance of “PMFMDA” by combining ailment or miRNA likeness from numerous data sources such as sequence similarity. Most prior miRNA-prediction methods, however, are still hampered by the data sparsity issue. As a result, predicting miRNAs with a few known illness connections is difficult. So many methods are proposed using the matrix factorization. In this paper, miRNA-disease association method is proposed using tri-factorization.

## 2 Materials and Method

Predicting “drug-disease associations using multi-task learning based on collective matrix factorization” was proposed by Huang et al. [18]. The recommended method is built on this method. We did, however, predict a connection between miRNA and ailment. The recommended method is depicted in Fig. 1.

The technique of the matrix tri-factorization involves several important steps. First, miRNA-disease association binary matrix( $A^P$ ) is built based on HMDD database. Second, the binary matrix is factorized into three low-dimensional matrices, which are served as underlying components for miRNAs and ailments, and coefficient matrices measuring the level of interaction between underlying components. Cai et al. [19] proposed graph Laplacian regularizations, which are dependent on



**Fig. 1** Matrix tri-factorization

the genetic features of miRNAs such as (miRNA similarity) and diseases (disease semantic similarity) are presented to further improve interpretability and overview.

The following is a brief description of various similarity computations which will be utilized as input for the algorithm.

Functional similarity of MiRNAs: These scores were obtained from Wang et al.'s proposed work, <http://cmbi.bjmu.edu.cn/misimas> [20]. A functional similarity score for each miRNA pair is derived in this dataset based on the finding that genes with similar functions are frequently associated with comparable illnesses. MFS [20] is the miRNA functional similarity matrix, where the entity  $MFS(i, j)$  in row  $I$  column  $j$  is the functional similarity score between miRNAs  $m_i$  and  $m_j$ .

MFS is the symbol for the matrix.

## 2.1 Environmental Factor-Based Similarity of miRNAs

Experimentally verified miRNA-EF association data from the MirEnvironment database [22] is also used in this work. Inspired by the work of Ha et al. [21], the MiRNA similarity based on *EFs* is also computed using the equation given here.

$$MEFS_{(i,j)} = \frac{MEFS1_{(i,j)}}{\sqrt{T_{(i,i)}}\sqrt{T_{(j,j)}}} \quad (1)$$

Entries in MEFS1 are the number of common environmental factors between any two miRNAs which are extracted from MirEnvironment database.  $MEFS_{(i,j)}$ , the  $(i, j)$  entry in the matrix, is the ratio of  $MEFS1_{(i,j)}$  and product of square roots of  $T_{(i,i)}$  and  $T_{(j,j)}$ .  $T_{(i,i)}$  denotes the total number of common *EFs* between  $m_i$  and all other miRNAs. This is nothing but sum of  $i$ th row elements of MEFS1.  $T_{(j,j)}$  is similarly computed as the  $j$ th column sum of MEFS1.

## 2.2 Disease-Based Similarity of miRNAs

The “human miRNA-disease database (HMDD)” [23] is used to obtain miRNA-disease data. The operation of combining the miRNAs of different copies that create the same mature miRNA into one group and unifying the names of distinct mature miRNAs as one miRNA has been completed, resulting in the 5430 miRNA association. The illness names are curated using standard MeSH disease words from the “National Center for Biotechnology Information (NCBI).”

$$\text{MDS}_{(i,j)} = \frac{\text{MDS1}_{(i,j)}}{\sqrt{T_{(i,i)}}\sqrt{T_{(j,j)}}} \quad (2)$$

Computation of entries in MDS is similar to those of MEFS in Eq. (1). MDS1 is a matrix whose entries are extracted from HMDD database.

Maximum miRNA similarity  $W^m$  is calculated by using the three similarity calculations above as in Eq. (3).

$$W^m(i, j) = \max(\text{MFS}_{(i,j)}, \text{MEFS}_{(i,j)}, \text{MDS}_{(i,j)}) \quad (3)$$

## 2.3 Semantic Similarity Between Diseases

Detailed description of the calculation of the DSS can be found in Wang et al. [20] work. The “MeSH” database (“<http://www.ncbi.nlm.nih.gov/>”) is used to read disease classifications. It is represented as a directed acyclic graph (DAG), with nodes representing specific diseases and links representing relationships between nodes. The disease in query using the equation, and  $D$  can be denoted as a predecessor node of  $d$  as well as  $d$  itself.

$$\begin{cases} D_d(d) = 1 & \text{if } d = D \\ D_d(d) = \max\{\Delta * D_d(d') | d' \in \text{children of } d\} & \text{if } d \neq D \end{cases} \quad (4)$$

$$DV(d) = \sum_{d \in T(D)} D_D(d) \quad (5)$$

The contribution of disease  $D$  decreases as the distance between it and its ancestral disease grows. Disease  $D$  contributes 1 to itself because it is on shift 0 and is far away. Disease  $D$  helps itself because it is far away on shift 0. The decay factor semantic contribution [0, 1] spreads the ancestor’s disease. The best value for is 0.5, according to the previous works [24, 25]. In terms of semantics, two diseases that share DAG components will be very similar. The disease’s semantic similarity is calculated to be steady with the formulation of Eq. (6).

$$W^d(d(i), d(j)) = \frac{\sum_{t \in T(d(i) \cap T(d(j)))} (D_{d(i)}(t) + D_{d(j)}(t))}{DV(d(i)) + DV(d(j))} \quad (6)$$

The  $A^P$  matrix is a binary matrix. In this matrix, we have taken 5430 miRNA-disease affiliations from the human miRNA-disease database; if there is an affiliation among miRNA and disease, it is allocated a value of 1; otherwise, it is allocated a value of 0. There are 642 miRNAs and 383 ailments in this matrix.  $U$  and  $V$  are matrices of order  $m \times k$  and  $n \times k$ , respectively, with elements randomly initialized between 0 and 1.  $R^P$  is a  $k \times k$  matrix that is updated using the conjugate gradient algorithm. In general latency representation, we use matrix tri-factoring to model  $A^P$  and map MiRNA (disease).

$$\min_{U, V, R^P} \frac{1}{2} (\|A^P - UR^PV^T\|)^2 \quad (7)$$

$U$  and  $V$  represent low-dimensional matrices representing MiRNA and diseases in the Frobenius norm, respectively.  $R^P$  is a matrix of coefficients that represents the connection between latent views and the interior of the related association type. The number of measurements in low-dimensional space is denoted by  $k \min(m, n)$ . Given that Eq. 7 maps miRNAs and disease into low-dimensional space, it is only logical that low-dimensional perspectives support the underlying data of miRNAs and disease connection. Close topological inference can considerably increase the overall effectiveness of the study, according to studies [26–28], spectral graph theory [29, 30], and its applications [31–36]. This is accomplished by the proposed Laplacian rules. To include information about the similarities between our versions, we present regularization phrases based primarily on miRNA biology and disease.  $W^m \in R^{m \times m}$  denotes the miRNA-miRNA similarity matrix, whereas,  $W^d \in R^{n \times n}$  denotes disease semantic similarity. The graph “Laplacian vectors” are then constructed as “ $L^U = D^m - W^m$ ” and “ $L^V = D^d - W^d$ ”, where  $D^m$  and  $D^d$  are diagonal matrices. It contains the diagonal elements which are row sums of  $W^m$  and  $W^d$ . Graph Laplacian regularizations are expressed as follows:

$$R1 = \text{tr} \begin{pmatrix} U & \\ U^T L & U \end{pmatrix} = \frac{1}{2} \sum_{i,j}^m \|U(i,:) - U(j,:)\|_2^2 w_{ij}^m \quad (8)$$

$$R2 = \text{tr} \begin{pmatrix} V & \\ V^T L & V \end{pmatrix} = \frac{1}{2} \sum_{i,j}^n \|V(i,:) - V(j,:)\|_2^2 w_{ij}^d \quad (9)$$

Here we can see that  $\text{tr}(\bullet)$  is the sum of diagonal elements a rectangular vector and “ $U I$ ” and “ $V I$ ” represent the  $i$ th row vectors of  $U$  and  $V$ , respectively. The reader is directed to [19] for additional information on Eq. (9) obviously, reducing R1 (or

R2) will highlight the existing similarities between miRNA  $mi$  miRNA  $mj$  (or the diseases  $di$  and  $dj$ ) within the low-dimensional space.  $W^m(i, j)$  (or  $W^d(i, j)$ ) is thus greater. In addition, L2 regularizations are introduced to improve the smoothness of  $U$ ,  $V$ , and  $R_p$ . L2 regularization (proposed method) is accomplished by combining Eqs. (7), (8), and (9).

$$\begin{aligned} \min_{U, V, R^P} & \frac{1}{2} \|A^P - UR^PV^T\|_F^2 + \frac{\alpha}{2} \text{tr}(U^T L^U U) \\ & + \frac{\beta}{2} (\text{tr}(V^T L^V V) + \frac{\lambda}{2} (\|U\|_F^2 + \|V\|_F^2 + \|R^P\|_F^2)) \end{aligned} \quad (10)$$

Here  $\alpha$ ,  $\beta$ , and  $\lambda$  represent regularization parameters.

## 2.4 Optimization

To solve the optimization problem in (Eq 10), we convert it into an equivalent constrained optimization problem.

$$\begin{aligned} \min L = & \frac{1}{2} \left( \|A^P - UR^PV^T\|_F^2 + \frac{\alpha}{2} \text{tr}(W^T L^U W) + \frac{\beta}{2} \text{tr}(J^T L^V J) \right. \\ & \left. + \frac{\lambda}{2} (\|U\|_F^2 + \|V\|_F^2 + \|R^P\|_F^2) + \text{tr}(Z^T (W - U)) + \frac{\rho^1}{2} \|W - U\|_F^2 \right. \\ & \left. + \text{tr}(Y^T (J - V)) + \frac{P^2}{2} \|J - V\|_F^2 \right) \end{aligned} \quad (11)$$

Here auxiliary variables are represented as  $J$  and  $W$ , the penalty parameters are  $\rho_1 > 0$ ,  $\rho_2 > 0$ , and Lagrange multipliers are denoted by  $Z$  and  $Y$ . To provide an alternative update rule for optimization, we used the “alternative direction method of multipliers (ADMM)” framework [39] in Eq. (11). Then,  $L$  is differentiated by keeping the partial derivative to 0 and corresponding to the auxiliary variables,  $U$  and  $V$ . We are able to get the update rule shown below.

$$J = (\beta L^V + \rho_2 I)^{-1} (\rho_2 V - \gamma) \quad (12)$$

$$W = (\alpha L^U + \rho_1 I)^{-1} (\rho_1 U - Z) \quad (13)$$

$$U = (A^P V R^{P^T} + Z + \rho_1 W) (R^P V^T V R^{P^T} + \lambda I + \rho_1 I)^{-1} \quad (14)$$

$$V = (A^{P^T} U R^P + Y + \rho_2 J) ((U R^P)^T U R^P + \lambda I + \rho_2 I)^{-1} \quad (15)$$

Here I represents the adaptive dimensionality identity matrix in distinct equations, if different variables are set; the  $R^P$  is simplified as follows:

$$\min_{R^P} \frac{1}{2} \|A^P - UR^PV^T\|_F^2 + \frac{\lambda}{2} \|R^P\|_F^2 \quad (16)$$

Proposed algorithm in [40] solves Eq. (13) which is leveraged by the conjugate gradient (CG) method [40].

Details of the algorithm are omitted; solution for the Eq. (13) is presented and solved by the algorithm as CG ( $R^P$ ). At last, the  $Z$ ,  $Y$  and the  $\rho_1$ ,  $\rho_2$  are updated as follows:

$$Y = Y + \rho_2(J - V) \quad (17)$$

$$Z = Z + \rho_1(W - U) \quad (18)$$

$$\rho_1 = \mu\rho_1 \quad (19)$$

$$\rho_2 = \mu\rho_2 \quad (20)$$

### Algorithm: Tri-factorize

**Input:** miRNA-disease association matrix  $A^P$  which is binary of size,  $m \times n$ ;  
 miRNA similarity matrix,  $W^m \in R^{m \times m}$ ;  
 disease similarity matrix,  $W^d \in R^{n \times n}$ ;  
 dimensionality of the embedded space,  $k < \min(m, n)$ ;  
 regularization parameters,  $\alpha > 0$ ,  $\beta > 0$  and  $\lambda > 0$

**Output:** the prediction matrices  $AP^*$

#### Initialization:

Entries of matrices  $V \in R^{n \times k}$  and  $U \in R^{m \times k}$  in the interval  $[0, 1]$   
 randomly;  $Y = 0$  and  $Z = 0$ ;

$$\rho_1 = \rho_2 = 1$$

#### Repeat

**Update  $R^P$  using**

$$\mathbf{R}^P = \text{CG}(\mathbf{R}^P)$$

**Update J, W, U, and V using Eqs. (12–15)**

**Update Y, Z,  $\rho_1$ , and  $\rho_2$  using Eqs. (17–20)**

**End until convergence of  $\mathbf{R}^P$**

**Output  $\mathbf{A}^{P*}$ , using**

$$\mathbf{A}\mathbf{p}^* = \mathbf{U}\mathbf{R}^P\mathbf{V}^T,$$

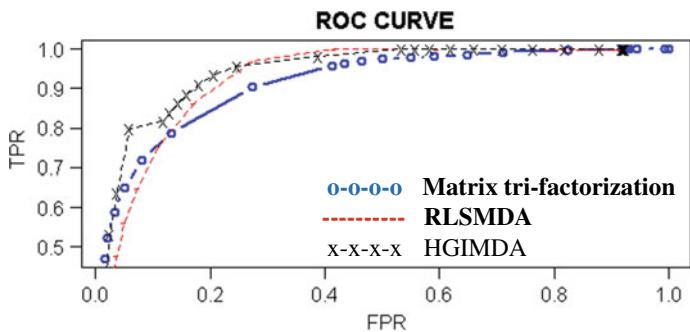
### 3 Performance Evaluation of Algorithm

Global leave one out cross-validation technique for performance forecast dependent on the known miRNA-disease affiliation is implemented. Here we have 5430 miRNA-disease associations. To test the efficacy of proposed method, global LOOCV is used. Each time one association is termed as test candidate, whereas the other associations are in the training dataset and the algorithm is run for 5430 times to get the ranking of each left out association. If the associated test miRNA has a value greater than the set threshold value, then it is treated true positive. Unassociated candidates who had the values greater than the set threshold value, they are treated as false positive. The test candidates with values less than the set threshold value are treated false negatives. In a similar way, the false positives and true positives are calculated at different thresholds and a graph is plotted with TPR as Y-axis and FPR as X-axis. We have compared the performance of proposed method with two methods “heterogeneous graph inference for miRNA-disease association (HGIMDA)” [9], “regression least square for miRNA-disease association (RLSMDA)” [12]. Note that the AUC obtained for these methods are 0.838724 and 0.8665411, whereas the AUC of the proposed method is 0.90967 (Fig. 2).

In this proposed model, six parameters  $\alpha$ ,  $\beta$ ,  $\lambda$ ,  $\rho_1$ ,  $\rho_2$ , and  $k$  are used, and it has been set as  $\alpha = 10$ ,  $\beta = 10$ ,  $\lambda = 4$ ,  $\rho_1 = 1$ ,  $\rho_2 = 1$ , and  $k = 10$  empirically tested for the different values of  $k$ . AUC obtained for different values is shown in Table 1.

The results of the experiment of the proposed algorithm for eight diseases are shown in Table 2.

This algorithm has been tested on eight prominent diseases, namely stomach cancer, ovarian cancer, breast cancer, kidney cancer, thyroid cancer, leukemia, and lung cancer. miRNAs predicted for eight diseases are confirmed from experimentally confirmed databases, namely dbDEMC [37], HMDD 3.0 [23], and MirCancer [38]. Each of these cancers has been proved to be detrimental to humans. miRNA predictions for each of the diseases help in the early diagnosis and providing the better treatment and helps in increasing the survival rate.

**Fig. 2** ROC curve**Table 1** Results for different values of  $K$ 

Value of $k$	AUC
10	0.909669556
15	0.8887
20	0.877317421
25	0.866955036
30	0.860196006
35	0.855369711
40	0.85144168
45	0.848803183
50	0.846472906

**Table 2** Confirmed miRNAs out of top 50

Disease	Confirmed miRNAs out of top 50
Stomach cancer	50
Ovarian neoplasm	49
Breast neoplasm	49
Kidney cancer	48
Thyroid cancer	46
Lung cancer	46
Leukemia	45
Brain neoplasm	42

## 4 Conclusion

In this work, we have considered the similarities between miRNAs based on functionality, environmental factors, and diseases also considered disease semantic similarity and adjacency matrix which consists of miRNA-disease associations. In this

matrix, tri-factorization method given adjacency matrix is tri-factorized by applying constraints, i.e., miRNA–miRNA-based similarity and disease semantic similarity. The matrix tri-factorization method has shown better performance when compared to contemporary methods. Matrix tri-factorization method can predict the miRNAs for all the diseases simultaneously. Further, it also predicts the miRNA for the diseases for which there are no predicted miRNAs in the literature, i.e., which does not have any related miRNAs. This method does not require any negative samples as in machine learning methods. In future, it is possible to improve the performance of the proposed method by integrating miRNA family information and miRNA target information.

**Declaration of Competing Interest** The authors declare that they have no competing interests.

## References

1. Ines, A.G., Miska, E.A.: Microrna functions in animal development and human disease. *Development* **132**(21), 4653–4662 (2005)
2. Ines, A.G., Miska, E.A.: Micrornas: genomics, biogenesis, mechanism, and function. *Cell* **116**(21), 281–297 (2004)
3. Lee, R.C., Feinbaum, R.L., Ambros, V.: The *C. elegans* heterochronic gene lin-4 encodes small RNAs with antisense complementarity to lin-14. *Cell* **75**(5):843–854 (1993)
4. Karp, X.: Encountering micrornas in cell fate signaling. *Science* **310**(5752), 1288–1289 (2005)
5. Cheng, A.M.: Antisense inhibition of human miRNAs and indications for an involvement of miRNA in cell growth and apoptosis. *Nucleic Acids Res.* **33**(5752), 1290–1297 (2005)
6. Miska, E.A.: How micrornas control cell division, differentiation and death. *Curr Opin Genet Dev.* **15**(5), 563–568 (2005)
7. Jopling, C.L., Minkyoung, Y., Lancaster, A.M., Lemon, S.M., Peter, S.: Modulation of hepatitis C virus RNA abundance by a liver-specific microRNA. *Science* **309**(5740), 1577–1581 (2005)
8. Chen, X., Liu, M., Yan, G.: RWRMDA: predicting novel human microRNA–disease associations. *Mol. BioSyst.* 2792–2798 (2012). <https://doi.org/10.1039/c2mb25180a>
9. Chen, X., Yan, C.C., Zhang, X., You, Z.-H., Huang, Y.-A., Yan, G.-Y.: HGIMDA: heterogeneous graph inference for miRNA disease association prediction. *Oncotarget* **7**(40), 65257–65269 (2016)
10. Jiang, Y., Liu, B., Yu, L., Yan, C., Bian, H.: Predict miRNA-disease association with collaborative filtering. *Neuroinformatics* **16**(3–4), 363–372 (2018)
11. You, Z.-H., Huang, Z.-A., Zhu, Z., Yan, G.-Y., Li, Z.-W., Wen, Z., Chen, X.: Pbmda: a novel and effective path-based computational model for miRNA-disease association prediction. *PLoS Comput. Biol.* **13**(3), 1005455 (2017)
12. Chen, X., Yan, G.Y.: Semi-supervised learning for potential human microRNA-disease associations inference. *Sci Rep.* **30**(4), 5501 (2014). <https://doi.org/10.1038/srep05501.PMID:24975600;PMCID:PMC4074792>
13. Li, J.Q., Rong, Z.H., Chen, X., Yan, G.Y., You, Z.H.: MCMDA: matrix completion for MiRNA-disease association prediction. *Oncotarget* **28**:8(13), 21187–21199 (2017). <https://doi.org/10.1863/oncotarget.15061>. PMID: 28177900; PMCID: PMC5400576
14. Gu, C., Liao, B., Li, X., et al.: Network consistency projection for Human miRNA-disease associations inference. *Sci Rep* **6**, 36054 (2016). <https://doi.org/10.1038/srep36054>
15. Shen, Z., Zhang, Y.-H., Han, K., Nandi, A.K., Honig, B., Huang, D.S.: miRNA-disease association prediction with collaborative matrix factorization. *Complexity* **2017**, 9 (2017). Article ID 2498957. <https://doi.org/10.1155/2017/2498957>

16. Gao, Y.-L., Cui, Z., Liu, J.-X., Wang, J., Zheng, C.-H.: NPCMF: nearest profile-based collaborative matrix factorization method for predicting miRNA-disease associations. *BMC Bioinform.* **20**(1), 353 (2019)
17. Mridha, K., et al.: Plant disease detection using web application by neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 130–136 (2021). <https://doi.org/10.1109/ICCCA52192.2021.9666354>
18. Huang, F., Qiu, Y., Li, Q., Liu, S., Ni, F.: Predicting drug-disease associations via multi-task learning based on collective matrix factorization. *Front. Bioeng. Biotechnol.* **8**, 218 (2020). <https://doi.org/10.3389/fbioe.2020.00218>
19. Cai, D., He, X., Han, J., Huang, T.S.: Graph regularized nonnegative matrix factorization for data representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**, 1548–1560 (2011). <https://doi.org/10.1109/TPAMI.2010.231>
20. Wang, D., Wang, J., Lu, M., Song, F., Cui, Q.: Inferring the human microRNA functional similarity and functional network based on microRNA-associated diseases. *Bioinformatics* **26**(13), 1644–1650 (2010). <https://doi.org/10.1093/bioinformatics/btq241>
21. Ha, J., Kim, H., Yoon, Y., Park, S.: A method of extracting disease-related microRNAs through the propagation algorithm using the environmental factor based global miRNA network. *Bio-Med. Mater. Eng.* **26**, S1763–S1772 (2015). <https://doi.org/10.3233/BME-151477>
22. Yang, Q., Qiu, C., Yang, J., Wu, Q., Cui, Q.: miREnvironment database: providing a bridge for microRNAs, environmental factors and phenotypes. *Bioinformatics* **27**, 3329–3330 (2011)
23. Li, Y., Qiu, C., Tu, J., Geng, B., Yang, J., Jiang, T., et al.: HMDD v2.0: a database for experimentally supported human microRNA and disease associations. *Nucleic Acids Res.* **42**(Database issue), D1070–D1074 (2014). <https://doi.org/10.1093/nar/gkt1023>
24. Chen, X., You, Z., Yan, G., Gong, D.: IRWRLDA: improved random walk with restart for lncRNA-disease association prediction. *Oncotarget* **7**, 57919±57931 (2016). PMID: 2751731. <https://doi.org/10.18632/oncotarget.11141>
25. Huang, Y., Chen, X., You, Z., Huang, D., Chan, K.: ILNCSIM: improved lncRNA functional similarity calculation model. *Oncotarget* **7**, 25902±25914 (2016). PMID. <https://doi.org/10.18632/oncotarget.8296>
26. Palimkar, P., et al.: Machine learning technique to prognosis diabetes disease: random forest classifier approach. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems*, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_19](https://doi.org/10.1007/978-981-16-2164-2_19)
27. Belkin, M., Partha, N., Sindhwani, V.: Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.* **7**, 2399–2434 (2006). Available online at: <http://www.jmlr.org/papers/v7/belkin06a.html>
28. Ma, Y., Fu, Y.: *Manifold Learning Theory and Applications*. CRC; Taylor & Francis distributor, Boca Raton, FL (2012). <https://doi.org/10.1201/b11431>
29. Zhang, W., Liu, X., Chen, Y., Wu, W., Wang, W., Li, X.: Feature derived graph regularized matrix factorization for predicting drug side effects. *Neurocomputing* **287**, 154–162 (2018). <https://doi.org/10.1016/j.neucom.2018.01.085>
30. Chung, F.R.K.: *Spectral graph theory*. Providence, R.I.: Published for the Conference Board of the mathematical sciences by the American Mathematical Society. (1997)
31. Rana, B., Juneja, A., Saxena, M., Gudwani, S., Kumaran, S.S., Behari, M., et al.: Graph-theory-based spectral feature selection for computer aided diagnosis of Parkinson's disease using T1-weighted MRI. *Int. J. Imag. Syst. Technol.* **25**, 245–255 (2015). <https://doi.org/10.1002/ima.22141>
32. Zhang, W., Chen, Y., Tu, S., Liu, F., Qu, Q.: Drug side effect prediction through linear neighborhoods and multiple data source integration. In: 2016 IEEE International Conference on Bioinformatics and Biomedicine (Shenzhen: BIBM), pp. 427–434 (2016a). <https://doi.org/10.1109/BIBM.2016.7822555>
33. Zhang, W., Chen, Y., Li, D.: Drug-target interaction prediction through label propagation with linear neighborhood information. *Molecules* **22**, 2056 (2017). <https://doi.org/10.3390/molecules22122056>

34. Mukhopadhyay, M., et al.: Facial emotion recognition based on textural pattern and convolutional neural network. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573860>
35. Zhang, W., Yue, X., Liu, F., Chen, Y.L., Tu, S.K., Zhang, X.N.: A unified frame of predicting side effects of drugs by using linear neighborhood similarity. BMC Syst. Biol. **11**(Suppl. 6), 101 (2017). <https://doi.org/10.1186/s12918-017-0477-2>
36. Zhang, W., Qu, Q., Zhang, Y., Wang, W.: The linear neighborhood propagation method for predicting long non-coding RNA–protein interactions. Neurocomputing **273**, 526–534 (2018). <https://doi.org/10.1016/j.neucom.2017.07.065>
37. Yang, Z., Ren, F., Liu, C., He, S., Sun, G., Gao, Q., et al.: DBDEMC: a database of differentially expressed miRNAs in human cancers. BMC Genomics **11**(Suppl. 4), S5 (2010). 10.1186%2F1471-2164-11-S4-S5
38. Xie, B., Ding, Q., Han, H., Di, W.: miRCancer: a microRNA–cancer association database constructed by text mining on literature. Bioinformatics **29**(5), 638–644 (2013). <https://doi.org/10.1093/bioinformatics/btt014>
39. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends Mach. Learn. **3**, 1–122 (2011). <https://doi.org/10.1561/2200000016>
40. Yu, H.-F., Jain, P., Kar, P., Dhillon, I.S.: Large-scale multi-label learning with missing labels. In: Proceedings of the 31st International Conference on International Conference on Machine Learning. (Beijing: JMLR.org) (2014)

# A Novel Framework for Malpractice Detection in Online Proctoring



Shaik. Mohissin Sultana and M. Kameswara Rao

**Abstract** Online exams have become increasingly popular in recent years for assessing students' content knowledge, especially during the COVID-19 outbreak. Proctoring for online tests is, however, challenging due to the lack of a face-to-face connection. Furthermore, according to a prior study, online assessments are more vulnerable to various types of cheating, putting their validity at risk. Suspicious student head and mouse movements are identified and depicted in three degrees of detail, allowing course instructors and professors to proctor online tests in a rapid, fast, and trustworthy manner. Our thorough evaluations, which include usage scenarios, well-designed user research, and expert interviews, indicate that our method is effective and practical.

**Keywords** Online proctoring · Face recognition · Head pose detection

## 1 Introduction

Many institutions and universities offer a variety of educational courses and training in actual classrooms and locations, with lectures, admissions exams, end exams, and other processes requiring presence. People will be able to attend lessons from the comfort of their own homes, at their own time and location, which will result in lower costs and a lesser environmental effect, among other benefits. Indeed, all of the above-mentioned concerns with in-class training can be alleviated or minimised with online learning. This study offers a revolutionary online monitoring deep learning-based system to continuously monitor physical locations without the need for a hardware monitor. Facial recognition utilising the OpenCV face identification method with the HOG face detector is two of the biometric approaches used by the system. To test the

---

Shaik. M. Sultana (✉) · M. Kameswara Rao

Department of Electronics and Computer Engineering, KLEF, Vaddeswaram, India  
e-mail: [180050043@kluniversity.in](mailto:180050043@kluniversity.in)

M. Kameswara Rao  
e-mail: [kamesh.manchiraju@kluniversity.in](mailto:kamesh.manchiraju@kluniversity.in)

system, which is implemented as a software system, the face detection data set and benchmark (FDDB) and Labeled Faces in the Wild (LFW) datasets are employed.

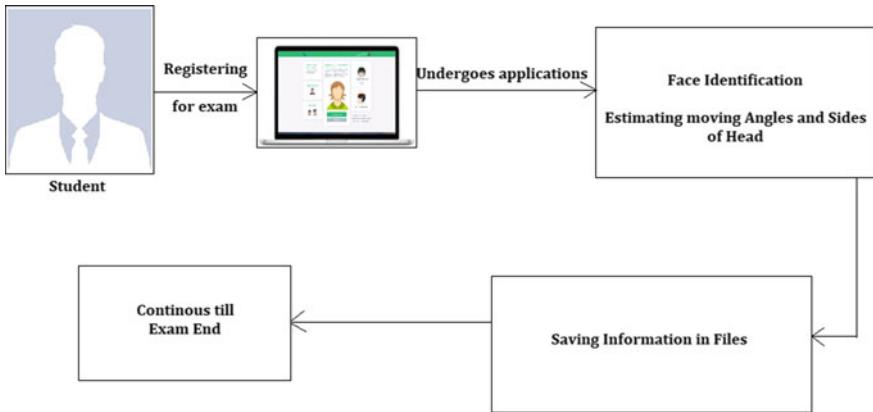
## 2 Related Works

The online exam encounters a number of challenges throughout the test. It discussed the many challenges with the online exam and presented a solution that includes grouping client for a specific location and time, hostnames or IPs, facial recognition and fingerprints, and a biometric solution. For the on-screen test, a profile-based authentication system architecture is supplied, which is based on a number of difficult questions, including a personal question, a favourite question, and a professional question. In 2018, the user's face, voice, touch, mouse and keyboard were all identified by a continuous multi-biometric authentication scheme. It builds and installs a network firewall system that uses demilitarised zone, digital protocol network, network address translation and other firewall technologies for intrusion detection. Face recognition based on deep learning is also a method for detecting the protected online test. The cheating and non-cheating status of students is used to evaluate their recommended strategies. For researchers and engineers, the online proctoring system has recently become a hurdle. The demand for online monitoring is growing, as are the challenges posed by the Coronavirus outbreak. The paid version of the proctoring system was commercialised by several industrial enterprises. This is the whole online proctoring process, from the start of the test to the end of the exam. Artificial intelligence (AI) has altered our environment and surroundings by infusing intelligence into many of our daily tasks [1–3], but it also has a number of drawbacks [4, 5]. Machine and deep learning innovation has benefitted education [6], transportation [7], communication networks [8], disaster management [9], smart cities [10] and many other sectors. Proctoring and online learning could be totally transformed by AI.

## 3 Proposed Model

The suggested online proctoring method that is accessible via the Internet is divided into two sections. There is the online proctoring, for starters. Figure 1 depicts the proposed design for the online monitoring system.

The online proctoring programme confirms the examinee's identification and prohibits any suspicious behaviour throughout the exam. Even before the test begins, the application ensures that the participant shares a screen with video and audio recording. The test does not begin until the proctors' identities have been confirmed. The main theme starts with online proctoring, which involves a student taking an exam from a monitor utilising the student's camera. The webcam captures frames using the HTTPS protocol and detects three types of techniques: face detection,



**Fig. 1** Proposed architecture

which uses facial landmarks to detect the face, face recognition, which uses loops to determine whether the face matches the student and head pose estimation, which uses facial landmarks and some functions and algorithms to estimate the student's head movement and angle. The data will be saved in a CSV file once it has been located. Throughout the exam, this strategy will be used. Throughout this method, the test will be constantly proctored. The data will be saved as a CSV file in a database. When the full inspection is completed, this operation will end, and the obtained data will be saved in a CSV file database.

#### Algorithm 1: Master Algorithm

```

1: procedure online Proctor
2:   while True do
3:     frame ← captureFrameFromWebCam
4:     getfaceDetect ← getfaceDetection(method “HOG”)
5:     findface ← findFacetForCurrentFrame
6:     facelandmarks ← findfacelandmarks for current frame
7:     detectmarks ← findfacemarks
8:     if getface == 0 then
9:       cancel Exam: face not found
10:    else if getface == 1 then
11:      continue: Exam
  
```

```

12:           end if
13:           face Match ← faceRecognition()
14:           if face Match == False then
15:               cancel Exam: Unjustified face
16:           end if
17:           angle ← HeadPoseDetection()
18:
19:           end if
20:
21:       end while
22.
23: end procedure

```

### *Face Identification*

Face identification is the most widely used biometric for internet authentication. In 1999, Intel released the OpenCV computer vision library. Face recognition algorithms provided by OpenCV include image representation, affect its operations and local binary patterns histograms (LBPH). These algorithms are based on Paul and Mi-Cascade chael's classifier Haar technology. In our suggested methodology, we use a photograph of the tutee as the input and use HOG methods to locate facial landmarks in the image. Calculate the locations of the 68 landmarks in the identified image. We have matched the photographs to people we have already recognised and saved in our database. The facial recognition pseudocode is shown in Algorithm 2.

### **Algorithm 2: Recognition of Face**

```

1: procedure Face Identification (ID, Name)
2:   while True do
3:
4:     Case ← present Case
5:     Locateface ← get all faces
6:     Encodingface ← get all faceEncoding
7:     faceMatch ← compare all faces with student face

```

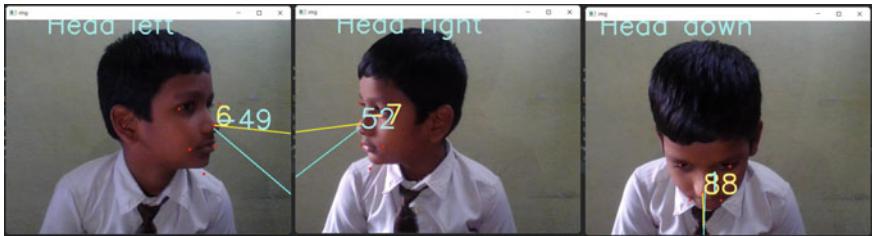
```
8:  
9:      if sameface == True then  
10:  
11: Face same  
12:  
13: else  
14: Face not same  
15:  
16:  
17: end if  
18:  
19: end while  
20:  
21: end procedure
```

### *Head Pose Estimation*

We computed at the tip of the nose, rational and translational vectors for the six parts of the face that we require using standard 3D coordinates of facial features. To create an accurate approximation, we now need to know the camera's intrinsic attributes, such as focal length, optical centre and radial distortion parameters. We can estimate the first two and pretend the last one is not there to make our job easier. We can project those 3D points onto a 2D surface, which is our picture, if we have the requisite vectors. In computer vision jargon, this is known as the Perspective-n-Point problem or PnP. The purpose of this job is to determine the posture of an item. The process of transferring the camera from its current 3D location ( $X, Y, Z$ ) to a new 3D location ( $X', Y', Z'$ ) is known as translation. There are three degrees of freedom in translation. We have the option of moving in one of three directions:  $X$ ,  $Y$ , or  $Z$ . Translation is represented by the vector  $t$ , which is equals to  $(X' - X, Y' - Y, Z' - Z)$ . The camera can be rotated on the  $X$ -,  $Y$ -, and  $Z$ -axes as well. To calculate the posture of a 3D object, you will need to know six numbers: three for translation and three for rotation.

**Algorithm 3: Head Pose Detection Algorithm**

```
1: procedure get_2d_points(rotationvector,translationvector,cameramatrix,)  
2:     while True do  
3:         frame ← currentFrame  
4:         getheadposepoints ← frame.img  
5:         facemodel ← getfacedetector()  
6:         landmark model ← get_lanidmark_model()  
7:         while True  
8:             If ret == True  
9:                 faces ← find_faces(img.face_model)  
10:                for p ← image points  
11:                    if ang1>=48:  
12:                        Print 'Head Down'  
13:                    elif ang2<=-48:  
14:                        Print('Head UP')  
15:                    elif ang2 <= 48:  
16:                        print ('Head right')  
17:                    elif ang2<=-48  
18:                        print ('Head Left')  
19:                cv2.waitKey(1)  
20:            end if  
21:        end for  
22:    end while  
23: end procedure
```



**Fig. 2** Estimating angles and sides

## 4 Results

See Fig. 2.

## 5 Conclusion

Face recognition and head pose detection techniques are used in this work to give complete information for online assessments. Our recommended technique will help to reduce disparity during the online Assessments. Human-induced detection is crucial when using an online proctoring system since it helps discover suspicious behaviour among students throughout the exam. Because for online monitoring, we only have head pose detection and face recognition algorithms. We would also like to use and study other human behaviours including multi-person detection, speaking with others, looking on various perspectives, and so on. With a significant number of false prediction positives in the majority of the tests, Dlib and MTCNN produce relatively similar results, with MTCNN having a slight edge, whereas Dlib is unable to recognise very small faces. As we saw when contrasting pictures, MTCNN may generate the good outcome shown in Fig. 2. It works well with occlusion and quick head motions and can recognise side faces. It had the fastest frame rate as well. With the help of online remote proctoring, an administrator can administer the exam from a faraway location. Proctoring ensures that online assessments are cheat-proof, allowing anyone taking the test from the comfort of their own home to pass without cheating. To put our proposed approach to the test, we chose two datasets. Future work will focus on a deeper examination and evolution of the system in a real-world setting. The goal of this study is to develop an unlock structure so that the sections can learn from one another, resulting in speedier breakthroughs in the field as a result of open-source advances.

## References

1. Yigitcanlar, T., Kankanamge, N., Regona, M., Maldonado, A., Rowan, B., Ryu, A., Desouza, K.C., Corchado, J.M., Mehmood, R., Li, R.Y.M.: Artificial intelligence technologies and related urban planning and development concepts: how are they perceived and utilized in Australia?" *J. Open Innov. Technol. Market Complex* **6**(4), 187 (2020). [Online]. Available: <https://www.mdpi.com/2199-8531/6/4/187>
2. Bedi, P., et al.: A Framework for Personalizing Atypical Web search sessions with concept-based user profiles using selective machine learning techniques. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems*, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_23](https://doi.org/10.1007/978-981-16-2164-2_23)
3. Janbi, N., Katib, I., Albeshri, A., Mehmood, R.: Distributed artificial intelligence-as-a-service (DAIaaS) for smarter IoE and 6G environments. *Sensors* **20**(20), 5796 (2020). [Online]. Available: <https://www.mdpi.com/1424-8220/20/20/5796>
4. Rajawat, A.S., et. al.: Efficient deep learning for reforming authentic content searching on big data. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems*, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_26](https://doi.org/10.1007/978-981-16-2164-2_26)
5. Yigitcanlar, T., Mehmood, R., Corchado, J.M.: Green artificial intelligence: towards an efficient, sustainable and equitable technology for smart cities and futures. *Sustainability* **13**:8952 (2021) [Online]. Available: <https://www.mdpi.com/2071-1050/13/16/8952>
6. Alam, F., Almaghthawi, A., Katib, I., Albeshri, A., Mehmood, R.: iResponse: an AI and IoT-enabled framework for autonomous COVID-19 pandemic management. *Sustainability* **13**(7), 3797 (2021). [Online]. Available: <https://www.mdpi.com/2071-1050/13/7/3797>
7. Aqib, M., Mehmood, R., Alzahrani, A., Katib, I., Albeshri, A., Altowaijri,S.: Smarter traffic prediction using big data, in-memory computing, deep learning and gpus. *Sensors (Switzerland)* **19**(9) (2019)
8. Alomari, E., Katib, I., Albeshri, A., Mehmood, R.: COVID-19: detecting government pandemic measures and public concerns from twitter arabic data using distributed machine learning. *Int. J. Environ. Res. Public Health* **18**(1), 282 (2021). [Online]. Available: <https://www.mdpi.com/1660-4601/18/1/282>
9. Rawat R., et al.: Analysis of darknet traffic for criminal activities detection using TF-IDF and light gradient boosted machine learning algorithm. In: Mekhilef S, Favorskaya M, Pandey RK, Shaw RN (eds) *Innovations in electrical and electronic engineering. Lecture notes in electrical engineering*, vol. 756. Springer, Singapore (2021). [https://doi.org/10.1007/978-981-16-0749-3\\_53](https://doi.org/10.1007/978-981-16-0749-3_53)
10. Mehmood, R., Alam, F., Albogami, N.N., Katib, I., Albeshri, A., Altowaijri, S.M.: UTiLearn: a personalised ubiquitous teaching and learning system for smart societies. *IEEE Access* **5**, 2615–2635 (2017)

# Classification of Land Cover Hyperspectral Images Using Deep Convolutional Neural Network



J. Arun Pandian, Saurav Kr. Gupta, Rohit Kumar, Shourjya Hazra, and K. Kanchanadevi

**Abstract** In this research, we proposed a novel deep convolutional neural network (DCNN) for land cover hyperspectral image classification. The Indian pines dataset was used to train and test the performance of the proposed model. Data augmentation, up-sampling, and zero-padding techniques were used to enhance the quality and quantity of the dataset. The proposed model was trained on the enhanced dataset using a graphical processing unit (GPU) environment. The trained model was tested using a test dataset and produced an accuracy of 99.3%. The testing accuracy of the proposed DCNN model was superior to other state-of-the-art machine learning techniques such as logistic regression (LR), decision tree (DT), support vector machine (SVM), and multilayer perceptron (MLP).

**Keywords** Land cover images · Hyperspectral images · Deep convolutional neural network · Data augmentation · Data up-sampling

## 1 Introduction

Hyperspectral imaging is used to take images over multiple bands of the electromagnetic spectrum. The reflectance bands of each pixel can give more clear information and carry particular properties of that object. Using human eyes, the images are divided into three spectra (long wavelength as red, medium wavelength as green, and short wavelength as blue) [1]. Whereas, hyperspectral imaging can divide the images into many more bands so that it can give information in depth by recording the fine resolution of the wavelength. There are many sensors available to record the images and much more application can be done with that. But in this model, hyperspectral imaging is used to give more accurate results while classification is done with several algorithms like support vector machine (SVM), convolutional neural network (CNN), and deep convolutional neural network (DCNN) [2]. The previous model is available multispectral images which can refer to up to 3–10 bands, whereas

---

J. A. Pandian · S. Kr. Gupta (✉) · R. Kumar · S. Hazra · K. Kanchanadevi  
Department of Computer Science & Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D  
Institute of Science and Technology, Chennai 600062, India  
e-mail: [vtu11817@veltech.edu.in](mailto:vtu11817@veltech.edu.in)

hyperspectral images are having hundreds to thousands of bands. Also, more bands can help to see the unseen. On basis of that most of the models now preferred hyperspectral imaging rather than any multispectral images. The experiment is conducted on hyperspectral images of  $145 \times 145$  pixels and 224 spectral reflectance bands in the wavelength range 0.4–2.5  $\mu\text{m}$  [3]. The dataset is called Indian Pines dataset and consists of two-thirds agriculture, and one-third forest or other natural vegetation. The dataset is classified into 16 classes which contain uneven and unorganized data. This dataset can be used for hyperspectral images classification. Apart from this, many other datasets are available for the same purpose. The Salinas scene dataset collected by the 224-band airborne visible/infrared imaging spectrometer (AVIRIS) sensor over Salinas Valley, California comprises 512 lines by 217 samples. The dataset contains data on vegetables, bare soils, and vineyard fields. The Indian Pines dataset is preferred as the classes are well-defined, and the dataset was found to be most suitable for the classification [4].

The main objective of our research is to build a classification model which can operate on hyperspectral images. However, there are various models already created using different algorithms for the same. The highest accuracy achieved by using a multilayer perceptron was 95.51%, but our model which was created using deep convolutional neural network (DCNN) algorithm provided 99.3% accuracy which is the highest attained accuracy value is ever obtained. The predicted output of for model was found to be the same as the ground truth concluding that the model worked perfectly.

This paper has discussed the process and implementation of hyperspectral image classification. The second part of the paper discusses the data collection and various preprocessing methods that were done on the dataset. The next part of the paper discusses the proposed model and the various methodologies that have been implemented. The model architecture has been explained with the importance of each layer in the proposed model. The next part of the paper is about the training and accuracy of the model. The validation and training accuracy, as well as loss, is plotted in form of a graph. This section is followed up by results and discussion where the predicted model is compared with the ground truth, and the accuracy of the model is depicted. The last section of the paper compared the proposed model with other existing models.

## 2 Literature Survey

This section discusses some recent innovations on hyperspectral image classification using image processing and artificial intelligence techniques. In [5], the authors have provided a slight introduction to hyperspectral images. It aims to convert the available RGB images into hyperspectral images using the convolutional neural network (CNN) algorithm. A two-stage CNN, also known as spectral super-resolution network (SSR-Net), was used to know about the transformation model between red, green, blue (RGB) images and hue saturation intensity (HIS). The authors in [6] proposed

an idea of using a novel principal component analysis (PCA) and segmented-PCA (SPCA)-based fusion method to perform feature extraction on hyperspectral images and also to classify them. In [7], the authors have implemented the CNN algorithm to gain high-level feature extraction from hyperspectral images. A self-looping convolutional network has been used to increase the efficiency of the classification process for the images.

The authors in [8] compared 3D CNN and support vector machine algorithms based on seed-based classification and pixel-based classification. 3D CNN was found to give higher accuracy in both classification methods. The authors in [9] have explained that hyperspectral images are a rich source of information but are not used much, this is due to the reason that these images commonly suffer from atmospheric and sensor noise. This makes preprocessing one of the most required steps before using the data for any model. Dimensionality and noise reduction are used to make the data ready for training. In [10], the authors have introduced a spatial-spectral method to classify hyperspectral images. The model is based on the combination of a deep convolutional network and intrinsic image decomposition. The raw data is first preprocessed, and the dimensionality of the hyperspectral image is reduced based on a band grouping technique.

In [11], the authors identified that deep convolutional neural network-based hyperspectral image classification is used when there are sufficient labeled samples during training. The method although gives a good accuracy but is used less due to the expensiveness of the process of labeling the data. A data mixture model has been used to argument the labeled training set quadratically and trains a deep convolutional neural network (DCNN)-based classifier. Through random sampling, the coefficient in the data mixture model is used to obtain several independent classifiers and fuse them with a voting strategy to produce final results. The authors in [12] discussed hyperspectral image classification of deep neural networks using stacked auto-encoders, deep belief networks, and convolutional neural networks. In [13], the authors have used hyperspectral image classification along with deep learning algorithms for the identification of the hybrid seeds. Principal component analysis and linear discrimination analysis were used for clustering. The authors in [14] have discussed hyperspectral image classification using convolutional neural networks (CNN). The 3D convolution extracts spectral–spatial features, whereas 2D convolution reduces the model’s complexity.

### 3 Materials and Methods

The proposed work was separated into three subdivisions such as data preparation, model design, and training. Each subdivision is discussed in detail in subsequent subsections. At first, the data preparation was discussed in the next subsection.

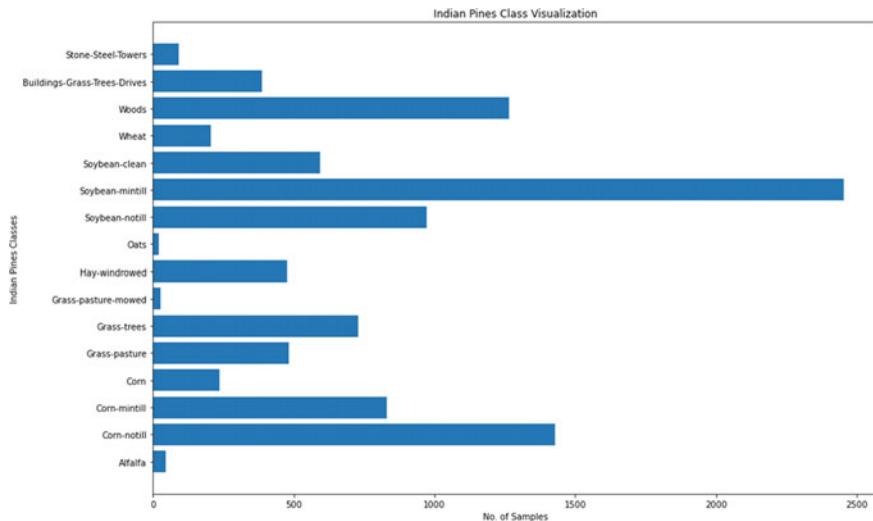
### 3.1 Data Preparation

Hyperspectral image of Indian pines is used as a segmentation dataset. The hyperspectral bands in this dataset consisting 145X145 pixels of Indian, US pines and contain 224 spectral reflectance bands for each pixel. The reflectance band is used for representing a different portion of this hyperspectral dataset with a wavelength of 0.4–2.5  $\mu\text{m}$ . In the dataset, different variants of pines are used with many samples for getting better accuracy. Now as a first step for using this dataset with this model, the mat file of corrected Indian pine and as well as ground truth Indian pine is loaded with the loadmat method. Ground truth data is used to get more accurate measurements from the satellite photographs while testing the system. The data is split randomly as 9:1 for training and testing the model.

After plotting the graph of the Indian pines dataset, it is seen that some classes have more data and some are having a very less amount of data which can affect the overall accuracy. So to balance the dataset following preprocessing techniques are used parallelly. This model used oversampling method for duplicating the unique values from the zip files and adding them to the training dataset. By using oversampling, the imbalanced dataset is ready for further processing and the new data is concatenated with X and Y values. The next method, standardization is implemented with the sklearn preprocessing package. Standard scalar is a utility class that is used for standardizing the array-like data structure and reshaping it.

Zero padding is used to give an extra border of zero's to get a larger view of the image with a black frame around it. Padding is mostly used with convolutional neural network (CNN) to preserve or restore the input values after dimensionality reduction. The patching method is also used as a preprocessing technique to reduce the noise of the image which is used after the zero-padding method. Firstly, this model split the patches with windows size and then reshaped them after removing the zero labels. Since deep neural network (DNN) gives more accurate results when it is trained with a large number of data, data augmentation is done to fit the model with the deep neural network by expanding the dataset with modified data artificially. In this model, flipping is done to add more images to the dataset. Finally, the preprocessed data is saved together to train the model (Fig. 1).

The principal component analysis is a dimensionality reduction technique used in machine learning applications. When comes to the classification of multi-dimensional images like hyperspectral images, principal component analysis (PCA) proves to give the highest accuracy. Also, it is suitable with all the methods like support vector machine (SVM), convolutional neural network (CNN), deep neural network (DNN) which are used in this model. PCA reduces all the extra dimensions of the image so that we can achieve the highest accuracy and the model works fine with the least loss. For the performed experiment, PCA works with sklearn and NumPy with a total of 75 components.



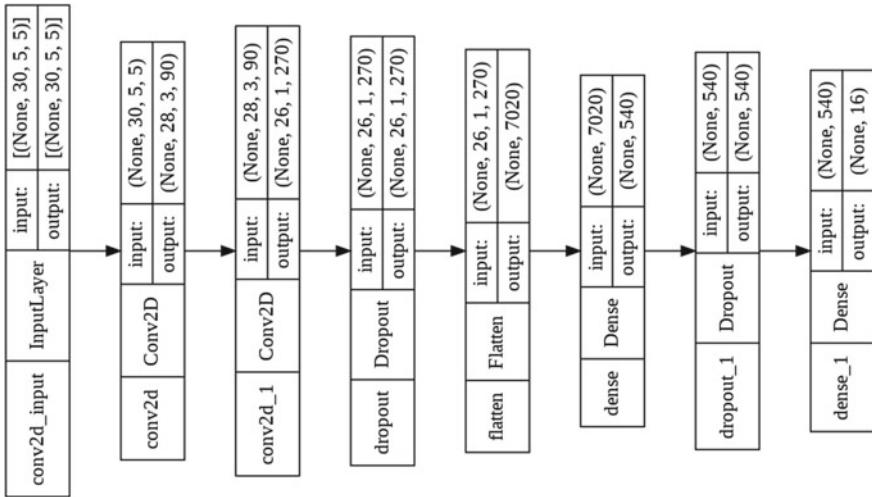
**Fig. 1** Indian pines class visualization

### 3.2 Model Design and Training

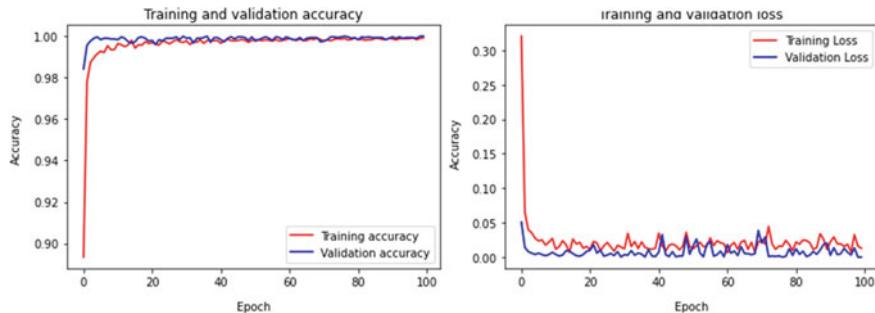
The proposed model consists of a deep convolutional neural network along with PCA. The sequential model application programming interface (API) is used where an instance of the class is created and model layers are created and added to it. The data which is initially in the shape of (30, 5, 5) is transformed into the output as (28, 3, 90) after the first layer of 2D convolution layer (Conv2D) is applied to it. The preceding layer of Conv2D transforms this in the scale of (26, 1, 270). Dropout layer is now applied to the dataset which randomly sets the input units to 0 with a frequency of 0.25 at each step during the time of training to prevent overfitting. This step also scales up the inputs that are not set to 0–1/(1-rate) such that the sum over all inputs is unchanged. Now for the ease of processing, all the dimensions of the model are removed except one layer. The data provided is reshaped to the equal number of elements contained in tensor non including the batch dimensions (Fig. 2).

For the given set of input, the model is transformed into a single-dimensional layer of 7020. The next layer is applied which is called a dense layer that changes the dimensions of the vector of 180 with rectified linear activation unit (ReLU). Now the input is in form of a vector of 540 which is again applied with a dropout of 0.5. The preceding layer applies dense on the input with a value of 16 with softmax activation function. The function is now compiled with the help of the Adam optimizer, and the loss and accuracy are observed. The compilation is done with a batch size of 32 and a validation split of 0.10 (Fig. 3).

The training and validation data count was 38,499 which was 90% of the dataset, whereas the test data count was 4277 which was 10% of the dataset. Hence, the total dataset size after all the preprocessing was 42,776.

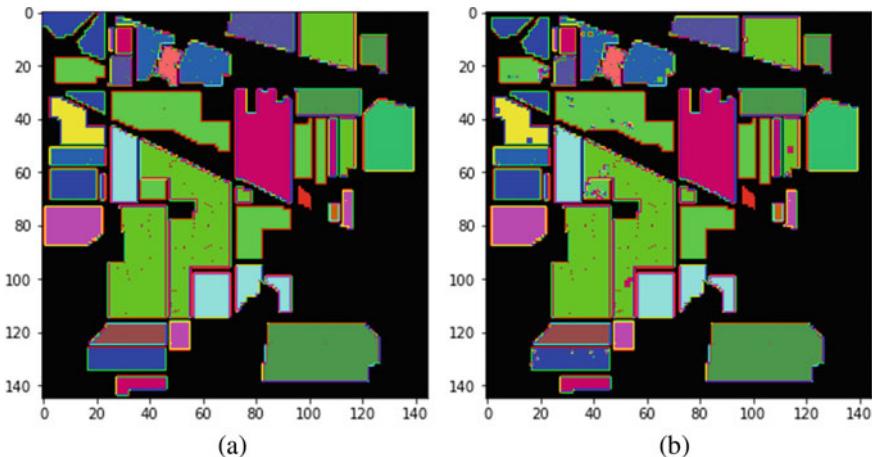


**Fig. 2** Proposed model architecture



**Fig. 3** Training and testing progresses

A graph between training and validation accuracy has been plotted, and it is concluded that the accuracy of the model increased with every increase in epoch value. The final accuracy obtained is 99.3%. It can also be seen that the training and validation accuracy plots have collided and almost follow the same path which shows that both are similar. It is observed that there is no change in training accuracy from 90 to 100 epoch values so the process is stopped for 100 epochs. Another graph is plotted between the training and validation loss. It can be observed that with the increase in epoch values, the training loss value decreases. When compared with the validation loss, the training loss is found to be a little bit above, but it follows the same path as the validation plot.



**Fig. 4** **a** Ground truth image and **b** model classified image

## 4 Results and Discussions

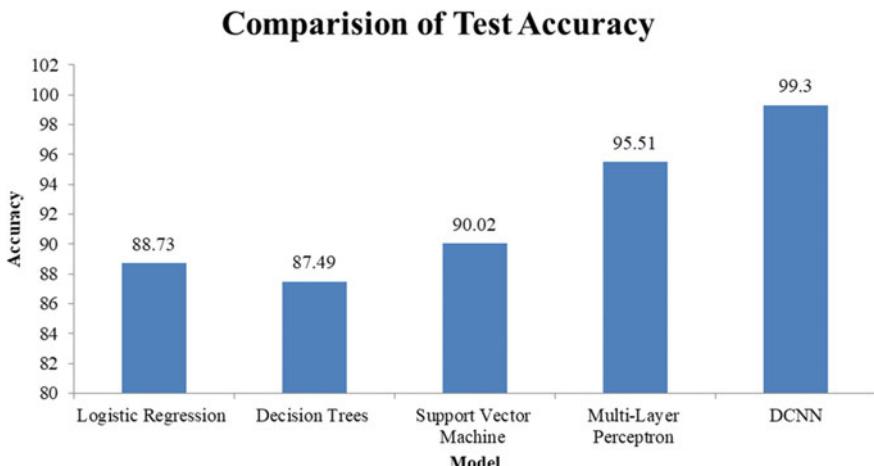
The ground truth acts as a standard that the proposed model has to follow. The below image depicts the predicted output of the model. On comparing both the images, it can be observed that both of them are almost similar to each other which means that the output is correct and the model worked correctly. Before performing the experiment, proper objective/provable data for the collected dataset was generated. The following image represents the ground truth of the image (Fig. 4).

After the experiment is performed, the statistical model automatically generates the output in terms of the efficiency of the model. The model shown approves the research hypothesis since the ground truth and predicted output coincide with each other (Fig. 5).

There are various algorithms available to obtain the desired output. Various tests have been performed to test the accuracy of several algorithms. The final model has been built after comparing the results of various other algorithms like logistic regression, decision trees, support vector machine, multilayer perception, and deep convolutional neural network. Among all of them, DCNN provided the highest accuracy of 99.3% so it was selected as the best algorithm suitable for the model. A graph showing the accuracy of various algorithms has been provided above for reference.

## 5 Conclusions

Hyperspectral images are widely used in agriculture and geographical information systems. Efficient classification of hyperspectral images is more useful to understand various objects which are present in the image. A novel deep convolutional neural



**Fig. 5** Testing result comparison

network was proposed in this research to classify hyperspectral images. The Indian pines data was used to train and test the model. The principal component analysis technique was used to reduce the dimensionality of the training and testing data. Also, the data augmentation and up-sampling techniques are used to normalize the data. The model was trained on the dataset upto 100 epochs. The model produced a testing accuracy of 99.3%. Also, the testing performance of the model was compared with other standard machine learning techniques and a fully connected neural network. The comparison result shows the performance of the proposed model was better than the other techniques. The model will be trained and tested with more complex hyperspectral data in the future. Based on the performance, the model architecture will be enhanced.

**Acknowledgements** The authors are grateful to Vel Tech Technology Business Incubator, Chennai.

## References

1. Wang, C., Liu, B., Liu, L., Zhu, Y., Hou, J., Liu, P., Li, X.: A review of deep learning used in the hyperspectral image analysis for agriculture. *Artif. Intell. Rev.* **54**, 5205–5253 (2021). <https://doi.org/10.1007/s10462-021-10018-y>
2. Kavitha, K., Arivazhagan, S., Kanaga Sangeetha, I.: Hyperspectral image classification using support vector machine in Ridgelet domain. *Natl. Acad. Sci. Lett.* **38**, 475–478 (2015). <https://doi.org/10.1007/s40009-015-0361-9>
3. Liu, L., Li, C., Lei, Y., Yin, J., Zhao, J.: Feature extraction for hyperspectral remote sensing image using weighted PCA-ICA. *Arab. J. Geosci.* **10**, 307 (2017). <https://doi.org/10.1007/s12517-017-3090-1>

4. Baumgardner, M.F., Biehl, L.L., Landgrebe, D.A.: 220 Band AVIRIS hyperspectral image data set, 12 June 1992 Indian Pine Test Site 3(2015). <https://purr.purdue.edu/publications/1947/1>. <https://doi.org/10.4231/R7RX991C>
5. Mei, S., Geng, Y., Hou, J., Du, Q.: Learning hyperspectral images from RGB images via a coarse-to-fine CNN. *Sci. China Inf. Sci.* **65** (2022). <https://doi.org/10.1007/s11432-020-3102-9>
6. Fu, H., Sun, G., Ren, J., Zhang, A., Jia, X.: Fusion of PCA and segmented-PCA domain multiscale 2-D-SSA for effective spectral-spatial feature extraction and data classification in hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* **60** (2022). <https://doi.org/10.1109/TGRS.2020.3034656>
7. Pande, S., Banerjee, B.: HyperLoopNet: Hyperspectral image classification using multiscale self-looping convolutional networks. *ISPRS J. Photogramm. Remote Sens.* **183**, 422–438 (2022). <https://doi.org/10.1016/j.isprsjprs.2021.11.021>
8. Gao, T., Chandran, A.K.N., Paul, P., Walia, H., Yu, H.: Hyperseed: An end-to-end method to process hyperspectral images of seeds. *Sensors* **21** (2021). <https://doi.org/10.3390/s21248184>
9. Mehalli, Z., Zigh, E., Loukil, A., Ali Pacha, A.: Hyperspectral data preprocessing of the Northwestern Algeria Region BT—Networking. *Intell. Syst. Secur.* (2022).
10. Mukhopadhyay, M. et al.: Facial emotion recognition based on textural pattern and convolutional neural network. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6. <https://doi.org/10.1109/GUCON50781.2021.9573860>
11. Wang, C., Zhang, L., Wei, W., Zhang, Y.: Hyperspectral image classification with data augmentation and classifier fusion. *IEEE Geosci. Remote Sens. Lett.* **17**, 1420–1424 (2020). <https://doi.org/10.1109/LGRS.2019.2945848>
12. Zeng, J., Hu, W., Huang, F.: Analysis of hyperspectral image classification technology and application based on convolutional neural networks. In: 2021 IEEE International Conference on Computer Science, Electronic Information Engineering and Intelligent Control Technology (CEI), pp. 409–414 (2021). <https://doi.org/10.1109/CEI52496.2021.9574493>
13. Bodapati, S., et al.: Comparison and analysis of RNN-LSTMs and CNNs for social reviews classification. In: Bansal, J.C., Fung, L.C.C., Simic, M., Ghosh, A. (eds.) *Advances in Applications of Data-Driven Computing. Advances in Intelligent Systems and Computing*, vol. 1319. Springer, Singapore (2021). [https://doi.org/10.1007/978-981-33-6919-1\\_4](https://doi.org/10.1007/978-981-33-6919-1_4)
14. Kumar, V., Singh, R.S., Dua, Y.: Morphologically dilated convolutional neural network for hyperspectral image classification. *Signal Process. Image Commun.* **101**, 116549 (2022). <https://doi.org/10.1016/j.image.2021.116549>

# A Comprehensive Comparative Study Between LBP and LBP Variants in Face Recognition



Shekhar Karanwal

**Abstract** LBP is renowned as one of the most powerful local descriptors for texture description. The merits of LBP are monotonic gray invariance property and less complex algorithm. Therefore, LBP was deployed successfully in diverse range of applications. The success of LBP has inspired researchers to develop new LBP variants for diverse range of applications. These LBP variants achieve good results with respect to the application they were developed. After observing merits of LBP and its variants, it is found that there is need for comprehensive comparative study among these LBP-based descriptors and chose best among all descriptors. With this note, the proposed work provides comprehensive comparative study between 15 LBP-based descriptors which includes LBP and 14 LBP variants. Apart from LBP, the other 14 are HELBP, VELBP, NI-LBP, AD-LBP, DLBP, tLBP, RD-LBP, MRELBP-NI, MB-ZZLBP, MBP<sub>6</sub> × 6 MB-LBP, OC-LBP, LDBP and LNDBP. For all descriptors, the features are extracted globally, and then, PCA is used for feature compaction. Ultimately classification is performed by RBF, the SVMs-based method. Experiments performed on ORL face dataset confirm that among all 15, it is MB-ZZLBP which secures superior accuracy than other 14 descriptors. MB-ZZLBP also out classes numerous methods from literature.

**Keywords** Feature extraction · Local descriptors · Global descriptors · Classifier · Grayscale images

## 1 Introduction

In recent times, local descriptors have shown promising results with regard to application they were developed. By performing operations on small image patches makes local descriptors more superior than global descriptors, in challenging conditions. Among all these applications, the local descriptors were exploited mostly in face recognition (FR). Some of popular local descriptors are local binary pattern (LBP)

---

S. Karanwal (✉)

Department of CSE, Graphic Era University (Deemed), Dehradun, Uttarakhand 248002, India  
e-mail: [shekhar.karanwal@gmail.com](mailto:shekhar.karanwal@gmail.com)

[1], scale invariant feature transform (SIFT) [2] and Gabor wavelet transform (GWT) [3]. Among 3, it is LBP which received attention most. The reason behind that is monotonic gray invariance property and less complex algorithm. Additionally, LBP takes much less time for execution compare to SIFT and GWT. In LBP, neighborhoods are transformed to 1 or 0 by comparing gray intensity of neighbors with center pixel. By putting weights (binomial) and aggregating values generates the LBP code. LBP showed impressive outcomes in texture study (TS) [4] and in FR [1]. This inspired researchers to develop LBP-based descriptors (variants) for diverse range of applications. Some introduced totally novel concept and some modifies the LBP methodology for developing their methodologies. All these descriptors attain good results in the considered conditions.

By conducting comprehensive survey, it is found that there is not much work reported in literature which provides the comparative study between LBP and LBP variants in FR. So, it is required/need to analyze these descriptors through comparative (experimental) study and chose the best among all. With this note, the proposed work provides the comprehensive comparative study between 15 LBP-based descriptors which includes LBP and 14 LBP variants. Apart from LBP, the other 14 are HELBP[5], VELBP [5], NI-LBP [6], AD-LBP [6], DLBP[7], tLBP [8], RD-LBP [6], MRELBP-NI [9], MB-ZZLBP [10], MB P [11], 6 × 6 MB-LBP [12], OC-LBP [13], LDBP [14] and LNDBP [14]. For all descriptors globally, feature extraction is done, and then, PCA [15] is used for feature compaction. Ultimately, classification is performed by RBF, the SVMs [16]-based method. Experiments performed on ORL [17] face dataset confirm that among all 15, it is MB-ZZLBP which secures superior accuracy than other 14 descriptors. MB-ZZLBP also outclasses various literature methods.

*Roadmap:* Sect. 2 discusses related works, and details of 15 LBP-based descriptors are disposed in Sect. 3, experiments are inclined in Sect. 4 with conclusions and future scope is elaborated in Sect. 5.

## 2 Related Works

In literature, plentiful of LBP variants were reported in various challenging conditions with regard to different applications. This section covers up some of the LBP variants (other than those mentioned earlier) which are invented after the LBP emergence.

Saidi et al. [18] invented novel LBP variant for TS so-called circular parts LBP (CPLBP). CPLBP was proposed to improve the LBP methodology, by expanding the neighborhood area from single to neighbors region, by utilizing the polar coordinates. The polar coordinates allow acquiring the discriminant features for classification. CPLBP attains good results on Outex datasets. Kar et al. [19] proposed the blocked LBP (B-LBP) for tropical cyclone investigation (TCI). B-LBP was the improved version of LBP in which center pixels are generated from the input image. Precisely, the sample image is dissolved into  $3 \times 3$  regions, and then, each block center pixels are used for forming the image pattern. B-LBP shows significant improvement over

LBP. Rasool et al. [20] develop the rotation invariant LBP-based descriptor for TS. In proposed descriptor, the radial and circumferential derivative information are considered for developing the feature size. The circumferential and radial derivatives generate information across the center pixel which is complementary in nature. The integration of both of them leads to effective descriptor for TS. Evaluation on two texture datasets proves the potent of the invented descriptor.

Vu et al. [21] take essentials of deep learning and LBP to impose the mask-based FR. Initially, retina face is used for feature learning, which is jointly extra-supervised and self-supervised learning (multi-task) detector. Additionally, LBP features are generated from masked face attributes such as eye, eyebrow and forehead locations. Further, retina face learned features and LBP are fused into 1 framework to recognize masked faces. The invented concept beats the results of various methods on different datasets. Kartheek et al. [22] invented novel descriptor radial mesh pattern (RMP) in facial expression recognition (FER). The RMP methodology is based on chess game rule. By considering the center position in  $5 \times 5$  image patch, the possible locations of bishop, knight and rook are determined, and then, feature extraction is performed from these locations. Beside binomial weights, other weights like prime, Fibonacci, squares, even and odd are used for extracting features. On 6 datasets, RMP show its relevance against several benchmark methods. Karanwal et al. [23] presented the discriminative color descriptor (DCD) in FR. In proposed descriptor, the color forms of LBP, HELBP and LPQ are generated and integrated into 1 framework. DCD comprehensively outperforms the gray version and color forms of LBP, HELBP and LPQ. DCD also outstrip various benchmark methods from literature. For evaluation, GT face dataset is used.

Kas et al. [24] discovered MLD-CBP for TS. The proposed MLD-CBP requires no training, have low dimension and robust descriptor for TS. MLD-CBP encodes the most descriptive directions available inside the multi radiiuses, which captures intensity variations that can fall in distinct directions. Results on 15 datasets confirms MLD-CBP efficacy. Karanwal et al. [25] invented the GBSBP in FR. In GBSBP, four graph structures are formed (2 each by considering  $3 \times 5$  and  $5 \times 3$  image patches). Each graph structure is filled with 3 different edges and each edge specifies unique measure. The values determined by these measures are conceived by comparison function to derive corresponding code. Finally, GBSBP code is formed by concatenating all 4 codes. GBSBP secure superb results on 4 datasets. Bedi et al. [26] proposed mean distance-LBP (MD-LBP) for lever ultrasound images. In MD-LBP, the mutual relationship between all the neighbors was transfigured to binary pattern by considering standard deviation and Euclidean distance, the 2 local statistics. Color and GLCM details are also owned in the presented work. Experiments confirm the potential of the proposed method.

### 3 Description of LBP-Based Descriptors

#### 3.1 Local Binary Pattern (LBP)

LBP [4] was introduced originally for TS. Since then, it is utilized in various applications. The reason behind LBP success is due to gray invariance monotonic property and low obscure algorithm. In LBP, the neighborhoods are transformed to 1 or 0, by comparing gray intensity of neighbors with center pixel (in  $3 \times 3$  patch). Label 1 is assigned for higher or same gray value otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation LBP code forms. By computing LBP codes in all places generates the LBP image. From LBP image, 256 size histogram is formed. The LBP code computation process is elaborated in Eq. 1. In Eq. 1,  $LBP_{S,K}$ ,  $U_{K,s}$  and  $U_C$  signify neighbors total values, radius, sole neighbor places and center pixel.

$$\begin{aligned} LBP_{S,K}(x_c) &= \sum_{s=0}^{S-1} i(U_{K,s} - U_C)2^s, \\ i(e) &= \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases} \end{aligned} \quad (1)$$

#### 3.2 Horizontal Elliptical LBP (HELBP)

HELBP [5] was introduced for FR. In HELBP, the eight elliptical neighborhoods (aligned horizontally) are transformed to 1 or 0, by comparing gray intensity of neighbors with center pixel (in  $3 \times 5$  patch). Label 1 is assigned for high error same gray value otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation HELBP code is formed for one-pixel position. By computing HELBP codes in all places generates the HELBP image. From HELBP image, the 256 size histogram is formed. The HELBP code computation process is portrayed in Eq. 2. In Eq. 2  $HELBP_{S,K_1,K_2}$ ,  $U_{K_1,K_2,s}$  and  $U_C$  specify neighbors total values, radius ( $K_1$ ), radius ( $K_2$ ), sole neighbor places and center pixel.

$$\begin{aligned} HELBP_{S,K_1,K_2}(x_c) &= \sum_{s=0}^{S-1} i(U_{K_1,K_2,s} - U_C)2^s, \\ i(e) &= \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases} \end{aligned} \quad (2)$$

### 3.3 Vertical Elliptical LBP (VELBP)

VELBP [5] was also introduced for FR. In VELBP, the eight elliptical neighborhoods (aligned vertically) are transformed to 1 or 0, by comparing gray intensity of neighbors with center pixel (in  $5 \times 3$  patch). Label 1 is assigned for high error same gray value otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation VELBP code is formed for one-pixel position. By computing VELBP codes in all places generates the VELBP image. From VELBP image, the 256 size histogram is formed. The VELBP code computation process is same as that of HELBP, only the vertical alignment of pixels is considered. Therefore, Eq. 2 can be regarded as VELBP code formation process by vertical alignment of pixels.

### 3.4 Neighborhood Intensity-Based LBP(NI-LBP)

NI-LBP [6] was introduced for TS. In NI-LBP, the neighborhoods are transformed to 1 or 0, by comparing gray intensity of neighbors with its mean (in  $3 \times 3$  patch). Label 1 is assigned for higher or same value to mean (i.e., neighbors  $\geq$  mean) otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation NI-LBP code is formed for one-pixel position. By computing NI-LBP codes in all places generates the NI-LBP image. From NI-LBP image, the 256 size histogram is formed. The NI-LBP code computation process is elaborated in Eq. 3. In Eq. 3,  $LBP_{S,K}$ ,  $U_{K,s}$  and  $\mu_{S,K}$  signify neighbors total values, radius, sole neighbor places and mean value. Equation 4 evaluates mean of the neighbors.

$$\text{NI-LBP}_{S,K}(x_c) = \sum_{s=0}^{S-1} (U_{K,s} - \mu_{S,K}), \quad (3)$$

$$i(e) = \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases}$$

$$\mu_{S,K} = \frac{1}{S} \sum_{s=0}^{S-1} U_{K,s} \quad (4)$$

### 3.5 Angular Difference-Based LBP(AD-LBP)

AD-LBP [6] was introduced for TS. In AD-LBP, the neighborhoods are transformed to 1 or 0, by comparing the gray difference of neighbors (placed according to angular displacement) with specified threshold (in  $3 \times 3$  patch). Label 1 is assigned for

high error same difference value (to threshold) otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation AD-LBP code is formed for one-pixel position. By computing AD-LBP codes in all places generates the AD-LBP image. From AD-LBP image, the 256 size histogram is formed. The AD-LBP code computation process is elaborated in Eq. 5. In Eq. 5,  $LBP_{S,K,\psi,\phi}$ ,  $U_{K,s}$  and  $U_{K,mod(s+\psi,S)}$  signify neighbors total values, radius, angular displacement, threshold, sole neighbor places and pixel positioned according to the angular displacement.  $\psi$  takes values between  $1 \leq \psi \leq 4$ . For this work  $\psi$  taken as 1, which means difference is generated between neighbors who are immediately placed (adjacent) to each, in direction clockwise.  $\phi$  used for comparison is .01.

$$\text{AD-LBP}_{S,K,\psi,\phi}(x_c) = \sum_{s=0}^{S-1} (U_{K,s} - U_{K, mod(s+\psi,S)})2^s, \quad (5)$$

$$i(e) = \begin{cases} 1 & e \geq \phi \\ 0 & e < \phi \end{cases}$$

### 3.6 Distance LBP (DLBP)

DLBP [7] was introduced for brain tumor classification (BTC). In DLBP, the neighborhoods are transformed to 1 or 0, by comparing the gray neighbors placed as per the specified distance (in  $3 \times 3$  patch). If distance is 1, then immediate neighbor (of the previous) is compared. If distance is 2, then next immediate neighbor (of previous to previous) is compared. Similarly, the comparison is performed with specific distance in clockwise direction. Label 1 is assigned for higher value otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation DLBP code is formed for one-pixel position. By computing DLBP codes in all places generates the DLBP image. From DLBP image, the 256 size histogram is formed. The DLBP code computation process is elaborated in Eq. 6. In Eq. 6,  $DLBP_{S,K}$ ,  $U_{K,s}$  and  $U_{K,(s+n)mod8}$  state the neighbors total values, radius, sole neighbor places and pixel placed as per the specified distance. For this work, the neighbor pixels are compared by using  $n = 1$ .

$$DLBP_{S,K}(x_c) = \sum_{s=0}^{S-1} i(U_{K,s} - U_{K,(s+1)mod8})2^s, \quad (6)$$

$$i(e) = \begin{cases} 1 & e > 0 \\ 0 & e \leq 0 \end{cases}$$

### 3.7 Transition LBP (tLBP)

tLBP [8] was developed for rapid object detection (ROD). In tLBP, neighborhoods are transformed to 1 or 0, by comparing the gray neighbors placed in clockwise direction. Precisely, each neighbor is compared to immediate neighbor in clockwise direction. Label 1 is assigned for higher or same gray value otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation tLBP code is formed for one-pixel position. By computing tLBP codes in all places generates the tLBP image. From tLBP image, the 256 size histogram is formed. The tLBP code formation is elaborated in Eq. 7. In Eq. 7,  $tLBP_{S,K}$ ,  $U_{K,s}$  and  $U_{K,(s+1)mod8}$  state neighbors total values, radius, sole neighbor places and immediate neighbor places.

$$tLBP_{S,K}(x_c) = \sum_{s=0}^{S-1} i(U_{K,s} - U_{K,(s+1)mod8})2^s, \\ i(e) = \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases} \quad (7)$$

### 3.8 Radial Difference-Based LBP(RD-LBP)

RD-LBP [6] was developed for TS. In RD-LBP, the neighborhoods are transformed to 1 or 0, by comparing radials pixel intensity (in  $5 \times 5$  patch). Precisely, the difference generated from radials pixels (higher-lower-order radial intensities) is transformed to label 1 for higher (+ve) or zero difference value otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation RD-LBP code is formed for one-pixel position. By computing RD-LBP codes in all places generates the RD-LBP image. From RD-LBP image, the 256 size histogram is formed. The RD-LBP code generation process is elaborated in Eq. 8. In Eq. 8,  $RD-LBP_{S,K_1,K_2}$ ,  $U_{K_2,s}$  and  $U_{K_1,s}$  state neighbors total values, radius ( $K_1$ ), radius ( $K_2$ ), pixel places at  $K_2$  and pixel places at  $K_1$ .

$$RD-LBP_{S,K_1,K_2}(x_c) = \sum_{s=0}^{S-1} i(U_{K_2,s} - U_{K_1,s})2^s, \\ i(e) = \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases} \quad (8)$$

### 3.9 Median Robust Extended LBP Based on NI(MRELBP-NI)

MRELBP-NI [9] was examined for TS. In MRELBP-NI, the median filter is utilized in regions (neighbor) of  $m \times n$  image patch. Then, neighborhoods are transformed to 1 or 0, by comparing the medians with mean of those. Label 1 is assigned for higher or same value (i.e., medians  $\geq$  mean) otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation MRELBP-NI code is formed for one-pixel position. By computing MRELBP-NI codes in all places generates the MRELBP-NI image. From MRELBP-NI image, the 256 size histogram is formed. The MRELBP-NI code generation process is elaborated in Eq. 9. In Eq. 9,  $MRELBP-NI_{S,K_2}$ ,  $\phi(U_{K_2,S,3,s})$  and  $\mu_{K_2,S,3}$  signify neighbors total values, radius, median filter and mean of neighbor medians. For this work,  $9 \times 9$  patch is used and medians are generated in eight neighborhood regions of size  $3 \times 3$ . Equation 10 generates the mean of neighborhood medians.

$$MRELBP-NI_{S,K_2}(x_c) = \sum_{s=0}^{S-1} i(\phi(U_{K_2,S,3,s}) - \mu_{K_2,S,3}) 2^s, \quad (9)$$

$$i(e) = \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases}$$

$$\mu_{K_2,S,3} = \frac{1}{S} \sum_{s=0}^{S-1} \phi(U_{K_2,S,3,s}) \quad (10)$$

### 3.10 Multiscale Block ZigZag LBP(MB-ZZLBP)

MB-ZZLBP [10] was introduced for FR. In MB-ZZLBP, the mean filter is utilized in all regions of  $6 \times 6$  patch, where each region size is  $2 \times 2$ . After obtaining mean patch, the eight pixel positions are transformed to 1 or 0, by comparing the pixels in zigzag manner. Precisely, the difference generated (by subtracting lower-higher-order pixel) is transformed to 1 for higher (+ve) or zero difference value otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation MB-ZZLBP code is formed for one-pixel position. By computing MB-ZZLBP codes in all places generates MB-ZZLBP image. From MB-ZZLBP image, 256 size histogram is formed. The MB-ZZLBP code formation process is elaborated in Eq. 12. Equation 11 computes mean in  $2 \times 2$  regions of  $6 \times 6$  patch, where  $L_{i,j}$  specifies the regions and  $U_{i,j}$  contains the mean values. In Eq. 12, MB-ZZLBP<sub>S,K</sub>,  $U_{K,s}$  and  $U_{K,s+1}$  signify neighbors total values, radius, lower order pixel and higher order pixel.

$$U_{i,j} = \text{mean}(L_{i,j}) \quad (11)$$

$$\begin{aligned} \text{MB-ZZLBP}_{S,K}(x_c) &= \sum_{s=0}^{S-1} i(U_{K,s} - U_{K,s+1})2^s, \\ i(e) &= \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases} \end{aligned} \quad (12)$$

### 3.11 Median Binary Pattern (MBP)

MBP [11] was proposed for TS. In MBP, the pixel values in  $3 \times 3$  patch are transformed to 1 or 0, by comparing the gray pixel intensities with median of those. Label 1 is assigned for higher or same value to median (i.e., gray intensities  $\geq$  median) otherwise label 0 is assigned. In MBP, there is choice to include or exclude center bit. In this work, center bit is excluded. The generated binary pattern is supplied by binomial weights and after values summation MBP code is formed for one-pixel position. By computing MBP codes in all places generates the MBP image. From MBP image, the 256 size histogram is formed. The MBP code computation process is elaborated in Eq. 13. In Eq. 13,  $\text{MBP}_{S,K}$ ,  $U_{K,s}$  and  $U_{\text{median}}$  signify the neighbors total values, radius, sole neighbor places and median value.

$$\begin{aligned} \text{MBP}_{S,K}(x_c) &= \sum_{s=0}^{S-1} i(U_{K,s} - U_{\text{median}})2^s, \\ i(e) &= \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases} \end{aligned} \quad (13)$$

### 3.12 $6 \times 6$ Multiscale Block LBP ( $6 \times 6$ MB-LBP)

MB-LBP [12] was developed for FR. In MB-LBP, mean filter is utilized in all regions (size  $a \times b$ ) of the mxn patch. Then, neighbors are compared with center pixel by using same thresholding concept as defined in LBP. For this work,  $6 \times 6$  patch window (so-called  $6 \times 6$  MB-LBP) is used which have nine regions of size  $2 \times 2$ . After mean computation, same thresholding concept is deployed (for neighborhood comparison with center pixel) as deployed in LBP. Therefore,  $6 \times 6$  MB-LBP builds the histogram size of 256.

### 3.13 Orthogonally Combined LBP (OC-LBP)

OC-LBP [13] was invented for diverse applications. In OC-LBP, the orthogonal pixels (of two groups in  $3 \times 3$  patch) are transformed to 1 or 0 by comparing gray intensity of orthogonal pixels with center pixel. Label 1 is assigned for high error same gray value otherwise label 0 is assigned. The generated binary patterns are supplied by binomial weights, and after values summation, OLBP1 and OLBP2 codes are formed for one-pixel position. By computing OLBP1 and OLBP2 codes in all places generates the OLBP1 and OLBP2 images. Both OLBP images forms the histogram size of 16. Therefore, OC-LBP size is 32. The OC-LBP code generation process is displayed in Eq. 14, Eq. 15 and Eq. 16. In Eq. 14,  $\text{OLBP1}_{S,K}$ ,  $U_{K,s1}$  and  $U_C$  signify neighbors total values (of first orthogonal group), radius, sole neighbor places and center pixel. In Eq. 15,  $\text{OLBP2}_{S,K}$ ,  $U_{K,s2}$  and  $U_C$  signify neighbors total values (of second orthogonal group), radius, sole neighbor places and center pixel. Equation 16 concatenates codes of OLBP 1 and OLBP 2 to form the OC-LBP code.

$$\text{OLBP1}_{S,K}(x_c) = \sum_{s1=0}^{(S/2)-1} i(U_{K,s1} - U_C)2^{s1} \quad (14)$$

$$i(e) = \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases}$$

$$\text{OLBP2}_{S,K}(x_c) = \sum_{s2=0}^{(S/2)-1} i(U_{K,s2} - U_C)2^{s2}, \quad (15)$$

$$i(e) = \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases}$$

$$\text{OC-LBP}_{S,K} = [\text{OLBP1}_{S,K} \quad \text{OLBP2}_{S,K}] \quad (16)$$

### 3.14 Local Difference Binary Pattern(LDBP)

LDBP [14] was presented for FR. In LDBP 2 variables are assumed, i.e.,  $A$  and  $B$ .  $A$  stores the value which is obtained by differentiating the immediate neighbor from the current neighbor and  $B$  stores the value which is obtained by differentiating the center pixel from the current neighbor. Ultimately,  $B$  is differentiated from  $A$  for obtaining the value for one neighborhood location. This concept is performed in all neighborhood locations of the  $3 \times 3$  patch, in clockwise direction. Label 1 is assigned for bigger (+ve) or zero difference value else label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation LDBP code is formed for one-pixel position. By computing LDBP codes in all places generates

LDBP image. From LDBP image, 256 size histogram is formed. The LDBP code computation process is elaborated in Eq. 17. In Eq. 17,  $\text{LDBP}_{S,K}$ ,  $U_{K,s}$ ,  $U_{K,(s+1)\bmod 8}$  and  $U_c$  signify neighbors total values, radius, current neighbor places, immediate neighbor places and center pixel.

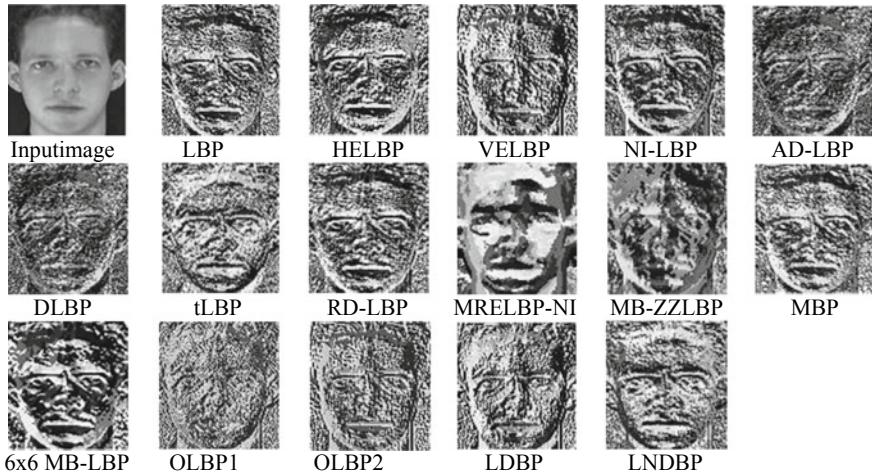
$$\begin{aligned} \text{LDBP}_{S,K}(x_c) &= \sum_{s=0}^{S-1} i((U_{K,s} - U_{K,(s+1)\bmod 8}) - (U_{K,s} - U_c))2^s \\ i(e) &= \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases} \end{aligned} \quad (17)$$

### 3.15 Local Neighborhood Difference Binary Pattern(LNDBP)

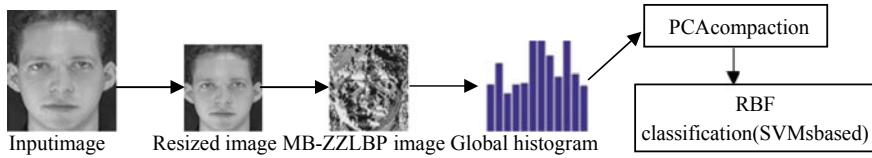
LNDBP [14] was introduced for FR. In LNDBP, the neighborhoods are transformed to 1 or 0, by comparing the gray neighbors (at distance of + 2) in direction clockwise. Precisely, each neighbor is compared to next immediate neighbor in clockwise direction. Label 1 is assigned for higher or same gray value otherwise label 0 is assigned. The generated binary pattern is supplied by binomial weights and after values summation LNDBP code is formed for one-pixel position. By computing LNDBP codes in all places generates the LNDBP image. From LNDBP image, the 256 size histogram is formed. The LNDBP code formation is elaborated in Eq. 18. In Eq. 18,  $\text{LNDBP}_{S,K}$ ,  $U_{K,s}$  and  $U_{K,(s+2)\bmod 8}$  state neighbors total values, radius, sole neighbor places and next immediate neighbor places.

$$\begin{aligned} \text{LNDBP}_{S,K}(x_c) &= \sum_{s=0}^{S-1} i(U_{K,s} - U_{K,(s+2)\bmod 8})2^s, \\ i(e) &= \begin{cases} 1 & e \geq 0 \\ 0 & e < 0 \end{cases} \end{aligned} \quad (18)$$

For all 15 descriptors, PCA is deployed to represent the features in low dimension. Then, compact size is conceived by the radial basis function (RBF) classifier for performance evaluation. RBF is SVMs-based classifier and it generally yields good results, therefore, it is utilized. The compact size details are mentioned in experiments section. Figure 1 shows transformed images of 15 descriptors which is then used for global feature extraction. The block diagram based on MB-ZZLBP is depicted in Fig. 2. Similarly, FR framework is generated with respect to each descriptor. MB-ZZLBP yields best results among all, therefore, its FR framework is displayed. In Fig. 2, the global histogram size (of MB-ZZLBP) is displayed for its identification. Its original size is declared in the defined section.



**Fig. 1** Transformed images of 15 descriptors from which global histograms are extracted



**Fig. 2** Block diagram based on MB-ZZLBP

## 4 Experiments

### 4.1 Characteristics with Regard to Used Dataset

The ORL dataset is used for evaluating all the descriptors. This dataset is also known as AT&T dataset. Since its development (1994), it has been used extensively by various researchers in their methodologies. The dataset encompasses 400 samples of 40 persons and each person contributes 10 non-identical samples, acquired in varied challenges. These challenges pertain to pose, emotion and light variations. The pose challenge is most diverse among all the challenges and image resolution is stick to  $112 \times 92$  for all the samples. Figure 3 delivers some images.



**Fig. 3** Some images of ORL

#### **4.2 The Feature Size Evolved from the Local Descriptors and Size Conceived by the RBF Classifier After Deployment of PCA**

ORL dataset encompasses bigger image resolution and if used straight way (for feature extraction) could take much time for execution. Therefore, these samples are resized to appropriate image resolution of  $52 \times 48$ . From this resized resolution, 15 descriptors are imposed for extracting features. These 15 are LBP, HELBP, VELBP, NI-LBP, A DLBP, DLBP, tLBP, RD-LBP, MRELBP-NI, MB-ZZLBP, MBP,  $6 \times 6$  MB-LBP, OC-LBP, LDBP and LNLP. Feature size obtained from 14 descriptors are 256 except OC-LBP, which forms the feature size of 32. Further, PCA is supplied to all 15 descriptors for representing the features in lower dimension. After PCA, size

conceived by RBF classifier is 25 for evaluating the performance of all descriptors. Compressed size remains same for all to make comparison fair. All simulation/coding is done in MATLAB R2018a settings.

### 4.3 Accuracy Evaluation

The accuracy is evaluated by formula inclined in Eq. 19. In Eq. 19,  $T_{st}$ , MMS and ACC signify the test size, mismatched samples and acquired ACC in %. The remaining element which is not initialized in Eq. 19, i.e.,  $T_{rg}$  represents the training size. As an example, the  $T_{rg} = 1$  (i.e., 1 sample from every person) constitutes the training size of 40 and rest 360 (i.e.,  $T_{st} = 9$  samples from every person) constitutes the test size. If the MMS found to be 3, then acquired ACC is  $397/400 = 99.25\%$ . Similarly, the ACC is formed on considered  $T_{rg}/T_{st}$  values.

$$ACC = \frac{T_{st} - MMS}{T_{st}} * 100 \quad (19)$$

On taken dataset (i.e., ORL), the  $T_{rg} = 1:3$  and  $T_{st} = 9:7$ . So by considering these values, 3 subsets are formed with respect to Trg/Tst values. On every subset, the best ACC is measured after running (RBF) the classifier 20 times. Table 1 provides the ACC comparison between all the descriptors. It is noticed from Table 1 that among all 15, the finest ACC is earned by MB-ZZLBP descriptor. Two other descriptors which come closer to MB-ZZLBP are RD-LBP and MRELBP-NI. The graph demonstration is shown in Fig. 4.

### 4.4 ACC Comparison with Literature Methods

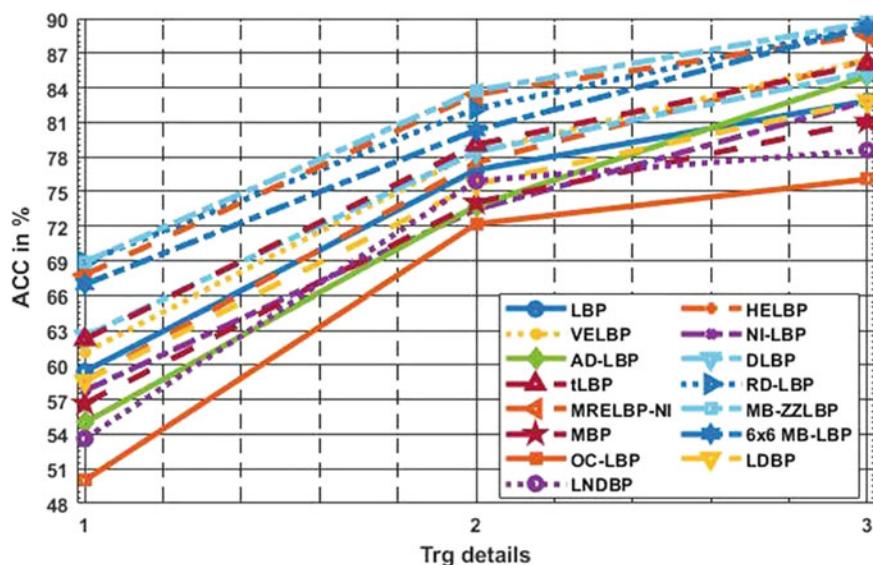
Among all 15, the finest ACC is secured by MB-ZZLBP, therefore, MB-ZZLBP is compared with literature methods. Total 10 methods are compared with MB-ZZLBP.

The ACC obtained from these are as follows: RCKSDPP [27] and KSSDPP [27] accomplish the ACC of [83.22% 90.90%] and [80.15% 88.11%] when  $T_{rg}=2:3$ .

Wavelet + 1D-PCA [28], 1D-PCA [28], SSA + NIG [29], LC-LBP [30], LPQ [30], DRC [31], ProCRC [31] and COCRC [31] obtains the ACC of [68.61% 79.06% 77.14%], [64.44% 73.43% 75.35%], [66.87% 71.87% 78.00%], [51.66% 68.75% 80.00%], [60.27% 78.12% 83.57%], [67.31% 83.44% 89.79%], [66.06% 80.94% 87.64%] and [61.78% 78.38% 87.25%] when  $T_{rg}=1:3$ . The MB-ZZLBP comprehensively outclass all the methods on  $T_{rg}=1$  and  $T_{rg}=2$ . On  $T_{rg}=3$ , two methods proves better than MB-ZZLBP and these are RCKSDPP [27] and DRC [31]. Remaining 8 are fully beaten. Table 2 provides the ACC comparison between MB-ZZLBP and the other methods. The other compared methods pertain to 5 different categories and

**Table 1** ACC comparison between all descriptors

Descriptors	$T_{rg}$ details		
	$T_{rg} = 1$	$T_{rg} = 2$	$T_{rg} = 3$
LBP	146/59.44	74/76.87	48/82.85
HELBP	149/58.61	72/77.50	38/86.42
VELBP	140/61.11	69/78.43	38/86.42
NI-LBP	152/57.77	85/73.43	48/82.85
AD-LBP	162/55.00	84/73.75	42/85.00
DLBP	135/62.50	69/78.43	41/85.35
tLBP	136/62.22	67/79.06	39/86.07
RD-LBP	112/68.88	57/82.18	30/89.28
MRELBP-NI	116/67.77	53/83.43	32/88.57
MB-ZZLBP	<b>112/68.88</b>	<b>52/83.75</b>	<b>29/89.64</b>
MBP	156/56.55	83/74.06	53/81.07
$6 \times 6$ MB-LBP	119/66.94	63/80.31	30/89.28
OC-LBP	180/50.00	89/72.18	67/76.07
LDBP	149/58.61	78/75.62	48/82.85
LNDBP	167/53.61	77/75.93	60/78.57

**Fig.4** Graph demonstration between all the descriptors

**Table 2** ACC comparison between MB-ZZLBP and others

Methods	Category	$T_{rg}$ details		
		$T_{rg} = 1$	$T_{rg} = 2$	$T_{rg} = 3$
RCKSDPP [27]	Dimension Reduction	N/A	83.22	<b>90.90</b>
KSSDPP [27]	Dimension Reduction	N/A	80.15	88.11
Wavelet + 1D-PCA [28]	Local method	68.61	79.06	77.14
1D-PCA [28]	Global method	64.44	73.43	75.35
SSA + NIG for PC2 [29]	Spectrum analysis	66.87	71.87	78.00
LC-LBP [30]	Local method	57.66	68.75	80.00
LPQ [30]	Local method	60.27	78.12	83.57
DRC [31]	CR method	67.31	83.44	<b>89.79</b>
ProCRC [31]	CR method	66.06	80.94	8764
COCRC [31]	CR method	67.78	78.38	87.25
MB-ZZLBP	Local method	<b>68.88</b>	<b>83.75</b>	<b>89.64</b>

these are dimension reduction, local, global, spectrum analysis and collaborative representation (CR).

## 5 Conclusions and Future Scope

This work presented the comprehensive comparative study between 15 LBP-based descriptors. Besides LBP, the other 14 LBP variants are HELBP, VELBP, NI-LBP, AD-LBP, DLBP, tLBP, RD-LBP, MRELBP-NI, MB-ZZLBP, MBP,  $6 \times 6$  MB-LBP, OC-LBP, LDBP and LNDBP. For all descriptors, the features are extracted globally, and then, PCA is used for feature compaction. Ultimately, classification is performed by RBF, and the SVMs-based method. Experiments performed on ORL dataset confirm that among all 15, it is MB-ZZLBP which secured superior accuracy than 14 other descriptors. MB-ZZLBP also achieves better accuracy than several literature methods. The future prospect of the proposed work is inclined in two categories: First, the regional feature extraction of all descriptors and second: Testing of these descriptors in other challenges such as noise, blur, harsh illuminations, occlusions and corruptions. The regional feature extraction will surely enhance the accuracy and to examine these descriptors on other challenges will exploit the descriptors in more tough conditions.

## References

1. Chen, T., Gao, T., Li, S., Zhang, X., Cao, J., Yao, D., Li, Y.: A novel face recognition method based on fusion of LBP and HOG. *IET Image Process.* (2021)
2. Gupta, S., Thakur, K., Kumar, M.: 2D-human face recognition using SIFT & SURF descriptors of face's feature regions. *Vis. Comp.* **37** (2021)
3. Li, C., Huang, Y., Huang, W., Qin, F.: Learning features from covariance matrix of gabor wavelet for face recognition under adverse conditions. *Pattern Recogn.* **119** (2021)
4. Ojala, T., Pietikainen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. *Pattern Recogn.* **29**(1), 51–59 (1996)
5. Nguyen, H.T., Caplier, A.: Elliptical LBPs for face recognition. In: ACCV (2012)
6. Liu, L., Zhao, L., Long, Y., Kuang, G., Fieguth, P.: Extended local binary patterns for texture classification. *Image Vis. Comput.* **30**(2), 86–99 (2012)
7. Kaplan, K., Kaya, Y., Kuncan, M., Ertunc, H.M.: Brain tumor classification using modified LBP feature extraction methods. *Med. Hypotheses* **139**, 1–12 (2020)
8. Trefny, J., Matas, J.: Extended set of LBPs for rapid object detection. In: CVWW, pp. 37–43 (2010)
9. Liu, L., Lao, S., Fieguth, P.W., Guo, Y., Wang, X., Pietikainen, M.: Median robust extended LBP for texture classification. *IEEE Trans. Image Process.* **25**(3), 1368–1381 (2016)
10. Karanwal, S., Diwakar, M.: MB-ZZLBP: multiscale block zig zag local binary pattern for face recognition. In: MARC, pp. 613–622 (2021)
11. Hafiane, A., Seetharaman, G., Zavidovique, B.: MBPs for textures classification. In: ICIAR, pp. 387–398 (2007)
12. Liao, S., Zhu, X., Lei, Z., Zhang, L., Li, S.Z.: Learning multi-scale block LBPs for face recognition. In: ICB, pp. 828–837 (2007)
13. Zhu, C., Bichot, C.E., Chen, L.: Image region description using orthogonal combination of LBP enhanced with color information. *Pattern Recogn.* **46**(7), 1949–1963 (2013)
14. Karanwal, S., Roka, S.: A robust fused descriptor under unconstrained conditions. In: SUSCOM (2021)
15. Xu, K., Fan, B., Yang, H., Hu, L., Shen, W.: Locally weighted PCA-based multimode modeling for complex distributed parameter systems. *IEEE Trans. Cybern.* 1–11 (2021)
16. Junior, P.R.M., Boult, T.E., Wainer, J., Rocha, A.: Open-set SVMs. *IEEE Trans. Syst. Man Cybern. Syst.* 1–14 (2021)
17. <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>.
18. Saidi, I.A., Rziza, M., Debayle, J.: A novel texture descriptor: circular parts LBP. *Image Anal. Sterol.* **40**(2), 105–114 (2021)
19. Kar, C., Banerjee, S.: Tropical cyclones classification from satellite images using blocked LBP and histogram analysis. In: SCTA, pp. 399–407 (2021)
20. Rasool, M., Kaur, A.: A novel rotation invariant descriptor for texture classification with LBPs. In: SCSP, pp. 385–396 (2021)
21. Vu, H.N., Nguyen, M.H., Pham, C.: Masked face recognition with convolutional neural networks and local binary patterns. *Appl. Intell.* (2021).
22. Kartheek, M.N., Prasad, M.V.N.K., Bhukya, R.: Radial mesh pattern: a handcrafted feature descriptor for facial expression recognition. *J. Amb. Intell. Human. Comput.* (2021)
23. Karanwal, S.: Discriminative color descriptor by the fusion of 3 novel color descriptors. *Optik* **244** (2021)
24. Kas, M., Khadiri, I.E., Merabet, Y.E., Ruichek, Y., Messoussi, R.: Multi level directional cross binary patterns: New handcrafted descriptor for SVM-based texture classification. *Eng. Appl. Artif. Intell.* **94** (2020)
25. Karanwal, S.: Graph based structure binary pattern for face analysis. *Optik* **241** (2021)
26. Ghosh, M., et al.: Robust face recognition by fusing fuzzy type 2 induced multiple facial fused image. In: IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573871>

27. Ding, M., Song, X., Yu, B.: An inexact proximal DC algorithm with sieving strategy for rank constrained least squares semi definite programming. [arXiv:2105.12389](https://arxiv.org/abs/2105.12389) (2021)
28. Dinariyah, I.: Alamsyah: accuracy enhancement in face recognition using 1D-PCA & 2D-PCA based on multilevel reverse-biorthogonal wavelet transform with KNN classifier. *J. Phys. Conf. Ser.* **1918**, 1–5 (2021)
29. Bodapati, S., et al.: Comparison and analysis of RNN-LSTMs and CNNs for social reviews classification. In: Bansal, J.C., Fung, L.C.C., Simic, M., Ghosh, A. (eds.) *Advances in Applications of Data-Driven Computing. Advances in Intelligent Systems and Computing*, vol. 1319. Springer, Singapore (2021). [https://doi.org/10.1007/978-981-33-6919-1\\_4](https://doi.org/10.1007/978-981-33-6919-1_4)
30. Karanwal, S.: A comparative study of 14 state of art descriptors for face recognition. *Multimedia Tools Appl.* **80**(8), 12195–12234 (2021)
31. Wang, Y., Tan, Y.P., Tan, Y.Y., Chen, H., Zhou, C., Li, L.: Generalized and discriminative collaborative representation for multiclass classification. *IEEE Trans. Cybern.* 1–12 (2020)

# Ten-Layered Deep Convolution Neural Network-Based Tea Leaf Disease Prediction



J. Arun Pandian, M. Shyamala Devi, K. Kanchanadevi, R. Aruna,  
Mupparaju Kavya Sree, Tella Malathi, and Kanagala Saideepak

**Abstract** Pests and pathogens have a significant impact on overall food production systems, resulting in massive losses in terms of volumes, reliability, survivability, and economy. Investigational evaluations and geometric experiments are two typical appropriate diagnostic methodologies designed to automatically reduce plant economic loss. Plant pathogens have quite a worldwide effect on the agricultural industry, leading to significant industrial and post-harvest damages. Timely identification of plant and diseases is essential to maintaining long-time survivability, and it makes it very difficult for research scientists. Deep learning techniques could be used to identify the tea leaf ailment using this overview. In this paper, the tea leaf images from the Mendeley data repository were used for implementation with 160 images under four classes as healthy, red scab, red leaf spot, and leaf blight. The augmentation process is done ending with 1040 augmented images, thereby forming input dataset with 1200 images. The training dataset is formed with 920 images, with validation dataset of 80 images, and system is tested with 200 images. The dataset is preprocessed starting with the train-equipped layers of ten-layered Conv2D base model. The base model is then pretrained with ImageNet with categorical cross-entropy as training loss function along with Adam optimizer and the

---

J. A. Pandian (✉) · M. Shyamala Devi · K. Kanchanadevi · R. Aruna · M. Kavya Sree ·  
T. Malathi · K. Saideepak  
Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of  
Science and Technology, Chennai, Tamilnadu, India  
e-mail: [jarunpandian@veltech.edu.in](mailto:jarunpandian@veltech.edu.in)

M. Shyamala Devi  
e-mail: [shyamaladev@gmail.com](mailto:shyamaladev@gmail.com)

K. Kanchanadevi  
e-mail: [kanchanadevicse@veltech.edu.in](mailto:kanchanadevicse@veltech.edu.in)

R. Aruna  
e-mail: [rarraruna@veltech.edu.in](mailto:rarraruna@veltech.edu.in)

M. Kavya Sree  
e-mail: [vtu13548@veltech.edu.in](mailto:vtu13548@veltech.edu.in)

T. Malathi  
e-mail: [vtu13439@veltech.edu.in](mailto:vtu13439@veltech.edu.in)

trainable parameters as 150,477,897. To obtain the high-performance key characteristics of tea leaf image data, the base model is added to develop custom layers using a transfer deep learning approach including several CNN models like VGG-19, 10-DCNN, Inception V3, ResNet-50, MobileNetV2, and DenseNet20. Performance metrics such as RoC curve, model loss, sensitivity, precision, specificity, accuracy, and F1-score are used to assess the effectiveness of ten-layered CNN frameworks with transfer learning. The project is implemented in Python and runs on NVidia Tesla V100 GPU server with 1000 training iteration and batch size of 32. Experimental results show that 10-DCNN model is found to exhibit 99.25% accuracy, precision and sensitivity of 98.5, and specificity and F1-score of 99.5.

**Keywords** Deep learning · CNN · Transfer learning · Optimizer accuracy

## 1 Introduction

Agriculture is the foundation of all economies on the planet. Agriculture is one of the most important factors that affect a country's overall market situation. Agricultural production is also a necessary condition for any country's economic growth. It is vital since it provides raw materials, work opportunities, and nourishment to a diverse range of citizens. A wide variety of such frameworks influences expected crop production, which differs throughout the world. Large amounts of chemical fertilizers, the presence of chemical compounds in water resources, non-periodic weather patterns are wildly differing soil nutrients. However, apart from these issues, one of the most difficult issues is the destruction of a large portion of production due to disease.

The presence of diseases in the plants that grow after ensuring efficient resources to the fields diminishes a substantial portion of the production. This results to a focus on critical disease detection methods in plants. The presence of numerous plant diseases is a foremost source of medium among agricultural people. Considerable efforts have been made in recent years to find leaf diseases affecting wide range of crops in different regions of the world.

Crop production is reduced because of various varieties of plants' illnesses and injuries in the aspect of durability or in the aspect of quality. Plant disease is defined as any type of modification that interferes with the normal processes of the plant. These diseases can have a significant impact on crop production, reducing both the overall yield quality and quantity. Managing a diverse crop yield encompasses a set of timely activities, such as following up of diseases that minimizes crop yield to undesirable stuff. It also consists of finding rapid solutions to a wide range of problems. Disease can have an effect on the overall functional capacity of the plant. It can result in growth retardation, reduced production, reduced leaf fall, and a diverse range of other issues.

## 2 Literature Review

Tea crop production and invention regions cover a total of 2.87 million hectares across China's 17 territories. Furthermore, the total output of manufacturing of tea plant leaves stopped 2.4 billion tons in 2016. As a result, tea plants are subjected to increased levels of heat and shade, and as a result, warm environments and abundant precipitation characterize areas where tea estates are commonly found. The current methods for diagnosing plant disease primarily recall tiny recognizable evidence for growth to subatomic natural and spectrometric processes. Atomic organic and spectroscopic distinguishing proof are more accurate, but they are labor-intensive and necessitate specialized and expensive equipment [1]. The rapid development of smart agriculture and precision agriculture in recent years has resulted in the frequent use of computer image processing techniques to solve a wide range of agricultural science problems. For example, these technologies have been used to measure plant nutrient content [2]. Support vector machines and artificial neural networks were used. ANNs are biological neural networks that self-learn, continually improving their knowledge and capabilities [3]. The multi-layer perceptron model is a multi-layer feedforward artificial neural network template that is effective for nonlinear system analysis. To trace the sum of the input vectors to the output of a neuron in the hidden layer, sigmoidal linear activation functions were used [4]. Support vector machines are a powerful classification technique that has been broadly used for a variety of pattern recognition applications, which include machine learning techniques based on statistical learning theory [5, 6].

Nonlinear transformation algorithms can be used to transform linear regression samples of relatively small space into high-dimensional feature spaces in order to make them linearly separable in the form of linear intangibility. The technique is based on structural risk minimization theory, which guides the development of an optimal hyperplane in feature space, ensuring that the learner is globally optimized and that the assumption in the complete sample space meets a certain upper bound with a certain probability. These two methods necessitate smaller sample sizes and an appropriate training rule [7]. Through stacked layers of convolutional filters, CNNs might gain enough knowledge to extract features from images on their own. CNNs are hierarchical neural networks consisting of multiple convolution layers, a pooling layer, and a full connection layer. The system begins with aspects of the networks that are local receptive fields, shared weights, and spatial sub-sampling. As a result of these characteristics, CNNs have high invariance for translation, scaling, shifting, and other types of deformation [8]. Furthermore, unlike traditional image processing algorithms, CNNs take the image directly as a network during training, ignoring the extraction of complicated components and the need for data reconstruction. Meanwhile, the high recognition accuracy of CNNs leads to widespread execution in computer vision-related fields, where development is accelerating [9]. In recent times, the development of computer vision technology has resulted in the

growing use of computational image processing and recognition methods for identifying diseases. The existing and most widely used method is based on trying to extract global features, including color features [10].

The disease spaces' available in local features are then processed using a variety of algorithms, which include local feature scale-invariant feature transform (SIFT), accelerated robust features (SURF), histogram of oriented gradient (HOG), dense scale-invariant feature transform (DSIFT), and pyramid histograms of visual words (PHOW) [11, 12]. The requirement to artificially obtain features in advance is a significant disadvantage of these methods. CNNs, on the other hand, learn data characteristics through convolution operations, making them better suited for image pattern recognition. As a result, CNNs are being used to detect plant diseases [13].

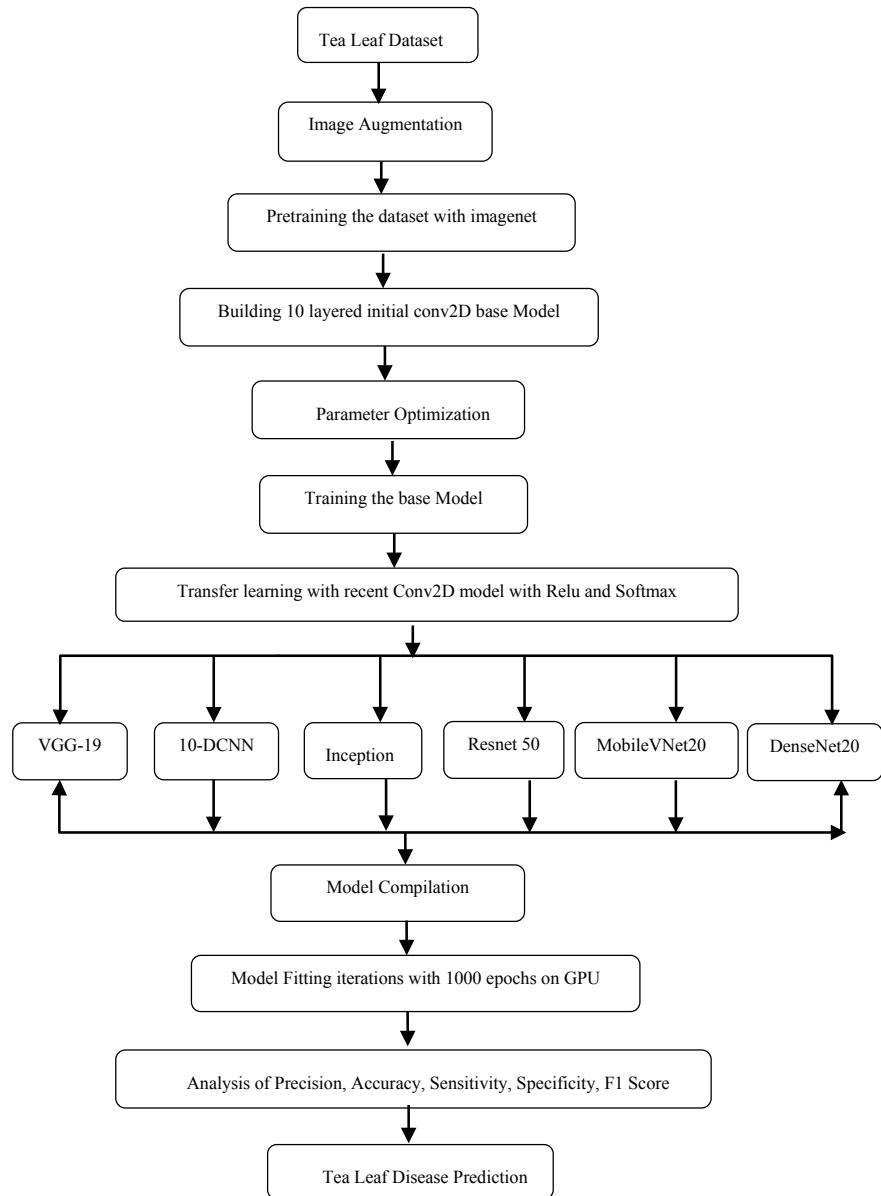
### 3 Ten-Layered Deep Convolutional Neural Network

The workflow of the proposed ten-layered deep convolutional neural network is shown in Fig. 1. The following contributions are carried out in this paper.

- The tea leaf images from the Mendeley data repository were used for implementation with 160 images.
- The augmentation process such as affine transformation, brightness, contrast, cropping, flipping, hue, padding, rotation, saturation, scaling, and translation is done ending with 1040 augmented images, thereby forming the input dataset with 1200 images.
- The training dataset is formed with 920 images, with validation dataset of 80 images, and the system is tested with 200 images. The dataset is preprocessed with the base initial trained layers of ten-layered Conv2D base model.
- The base model is then pretrained with ImageNet with categorical cross-entropy as training loss function along with Adam optimizer and the trainable parameters as 150,477,897.
- The base model is updated to the custom layers developed with transfer learning deep learning approach with several convolutional neural network models like VGG-19, 10-DCNN, Inception V3, ResNet-50, MobileNetV2, and DenseNet20 to extract the high-level general features of tea leaf images.
- The performance of ten-layered convolutional neural network models along with transfer learning is analyzed with the metrics like RoC curve, model loss, sensitivity, precision, specificity, accuracy, and F1-score.

### 4 Implementation Setup

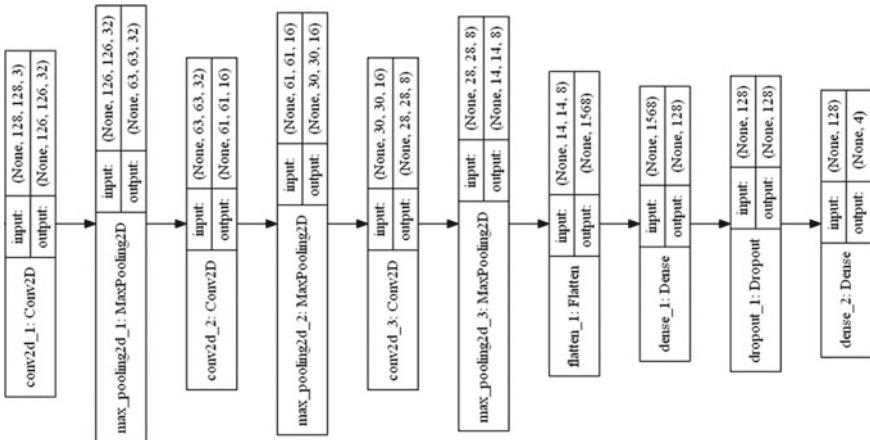
This paper uses Open Tea Leaf dataset downloaded from Mendeley data repository. The Open Tea Leaf dataset consists of 160 tea leaf images with 4 classes as healthy, red scab, red leaf spot, and leaf blight. The augmentation process consists such as



**Fig. 1** Ten-layered deep CNN system workflow



**Fig. 2** Random sample images from tea leaf disease dataset



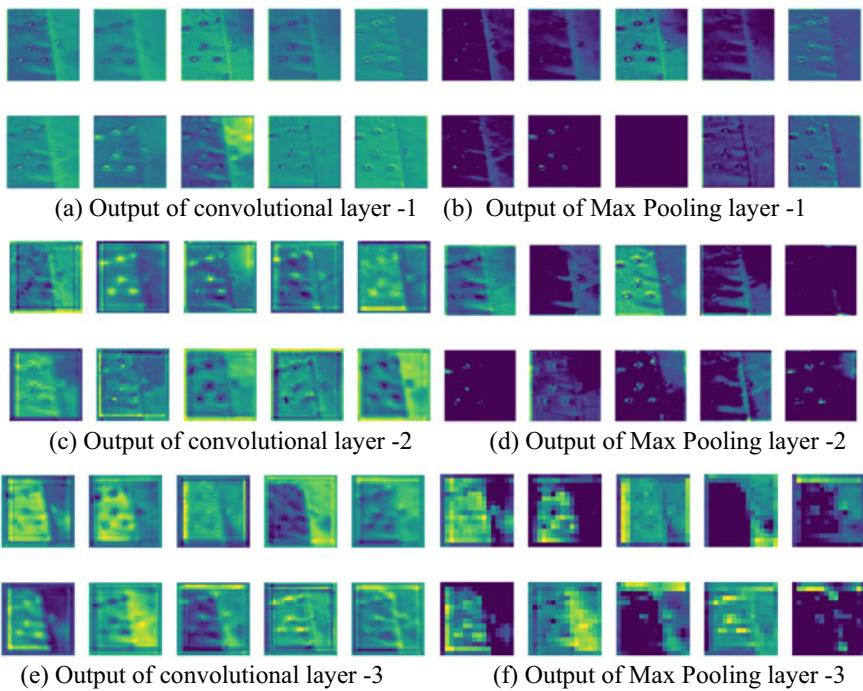
**Fig. 3** Model development of ten-layered DCNN system

affine transformation, brightness, contrast, cropping, flipping, hue, padding, rotation, saturation, scaling, and translation. The augmentation is done ending with 1040 augmented images, thereby forming the input dataset with 1200 images. The training dataset is formed with 920 images, with validation dataset of 80 images, and the system is tested with 200 images. The project is implemented with Python under NVidia Tesla V100 GPU server with training epochs of 1000 and batch size of 32. The sample leaf images is shown in Fig. 2.

The model development along with input and output size of each layers is shown in Fig. 3.

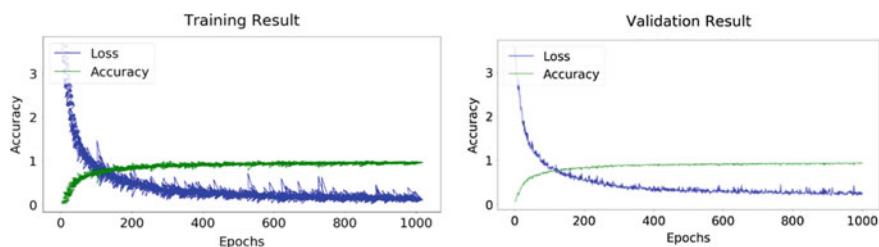
## 5 Results and Discussion

The output of the convolutional and max pooling layer is shown in Fig. 4.

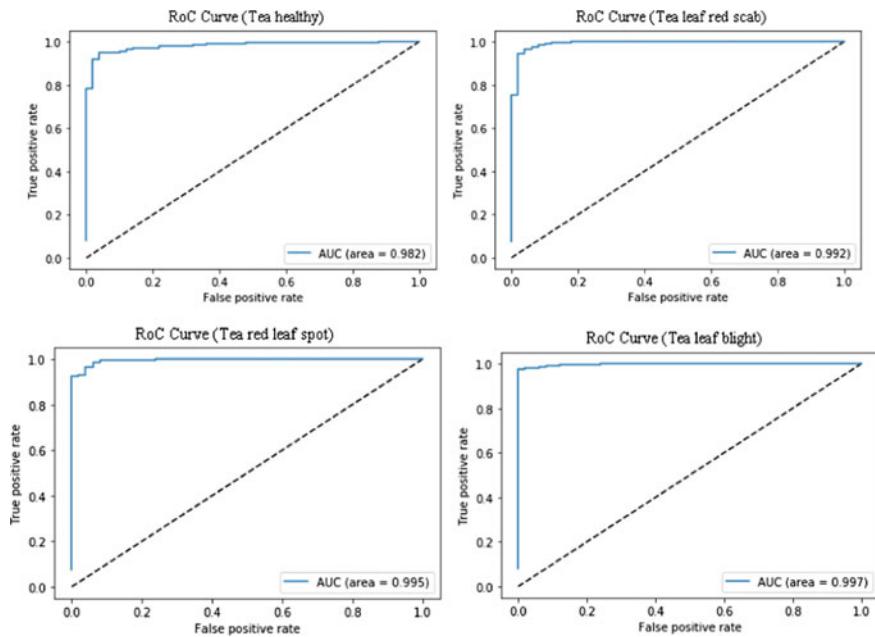


**Fig. 4** Convolutional and max pooling layers output

The dataset is preprocessed with the initial trained layers of ten-layered Conv2D base model. The base model is then pretrained with ImageNet with categorical cross-entropy as training loss function along with Adam optimizer and the trainable parameters as 150,477,897. The base model is added to the custom layers developed with transfer learning deep learning approach with several convolutional neural network models like VGG-19, 10-DCNN, Inception V3, ResNet-50, MobileNetV2, and DenseNet20 to extract the high-level general features of tea leaf images. The training validation loss and training validation accuracy are shown in Fig. 5. The RoC curve obtained for each of the tea leaf classes is shown in Fig. 6, and the readings



**Fig. 5** Training validation loss and accuracy



**Fig. 6** AuC–RoC curves of 10-DCNN model on tea leaf disease detection

are shown in Table 1. The performance metrics obtained from the ten-layered CNN toward tea leaf disease detection are shown in Figs. 11, 12, and 13 (Figs. 7, 8, 9, and 10).

## 6 Conclusion

This paper attempts to provide the detection of tea leaf disease using ten-layered deep convolutional neural network. The output of the convolutional and max pooling layers is also analyzed. The performance of ten-layered convolutional neural network models along with transfer learning is analyzed with the metrics like RoC curve, model loss, sensitivity, precision, specificity, accuracy, and F1-score. The project is implemented with Python under NVidia Tesla V100 GPU server with training epochs of 1000 and batch size of 32. Experimental results show that 10-DCNN model is found to exhibit 99.25% accuracy, precision and sensitivity of 98.5, and specificity and F1-score of 99.5. Future direction of the research is to include other possible tea leaf diseases for detection.

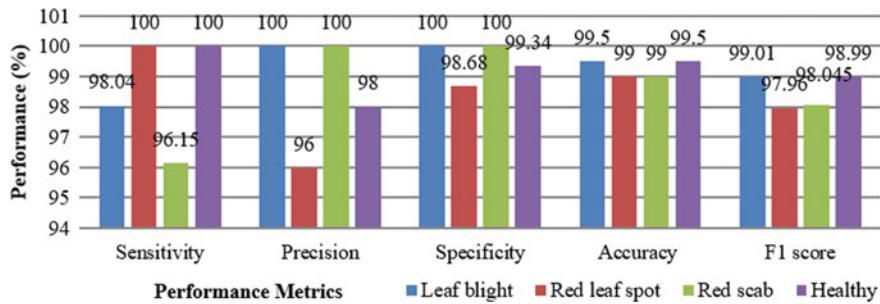
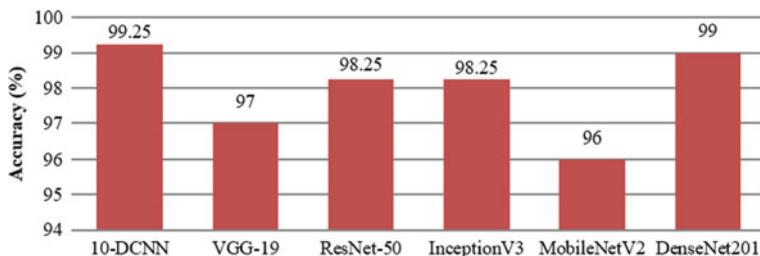
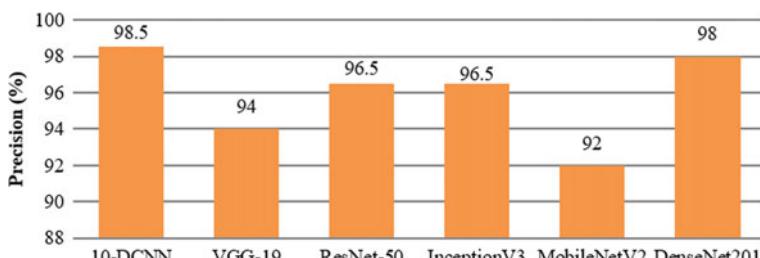
**Table 1** Performance metrics of tea leaf disease detection

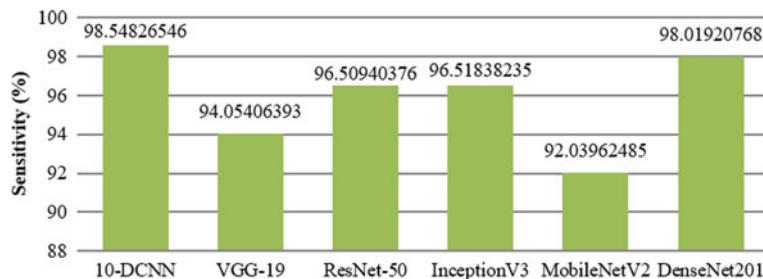
Class type	10-DCNN	VGG-19	ResNet-50	Inception V3	MobileNetV2	DenseNet20
<i>Sensitivity</i>						
Leaf blight	95.83333	97.95918	96.07843	93.61702	100.00000	95.83333
Red leaf spot	92.30769	96.00000	96.00000	90.38462	97.95918	92.30769
Red scab	92.15686	96.00000	97.91667	92.15686	98.03922	92.15686
Healthy	95.91837	96.07843	96.07843	92.00000	96.07843	95.91837
Average sensitivity	94.05406	96.50940	96.51838	92.03962	98.01921	94.05406
<i>Precision</i>						
Leaf blight	100.0000	92.0000	96.0000	98.0000	88.0000	98.0000
Red leaf spot	96.0000	96.0000	96.0000	96.0000	94.0000	96.0000
Red scab	100.0000	94.0000	96.0000	94.0000	94.0000	100.0000
Healthy	98.0000	94.0000	98.0000	98.0000	92.0000	98.0000
Average precision	98.5000	94.0000	96.5000	96.5000	92.0000	98.0000
<i>F1-score</i>						
Leaf blight	99.0099	93.8776	96.9697	97.0297	90.7216	98.9899
Red leaf spot	97.9592	94.1176	96.0000	96.0000	92.1569	96.9697
Red scab	98.0392	93.0693	96.0000	95.9184	93.0693	99.0099
Healthy	98.9899	94.9495	97.0297	97.0297	92.0000	97.0297
Average F1-score	98.4995	94.0035	96.4998	96.4944	91.9870	97.9998
<i>Specificity</i>						
Leaf blight	100.0000	97.3684	98.6755	99.3289	96.0784	99.3377
Red leaf spot	98.6842	98.6486	98.6667	98.6667	97.9730	98.6755
Red scab	100.0000	97.9866	98.6667	98.0263	97.9866	100.0000
Healthy	99.3377	98.0132	99.3289	99.3289	97.3333	99.3289
Average specificity	99.5055	98.0042	98.8344	98.8377	97.3428	99.3355
<i>Accuracy</i>						
Leaf blight	99.5000	97.0000	98.5000	98.5000	95.5000	99.5000
Red leaf spot	99.0000	97.0000	98.0000	98.0000	96.0000	98.5000
Red scab	99.0000	96.5000	98.0000	98.0000	96.5000	99.5000
Healthy	99.5000	97.5000	98.5000	98.5000	96.0000	98.5000

(continued)

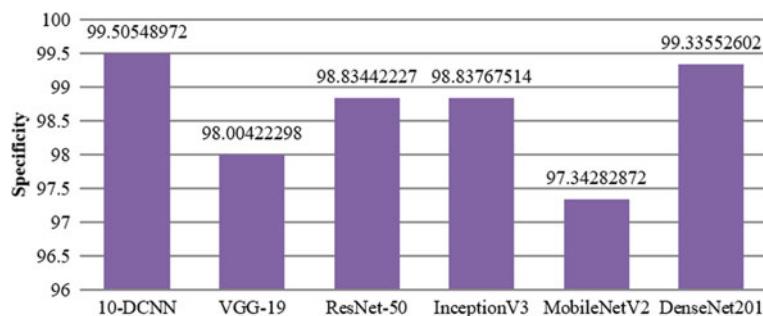
**Table 1** (continued)

Class type	10-DCNN	VGG-19	ResNet-50	Inception V3	MobileNetV2	DenseNet20
Average sensitivity	99.2500	97.0000	98.2500	98.2500	96.0000	99.0000

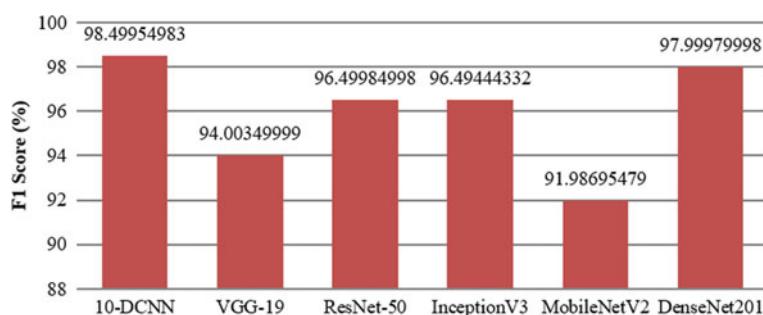
**Fig. 7** Performance of 10-DCNN on tea leaf disease detection**Fig. 8** Average accuracy comparison of 10-DCNN on tea leaf disease detection**Fig. 9** Average precision comparison of 10-DCNN on tea leaf disease detection



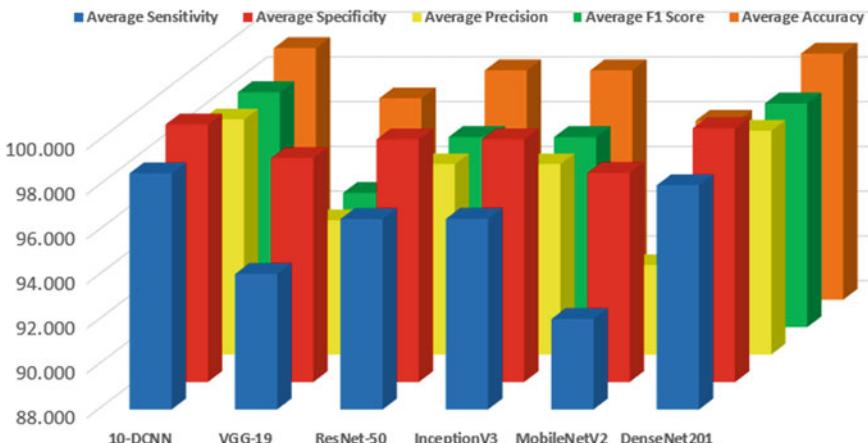
**Fig. 10** Average sensitivity comparison of 10-DCNN on tea leaf disease detection



**Fig. 11** Average specificity comparison of 10-DCNN on tea leaf disease detection



**Fig. 12** Average F1-score comparison of 10-DCNN on tea leaf disease detection



**Fig. 13** Average performance metrics of 10-DCNN on tea leaf disease detection

## References

1. Yang, N., Yuan, M.F., Wang, P., Zhang, R. B., Sun, J., Mao, H.P.: Tea diseases detection based on fast infrared thermal image processing technology. *J. Sci. Food Agric.* (2019)
2. Sulistyo, S.B., Wu, D., Woo, W.L., Dlay, S.S., Gao, B.: Computational deep intelligence vision sensing for nutrient content estimation in agricultural automation. *IEEE Trans. Autom. Sci. Eng.* **15**, 1243–1257 (2018)
3. Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., Stefanovic, D.: Deep neural networks based recognition of plant diseases by leaf image classification. *Comput. Intell. Neurosci.* 3289801 (2016)
4. Abdullah, N.E., Rahim, A.A., Hashim, H., Kamal, M.M.: Classification of rubber tree leaf diseases using multilayer perceptron neural network. In: Proceedings of the 5th Student Conference on Research and Development-SCOReD, Shah Alam, Malaysia (2007)
5. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.* **2**, 121–167 (1998)
6. Vapnik, V.: Statistical Learning Theory. Wiley, New York, NY, USA (1998)
7. Mukhopadhyay, M., et al.: Facial emotion recognition based on textural pattern and convolutional neural network. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). [https://doi.org/10.1109/GUCON5\\_0781.2021.9573860](https://doi.org/10.1109/GUCON5_0781.2021.9573860)
8. Mridha, K., et al.: Plant disease detection using web application by neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 130–136 (2021). <https://doi.org/10.1109/ICCCA52192.2021.9666354>
9. Ferentinos, K.P.: Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* **145**, 311–318 (2018)
10. Mridha, K., et al.: Deep learning algorithms are used to automatically detection invasive ducal carcinoma in whole slide images. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 123–129 (2021). <https://doi.org/10.1109/ICCA52192.2021.9666302>
11. Pires, R.D.L., Goncalves, D.N., Orue, J.P.M., Kanashiro, W.E.S., Rodrigues, J.F., Jr., Machado, B.B., Goncalves, W.N.: Local descriptors for soybean disease recognition. *Comput. Electron. Agric.* **2016**(125), 48–55 (2016)

12. Zhang, S.W., Zhu, Y.H., You, Z.H., Wu, X.W.: Fusion of superpixel, expectation maximization and PHOG for recognizing cucumber diseases. *Comput. Electron. Agric.* **2017**(140), 338–347 (2017)
13. Pichayoot, O.: Corn disease identification from leaf images using convolutional neural networks. In: Proceedings of the 21st International Computer Science and Engineering Conference (ICSEC), Bangkok, Thailand (2017)

# Maximum Relevancy and Minimum Redundancy Based Ensemble Feature Selection Model for Effective Classification



A. Saravanan<sup>ID</sup>, C. Stanly Felix, and M. Umarani

**Abstract** Feature selection algorithms are widely used in most of the critical applications of machine learning, pattern recognition, disease diagnosis and fraud identification. Specifically, due to the existence of redundant and irrelevant attributes, the performance of the underlying model will be degraded with higher time complexity and lower accuracy. Thus, feature selection has become one of the most widely used pre-processing techniques as it improves performance concerning time and accuracy. Several methods exist in identifying the optimal subset of features from the large feature set. However, none of the single feature selection techniques is proved to be effective for all types of the training set. Each method will be effective with specific classifiers and for specific datasets. This paper utilizes the maximum relevancy and minimum redundancy based ensemble feature selection for effective classification with maximized accuracy. The proposed model collects the ranks from different filtering techniques and convert them to scores using the rank-sum approach. The accumulated scores signify the feature importance for which the mutual information for each selected feature with the class variable as well as the selected optimal features are computed. The model has been validated using various experiments by different classifiers, and it is proved to be more effective in improving the performance of the classification results than the existing single rank techniques.

**Keywords** Ensemble model · Feature selection · Mutual information · Minimum redundancy · Maximum relevancy

## 1 Introduction

Data mining has become one of the most significant fields in recent decades, and it is the base for multidisciplinary research evolution. It is a most common field which is often used with machine learning, statistics, soft computing, pattern recognition

---

A. Saravanan (✉) · C. S. Felix · M. Umarani

Department of Computing, Coimbatore Institute of Technology, Coimbatore, Tamilnadu 641 014, India

e-mail: [a.saravanan21@gmail.com](mailto:a.saravanan21@gmail.com)

and artificial intelligence [1]. As data mining is intended to extract useful and meaningful knowledge from the vast amount of data and as the large amount of data is getting increased day by day that requires to be transformed into a piece of useful information, data mining and its related fields are always a top-grade research topic in almost all domains [2]. Though data mining is considered as one of the iterative sequence of steps in knowledge discovery in database (KDD) process that includes data selection, data pre-processing, data transformation, data mining and evaluation [3], it is considered to be the most essential part as it transforms the formatted data into knowledge using various tasks [4].

Often the data mining tasks can be either predictive that includes classification and prediction or descriptive that includes association rule mining, outlier mining and clustering [5]. With the existence of various data mining tasks, the most commonly used task is a classification that aims at assigning the given observation to a target class. The main challenge is the accuracy in predicting the target class for the test instances [6]. The classification process has wide usage in various fields including disease diagnosis [7], credit risk analysis [8–10], predicting student's performance [11], categorizing musical items [12], detection network anomalies [13], identifying outliers [14], noise detection [15] and more. The classification process can be implemented in two steps in which the first step is the model construction that utilizes pre-defined class labels from the training data to train the classification model, and the second step is the model usage in which it predicts the unknown labels for the test data. The classification model is always evaluated using the prediction accuracy and the time complexity. An enormous amount of classification algorithms has been proposed so far in which most of them are widely used in many applications including neural networks, decision trees, naïve Bayes, nearest neighbour, random forest, support vector machine and more [16].

However, the quality of the results often depends on the quality of the data given as an input for the underlying model. Thus, to improve the quality of the input, several pre-processing steps are proposed in which feature selection plays a vital role as it selects the significant feature subset from the huge feature set for the underlying study based on the relevancy and non-redundancy [17]. Several machine learning algorithms are available to select the optimal subset of the features that improves the prediction accuracy. The main objective of any feature selection model is to improve the accuracy of the underlying model, remove the redundant attributes and improve the relevancy as well as minimize the time complexity [18]. These feature selection techniques can be either filter based or wrapper based [19]. Filter-based models identify the significant features using specific methods like information gain (IG), relief and gain ratio, whereas wrapper-based feature selection models utilize a learning algorithm for selecting optimal subset [20]. Filter-based models are much faster than wrapper-based models since they utilize cheaper computations including the metrics such as distance [4].

Several feature selection algorithms were proposed in the literature for the past two decades. However, the accuracy of the classification is still needs to be increased and execution time to be decreased. This is due to the presence of more redundant features that exist in the selected features. Also, another significant problem to be

addressed is the biasness created by the individual models [17]. Thus, to overcome the issues in the existing feature selection models specifically in filter-based models, the paper introduces an ensemble-based feature selection model that makes use of various filtering techniques such as chi-square, symmetric uncertainty, information gain, gain ratio and relief. The model accumulates the results obtained from various filtering models and evaluates the relevancy and redundancy between the class attribute and other optimal features through mutual information computation, respectively. The experimental analysis shows that the proposed model offers better classification accuracy than other individual filtering techniques and existing ensemble models.

The organization of the paper is as follows. Section 2 presents the existing literature works related to the field of study. Section 3 describes the proposed model with the architectural design as well as algorithm pseudocode with proper explanation. Section 4 explains the experimental study carried out for the proposed model along with the datasets used. The performance analysis of the proposed model and the comparison of results with the existing models are presented in Sect. 5. Section 6 concludes the paper with the research findings and the future work to be carried out.

## 2 Related Works

As the data is increasing data by data, it not only leads to an increase in the size of the samples but also increases the dimensionality of the data due to usage of the internet and digitalization. Thus, dimensionality reduction has gained popularity by removing redundant features and selecting significant optimal feature subsets. In the last few decades, a various number of research ideas have been proposed for selecting significant features that are effective for classification. A survey on semi-supervised feature selection models was reviewed that covers the basic as well as semi-supervised feature selection representing hierarchical structure [21]. Similarly, 22 different filter-based feature selection models were analysed using several benchmark datasets. Based on the analysis, the authors conclude that there is no single group of filters that outperforms all the datasets and always produces different results for different datasets [22].

In general, ensemble methods are used in feature selection as it makes use of more than one feature selection model. However, the feature selection and ensemble model are often used in two ways in which one is used for successive ensemble methods [23, 24] and the other utilizes an ensemble of feature selection to improve scalability [25]. The main aspects to be considered in proposing the ensemble model for feature selection are (1) type of feature selection techniques [26], (2) the number of feature selection techniques [27, 28], (3) aggregation method used to combine the results [29] and (4) the threshold method for selecting the number of features [30].

An ensemble model that utilizes various classifiers such as CART, CHAID and QUEST was proposed in which the model makes use of a confidential-based weighted voting scheme for combining the results [31]. Mutual information-based ensemble feature selection was proposed in which it utilizes five filtering techniques such as

gain ratio, info gain, relief [32], chi-square and symmetric uncertainty [33], and it applies mutual information for combining the result and eliminate the redundancy. However, the model fails to compare the results with other existing ensemble-based models [17]. An approach that utilizes criteria derived from mutual information was proposed where the criterion is intended to evaluate the correlation between two features and it applies the firefly algorithm for effective results [34].

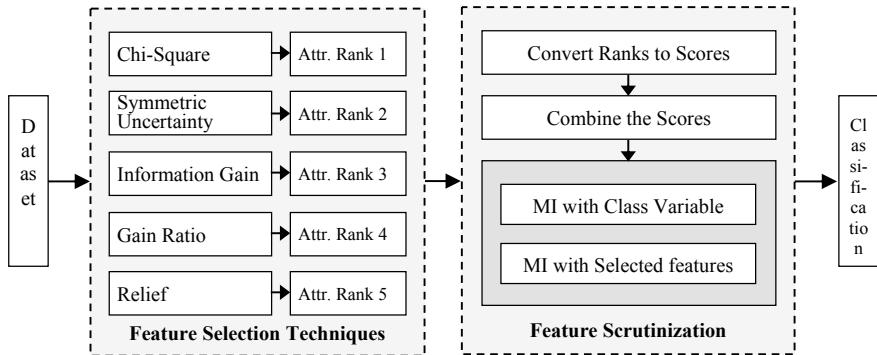
A model has been built that utilizes an ensemble of simple Bayesian classification. The model employs a hill-climbing-based refinement cycle as well as a variety of base classifiers to select optimal feature subsets for increasing accuracy [35] and was evaluated using various datasets. Another analysis was carried out using various feature selection techniques such as symmetric uncertainty, relief, support vector machine [36] and random forest [37] with individual model and ensemble-based analysis was carried out using instance perturbation with bagging model containing 40 bags [38]. A minimum redundancy maximum relevance (mRMR)-based feature selection was proposed to find optimal feature subsets to eliminate feature redundancy and improve relevancy concerning target class [39].

A feature selection model that employs optimized multiple rank scores for selecting features was proposed in which it makes use of various attribute evaluators including decision table, correlation, zeroR classifier, M5Rule classifier and relief. Here, the model fails to evaluate the results using a wide range of datasets and is evaluated using four credit score datasets [40]. Though there exist several works in the literature, the accuracy of the model needs to be improved. Thus, a model has been proposed that focuses on minimum redundancy and maximum relevancy with the ensemble model in selecting a significant optimal subset of features to improve the classification performance.

### 3 Proposed Feature Selection Model

The maximum relevancy and minimum redundancy based ensemble feature selection technique has been proposed, and the architecture of the proposed model is presented in Fig. 1. The model initially applies various filtering techniques using the greedy search method to identify the subset of features that seem to be significant. Each filtering technique results in a set of features based on its relevancy towards the study for which the ranks are assigned. The union of a subset of attributes selected by all the filtering techniques is the features with different ranks. The next process in the proposed model is the feature scrutinization in which for each feature, the assigned ranks are then converted to scores which are then summed that represents the feature significance. The features are arranged based on their attribute significance.

Then each feature is compared with the class variable as well as with the final selected optimal features by computing mutual information for evaluating the relevancy and redundancy. Initially, the dataset is given as an input for filtering techniques that are intended to arrange the features in a list based on the order of its significance. The proposed model uses various filtering techniques such as chi-square



**Fig. 1** Framework of the proposed feature selection model

(CS), symmetric uncertainty (SU), information gain (IG), gain ratio (GR) and relief (RF) that are utilized by Hoque et al. [17]. Each filtering technique utilizes a greedy search to rank the features based on the contribution of the feature in classification accuracy. Thus, for ‘ $m$ ’ features in the given input dataset represented as  $A_1, A_2, \dots, A_m$ , the output after applying the five filtering techniques produces five feature lists that are ranked based on their importance as in Eq. 1.

$$\{A_1, A_2, \dots, A_m\} \rightarrow \begin{pmatrix} R_{11} & \dots & R_{15} \\ \vdots & \ddots & \vdots \\ R_{m1} & \dots & R_{m5} \end{pmatrix} \quad (1)$$

The input dataset with ‘ $m$ ’ attributes for the filtering techniques results in  $m \times 5$  matrix having  $m$  rows and five columns representing  $m$  attributes and five filtering methods in which each row represents the various ranks  $\{R_{i1}, R_{i2}, R_{i3}, R_{i4}, R_{i5}\}$  of an attribute  $i$  provided by different filtering techniques.

The next phase focuses on scrutinizing the features by computing the single score of an attribute from various ranks and applying mutual information for evaluating the relevancy concerning the class attribute and redundancy concerning other selected attributes. Thus, the obtained ranks of the feature are converted to score. The normalized score for each attribute can be evaluated by inverting the obtained ranks from various filtering techniques which are then summed to identify the feature significance [40]. Then, the attributes are arranged in a list based on the accumulated scores and are represented as SF. This is the first level of selected attributes. It is given in Eq. (2).

$$\begin{pmatrix} S_{11} & \cdots & S_{15} \\ \vdots & \ddots & \vdots \\ S_{m1} & \cdots & S_{m5} \end{pmatrix} = \begin{pmatrix} \frac{1}{R_{11}} & \cdots & \frac{1}{R_{15}} \\ \vdots & \ddots & \vdots \\ \frac{1}{R_{m1}} & \cdots & \frac{1}{R_{m5}} \end{pmatrix} \rightarrow \begin{pmatrix} \sum_{i=1}^5 S_{1i} \\ \vdots \\ \sum_{i=1}^5 S_{mi} \end{pmatrix} = \begin{pmatrix} S_1 \\ \vdots \\ S_m \end{pmatrix} \quad (2)$$

Once the features are arranged based on the accumulated scores obtained from Eq. (2), then the next process is to identify the optimal feature subset by evaluating the mutual information concerning the class variable and the selected features from the optimal subset for maximizing the relevancy and minimizing the redundancy, respectively. Initially, the optimal feature subset OFS is empty. The mutual information is computed for the first feature from the list  $sf_1$  with that of the class variable and if the obtained value is greater than 0.70, then the feature is added in the final list of  $ofs_1$  as it represents that the attribute is highly relevant. Next, the mutual information is applied for the second feature in the list  $sf_2$  with that of the class attribute, and if the obtained value is greater than the threshold of 0.70, then the attribute is relevant. However, to minimize the redundancy, again the mutual information is applied between  $sf_2$  and  $ofs_1$  in which if the obtained value is less than 0.70 then it represents that the attribute is not redundant which can be added to the optimal subset. On the other hand, if the value is greater than 0.70, the feature exhibits higher redundancy and therefore the attribute in  $sf_2$  will not be added to the optimal list OFS. Thus, each feature from the initial list is evaluated with the class attribute and the selected features in the optimal subset to maximize relevancy and minimize redundancy. This process continues until all the features in the set SF are evaluated or the specified number of attributes are selected.

The algorithm pseudocode for the proposed maximum relevancy and minimum redundancy based ensemble feature selection technique is given in Fig. 2.

As the proposed model takes the merits of various feature selection techniques by combining the various filtering techniques, the final aggregated selected features will be highly effective. Also, the single score obtained from different models extremely signifies the feature significance. All the features or the features that scores above the pre-defined threshold are then evaluated by applying mutual information to increase the relevancy of the feature in classification with respect to the class variable and to decrease the redundancy of the feature with the other features which are already been selected. The proposed model introduces attribute significance that are calculated from the filtering techniques that help to reduce the computational complexity of the feature selection process.

## 4 Experimental Analysis

This section presents the various experimental analysis performed to evaluate the performance of the proposed maximum relevancy and minimum redundancy based

**Algorithm: MRMR based Ensemble Feature Selection Model**

Input: Training data with  $m$  attributes,  $n$  number of features to be selected,  $k$  set of filtering techniques, threshold  $h$   
 Output: Optimal subset of features  
 Procedure MRMR\_EFS()  
 Begin  
   For each filter technique  $i$  from 1 to  $k$   
     Apply feature selection technique  $i$   
     Identify the subset  $S_i$  with  $n$  features  
     Generate scores for the attribute ranks  $R_{ij}$   
   End for  
   For each selected attribute  $l$   
     Compute the sum of rank scores  
   End for  
   For each selected feature  $f_j$  from 1 to  $l$   
     Compute mutual information for  $f_j$  with class  
     If the value is greater than 0.70 then  
       For each element  $i$  from 1 to  $n$   
         If  $R_i$  is NULL then  
            $R_i = R_i \cup f_j$   
         Else  
           Compute mutual information for  $f_j$  with features in  $R_i$   
           If the value is lesser than 0.70 then  
              $R_i = R_i \cup f_j$   
           End if  
         End if  
       End For  
     End if  
   End for  
 Return selected optimal subset of features  $R_i$   
 End algorithm

**Fig. 2** Algorithm pseudocode for the proposed feature selection model

ensemble feature selection model with that of the existing model. To evaluate the performance, the system with the following characteristics is used. A system with 4 GB RAM has an Intel Core i3-4005 U CPU @ 1.70 GHz and 64-bit Windows 8 operating system. To analyse the various filtering-based feature selection techniques used in the proposed model, the Waikato Environment for Knowledge Analysis (WEKA) tool, a free open source software that consists of effective feature selection and classification techniques readily available, has been used. The proposed model has been implemented in Python which is then evaluated using a wide range of datasets with a count of 23 that comprises gene expression datasets, credit datasets and more. These datasets are publicly available at various sources including UCI and KAGGLE repository. The datasets used for the analysis of the proposed model along with the number of features and number of instances are listed in Table 1.

The proposed feature selection model has been analysed with various datasets in which the datasets are processed by different filtering techniques such as chi-square

**Table 1** Dataset description

S. No.	Dataset	#Features	#Instances
1	Accut1	7	120
2	Accute2	5	120
3	Abalone	7	4177
4	Australian credit approval	14	690
5	German credit	20	1000
6	Glass	9	214
7	HMEQ	12	5960
8	Wine	13	178
9	Iris	4	150
10	Japanese credit approval	15	690
11	Monk1	6	432
12	Monk2	6	432
13	Monk3	6	432
14	Tic-tac-toe	9	958
15	Zoo	16	101
16	Diabetes	9	768
17	Car	7	1728
18	Heart statlog	14	270
19	Breast cancerW	10	699
20	Colon cancer	2000	62
21	Lung cancer	325	73
22	Lymphoma	4026	45
23	SRBCT	2308	83

(CS), symmetric uncertainty (SU), information gain (IG), gain ratio (GR) and relief (RF). Each technique ends up with a list of attributes that are ordered based on the feature significance. The attributes are then assigned with five ranks from five different feature selection methods which are then converted to scores. The scores of each attribute are accumulated, and the attributes are arranged based on the highest scores. Each attribute in the list is evaluated with class attributes and other attributes being selected for the optimized feature subset using mutual information. Thus, the attributes having a value greater than 0.70 with class and lesser than 0.70 with optimal feature set is selected for the optimal subset. The results are analysed with various classification algorithms such as decision tree (DT), random forest (RF), k-nearest neighbour (KNN), support vector machine (SVM) and repeated incremental pruning to produce error reduction (RIPPER) for different datasets.

## 5 Result Analysis

The performance of the proposed maximum relevancy and minimum redundancy-based ensemble (MRMRE) feature selection model has been analysed to evaluate the performance. Initially, the analysis has been made by using various test options including ten-fold cross-validation (TO1), percentage split with 66% training set (TO2) and full training set (TO3) with different credit datasets such as Australian credit approval, German credit, HMEQ and Japanese credit approval and evaluated using repeated incremental pruning to produce error reduction (RIPPER) classifier algorithm. The results of the proposed model have been compared with the exiting optimized multiple rank score (OMRS) feature selection model and without applying any feature selection model (WFS). The various metrics such as percentage of correctly classified instances and incorrectly classified instances, true positive rate, false-positive rate, precision, recall and f-measure are used for the analysis [41]. The results obtained for the classification without applying feature selection and after applying the OMRS model and proposed MRMRE model are given in Table 2.

From the analysis, considering the average classification and false-positive rates, for the Australian credit dataset, WFS obtains 87.35% and 0.129, OMRS acquires 87.29% and 0.122 and proposed MRMRE obtains 89.59% and 0.111, respectively. For the German credit dataset, WFS obtains 73.36% and 0.496, OMRS acquires 73.93% and 0.496, and proposed MRMRE obtains 76.64% and 0.43, respectively. With the Japanese credit dataset, WFS obtains 86.58% and 0.135, OMRS acquires 86.07% and 0.142, and proposed MRMRE obtains 88.07% and 0.119 and that of for HMEQ dataset, WFS obtains 87.35% and 0.126, OMRS acquires 87.39% and 0.126, and proposed MRMRE obtains 87.79% and 0.122. Also, the average values of true positive rate, precision, recall and f-measure for the models WFS, OMRS and MRMRE are shown as a graph in Fig. 3.

An analysis has been made with various datasets listed in Table 1. The classification accuracies with different filtering techniques used in selecting features such as symmetric uncertainty (SU), gain ratio (GR), information gain (IG), relief (RF), chi-square (CS), ensemble-based feature selection (EFS) [17] and the proposed model (P) has been analysed using four different classifiers including decision tree (DT), random forest (RF), k-nearest neighbour (KNN) and support vector machine (SVM). The values for classification accuracy for all the seven feature selection models with the decision tree and the random forest are shown in Table 3.

The average classification accuracy with the feature selection models such as SU, GR, IG, RF, CS, EFS and P using DT are 78.8%, 77.9%, 78.3%, 78.3%, 80.3%, 77.8% and 80.6%, whereas for random forest, the classification accuracy for different feature selection models are 82.1%, 81.6%, 81.7%, 81.4%, 83.4%, 81.1% and 84%, respectively. Thus, the proposed feature selection model offers better average accuracy with a random forest classifier than the decision tree model. Similarly, the results of seven different feature selection models are also analysed using KNN and SVM classifiers. The values for classification accuracy for all the specified seven feature selection models with KNN and SVM are presented in Table 4.

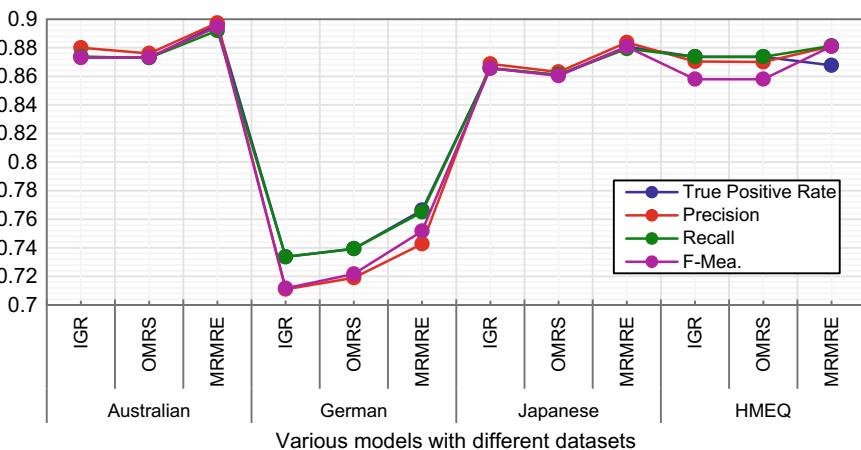
**Table 2** Performance evaluation with credit dataset

Test options	Models	Correct classi. %	Incorrect classi. %	TPR	FPR	Prec.	Rec.	FMea.
<i>Australian credit approval dataset</i>								
TO1	WFS	84.93	15.07	0.849	0.152	0.850	0.849	0.849
	OMRS	85.22	14.78	0.852	0.142	0.856	0.852	0.853
	MRMRE	86.81	13.19	0.868	0.131	0.872	0.868	0.870
TO2	WFS	85.96	14.04	0.860	0.131	0.873	0.860	0.859
	OMRS	88.09	11.91	0.881	0.117	0.882	0.881	0.881
	MRMRE	89.79	10.21	0.898	0.103	0.898	0.890	0.894
TO3	WFS	91.16	08.84	0.912	0.104	0.917	0.912	0.911
	OMRS	88.55	11.45	0.886	0.107	0.890	0.886	0.886
	MRMRE	92.17	07.83	0.922	0.099	0.922	0.918	0.920
<i>German credit dataset</i>								
TO1	WFS	71.00	29.00	0.710	0.501	0.686	0.710	0.689
	OMRS	71.60	28.40	0.716	0.499	0.692	0.716	0.694
	MRMRE	73.20	26.80	0.732	0.432	0.715	0.734	0.718
TO2	WFS	76.18	23.82	0.762	0.484	0.741	0.762	0.744
	OMRS	75.00	25.00	0.750	0.510	0.726	0.750	0.730
	MRMRE	78.82	21.18	0.788	0.469	0.756	0.782	0.769
TO3	WFS	72.90	27.10	0.729	0.503	0.706	0.729	0.702
	OMRS	75.20	24.80	0.752	0.413	0.739	0.752	0.741
	MRMRE	77.90	22.10	0.779	0.389	0.757	0.779	0.768
<i>Japanese credit approval dataset</i>								
TO1	WFS	85.80	14.20	0.858	0.142	0.859	0.858	0.858
	OMRS	85.80	14.20	0.858	0.135	0.863	0.858	0.858
	MRMRE	87.25	12.75	0.872	0.128	0.871	0.869	0.870
TO2	WFS	85.53	14.47	0.855	0.138	0.862	0.855	0.855
	OMRS	82.55	17.45	0.826	0.178	0.825	0.826	0.825
	MRMRE	86.38	13.62	0.864	0.136	0.871	0.861	0.866
TO3	WFS	88.41	11.59	0.884	0.125	0.885	0.884	0.884
	OMRS	89.86	10.14	0.899	0.114	0.901	0.899	0.898
	MRMRE	90.58	0.942	0.906	0.094	0.909	0.908	0.906
<i>HMEQ dataset</i>								
TO1	WFS	87.08	12.92	0.871	0.129	0.868	0.871	0.854
	OMRS	86.90	13.10	0.869	0.131	0.867	0.869	0.851
	MRMRE	87.38	12.61	0.873	0.126	0.875	0.878	0.876
TO2	WFS	85.88	14.12	0.859	0.141	0.848	0.859	0.843

(continued)

**Table 2** (continued)

Test options	Models	Correct classi. %	Incorrect classi. %	TPR	FPR	Prec.	Rec.	FMea.
TO3	OMRS	86.13	13.87	0.861	0.138	0.854	0.861	0.842
	MRMRE	86.48	13.52	0.865	0.135	0.871	0.867	0.869
	WFS	89.09	10.91	0.891	0.109	0.895	0.891	0.877
	OMRS	89.14	10.86	0.891	0.108	0.889	0.891	0.881
	MRMRE	89.53	10.47	0.865	0.105	0.897	0.899	0.898

**Fig. 3** Performance comparison on credit datasets

The average classification accuracy with the feature selection models such as SU, GR, IG, RF, CS, EFS and P using KNN is 78%, 77.8%, 77.4%, 77.6%, 77.6%, 75.9% and 80.8%, whereas for SVM, the classification accuracy for different feature selection models are 77.1%, 76.6%, 76.9%, 75.9%, 78.3%, 76.3% and 80.3%, respectively. Thus, the proposed feature selection model offers better average accuracy with KNN and SVM classifiers. The average classification obtained for the classifiers such as DT, RF, KNN and SVM with 23 datasets used for the analysis is depicted as a graph in Fig. 4. It can be seen that the proposed model offers better results similar to the statistical model such as chi-square based feature selection.

The overall average accuracy values of the different feature selection models used in the analysis with all the four classifiers such as DT, RF, KNN and SVM are shown as a graph in Fig. 5.

From the above analysis, it is also clear that the proposed model offers better classification accuracy for the datasets having a large number of features including Australian Credit Approval, German Credit, Japanese Credit Approval, Diabetes, Car, Heart statlog, colon cancer, lung cancer, lymphoma and SRBCT than other datasets having a minimum number of features. The average time complexity for

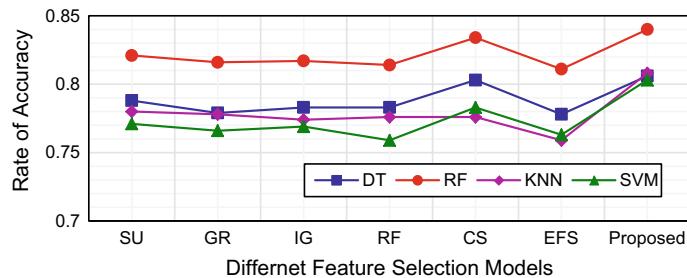
**Table 3** Performance evaluation with DT and RF

D	Decision tree (DT)							Random forest (RF)						
	SU	GR	IG	RF	CS	EFS	P	SU	GR	IG	RF	CS	EFS	P
1	0.94	0.94	0.94	0.92	0.94	0.94	0.94	0.94	0.94	0.94	0.92	0.94	0.94	0.92
2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
3	0.21	0.21	0.21	0.21	0.21	0.22	0.21	0.25	0.24	0.24	0.24	0.24	0.24	0.24
4	0.84	0.85	0.84	0.85	0.84	0.86	0.86	0.84	0.85	0.84	0.85	0.84	0.85	0.85
5	0.75	0.78	0.76	0.71	0.76	0.77	0.77	0.71	0.71	0.71	0.69	0.68	0.75	0.76
6	0.60	0.53	0.61	0.60	0.67	0.59	0.60	0.63	0.56	0.65	0.62	0.62	0.78	0.62
7	0.83	0.86	0.79	0.86	0.86	0.86	0.86	0.84	0.86	0.81	0.86	0.87	0.87	0.87
8	0.86	0.83	0.87	0.83	0.89	0.88	0.89	0.89	0.87	0.91	0.86	0.97	0.91	0.92
9	0.94	0.94	0.93	0.94	0.93	0.93	0.93	0.93	0.95	0.94	0.95	0.94	0.92	0.92
10	0.86	0.85	0.86	0.85	0.86	0.86	0.92	0.86	0.85	0.85	0.85	0.85	0.86	0.85
11	0.83	0.83	0.75	0.74	0.84	0.75	0.84	0.82	0.82	0.75	0.73	0.95	0.75	0.75
12	0.66	0.65	0.66	0.66	0.90	0.66	0.83	0.67	0.67	0.67	0.67	0.74	0.67	0.67
13	0.94	0.94	0.93	0.92	1.00	0.77	0.92	0.93	0.93	0.92	0.92	0.98	0.77	0.92
14	0.84	0.84	0.84	0.84	0.84	0.74	0.84	0.95	0.95	0.95	0.95	0.95	0.74	0.95
15	0.87	0.87	0.87	0.87	0.87	0.86	0.86	0.95	0.94	0.95	0.95	0.94	0.80	0.94
16	0.70	0.70	0.70	0.70	0.70	0.71	0.71	0.76	0.76	0.76	0.76	0.76	0.72	0.76
17	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96
18	0.80	0.76	0.79	0.78	0.75	0.77	0.82	0.80	0.78	0.79	0.77	0.82	0.77	0.86
19	0.94	0.93	0.94	0.94	0.93	0.94	0.93	0.96	0.96	0.96	0.96	0.96	0.97	0.96
20	0.78	0.78	0.78	0.78	0.78	0.78	0.78	0.83	0.83	0.84	0.84	0.84	0.85	0.86
21	0.55	0.53	0.55	0.56	0.54	0.56	0.56	0.77	0.76	0.76	0.76	0.73	0.77	0.82
22	0.66	0.60	0.69	0.71	0.64	0.71	0.72	0.71	0.70	0.71	0.73	0.70	0.83	0.86
23	0.78	0.76	0.76	0.79	0.79	0.78	0.79	0.89	0.88	0.89	0.88	0.91	0.91	0.93
Avg.	0.79	0.78	0.78	0.78	0.80	0.78	0.81	0.82	0.82	0.82	0.81	0.83	0.81	0.80

the proposed model is 0.345 s and is higher than other single-feature selection models and OMRS. Though the time taken by the proposed feature selection model is higher than other models, the time taken for classification is minimum with increased accuracy. These characteristics appreciate the proposed model in obtaining effective classification results.

**Table 4** Performance evaluation with KNN and SVM

D	KNN							SVM						
	SU	GR	IG	RF	CS	EFS	P	SU	GR	IG	RF	CS	EFS	P
1	0.91	0.90	0.93	0.90	0.92	0.92	0.90	1.00	1.00	1.00	1.00	1.00	1.00	1.00
2	0.95	0.96	0.96	0.93	0.96	0.96	0.96	1.00	1.00	1.00	1.00	1.00	1.00	1.00
3	0.20	0.20	0.20	0.20	0.20	0.18	0.32	0.14	0.14	0.14	0.14	0.14	0.15	0.22
4	0.84	0.85	0.84	0.85	0.84	0.85	0.91	0.86	0.85	0.84	0.84	0.85	0.85	0.88
5	0.86	0.87	0.87	0.88	0.86	0.86	0.88	0.86	0.86	0.86	0.89	0.87	0.87	0.89
6	0.60	0.55	0.65	0.60	0.72	0.53	0.65	0.45	0.40	0.44	0.44	0.36	0.23	0.42
7	0.81	0.84	0.79	0.84	0.85	0.85	0.85	0.84	0.85	0.81	0.84	0.85	0.86	0.84
8	0.76	0.75	0.71	0.71	0.74	0.86	0.82	0.88	0.86	0.90	0.86	0.97	0.89	0.95
9	0.95	0.95	0.95	0.95	0.95	0.92	0.90	0.95	0.95	0.95	0.95	0.95	0.72	0.95
10	0.86	0.85	0.85	0.85	0.85	0.86	0.92	0.85	0.85	0.85	0.85	0.85	0.86	0.92
11	0.79	0.78	0.68	0.75	0.78	0.67	0.75	0.66	0.66	0.66	0.50	0.66	0.67	0.67
12	0.58	0.60	0.59	0.60	0.60	0.56	0.60	0.51	0.53	0.51	0.48	0.67	0.67	0.67
13	0.93	0.93	0.91	0.90	0.87	0.68	0.87	0.79	0.80	0.75	0.75	0.80	0.72	0.75
14	0.74	0.75	0.75	0.75	0.75	0.65	0.75	0.65	0.63	0.64	0.63	0.63	0.64	0.77
15	0.96	0.96	0.97	0.97	0.96	0.79	0.96	0.94	0.94	0.94	0.94	0.93	0.95	0.95
16	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.76	0.76	0.76	0.76	0.76	0.74	0.76
17	0.78	0.78	0.78	0.78	0.78	0.82	0.78	0.65	0.65	0.65	0.65	0.65	0.71	0.70
18	0.79	0.74	0.76	0.76	0.57	0.68	0.82	0.80	0.80	0.79	0.79	0.82	0.78	0.80
19	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.97	0.97
20	0.80	0.79	0.76	0.77	0.77	0.81	0.80	0.81	0.79	0.82	0.82	0.83	0.85	0.86
21	0.72	0.73	0.72	0.72	0.70	0.74	0.82	0.79	0.75	0.77	0.77	0.75	0.70	0.77
22	0.62	0.63	0.63	0.63	0.63	0.75	0.79	0.70	0.72	0.76	0.72	0.77	0.80	0.82
23	0.88	0.87	0.89	0.89	0.91	0.89	0.92	0.90	0.87	0.88	0.88	0.93	0.92	0.92
Avg.	0.78	0.78	0.77	0.77	0.77	0.76	0.81	0.77	0.77	0.77	0.76	0.78	0.76	0.80

**Fig. 4** Performance comparison with different classifiers



**Fig. 5** Performance comparison of average accuracy

## 6 Conclusion

This paper presents the maximum relevancy and minimum redundancy based ensemble feature selection for effective classification acquiring maximized accuracy. The proposed model utilizes various filtering-based feature section techniques in which each method returns the features in order concerning evaluation results. Based on the ranks of the features, the scores are assigned and finally, mutual information is computed with the target variable as well as other selected attributes to evaluate the relevancy and redundancy. The experimental analysis has been performed for 23 datasets that are available publicly. The proposed model outperforms other existing individual feature selection techniques and ensemble-based methods with the average accuracy of 80.6%, 84%, 80.8% and 80.3% for different classifiers such as decision tree, random forest, k-nearest neighbour and support vector machines, respectively. The model results with an average accuracy of 81.5% which is more effective than various other models under comparison specifically for the datasets having a large number of feature sets. The time complexity is also satisfactory when compared with the accuracy of the results. The future work focuses on improving the accuracy of the classification to 100%, verifying the model with real-time datasets as well as exploring other feature selection techniques in ensemble models.

## References

- Charitopoulos, A., Rangoussi, M., Koulouriotis, D.: On the use of soft computing methods in educational data mining and learning analytics research: a review of years 2010–2018. *Int. J. Artif. Intell. Educ.* **30**(3), 371–430 (2020)
- Verma, D., Mishra, N.: Analysis and prediction of breast cancer and diabetes disease datasets using data mining classification techniques. In: International Conference on Intelligent Sustainable Systems, pp. 533–538. IEEE (2017)
- Han, J., Pei, J., Kamber, M.: Data Mining: Concepts and Techniques. Elsevier (2011)
- Mafarja, M.M., Mirjalili, S.: Hybrid whale optimization algorithm with simulated annealing for feature selection. *Neurocomputing* **260**, 302–312 (2017)

5. Nousi, C., Belogianni, P., Koukaras, P., Tjortjis, C.: Mining data to deal with epidemics: case studies to demonstrate real world AI applications. In: *Handbook of Artificial Intelligence in Healthcare*, pp. 287–312. Springer, Cham (2022)
6. Zhang, C., Liu, C., Zhang, X., Almanidis, G.: An up-to-date comparison of state-of-the-art classification algorithms. *Expert Syst. Appl.* **82**, 128–150 (2017)
7. Keleş, M.K.: Breast cancer prediction and detection using data mining classification algorithms: a comparative study. *TehnickiVjesnik* **26**(1), 149–155 (2019)
8. Gahlaut, A., Singh, P.K.: Prediction analysis of risky credit using data mining classification models. In: *8th International Conference on Computing, Communication and Networking Technologies*, pp. 1–7. IEEE (2017)
9. Moradi, S., MokhatabRafie, F.: A dynamic credit risk assessment model with data mining techniques: evidence from Iranian banks. *Financ. Innov.* **5**(1), 1–27 (2019)
10. Rajaleximi, P., Ahmed, M., Alenezi, A.: Classification of imbalanced class distribution using random forest with multiple weight based majority voting for credit scoring. *Int. J. Recent Technol. Eng.* **7**(6S5), 517–526 (2019)
11. Jalota, C., Agrawal, R.: Analysis of educational data mining using classification. In: *International Conference on Machine Learning, Big Data, Cloud and Parallel Computing*, pp. 243–247. IEEE (2019)
12. Oramas, S., Nieto, O., Barbieri, F., Serra, X.: Multi-label music genre classification from audio, text, and images using deep features. In: *18th International Society for Music Information Retrieval Conference*. Suzhou, China (2017)
13. Garg, S., Singh, A., Batra, S., Kumar, N., Obaidat, M.S.: EnClass: ensemble-based classification model for network anomaly detection in massive datasets. In: *Global Communications Conference*, pp. 1–7. IEEE (2017)
14. Sathya Bama, S., Ahmed, M.I., Saravanan, A.: Relevance re-ranking through proximity based term frequency model. In: *International Conference on ICT Innovations*, pp. 219–229. Springer, Cham (2016)
15. Saravanan, A., Sathya Bama, S.: Extraction of core web content from web pages using noise elimination. *J. Eng. Sci. Technol. Rev.* **13**(4), 173–187 (2020)
16. Sathya Bama, S., Saravanan, A.: Efficient classification using average weighted pattern score with attribute rank based feature selection. *Int. J. Intell. Syst. Appl.* **10**(7), 29 (2019)
17. Hoque, N., Singh, M., Bhattacharyya, D.K.: EFS-MI: an ensemble feature selection method for classification. *Complex Intell. Syst.* **4**(2), 105–118 (2018)
18. Rostami, M., Berahmand, K., Nasiri, E., Forouzande, S.: Review of swarm intelligence-based feature selection methods. *Eng. Appl. Artif. Intell.* **100**, 104210 (2021)
19. Mafarja, M., Mirjalili, S.: Whale optimization approaches for wrapper feature selection. *Appl. Soft Comput.* **62**, 441–453 (2018)
20. Got, A., Moussaoui, A., Zouache, D.: Hybrid filter-wrapper feature selection using Whale optimization algorithm: a multi-objective approach. *Expert Syst. Appl.* **115**312 (2021)
21. Sheikhpour, R., Saram, M.A., Gharaghani, S., Chahooki, M.A.Z.: A survey on semi-supervised feature selection methods. *Pattern Recogn.* **64**, 141–158 (2017)
22. Bommert, A., Sun, X., Bischl, B., Rahnenführer, J., Lang, M.: Benchmark for filter methods for feature selection in high-dimensional classification data. *Comput. Stat. Data Anal.* **143**, 106839 (2020)
23. Palimkar P. et al.: Machine learning technique to prognosis diabetes disease: random forest classifier approach. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems*, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_19](https://doi.org/10.1007/978-981-16-2164-2_19)
24. Bolón-Canedo, V., Alonso-Betanzos, A.: Ensembles for feature selection: a review and future trends. *Inf. Fusion* **52**, 1–12 (2019)
25. Das, A.K., Das, S., Ghosh, A.: Ensemble feature selection using bi-objective genetic algorithm. *Knowl.-Based Syst.* **123**, 116–127 (2017)
26. Seijo-Pardo, B., Porto-Díaz, I., Bolón-Canedo, V., Alonso-Betanzos, A.: Ensemble feature selection: homogeneous and heterogeneous approaches. *Knowl.-Based Syst.* **118**, 124–139 (2017)

27. Kumar A. et al.: Analysis of classifier algorithms to detect anti-money laundering. In: Bansal, J.C., Paprzycki, M., Bianchini, M., Das, S. (eds.) Computationally Intelligent Systems and their Applications. Studies in Computational Intelligence, vol. 950. Springer, Singapore (2021). [https://doi.org/10.1007/978-981-16-0407-2\\_11](https://doi.org/10.1007/978-981-16-0407-2_11)
28. Ferro, C.A., Jupp, T.E., Lambert, F.H., Huntingford, C., Cox, P.M.: Model complexity versus ensemble size: allocating resources for climate prediction. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **370**(1962), 1087–1099 (2012)
29. Torres-Sospedra, J., Fernandez-Redondo, M., Hernandez-Espinosa, C.: A research on combination methods for ensembles of multilayer feedforward. In: Proceedings of International Joint Conference on Neural Networks, vol. 2, pp. 1125–1130. IEEE (2005)
30. Seijo-Pardo, B., Bolón-Canedo, V., Alonso-Betanzos, A.: On developing an automatic threshold applied to feature selection ensembles. *Inf. Fusion* **45**, 227–245 (2019)
31. Pujari, P., Gupta, J.B.: Improving classification accuracy by using feature selection and ensemble model. *Int. J. Soft Comput. Eng.* **2**(2), 380–386 (2012)
32. Mridha, K. et al.: Phishing URL classification analysis using ANN algorithm. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–7 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573797>
33. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical recipes in C. Probab. Eng. Informational Sci. **2**(1) (1988)
34. Zhang, L., Shan, L., Wang, J.: Optimal feature selection using distance-based discrete firefly algorithm with mutual information criterion. *Neural Comput. Appl.* **28**(9), 2795–2808 (2017)
35. Tsymbal, A., Puuronen, S., Patterson, D.W.: Ensemble feature selection with the simple Bayesian classification. *Inf. Fusion* **4**(2), 87–100 (2003)
36. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. *Mach. Learn.* **46**(1), 389–422 (2002)
37. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
38. Saeys, Y., Abeel, T., Van de Peer, Y.: Robust feature selection using ensemble feature selection techniques. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 313–325. Springer, Berlin, Heidelberg (2008)
39. De Jay, N., Papillon-Cavanagh, S., Olsen, C., El-Hachem, N., Bontempi, G., Haibe-Kains, B.: mRMRE: an R package for parallelized mRMR ensemble feature selection. *Bioinformatics* **29**(18), 2365–2368 (2013)
40. Rajaleximi, P., Ahmed, M., Alenezi, A.: Feature selection using optimized multiple rank score model for credit scoring. *Int. J. Intell. Eng. Syst.* **12**(04), 74–84 (2019)
41. Bama, S.S., Ahmed, M.I., Saravanan, A.: A survey on performance evaluation measures for information retrieval system. *Int. Res. J. Eng. Technol.* **2**(2), 1015–1020 (2015)

# Comparative Study of Pre-trained Models on Cotton Plant Disease Detection Using Transfer Learning



Satyam Kumar, Dayima Musharaf, and Anil Kumar Sagar

**Abstract** The world of computer intelligence also known as artificial intelligence evolved to a great extent. Almost every sector is taking its help for whether to make production cost-efficient or to meet the rise of high demand and supply chain. An increase in population is directly proportional to an increase in the demand chain. Effective and efficient food production is required to feed this much of the population. Two factors of poor plant yield are bad climatic conditions and plant disease. Early detection of plant disease can lead to efficient food production. Depending on labs and human expertise for disease detection makes production cost in-efficient. The availability of smartphones globally has a very important role in plant disease detection. Through phone cameras and automated images can make plant disease detection cost-efficient. In this paper, we compared ten pre-trained models' accuracy by implementing transfer learning on our dataset of cotton plants.

**Keywords** Cotton disease detection · Transfer learning · KERAS · ResNet50 · VGG16 · MobileNet · DenseNet

## 1 Introduction

In India, 58% of the population has agriculture as the primary source of livelihood. According to the report of 2020–21, the share of agriculture in GDP has reached 20%. In the COVID time, every sector showed negative growth but agriculture was the only sector having positive growth of 3.4%. In the time span of 2017–20, India has witnessed 1 billion USD as funding in Agrotech. It is predicted that from

---

S. Kumar (✉) · D. Musharaf

School of Engineering and Technology, Sharda University, Greater Noida, India

e-mail: [2020540155.satyam@ug.sharda.ac.in](mailto:2020540155.satyam@ug.sharda.ac.in)

D. Musharaf

e-mail: [2020442645.dayima@ug.sharda.ac.in](mailto:2020442645.dayima@ug.sharda.ac.in)

A. K. Sagar

Department of Computer Science, Sharda University, Greater Noida, India

e-mail: [anil.sagar@sharda.ac.in](mailto:anil.sagar@sharda.ac.in)

2020 to 25, the Aggrotech industry of India will witness an investment of 30–35 billion USD. The Government of India is propagating the message and scheme of digital agriculture mission for agricultural projects with wide applications of robotics, artificial intelligence, machine learning, remote sensing, blockchain, etc. Drones are widely used in agriculture for keeping an eye on crops and their growth. Technology is on the boom, and every sector is emerging with an application of technology like Finance + technology = Fintech, Information + technology = Infotech. So, it is the perfect time to merge and promote agriculture with technology (Agriculture + technology = Aggrotech). Aggrotech has not seen much boom, so developing technology based on Aggrotech can be a game-changer for farmers as well as the industry of agriculture and technology. Farmers are the ones who are continuously facing problems whether in the form of bad climatic conditions or plant disease. It is very important to identify the disease at the right time so that it doesn't fester and spread to other plants on the farm. Although demand for food is rising to feed the growing population, people choosing agriculture as an option for a career is decreasing overtime. Technology should be introduced to agricultural fields for cost-efficient production. If we manage to identify the disease at an early stage, then the farmers can manage better the yield of crops. The disease can vary from plant to plant itself. It is hard to recognize each disease with the naked eye, and taking each plant sample for testing to the lab is a time- and money-consuming process. That's why AI is playing an important role, here in reducing time and money consumption. Computer vision with the help of automated systems is helping farmers to identify the disease. Models are trained differently on different algorithms to predict disease. Whenever we talk of prediction, the first thing that clicks to mind is accuracy. A more accurate prediction means a more accurate model. We can't expect 100% accuracy but we can't accept 50–70% accuracy also. We are performing comparisons between different pre-trained models using transfer learning on cotton plant disease.

## 2 Literature Review

Initially, machine learning was used for the classification of images with an accuracy of around (80–86%). After machine learning, deep learning came into the picture with a promising result. With the help of convolutional neural networks (CNNs), image classification showed an accuracy of 92–93%. CNN proved to be powerful in the area of plant disease detection. Kathiresan et al. [1] present straining based on MobileNet, ResNet50, ResNet101, InceptionV3, Xception, and DenseNet169 custom models using evaluation metrics. The proposed model showed 98.79% accuracy. Vimal et al. [2] proposed a methodology for using the AlexNet CNN model for input dimensions of  $227 \times 227 \times 3$ . The AlexNet model was used for feature extract or rather than for the task of classification. It is important to keep an eye on model over fitting as it can lead to great accuracy but only on training results. Srivastava et al. [3] proposed the methodology for preventing the model over fitting during the training. Hassan et al. [4] compared the transfer learning approach and deep learning approach on

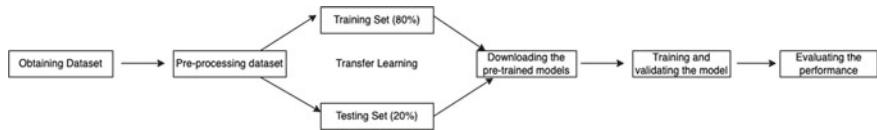
the identification of plant leaf disease. Rubini et al. [5] and Akshai and Anitha [6] proposed the methodology for transfer learning on the pre-trained deep learning models for plant disease detection. Khamparia et al. [7] proposed a model to predict seasonal crops disease and classify it through a deep convolutional encoder network. Hamuda et al. [8] proposed a model for improved image processing-based crop detection using Kalman filtering and the Hungarian algorithm. Ioffe and Szegedy [9] proposed a model which accelerates the deep network training by reducing internal covariate shift.

Gupta et al. [10] proposed a model for feature optimization using MWOA. Caglayan et al. [11] proposed a method for plant recognition approach using shape and color features in leaf images. Mridha et al. [12] proposed a model for the classification of pomegranate disease based on back propagation neural network. Rastogi et al. [13] proposed a model for leaf disease detection and grading computer vision technology. Savita et al. [14] proposed a model for detection and classification of plant leaf disease using image processing techniques. Pokrajac et al. [15] proposed a model on image processing in precision agriculture. Dubey et al. [16] proposed a model for infected fruit part detection using K-means clustering segmentation technique. Patil and Kumar [17] proposed a model for which advances in image processing for detection of plant diseases. Satti et al. [18] proposed a model of an automatic leaf recognition system for plant identification using machine vision technology. Phadikar and Sil [19] proposed a model for rice disease identification using the pattern recognition technique.

Meunkaewjinda et al. [20] proposed a model on grape leaf disease detection from a color imagery system using a hybrid intelligent system. Patil and Zambre [21] proposed a model on classification of cotton leaf spot disease using support vector machine. Tijare and Badnerkar [22] proposed a model on an image recognition-based crop disease identification system.

### 3 Proposed Methodology

Implementing the human neural network can do wonders in machine learning. We implement a neural network that consists of a dense hidden layer which is also known as deep learning. Deep learning has a wide range of dense neural networks for predictive analysis. Deep learning consists of hierarchical stacked algorithms of increasing abstraction and complexity. Deep learning has many perks over machine learning like accuracy and feature extraction. In deep learning, the feature set is built by the program itself without any requirement of supervision. To train models on the deep learning algorithm, we require a large dataset and high resources in the form of processing power. If the dataset is not very large, then there is always a chance of over fitting. So, on the small dataset deep learning does not provide great accuracy.



**Fig. 1** Flowchart of the working mechanism

### 3.1 Working Mechanism

Firstly, we obtain the dataset either manually, or sites like [www.kaggle.com](http://www.kaggle.com), etc. To eliminate unwanted details in the data, we do data preprocessing. Then, the dataset is split into two sets of which one is the training set (80%) and the other is the validation set (20%). Then, we download the pre-trained models consisting of lots of parameters. Then, we operate that model on our dataset. After training and validating the model on our dataset, we evaluate the models on the basis of their performance.

### 3.2 Dataset

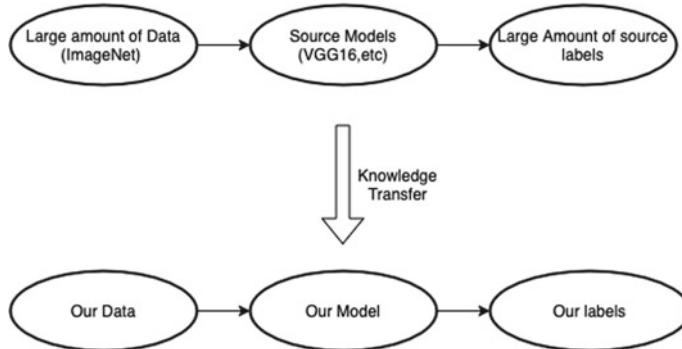
Dataset was collected manually by phone camera. The dataset consists of 2327 images in the form of  $256 \times 256$  pixels. We used colored RGB images. Diseased plant as well as fresh plant images were taken in ample amount. After taking images, the diseased plant images were stored in diseased folder and the fresh plant was stored in fresh plant folder.

### 3.3 Preprocessing Dataset

To remove redundancy or noise with a faster execution time, we preprocess the dataset as it plays a major role in training. Preprocessing the dataset also increases the accuracy of the model. Preprocessing of images is done with the help of data augmentation. For data augmentation, we used a library of deep learning of KERAS in Python. Rotating, filling, flipping, scaling, cropping, and zooming were the techniques applied to the data.

### 3.4 Transfer Learning

The technique of using a model which was created to perform a particular task or function as in another model for performing another task or subsequent function is



**Fig. 2** Transfer learning model

known as transfer learning. These models are already built by training models on millions of parameters as well as training on a large number of datasets.

Organizations like KERAS have built many pre-trained models for research as well as user-friendly applications. We have compared a total of ten pre-trained models which are named as follow: ResNet50, ResNet101, InceptionV3, MobileNet, DenseNet121, MobileNetV2, ResNet152V2, ResNet101V2, VGG16, and VGG19.

These ten pre-trained models were fine-tuned to our datasets on the last layer of its network. When the validation accuracy does not improve with time, we use early stopping to stop the training. We use label encoding to get a categorical output. Then, we conjoined the base model by a global average pooling layer whose function is to reduce the data and the model preparation for the final layer classification. For stability and quicker execution, the batch normalization layer is added. A dense layer having a weight value equal to nine (there were nine types of disease in cotton plant disease dataset which model need to classify) with SoftMax (probability) as an activation function is used for the process of withdrawing and classifying the output in the final layer. For updating the weights, Adam optimizer having a learning rate of 0.001 is used. We used a categorical function for loss to determine the performance of the model. The models have been trained again on the Epoch of 30 and batch size of 100. Finally, when we complete the training on the training dataset with avoiding over fitting with the help of dropout, we determine the result on a testing dataset in the categories of training loss, training accuracy, F1-score, precision, validation loss, validation accuracy, and recall for the performance of the model.

## 4 Experiments and Results

ResNet152V2 performed best among all the models we trained using transfer learning. After ResNet152V2, VGG19 performed better as mentioned in Table 1. Validation and training accuracy of VGG16 and MobileNet are the same as shown

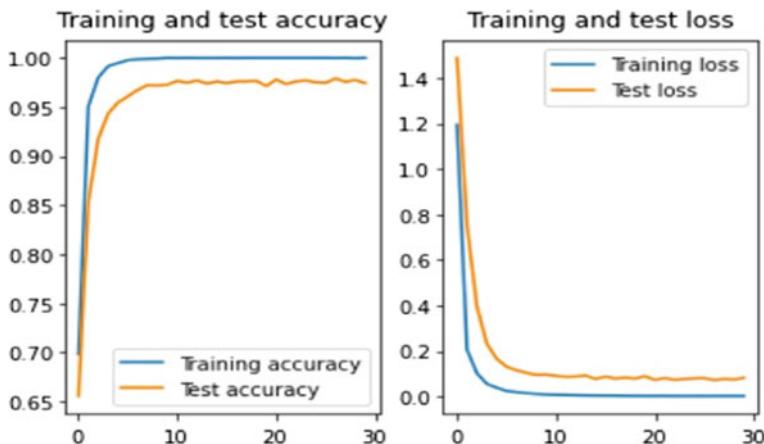
**Table 1** Training and validation accuracy comparison on test data

Pre-trained models	Training accuracy	Validation accuracy
ResNet50	0.9840	0.9303
ResNet101	1.0000	0.9567
ResNet101V2	1.0000	0.9793
ResNet152V2	1.0000	0.9895
InceptionV3	1.0000	0.9436
MobileNet	1.0000	0.9783
MobileNetV2	1.0000	0.9742
DenseNet121	1.0000	0.9671
VGG16	1.0000	0.9783
VGG19	1.0000	0.9895

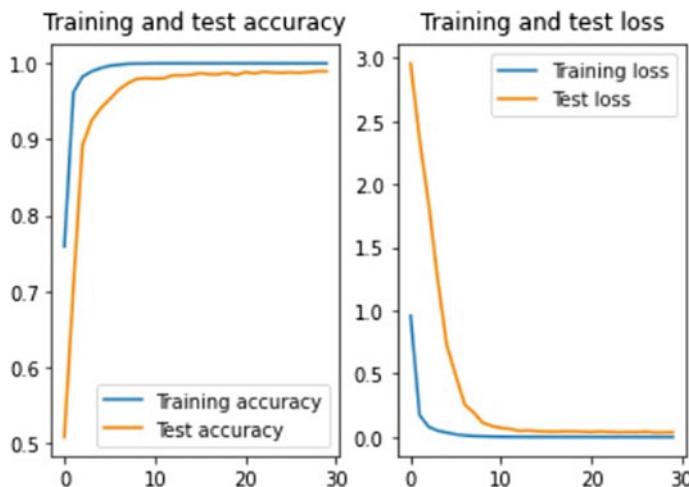
**Table 2** Pre-trained models comparison on test data

Pre-trained models	Accuracy	Precision	Recall	F1-score
ResNet50	0.93	0.88	0.87	0.87
ResNet101	0.95	0.84	0.86	0.85
ResNet101V2	0.97	0.96	0.94	0.95
ResNet152V2	0.99	0.98	0.98	0.98
InceptionV3	0.96	0.95	0.93	0.94
Mobile Net	0.98	0.97	0.95	0.96
Mobile NetV2	0.95	0.84	0.86	0.85
DenseNet121	0.94	0.95	0.93	0.94
VGG16	0.98	0.94	0.94	0.94
VGG19	0.99	0.95	0.95	0.95

in Table 2, but the VGG16 F1-score is less than the MobileNet F1-score. Because of this, MobileNet achieved better results. F1-score is directly proportional to the betterment of the model. Higher the F1-score, better the model is. We can conclude that ResNet152V2 and VGG19 succeed with a decent fit. We predicted a few images which don't belong to training as well as testing dataset and used our pre-trained models ResNet152V2 and VGG16 well. From Figs. 3 and 4, we can conclude that our model is trained perfectly as within less epoch, our model reached a well-trained and test accuracy and training and test loss decreased gradually.



**Fig. 3** Accuracy(y) versus epochs(x) and loss(y) versus epochs(x) of ResNet152V2



**Fig. 4** Accuracy(y) versus epochs(x) and loss(y) versus epochs(x) of VGG19

## 5 Conclusion

By all this, we can finally conclude that we have come to an era where artificial intelligence is the need of the hour. Machine learning, deep learning, as well as transfer learning are part of artificial intelligence. With the machine learning classical algorithm, the accuracy does not go over 86%. With the convolutional neural network (CNN) of deep learning, the accuracy does not go over 93% in plant disease detection. But with the help of transfer learning, we managed to achieve 99% of accuracy on

ResNet152V2. This study can further extend with the training of models on more categories of disease. It can also be deployed in applications for farmer's use.

## 6 Future Scope

This model was trained for cotton plant disease only. Further challenges which we have to tackle is making this methodology scalable with maintaining a great accuracy. Training models for different plant disease which will require wide variety of dataset and maintaining accuracy on these datasets is going to be challenging. There is no doubt that if we make this methodology scalable, it can make impact to farmers in farming sectors as well as people who are working toward kitchen gardening. Training models and deploying it on web as well as mobile application can do wonders in this sector.

## References

1. Kathiresan, G., Anirudh, M., Nagharjun, M., Karthik, R.: Disease detection in rice leaves using transfer learning technique. In: International Conference on Innovative Technology for sustainable development 2021 (ICITSD 2021) 27–29 Jan 2021. Chennai, India (2021)
2. Srivastava, V.K., Pradhan, M.K., Minz, S., Thakur, M.P.: Rice plant disease classification using transfer learning of deep convolutional neural networks. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XLII-3/W6, 2019 ISPRS-GEOGLAM-ISRS Joint Int. Workshop on "Earth Observations for Agricultural Monitoring", 18–20 Feb 2019. New Delhi, India (2019)
3. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from over fitting. *J. Mach. Learn. Res.* **15**(1), 1929–1958 (2014)
4. Hassan, S.M., Maji, A.K., Jasiński, M., Leonowicz, Z., Jasińska, E.: Identification of plant-leaf diseases using CNN and transfer-learning approach. *Electronics* **10**, 1388 (2021). <https://doi.org/10.3390/electronics10121388>
5. Rubini, P.E., Dr. Kavitha P.: Deep Learning model for early prediction of plant disease. In: Proceedings of the Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV 2021) (2021)
6. Akshai, K.P., Anitha, J.: Plant disease classification using deep learning. In: 2021 3rd International Conference on Signal Processing and Communication (ICPSC), 13–14 May 2021. Coimbatore (2021)
7. Khamparia, A., Saini, G., Gupta, D., Khanna, A., Tiwari, S., de Albuquerque, V.H.C.: Seasonal crops disease prediction and classification using deep convolutional encoder network. *Circ. Syst. Sig. Process.* Springer Publication (2019)
8. Hamuda, E., Mc Ginley, B., Glavin, M., Jones, E.: Improved image processing-based crop detection using Kalman filtering and the Hungarian algorithm. *Comput. Electron. Agric.* **148**, 37–44 (2018)
9. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. Wiley (2015)
10. Jain, R., Gupta, D., Khanna, A.: Usability feature optimization using MWOA. In: International Conference on Innovative Computing and Communication (ICICC), vol. 2 (2018)

11. Caglayan, A., Guclu, O., Can, A.B.: A plant recognition approach using shape and color features in leaf images. In: International Conference on Image Analysis and Processing, pp. 161–170. Springer, Berlin (2013)
12. Mridha, K. et al.: Plant disease detection using web application by neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 130–136 (2021). <https://doi.org/10.1109/ICCCA52192.2021.9666354>
13. Rastogi, A., Arora, R., Sharma, S.: Leaf disease detection and grading using computer vision technology and fuzzy logic. In: 2nd International Conference on Signal Processing and Integrated Networks (SPIN) (2015)
14. Ghaiwat, S.N., Arora, P.: Detection and classification of plant leaf diseases using image processing techniques: a review. *IJRAET* **2**(3) (2014)
15. Pokrajac, D., Lazarevic, A., Vucetic, S., Fiez, T., Obradovic, Z.: Image processing in precision agriculture. In: 4th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services, pp. 616–619 (2009)
16. Dubey, S.R., Dixit, P., Singh, N., Gupta, J.P.: Infected fruit part detection using k-means clustering segmentation technique. *IJIMAI* **2**(2), 65–72 (2013)
17. Patil, J.K., Kumar, R.: Advances in image processing for detection of plant diseases. *J. Adv. Bioinform. Appl. Res.* **2**(2), 135–141 (2011). ISSN: 0976-2604
18. Satti, V., Satya, A., Sharma, S.: An automatic leaf recognition system for plant identification using machine vision technology. *Int. J. Eng. Sci. Technol. (IEST)* ISSN:0975-5462, **5**(4), 874–879 (2013)
19. Phadikar, S., Sil, J.: Rice disease identification using pattern recognition techniques. In: Proceedings of 11th International Conference On Computer And Information Technology, pp. 25–27 (2008)
20. Meunkaewjinda, A., Kumsawat, P., Attakitmongcol, K., Srikaew, A.: Grape leaf disease detection from color imagery system using hybrid intelligent system. In: Proceedings of ECTICON, pp. 513–516. IEEE (2008)
21. Patil, S.P., Zambre, R.S.: Classification of cotton leaf spot disease using support vector machine. *IJERA* **4**(5), 92–97 (2014)
22. Tijare, N.S., Prof. Badnerkar, S.S.: Image recognition based crop disease identification system: a survey. *IJCSMC* **3**(4), 868–873 (2014)

# A Comparative Study of Time Series, Machine Learning, and Ensemble Models for Crude Oil Price Prediction



Ankit Prakash and Sunil Kumar Singh

**Abstract** Fuel costs are falling worldwide during the pandemic, but not in India. In recent years, the petrol's price has risen dramatically in all across the states of India, and reaching in three digits. The government is also targeting it as their best revenue source. It indicates that the fuel costs causes more inflation and common people have to suffer. These increased prices have robbed Indian citizens of living a healthy life. This paper analyzes Delhi's petrol price and forecasting by examining daily petrol prices' forecasting performance from June 16, 2017, to December 15, 2021. Various time series, machine learning, deep learning, and ensemble learning models are used to find the best model for forecasting the results. We have analyzed the performance of the models with the help of the performance metrics such as MSE, RMSE, MAPE, and NRMSE. The outcomes of the models indicate that time series-based models are pretty compelling.

**Keywords** Machine learning · Deep learning · Statistical models · Crude oil · Time series

## 1 Introduction

Prices of petrol in India are increasing day by day. This is happening since December 2020. The current price of petrol is around Rs. 95 per liter<sup>1</sup>. So, if we talk about what is wrong, we have to analyze it. During the COVID pandemic, when it hit India in March 2020, so many restrictions were imposed in terms of traveling and, of course, traveling domestically and internationally. So, there were so many impositions of different practices for social distancing that people were nowhere to go, and due to this, the price of crude oil crashed, resulting in no further increment on petrol or

---

A. Prakash · S. K. Singh (✉)

Department of Computer Science and Information Technology, Mahatma Gandhi Central University, Motihari, Bihar, India

e-mail: [sksingh@mgcub.ac.in](mailto:sksingh@mgcub.ac.in); [sunilsingh.jnu@gmail.com](mailto:sunilsingh.jnu@gmail.com)

<sup>1</sup> <https://economictimes.indiatimes.com/wealth/fuelprices/fuel-petrol,citystate-delhi.cms>.

oil prices for 80+ days.<sup>2</sup> On the other hand, when travel restriction was imposed, demand for petrol was less. So, they were hit with a double whammy first of all where to go, and because of where to go, demand fall, and they couldn't tap the benefit of the lower oil prices.

When we talk about oil marketing companies since November 2020, they couldn't tap the benefit. When the economy tried to get back on track after the lockdown was over, then the recovery of the economy started. The oil prices companies increased the petrol and diesel prices from November 20, 2020. Since then, there has been a higher trend, and if we talk about 2020 only in 18 days the petrol price has gone up by Rs. 2.65 per liter. Today you can see the price reached Rs. 100+ in many states. Consumers have to pay more because of the decontrolled price of fuel. The consumer will be at the negative trend and be at the opposing end. In this sense, oil was initially regulated, which came under the government's regulation. But in the year 2010, the government decided that they will not regulate the prices of petrol and oil marketing company would do it. In 2014, deregulation was also imposed on diesel. If we talk about that since then the marketing companies started revising the oil prices in a fortnight but after or on June 16, 2017, the revision of prices is being done daily [1]. Because the controlled pricing on fuel is happening the oil marketing companies to increase their profit margin have not let the consumers get the benefits of the lower crude oil prices. If we compare with countries, India has raised more than China, USA, and Brazil.

The reason may be due to global crude oil prices, the post-pandemic effect, and the exchange rate between the Indian rupees. The price rise is not that much, but Indians are paying more because of the post-pandemic. We can see that country is trying for recovery. When GDP has fallen, it has crashed to a low level, so they want to recover in a V-shaped recovery.

## 1.1 *Challenges*

GST leaves very little space for the government to maneuver for the taxation, when we talk about the commodities available for taxation. In that case, there are much fewer commodities available for taxation after the GST came into the practice and petrol is the single most lucrative item for any kind of imposition for taxes. If the price is high for crude oil in the international market, the negative impacts of the hike in price will pass on to the customer. The oil marketing companies will try to increase the profit margin. Even if the prices are low, the government tries to tap the benefits from that situation of low price of international crude oil by getting up its revenue by putting more and more taxes on it. The consumer will be on the losing side. There will be a rise in inflammation rate, affecting their household income [2].

---

<sup>2</sup> <https://indianexpress.com/article/india/petrol-diesel-price-hiked-by-60-paisa-per-litre-after-83-day-hiatus-6447499/>.

Lack of energy security infrastructure: When crude oil prices were down in the international market, India would have purchased it at that cost and stored it in storage. If India has proper physical infrastructure for storing crude oil when low prices, it could tap its benefits.

## 1.2 Way Forward

We need a taxation system in which there should be an exemption on tax when a pandemic-like situation occurs. The poor people should not suffer from a high tax on products.

The overview of this paper is as follows. In Sect. 2, we have discussed various research work was done in this area. Section 3 addresses the dataset processing and time series analysis, followed by discussing various models in Sect. 4. The performance analysis is detailed in Sect. 5 and the comparative study of various machine learning models. Section 6 states the comparative study and end up with a conclusion in Sect. 7.

## 2 Literature Survey

Crude oil price prediction has always been a key concern for researchers because it plays a quite essential role in the country's economic development. A few statistical and machine learning-based works for crude oil prediction are as follows.

A machine learning-based crude oil price prediction has been done in [3]. This work has applied the AR, MA, and ARIMA models for crude oil price prediction. They have shown that ARIMA is one of the best-performing models because mean-squared error (MSE) for the ARIMA model is the lowest among all the three models. Similar work was done in [4]; in this work, they have applied ARIMA-GARCH for crude oil price prediction. They have also used the google index concept in crude oil prediction that shows the negative impact.

A time series-based flexible neural tree (FNT) is introduced in [5]. The working of this model is based on a flexible multi-layer feed-forward neural network and evolutionary procedures. In this work, they have shown that the model can handle the tasks automatically, and outcomes of the model show the high accuracy that indicates the model's effectiveness.

Deep learning-based models [6] are also prominently applied for crude oil price prediction. A deep belief network and the recurrent neural network have been applied in [7] to determine the nonlinear movement in crude oil price prediction. The outcomes of this work indicated that parameter selection is quite important for better prognosis.

A deep long short-term memory (DLSTM)-based architecture is proposed in [8] for time series forecasting of the petroleum data. A comparative study of the model

shows that DLSTM is performing better in comparison with other existing models in terms of MAE, RMSE, and RMSPE. In this work, they have used the genetic algorithm to optimize DLSTM architecture.

A crude oil forecasting-based work done in [7], in which, they have applied ensemble learning techniques along with a few deep learning-based models. The ensemble technique is not much preferable as compared to the time series model. But outcomes indicate that results obtained by time series models are comparatively better.

### 3 The Data Collection and Preprocessing

We have presented the data collection and the required preprocessing in the proposed work. The data used in this model is day-wise time series data.

#### 3.1 Data Acquisition

The data used in the model is collected from Petroleum Planning and Analysis Cell (PPAC)<sup>3</sup>, Delhi, India. The data is from the June 16, 2017, to the December 15, 2021. It has two attributes named date (in days) and the rate (in Indian rupees) for the crude oil dataset.

#### 3.2 Data Preprocessing

The date column is converted into timestamp format and made a date as an index. There is no missing value in it as it has covered day-wise data. If missing values occur, they can be filled by taking the mean of the previous two values or applying other techniques to overcome this scenario.

#### 3.3 Data Analysis

Time series data has a pattern as well as structure in it. The pattern indicates the seasonality in the data, trend can be increasing/decreasing, and data is stationary or non-stationary.

Decomposing the time series into seasonal, residual, and trend is done to analyze the data. The simple polynomial model can generate the seasonality of the data.

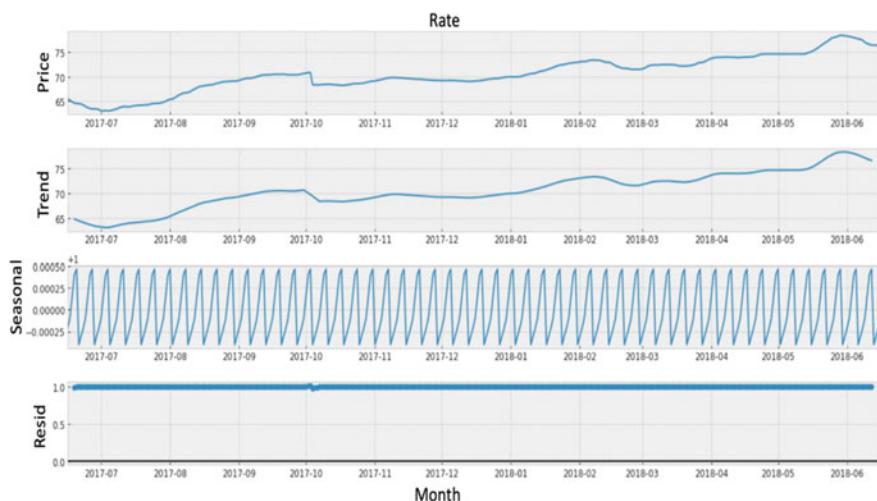
---

<sup>3</sup> [https://www.ppac.gov.in/content/149\\_1\\_PricesPetroleum.aspx](https://www.ppac.gov.in/content/149_1_PricesPetroleum.aspx).

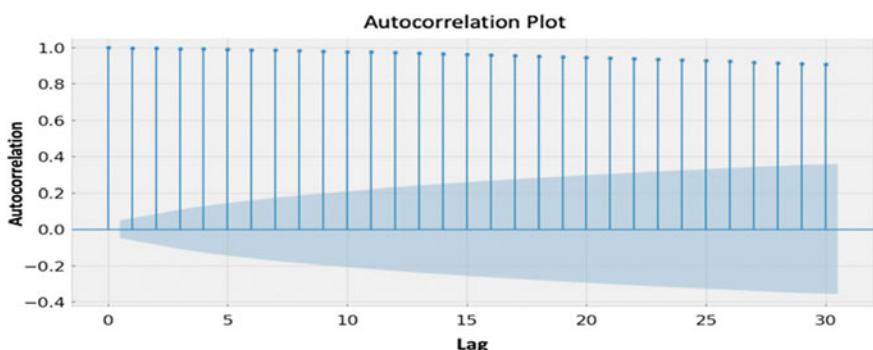
From Fig. 1, we can observe that there is an upward trend with some seasonality. In seasonality, four spikes can be seen in every month. The trend is not that strong here, and it is affected mainly due to the seasonality.

One important characteristic of time series is the stationary behavior of the data. Stationary data shows the constant mean and constant variance. To visualize the stationarity of the data, we have plotted the autocorrelation and partial autocorrelations. It shows how  $t$  is correlated to  $t + 1, t + 2, \dots, t + n$ . Where  $t$  and  $n$  indicate the time lag and lag value, respectively.

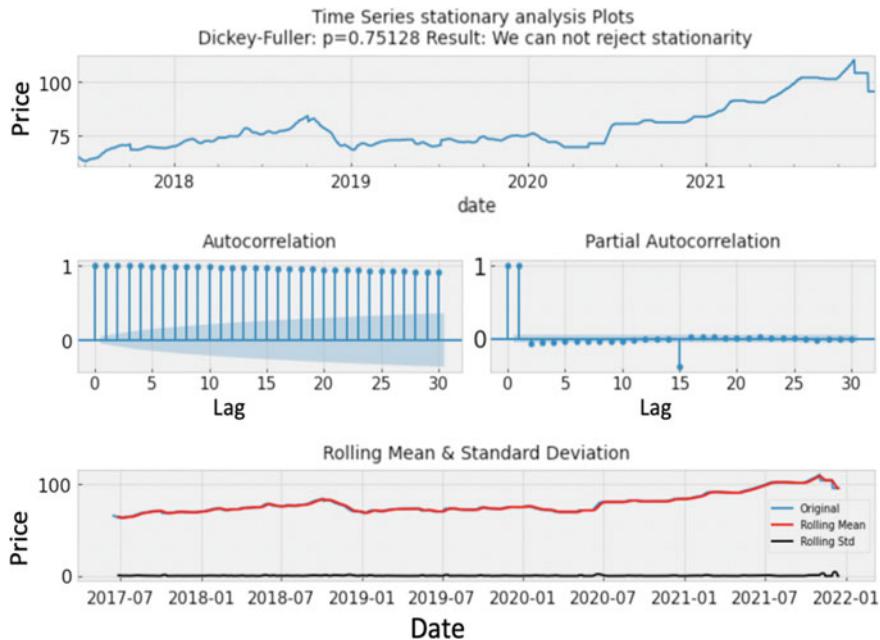
From Fig. 2, we can visualize that our data is highly correlated, which is non-stationary. Further, we have also applied the augmented Dickey–Fuller test to verify whether data is stationary or non-stationary. From Fig. 3, it can be seen that Dickey–Fuller value  $p$  ( $0.7512 > 0.05$ ) indicates that the data is non-stationary.



**Fig. 1** Trend, seasonality, and residual



**Fig. 2** Autocorrelation plot



**Fig. 3** Augmented Dickey–Fuller test

By observing the various metric from Fig. 3, it can be concluded that data is non-stationary, indicating that we cannot use the ARMA model because it works for stationary data [7]. Therefore, the ARIMA model is used, which has an integrated part to overcome from stationarity data. This stationarity is removed using difference transform or log scale transformation followed by smoothing.

## 4 The Model

This section briefly introduces the applied forecasting models and the proposed model's flowchart.

**AR(p) model:** AR models are models where forecasting is done with the help of past values. If  $X_t$  is current value, then  $X_{t-1}$ ,  $X_{t-2}$ ,  $X_{t-3}$ , etc., are past values.  $X_t$  is a function of different past values [8].

$$X_t = f(X_{t-1}, X_{t-2}, X_{t-3}, \dots, X_{t-k}) \quad (1)$$

in common representation, AR models come with the parameter ' $p$ '. When we say AR( $p$ ) models, that means there are parameters.

$$X_t = \beta_0 + \beta_1 * X_{t-1} + \beta_2 * X_{t-2} + \beta_3 * X_{t-3} + \dots + \beta_k * X_{t-k} \quad (2)$$

When AR (0) then it means  $X_t = \beta_0$ , if AR (1), then it will take up till  $\beta_1 * X_{t-1}$  and so on. Most of the time it takes up till  $t - 2$  or  $t - 3$ . Here  $\beta$  indicates the coefficient of autoregressive process.

**MA(q) Model:** MA models are the one where we forecast by taking the error term. If we regress the series with its immediate past series, we get error terms like  $\varepsilon_1, \varepsilon_2, \varepsilon_3, \dots, \varepsilon_n$ . Now instead of past values, you can use error terms for forecasting.

$$X_t = \phi_0 + \varepsilon_t + \phi_1 * \varepsilon_{t-1} + \phi_2 * \varepsilon_{t-2} + \phi_3 * \varepsilon_{t-3} + \dots + \phi_k * \varepsilon_{t-k} \quad (3)$$

$\varepsilon_t$  is assumed to be white noise means, when we take the mean of  $\varepsilon_t$  it will be zero and if you take the variance of  $\varepsilon_t$ , it will be constant [8],  $\phi$  represents the nematic constant.

**ARMA model:** This contains both AR and MA models which means they have past values and error terms [8].

$$\begin{aligned} X_t = & \beta_0 + \beta_1 * X_{t-1} + \beta_2 * X_{t-2} + \beta_3 * X_{t-3} + \dots + \beta_k * X_{t-k} \\ & + \varepsilon_t + \phi_1 * \varepsilon_{t-1} + \phi_2 * \varepsilon_{t-2} + \phi_3 * \varepsilon_{t-3} + \dots + \phi_k * \varepsilon_{t-k} \end{aligned} \quad (4)$$

**ARIMA ( $p, d, q$ ) model:** ARMA cannot work on non-stationary data so to make it work on stationary ARIMA was introduced. We have already discussed AR and MA other than that here ' $I$ ' stands for Integrated. It is denoted by  $d$  which makes non-stationary data into stationary using integration [15].

**SARIMA:** As we have read, ARIMA helps reduce or adjust trends to make it stationary data, but to accommodate seasonality, we need SARIMA. Here  $S$  stands for seasonal [15].

**HWES:** Holt-Winters exponential smoothing best fits the trend and seasonality data. It is easy to learn and easy to implement. It uses much less hypertunning as compared to models like ARIMA and SARIMA. Its forecasting is robust despite its simplicity. It can be used when our series has trend and seasonality [9].

**LSTM:** In short-term dependencies, RNN works well but for long-term dependency problems LSTM works properly. RNN fails in long-term dependencies. LSTM is a specific part of RNN. RNN has changed some things like it has introduced a memory unit known as cell unit or memory unit. Memory unit changes in every time zone and whatever the output is predicted, it considers it with the current input, previous output, and previous memory [10].

**SVM RBF:** RBF is a radial basis function kernel that is very powerful. It is very powerful at a high level because it can take into account weird strange non-regular decision boundaries where it is not linear or it is not quadratic. But it can still solve

the SVM problem and separate two classes fairly well in many of those situations. The goal of this kernel for natural 2D data is that it wants to consider the infinite number of interactions between variables. The radial basis kernel function of  $X_i$  and  $X_j$  is given by which is Euler's number [11].

**DeepAR:** This is a supervised learning algorithm for time series forecasting that uses a recurrent neural network to produce both point in time and probabilistic forecasts. It is a part of Gluonts [12]. The best thing about DeepAR is that it can work seamlessly on multiple time series. It develops one single global model. Initially, when you don't have data, it can see the closest model that can be applied and used for unseen data. While it may not be reliable but it is a good starting point to estimate the data over time.

**Bayesian Ridge Regression:** The main idea behind ridge regression is to find a new line that doesn't fit the training data. In other words, we introduce a small amount of bias to have the new line fit the data, but in return for that small amount of bias, we get a significant drop in variance, in other words by starting with slightly worse fit ridge regression provide better long-term predictions.

**Lasso:** It is very similar to ridge regression. In ridge regression, slope was getting squared, so instead of squaring the slope, we take the absolute value then it is Lasso regression. Just like ridge regression lambda can range between any value from positive to infinity and is determined using cross-validation.

**XGBoost:** It is a framework that can run on multiple programming languages like C++, R, and Python. It is platform free like we can run on any OS. It runs at a very fast speed, and due to its performance, it is most favorite [13].

**LightGBM:** It is another variation of gradient boosting and the light stands for the light version which supposedly makes this faster and more efficient. It is fast and gives high performance. It is based on decision tree also like XGBoost. It splits the tree leaf-wise [14].

**Ensembling:** It involves combining various models for better results. Here from Table 1, it can be seen that the residual correlation of the models is relatively high, which concludes that the ensemble of these models might be not effective.

**Table 1** Residual correlation

	TensorFlow simple LSTM	LightGBM	XGBoost
TensorFlow simple LSTM	1.000	—	—
LightGBM	0.704	1.000	—
XGBoost	0.719	0.995	1.000

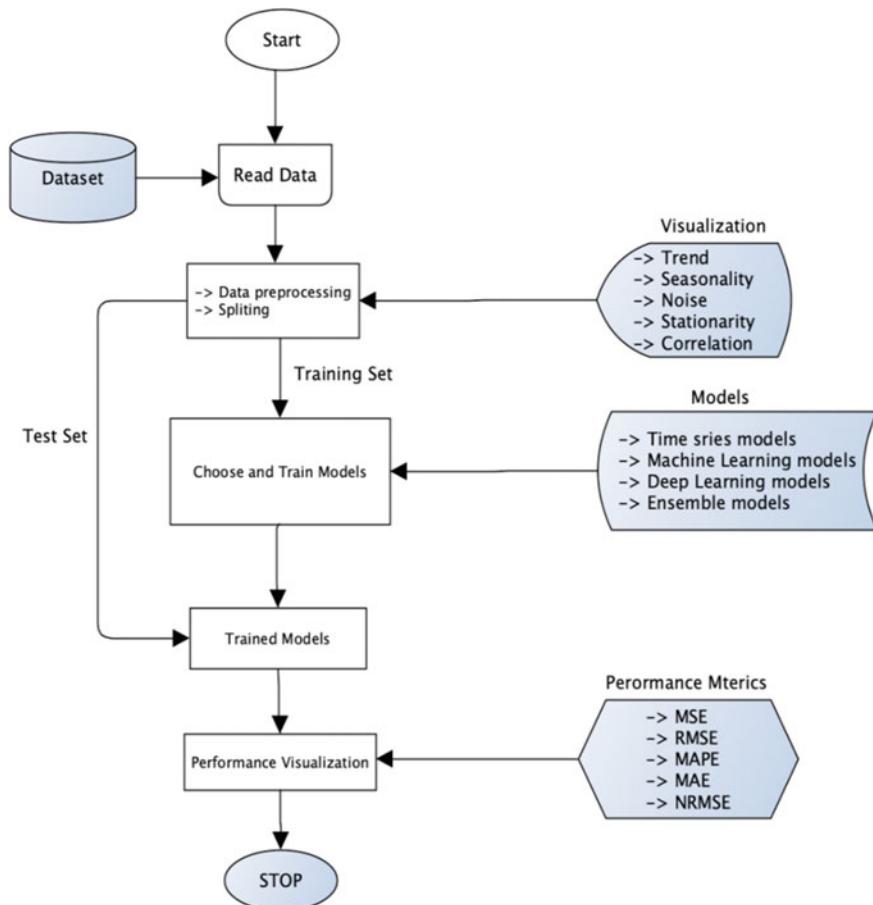
However, we have still applied the ensemble of these models to observe the outcomes for crude oil price prediction.

### Flowchart

The flowchart of the proposed model indicates the working as shown in Fig. 4. It specifies the step-by-step approach of the work in which, and we have taken petrol price dataset where it shows the price of petrol in each date.

Visualizing the time series data gives us an idea about dataset behavior. The dataset is read, and then it goes through the preprocessing part, where data visualization takes place. We visualize trends, seasonality, noise, stationarity, and correlation of records with one another. Then data is divided into test and train data.

In training data, various models are assigned, and data fits into the model. After fitting the data model predicts the test data.



**Fig. 4** Flowchart of the complete process

The performance of various models is measured with the help of various performance metrics. Then the best one may be considered for crude oil price forecasting.

## 5 Performance Analysis

The performance of the proposed model is tested by writing the program using Python 3.10.0 on Jupyter Notebook 6.4.6. We have used visualization techniques and libraries: pyplot, matplotlib, sklearn, and statsmodels.

In this work, the dataset contains 1296 records for training and 348 days for testing. We have applied various models to measure the performance with the help of actual and predicted values.

Further in this section, we have analyzed different models with their prediction measures. In the subpart of this section, there will be a comparative study on the time series model, tree models, and ensemble models.

### 5.1 Performance Metrics

After the model prediction, the models are measured with the help of performance metrics. This shows that if our model is good or not. In this section, we will see the regression metrics that is if we build a regression model they will be accessed with the performance. We will cover some important metrics.

#### 5.1.1 MAE

Mean absolute error (MAE) is the absolute difference between the model predictions and the actual values [9].

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (5)$$

From the equation, it says to calculate the difference between true values and predicted values. Calculate the average of this point after summing it. If  $\text{MAE} = 0$ , this means the model is perfect for prediction. The error is increased in proportional order.

### 5.1.2 MSE

Mean square error (MSE) is similar to MAE but instead of obtaining absolute value squaring of the value takes place [5].

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (6)$$

### 5.1.3 RMSE

Root mean square error (RMSE) is simply the square root of the MSE value. The problem with MSE is that the units are very different from the actual data points because data is squared and hard to compare [5].

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (7)$$

If you recall from statistics that standard deviation is a measure of the dispersion from the mean, which shows that how much the data is away from the mean. It estimates how large the residuals are dispersed which is the standard deviation.

### 5.1.4 MAPE

Mean absolute percentage error (MAPE) is similar to MAE. One problem with MAE values was that they can range from zero to infinity, making it difficult to interpret the result compared to the training data. This is the reason why answering a form of percentage is very beneficial because everyone can understand it [5].

$$\text{MAPE} = \frac{100\%}{n} \sum_{i=1}^n |(y_i - \hat{y}_i)/y_i| \quad (8)$$

### 5.1.5 NRMSE

Normalized root mean squared error (NRMSE) is another metric for analyzing the performance of models [10].

$$\text{NRthe MSE} = \frac{\sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}}{\text{SD}_y} \quad (9)$$

## 5.2 Comparative Analysis

In this section, we have analyzed the comparative performance of the various statistical and machine learning models.

From Table 2, it can be observed that Holt-Winters exponential smoothing (HWES) is one of the best-performing model along with ARIMA and SARIMAX. HWES works well on both trend and seasonality due to its triple exponential smoothing. The data is in regression form and is non-stationary, so the ARMA model fails to predict the test value. Models used here are from time series, machine learning, and deep learning. In the further discussion, it is shown that how the time series models are better than other models.

**Table 2** Performance of the models w.r.t performance metrics

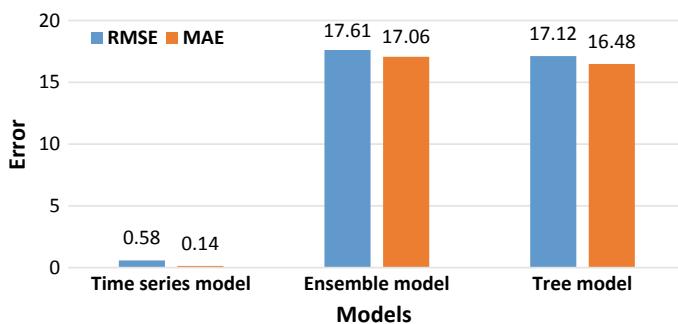
	MSE	RMSE	MAPE	MAE	NRMSE
EnsembleXG + TF	267.63	16.36	0.16	15.74	0.62
EnsembleLIGHT + TF	301.08	17.35	0.17	16.76	0.66
EnsembleXG + LIGHT + TF	309.65	17.60	0.18	17.06	0.67
EnsembleXG + LIGHT	367.21	19.16	0.19	18.69	0.73
Random forest tunned	404.02	20.10	0.2	19.72	0.76
DeepAR	53.88	7.34	0.07	6.19	0.28
TensorFlow simple LSTM	220.26	14.84	0.14	13.81	0.56
SVM RBF	67.95	8.24	0.08	7.13	0.31
LightGBM	406.94	20.17	0.2	19.71	0.77
XGBoost	329.67	18.16	0.18	17.67	0.69
Random forest	480.84	21.93	0.22	21.41	0.83
Lasso	332.43	18.23	0.18	17.48	0.69
Bayesian Ridge	324.65	18.02	0.18	17.28	0.68
SARIMAX	0.34	0.58	0	0.13	0.02
ARIMA	0.34	0.58	0	0.13	0.02
MA	113.62	10.66	0.1	10.17	0.4
AR	0.37	0.61	0	0.17	0.02
HWES	0.34	0.58	0	0.13	0.02

## 6 Discussion

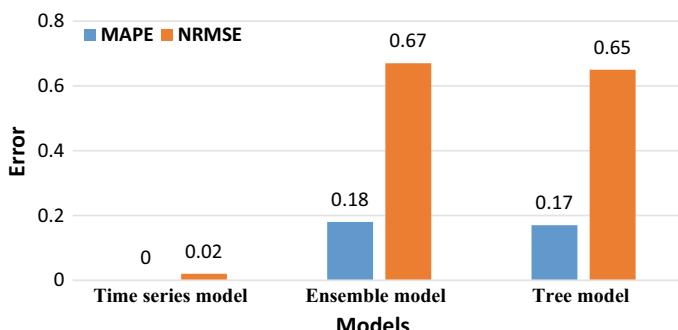
This section summarizes the performance of time series models, ensemble models, and tree models over crude oil price forecasting. For the same, we have used the RMSE, MAE, MAPE, and NRMSE as performance metrics to compare the models' performances.

To observe the performance of time series, ensemble, and tree models, we have taken the average of RMSE, MAE, MAPE, and NRMSE values of four best-performing models in each category. For time series models, we have considered HWES, AR, ARIMA, and SARIMAX as they are best-performing models; it can be seen from Table 2. Similarly, Random forest, XGBoost, LightGBM, and SVM RBF are used for tree models. In ensemble models, we have taken the ensemble of TensorFlow simple LSTM, XGBoost, and LightGBM.

The outcomes of the various models show that some of the models have performed well over training data but failed to predict the test data. Time series models are well-suited for predicting crude oil prices as observed in Figs. 5 and 6.



**Fig. 5** Error comparison of RMSE and MAE



**Fig. 6** Error comparison of MAPE and NRMSE

From Fig. 5, it can be observed that RMSE and MAE values are 0.58 and 0.14, respectively, for time series models, which is lowest in comparison with ensemble models and tree models. When we compare the tree models to the ensemble models, it is found that the tree model is slightly better.

Similarly, from Fig. 6, we can observe that MAPE and NRMSE are lowest for time series models which are 0.00 and 0.02, respectively. Once again tree models and ensemble almost equally performing models for forecasting the crude oil prices.

Therefore, from Figs. 5 and 6, we can conclude that time series models are best-performing models for predicting the crude oil prices in comparison with ensemble models and tree models.

## 7 Conclusion

Time series models are good in predicting univariate data. In this work, we have analyzed the various models for prediction on the petrol price of Delhi. The model used was machine learning, deep learning, and time series. Ensemble modeling was also done for checking the performance but it did not give a good result. HWES and AR are comparatively good in predicting the test data.

We have also performed the comparative study using the performance metrics which are MSE, RMSE, MAPE, MAE, and NRMSE. The values of the metrics give us a clear view that the time series-based models are performing comparatively better. The benefit of this work is that future data scientists and researchers could take note that if the dataset has increasing trend and seasonality, they should prefer these proposed models without getting into others.

## References

1. Gabhane, D., Gabhane, M.: Rising prices of petrol and diesel in India since 2014: an analytical study. *PalArch's J. Archaeol. Egypt/Egyptology* **18**(7), 2309–2315 (2021)
2. Bhattacharya, B., Batra, A.: Fuel pricing policy reform in India: implications and way forward. *Econ. Polit. Wkly.* 77–86 (2009)
3. Shambulingappa, H.: Crude oil price forecasting using machine learning. *Int. J. Adv. Sci. Innov.* **1**(1) (2020)
4. Yao, T., Zhang, Y.-J.: Forecasting crude oil prices with the Google index. *Energy Procedia* **105**, 3772–3776 (2017)
5. Chen, Y., Yang, B., Dong, J., Abraham, A.: Time-series forecasting using flexible neural tree model. *Inf. Sci.* **174**(3–4), 219–235 (2005)
6. Kumar, S., Mishra, S., Singh, S.K.: Deep transfer learning-based COVID-19 prediction using chest X-rays. *J. Health Manag.* **23**(04), 730–746 (2021). <https://doi.org/10.1177/09720634211050425>
7. Chen, Y., He, K., Tso, G.K.: Forecasting crude oil prices: a deep learning based model. *Procedia Comput. Sci.* **122**, 300–307 (2017)
8. Sagheer, A., Kotb, M.: Time series forecasting of petroleum production using deep LSTM recurrent networks. *Neurocomputing* **323**, 203–213 (2019)

9. Garg, C. et al.: Adaptive fuzzy logic models for the prediction of compressive strength of sustainable concrete. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_47](https://doi.org/10.1007/978-981-16-2164-2_47)
10. Chakraborty, R. et al.: Study and prediction analysis of the employee turnover using machine learning approaches. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573759>.
11. Claesen, M., De Smet, F., Suykens, J.A., De Moor, B.: Fast prediction with SVM models containing RBF kernels. arXiv preprint [arXiv:1403.0736](https://arxiv.org/abs/1403.0736) (2014)
12. Salinas, D., Flunkert, V., Gasthaus, J., Januschowski, T.: DeepAR: probabilistic forecasting with autoregressive recurrent networks. Int. J. Forecast. **36**(3), 1181–1191 (2020)
13. Kumar, S., Mishra, S., Singh, S.K.: A machine learning-based model to estimate PM<sub>2.5</sub> concentration levels in Delhi's atmosphere. Heliyon **6**(11), e05618 (2020)
14. Ju, Y., Sun, G., Chen, Q., Zhang, M., Zhu, H., Rehman, M.U.: A model combining convolutional neural network and lightgbm algorithm for ultra-short-term wind power forecasting. IEEE Access **7**, 28309–28318 (2019)
15. Kumari, S., Singh, S.K.: Machine learning-based time series models for effective CO<sub>2</sub> emission prediction in India (2022). Available on: <https://doi.org/10.21203/rs.3.rs-1265771/v1>

# Performance Analysis of Improved Mobility Models to Check Their Impact on QoS in MANET



Munsifa Firdaus Khan and Indrani Das

**Abstract** In mobile ad hoc networks (MANETs), nodes are mobile and communicate with each other without any help of base station. Mobility models play a useful role in providing QoS support in MANET. In this paper, we have investigated the effects of various mobility models which include modified Gauss–Markov (MGM), enhanced modified Gauss–Markov (EMGM), and random direction-3D (RD-3D), on routing protocol AODV and DSDV with parameters PDR, delay, and throughput. The enhanced modified Gauss–Markov mobility model outperforms well in comparison with the other mobility models with respect to PDR and delay, while the random direction-3D mobility model performs better in comparison with other mobility models with respect to throughput for both AODV and DSDV routing protocols.

**Keywords** AODV · DSDV · MANET · Mobility models · QoS

## 1 Introduction

In MANET nodes are self-organizing, self-describing, independent, and adaptive. Due to the lack of a base station, nodes in MANET communicate with each other by broadcasting messages to their neighbors. The MANET's distinct properties, such as changeable node architecture, a lack of central coordination, a lack of accurate state information, node mobility, and insufficient resources such as bandwidth and battery power, make it an important area for research [7, 20–22]. The performance level of service offered by a network to a user during communication is referred to as QoS. In MANET, improved QoS is required to achieve increased network performance [9, 22]. We investigated on the impact of QoS in MANETs using a novel rectangular-3D position allocator on standard routing protocols like AODV and DSDV, as well as improved mobility models like modified Gauss–Markov (MGM), enhanced modified Gauss–Markov (EMGM), and random direction-3D (RD-3D). The mobility of nodes has a significant impact on network performance [23]. The

---

M. F. Khan (✉) · I. Das  
Assam University, Silchar, Cachar, Assam 788011, India  
e-mail: [munsifa737@gmail.com](mailto:munsifa737@gmail.com)

performance of the network can be affected by abrupt changes in the topology of nodes. The EMGM mobility model outperforms the other mobility models in terms of PDR and delay, while the RD-3D mobility model outperforms the other mobility models in terms of throughput for both AODV and DSDV routing protocols.

The following is how the paper is laid out: Sect. 2 provides a high-level review of the mobility models, Sect. 3 examines conventional routing protocols, and Sect. 4 depicts simulation and performance analysis. The paper comes to a close with Sect. 5.

## 2 Mobility Models

The mobility model aids in the comprehension of node movement. To mimic more realistically, it is critical to select the proper mobility model for the environment. This paper presents a quick overview of several mobility models and their effects on routing protocols such as AODV and DSDV. It also shows how it affects MANET QoS. This study provides a clear picture of how to choose between mobility models and their settings. We have shown the RD-3D, MGM, and EMGM mobility models, as well as the unique rectangular-3D position allocator.

### 2.1 Rectangular-3D Position Allocator

This position allocator orients nodes in a three-dimensional plane. Initially, nodes will be placed in both the  $X$ -axis and  $Y$ -axis like grid position allocator. After the grid is full, then the value at  $Z$  will increase, and the next node will be placed in the next grid on  $Z$ -axis. With an increasing number of nodes, the length of the  $Z$ -axis increases. The node positions are calculated using the mathematical formula:

$$X = x_0 + \Delta x * (m \% n_x) \quad (1)$$

$$Y = y_0 + \Delta y * ((m / n_x) \% n_y) \quad (2)$$

$$Z = z_0 + \Delta z * (m / (n_x * n_y)) \quad (3)$$

where  $x_0$ ,  $y_0$ ,  $z_0$  represent the minimum values of  $X$ -,  $Y$ -, and  $Z$ -axes,  $m$  represents the current node,  $n_x$ ,  $n_y$  represent the number of nodes at  $X$ -axis and  $Y$ -axis,  $\Delta x$  represents the spaces between the two successive nodes at  $X$ -axis,  $\Delta y$  represents the spaces between the two successive nodes at  $Y$ -axis and  $\Delta z$  represents the spaces between the two successive nodes at  $Z$ -axis. This position allocator allocates the nodes uniformly in the simulation area. This is one of the main advantages of this

allocator for static nodes. When stable nodes are uniformly distributed in a simulation area, then the performance of the network increases automatically [11].

## 2.2 Gauss–Markov Mobility Model

This mobility model is proposed by Liang and Haas [4]. “At a constant interval of time ‘ $t$ ,’ the values of nodes with respect to speed, direction, and pitch are calculated based on the former value of pitch, direction, and speed at  $(t-1)$ th time interval. The speed, direction, and pitch values are calculated by the following equations:

$$S_t = \alpha S_{t-1} + (1 - \alpha) \tilde{S} + \sqrt{(1 - \alpha^2)} Sx_{t-1} \quad (4)$$

$$D_t = \alpha D_{t-1} + (1 - \alpha) \tilde{D} + \sqrt{(1 - \alpha^2)} Dx_{t-1} \quad (5)$$

$$P_t = \alpha P_{t-1} + (1 - \alpha) \tilde{P} + \sqrt{(1 - \alpha^2)} Px_{t-1} \quad (6)$$

where  $S_t$ ,  $D_t$ , and  $P_t$  are the new speed, direction, and pitch at time interval  $t$ ,  $\tilde{S}$ ,  $\tilde{D}$ , and  $\tilde{P}$  are the mean speed, mean direction, and mean pitch,  $Sx_{t-1}$ ,  $Dx_{t-1}$ , and  $Px_{t-1}$  are random variables and  $\alpha$  is a random variable whose value lies within the range of  $0 < \alpha < 1$ ” [3–5, 16, 17].

## 2.3 Modified Gauss–Markov Mobility Model

“The original GM mobility model has been changed. The speed, pitch, and direction of each node are calculated in this model at a predetermined distance ‘ $d$ ’ based on the prior value of pitch, direction, and speed at  $(d-1)$ th distance. The following mathematical formulas are used to calculate the speed, direction, and pitch values:

$$S_d = \alpha S_{d-1} + (1 - \alpha) \tilde{S} + \sqrt{(1 - \alpha^2)} Sx_{d-1} \quad (7)$$

$$D_d = \alpha D_{d-1} + (1 - \alpha) \tilde{D} + \sqrt{(1 - \alpha^2)} Dx_{d-1} \quad (8)$$

$$P_d = \alpha P_{d-1} + (1 - \alpha) \tilde{P} + \sqrt{(1 - \alpha^2)} Px_{d-1} \quad (9)$$

where  $S_d$ ,  $D_d$ , and  $P_d$  are the new speed, direction, and pitch at distance interval  $d$ ,  $\tilde{S}$ ,  $\tilde{D}$ , and  $\tilde{P}$  are the mean speed, mean direction, and mean pitch,  $Sx_{d-1}$ ,  $Dx_{d-1}$ , and

$Px_{d-1}$ , are random variables and  $\alpha$  is a random variable whose value lies within the range of  $0 < \alpha < 1$ . Randomness is determined by altering the value of  $\alpha$ " [12].

## 2.4 Enhanced Modified Gauss–Markov Mobility Model

"It is enhanced version of the MGM mobility model. In this model, at a fixed distance ' $d$ ,' the speed, pitch, and direction of each node is estimated based on the previous value of pitch, direction, and speed at  $(d-1)$ th distance. After traveling for a fixed distance, here the node pauses for a fixed time called pause time. The pause time is chosen from a predefined range  $[T_{\min}, T_{\max}]$ . The speed, direction, and pitch value are calculated by the mathematical formulas as shown in Eqs. (7), (8), and (9)" [13].

## 2.5 Random Direction-3D Mobility Model

"This mobility model is a 3D version of the existing random direction-2D mobility model where the node moves based on the random direction  $d$ . At a specific period, time ' $t$ ,' the speed  $v(t)$  of a node is determined as a Gaussian distribution from the interval [minimum\_speed, maximum\_speed], and an angular direction  $d$  is chosen from the interval  $[0, 2\pi]$ . Considering the node speed, pause time, and angular direction, each node moves toward a definite direction until it arrives at the boundary of the model. When it arrives at the boundary, it pauses, selects a new direction and speed, and starts moving toward a new direction" [14].

## 3 Routing Protocols

We have looked at two different forms of routing protocols, one proactive (DSDV) and the other reactive (AODV). Reactive routing protocols do not have route information, but proactive routing protocols keep a routing table with the route to the destination [8–10].

### 3.1 DSDV

This protocol is a modification of the traditional Bellman–Ford routing algorithm [1]. In this protocol, a routing table is exchanged among neighbor nodes to keep track of up-to-date information about the network's topology. Due to the usage of a sequence number, this approach primarily resolves the count to infinity problem [21].

### 3.2 AODV

“In this protocol, a path is discovered whenever a node wants to exchange information with another node. It provides loop-free paths and is scalable for massive networks” [2, 10, 15, 18, 19]. “When a path is unavailable, then the path discovery procedure is initiated, and the source node broadcasts RREQ packets to all the neighbor nodes. When neighbor nodes receive RREQ packets, they either transmit a RREP packet or broadcast the RREQ packet to the rest of the network” [6]. “When a node detects a path failure, it sends a RERR packet to its related neighbor nodes, informing all of the related nodes of the broken path” [2, 6, 19].

## 4 Performance Evaluation

### 4.1 Simulation Environment

Simulation is done in NS-3 [15]. The parameters and their values for these mobility models have been given in Tables 1, 2, 3, and 4, respectively.

**Table 1** Simulation environment

Parameters	Values
Routing protocols	AODV, DSDV
Mobility models	RD-3D, MGM, and EMGM
Propagation delay model	Constant speed propagation delay model
Propagation loss model	Friss propagation loss model
Position allocator	Rectangular 3D (i) For 25 nodes, length = width = 3, min X = min Y = min Z = 0, $\Delta X = \Delta Y = \Delta Z = 166$ (ii) For 50 nodes, length = width = 4, min X = min Y = min Z = 0, $\Delta X = \Delta Y = \Delta Z = 125$ (iii) For 75 nodes, length = width = 5, min X = min Y = min Z = 0, $\Delta X = \Delta Y = \Delta Z = 100$ (iv) For 100 nodes, length = width = 6, min X = min Y = min Z = 0, $\Delta X = \Delta Y = \Delta Z = 100$
Number of nodes	25, 50, 75 and 100
Total simulation time	120 s
Number of flows	10
Type of traffic	CBR
Data rate	512 kbps
Bit rate	2048 kbps
Transmission power	7.5 dBm

**Table 2** Parameters for MGM mobility model

Parameters	Values
Bounds	$X [0, 500]$ , $Y [0, 500]$ , and $Z [0, 500]$
Distance	12 m
Tuning parameter, $\alpha$	0.65
Mean velocity	[0, 20] m/s
Mean direction	[0, 6.283185307]
Mean pitch	[0.05, 1.0]
Normal velocity	Mean = 0.5, Variance = 0.4, and Bound = 0.6
Normal direction	Mean = 0.5, Variance = 0.4, and Bound = 0.6
Normal pitch	Mean = 0.5, Variance = 0.4, and Bound = 0.6

**Table 3** Parameters for EMGM mobility model

Parameters	Values
Bounds	$X [0, 500]$ , $Y [0, 500]$ , and $Z [0, 500]$
Distance	12 m
Pause time	[1, 5] s
Tuning parameter, $\alpha$	0.65
Mean velocity	[0, 20] m/s
Mean direction	[0, 6.283185307]
Mean pitch	[0.05, 1.0]
Normal velocity	Mean = 0.5, Variance = 0.4, and Bound = 0.6
Normal direction	Mean = 0.5, Variance = 0.4, and Bound = 0.6
Normal pitch	Mean = 0.5, Variance = 0.4, and Bound = 0.6

**Table 4** Parameters for RD-3D mobility model

Parameters	Values
Bounds	$X [0, 500]$ , $Y [0, 500]$ , and $Z [0, 500]$
Speed	[10, 20] m/s
Pause time	5 s

## 4.2 Experimental Results

Huge experiments on AODV and DSDV with 25, 50, 75, and 100 sets of nodes with various improved mobility models such as RD-3D, MGM, and EMGM employing

**Table 5** Throughput for AODV

No. of nodes	MGM	EMGM	RD-3D
25	0.8119	0.2323	7.5299
50	20.5650	6.5396	25.3575
75	20.3472	13.8718	26.1673
100	19.9955	8.3585	20.4361

**Table 6** Throughput for DSDV

No. of nodes	MGM	EMGM	RD-3D
25	1.5329	0.1918	55.2263
50	0.4398	1.3210	744.1532
75	0.7016	1.4645	0.7419
100	1.0080	0.9228	816.5877

Rectangular-3D position allocator are done to examine the influence of QoS in MANET.

#### 4.2.1 Throughput

Tables 5 and 6 demonstrate the results of AODV throughput with various sets of nodes and different mobility models. Table 5 shows that the maximum throughput for AODV is 26.1673 kbps for 75 nodes using RD-3D, while the lowest is 0.2323 kbps for 25 nodes using EMGM. For all set of nodes considered, RD-3D has a higher throughput than EMGM, whereas EMGM has a lower throughput. In comparison with the EMGM and MGM mobility models, the RD-3D mobility model has a higher average throughput for AODV. Table 6 shows that the highest throughput for DSDV is 816.5877 kbps for 100 nodes when using RD-3D, while the lowest is 0.1918 kbps for 25 nodes when using EMGM. For all the nodes, RD-3D provides higher DSDV throughput, whereas MGM provides lesser throughput. In comparison with the EMGM and MGM mobility models, the RD-3D mobility model has a higher average throughput for DSDV.

#### 4.2.2 Delay

Tables 7 and 8 demonstrate the delay values for AODV and DSDV, respectively, utilizing different combinations of nodes and our suggested mobility models. Table 7 shows that the largest delay value for AODV obtained for 25 nodes is 1231.6941 s, while the shortest is 320.4229 s using RD-3D for 75 nodes. Between the three mobility models, RD-3D yields lower delay values for 75 and 100 sets of nodes, MGM delivers lower delay values for 50 sets of nodes, and EMGM delivers lower delay values for 25 sets of nodes. Furthermore, using the EMGM mobility model,

**Table 7** Delay for AODV

No. of nodes	MGM	EMGM	RD-3D
25	884.6051	333.6364	1231.6941
50	327.7578	461.6186	398.5152
75	439.3279	480.1246	320.4229
100	402.1063	684.2848	335.6285

**Table 8** Delay for DSDV

No. of nodes	MGM	EMGM	RD-3D
25	8481.3999	2.1999	11944.3999
50	1558.1999	3473.8999	3141.7999
75	5630.6999	5042.2999	7552.5
100	3888.6999	2717.3999	4750.2999

the average delay for AODV is smaller, whereas using the RD-3D mobility model, the average delay is larger. Table 8 shows that the largest delay obtained for AODV using RD-3D mobility models for 25 nodes is 11944.3999 s, whereas the shortest delay obtained using EMGM for 25 nodes is 2.1999s. The average delay for DSDV is shown to be larger when using the RD-3D mobility model and lower when utilizing the EMGM mobility model.

#### 4.2.3 Packet Delivery Ratio

As shown in Tables 9 and 10, one of the performance indicators used to examine the influence of various mobility models using different sets of nodes on AODV and DSDV is PDR. Table 9 shows that the maximum PDR for AODV obtained

**Table 9** PDR for AODV

No. of nodes	MGM	EMGM	RD-3D
25	0.7252	0.2467	0.3577
50	0.8129	0.9705	0.6196
75	0.8761	0.9317	0.8369
100	0.8688	0.9037	0.8619

**Table 10** PDR for DSDV

No. of nodes	MGM	EMGM	RD-3D
25	0.2571	0.0245	0.0999
50	0.0571	0.2979	0.0959
75	0.0816	0.4592	0.0612
100	0.1061	0.0796	0.0245

using the EMGM mobility model for 50 nodes is 0.9705, whereas the lowest PDR obtained using the EMGM mobility model for 25 nodes is 0.2467. It is worth noting that EMGM delivers greater PDR values for 50, 75, and 100 nodes, whereas MGM delivers greater PDR values for 25 nodes. Furthermore, the EMGM mobility model is found to be superior to the MGM and RD-3D mobility models in terms of PDR for AODV. Table 10 shows that the greatest PDR for DSDV achieved using the EMGM mobility model for 75 nodes is 0.4592, while the lowest is 0.0245 using EMGM for 25 nodes and RD-3D for 100 nodes. It is worth noting that EMGM delivers greater PDR values for 50 and 75 nodes, whereas MGM delivers higher PDR values for 25 and 100 nodes. Furthermore, the EMGM mobility model is found to be superior to the MGM and RD-3D mobility models in terms of PDR for DSDV.

## 5 Conclusion and Future Works

We conducted extensive tests on AODV and DSDV with a variety of nodes, QoS parameters, and mobility models to check their impact on QoS in MANET. In comparison with MGM and EMGM mobility models, the throughput for both AODV and DSDV is higher when employing the RD-3D mobility model, as a result, QoS support is improved. It is also analyzed that delay is better for both the routing protocols using EMGM mobility models, and it is worst using RD-3D mobility models. So, it is concluded that EMGM outperforms well to the parameter delay compared to other mobility models. It is also noticed that EMGM gives higher PDR values than the other mobility models for both AODV and DSDV routing protocols. Therefore, it is concluded that EMGM gives higher QoS support with respect to PDR and delay in comparison to MGM and RD-3D mobility models, whereas RD-3D mobility models give better QoS support to throughput. It is concluded that different mobility models give different results and have huge impact on QoS support in MANET. One of the future directions of this paper is to decrease the delay values and increase the PDR of the RD-3D mobility model by enhancing the model with better border solutions when the node reaches the boundary. Another way is to improve the MGM and EMGM mobility models so that the throughput can be increased. This paper is helpful to the researchers who are working on MANET. This research will help them to identify a suitable mobility model for enhancing their research work.

## References

1. Ahmed, G., Barskar, R., Barskar, N.: An improved DSDV routing protocol for wireless ad hoc networks. *Procedia Technol.* **6**, 822–831 (2012)
2. Ahmed, A., Hanan, A., Osman, I.: AODV routing protocol working process. *J. Convergence Inf. Technol.* **10**(2), 1–7 (2015)
3. Biomo, J., Thomas, K., Marc, S.: An enhanced Gauss-Markov mobility model for simulations of unmanned aerial ad hoc networks. In: 7th IFIP Wireless and Mobile Networking Conference (WMNC). IEEE (2014)

4. Broyles, D., Jabbar, A., Sterbenz, J.P.G.: Design and analysis of a 3-D Gauss-Markov mobility model for highly-dynamic airborne networks. In: Proceedings of the International Telemetering Conference (ITC). San Diego, CA (2010)
5. Camp, T., Boleng, J., Davies, V.: A survey of mobility models for ad hoc network research. *Wirel. Commun. Mob. Comput.* **2**(5), 483–502 (2002)
6. Chakeres, I., Elizabeth, M.: AODV routing protocol implementation design. In: 24th International Conference on Distributed Computing Systems Workshops, Proceedings. IEEE (2004)
7. Dhungana, A., Bulut, E.: Peer-to-peer energy sharing in mobile networks: applications, challenges, and open problems. *Ad Hoc Netw.* **97**(102029), 1–16 (2020)
8. Khan, M.F., Das, I.: A study on Quality-of-Service routing protocols in mobile ad hoc networks. In: 2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN), pp.95–98 (2017)
9. Khan, M.F., Das, I.: Effect of different propagation models in routing protocols. *Int. J. Eng. Adv. Technol. (IJEAT)* **9**(2), 3975–3980 (2019)
10. Khan, M.F., Das, I.: An investigation on existing protocols in MANET. In: Innovations in Computer Science and Engineering, Lecture Notes in Networks and Systems, vol. 74, pp. 215–224 (2019)
11. Khan, M.F., Das, I.: Design and Analysis of a novel Rectangular-3D Position Allocator to improve QoS in MANET. (Manuscript submitted for publication) (2020)
12. Khan, M.F., Das, I.: Modified Gauss Markov Mobility Model to improve QoS in MANET. (Manuscript submitted for publication) (2020)
13. Khan, M.F., Das, I.: An enhanced modified Gauss Markov mobility model to achieve better QoS support in MANET. (Manuscript submitted for publication) (2020)
14. Khan, M.F., Das, I.: Implementation of random direction-3D mobility model to achieve better QoS support in MANET. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, **11**(10), 195–203 (2020)
15. Khan, M.F., Das, I.: Performance evaluation of routing protocols in NS-2 and NS-3 simulators. *Int. J. Adv. Trends Comput. Sci. Eng.* **9**(4), 6509–6517 (2020)
16. Khan, M.F., Das, I.: Analysis of various mobility models and their impact on QoS in MANET. In: Computationally Intelligent Systems and their Applications, vol. 950, pp.131–141 (2021)
17. Khan, M.F., Das, I.: Impact of various parameters on Gauss Markov mobility model to support QoS in MANET. In: Cloud Computing Enabled Big-Data Analytics in Wireless Ad-hoc Networks, pp. 85–101 (2022)
18. Landge, P., Nigavekar, A.: Modified AODV protocol for energy efficient routing in manet. *Int. J. Eng. Sci. Res. Technol.* **5**(3), 523–529 (2016)
19. Maurya, P.K., Sharma, G., Sahu, V., Roberts, A., Srivastava, M.: An overview of AODV routing protocol. *Int. J. Mod. Eng. Res. (IJMER)* **2**(3), 728–732 (2012)
20. Mikitiuk, A., Trojanowski, K.: Maximization of the sensor network lifetime by activity schedule heuristic optimization. *Ad Hoc Netw.* **96**(101994), 1–15 (2020)
21. Murthy, C.S.R., Manoj, B.S.: Ad hoc Wireless Networks: Architectures and Protocols. Prentice Hall PTR (2004)
22. Sichitiu, L.M.: Mobility models for ad hoc networks. In: Guide to Wireless Ad Hoc Networks, pp. 237–254. Springer, London (2009)
23. Zhong, X., Chen, F., Guan, Q., Ji, F., Yu, H.: On the distribution of nodal distances in random wireless ad hoc network with mobile node. *Ad Hoc Netw.* **97**, 102026, 1–30 (2020)

# A Comprehensive Analysis of Application-Based MAC Protocol for Wireless Sensor Network



Mamta Mann and Rishipal Singh

**Abstract** Wireless sensor network has revolutionized the field of science and technology due to its ubiquitous deployment attribute. The unprecedented growth in WSNs is credited to its miniature size and cost-effectiveness. However, one of the limiting factors of sensor nodes is that they are battery-operated and providing longevity to the network is one of the significant concerns while designing WSNs. The medium access control (MAC) protocol is seen as an important part as it handles the coordination and communication amongst the sensor nodes by controlling access to the channel. Moreover, its design can have a considerable effect on several parameters related to the sensor nodes such as battery consumption, collision, network lifetime and latency. Therefore, it becomes of great importance to design the MAC protocol as per the requirements of network in order to enhance the lifetime of the network. In this paper, we discuss and categorize various MAC protocols on the basis of their design for different areas of application.

**Keywords** MAC protocol · Wireless sensor network · Access mechanism

## 1 Introduction

The up-gradation in the field of information and communications technology (ICT) has been taking place ever since its inception, which has led to the evolution of various areas of development. One such example is WSN-based applications. The miniature wireless sensors/detectors employed in a network have unlocked an unparalleled prospect for a large number of real-time applications. Due to small size of sensor devices and trouble-free deployment, they can be placed in different scenarios. These miniatures can be positioned in areas, where human's ingress/entrance is unsafe or continuing unattended observation is needed. The information gathered is vital as it could help in predicting any natural hazard or calamity, accidents, for example volcano monitoring, mine monitoring, health care, smart management of a firm,

---

M. Mann (✉) · R. Singh  
Guru Jambheshwar University Hisar, Hisar 125001, India  
e-mail: [mamtamor12121990@gmail.com](mailto:mamtamor12121990@gmail.com)

International border security, environmental monitoring. The capacity of sensors for processing data is not unlimited as they are mostly battery-operated. But, information gathered from a great number of sensing/detecting nodes at a central location like base stations (BSs) or sink nodes, enable complete and comprehensive demonstration of a given physical environment [1]. Therefore, it can be concluded that the sensors work collectively in large numbers to form a network or to implement an application. This characteristic of efficiently coordinating together even after being so densely deployed is what makes it different from the traditional networks. However, the battery-operated nature of sensor nodes acts as a limiting factor and needs to be addressed efficiently. Therefore, designing an energy-efficient network is of great importance. In context of the seven-layer approach, MAC layer is considered to be the most essential layer for energy conservation. The transceiver part of sensor node responsible for sensing and transmitting data consumes enough energy and is controlled by MAC layer mechanism. It is due to this reason researchers have shown great interest in designing MAC protocol as per the network/application requirement. This paper provides a comprehensive classification of MAC protocols on the basis of their area of application. Such type of classification has not been done so far to the best of our knowledge.

The rest of the paper is organized as follows: Sect. 2 discusses the general challenges and design issues related with MAC protocols. Section 3 provides a detailed literature review. Section 4 discusses the future scope and conclusion.

## 2 Challenges and Design Issues

The MAC protocol is given the responsibility of establishing communication link between sensor nodes efficiently. While designing a MAC protocol certain attributes needs to be considered [2].

### 2.1 Attributes

- **Energy consumption:** As the nodes are battery constrained and mostly placed in environments where they are difficult to replace. Therefore, designing a MAC protocol which is energy-efficient is of significant importance.
- **Quality of service:** To enhance the longevity of the network, the quality of service should not be compromised. The access mechanism designed should be fair.
- **Data delivery performance:** Delay in packet is usually application-dependent. WSN is used for implementing real-time events which need on time reporting of information gathered by sensor nodes to sink node.
- **Self-Stabilizing/Adaptability:** The protocol should be distributed and contention-free which means it should self-stabilize or adapt to the changes in the network like change in topology, node density or traffic characteristics.

- **Channel utilization:** It indicates how efficiently the whole bandwidth is put to use in the course of communication process. Channel utilization is often called bandwidth utilization or channel capacity.

## 2.2 *Issues*

Major sources of energy waste in wireless sensor network are basically of four types [3, 4].

- **Collision:** Interference occurs when multiple nodes access medium at the same time corrupting the data packet. Consequently, the packet needs to be discarded and retransmitted which increases energy consumption and latency also.
- **Overhearing:** It happens when an irrelevant frame is received and decoded by an intended node.
- **Packet Overhead:** Transmission and reception of control packets. For example, RTS packets for alleviating hidden terminal problem consume energy.
- **Idle listening:** It refers to the process of listening to the channel while there is no actual transmission occurring resulting in unnecessary energy dissipation.
- **Over emitting:** When frame is propagated during the time when sink node or target node is inactive resulting in retransmission.
- **Hidden/Exposed Terminal:** It happens when two nodes consequently send data to a third node without knowing about each other degrading throughput.

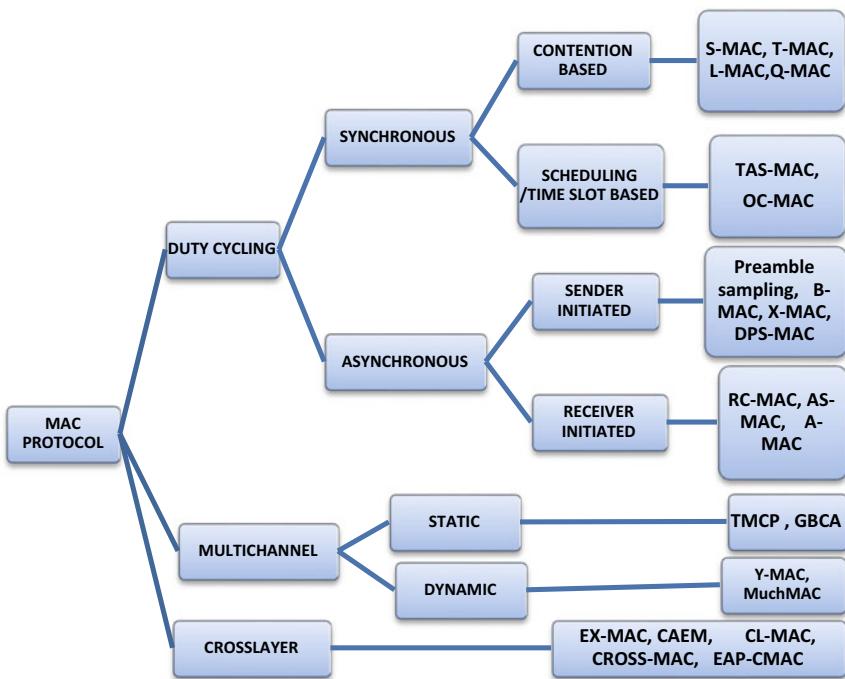
## 2.3 *Classification of MAC Protocol*

MAC protocol design is application-dependent; it can be time critical in nature, while some needs to be energy-efficient. Based on several aspects, the general classification of MAC protocol is presented in Fig. 1 [5, 6]

## 3 Analysis of MAC Protocols on the Basis of Application Area

### 3.1 *MAC Protocol for Vehicular Ad Hoc Network*

Vehicular ad hoc network is a type of MANET in which the vehicles serve the purpose of mobile nodes. As the network topology, recurrently keeps on changing due to high mobility rate of sensor nodes, the conventional contention-based MAC layer implementation 802.11p is not suitable for VANETs. It suffers from problems like collision, hidden/exposed terminal problem and interference [7]. Therefore, several



**Fig. 1** Classification of MAC protocol

MAC protocols have been specifically proposed for VANETs for tackling the above-mentioned issues in an energy-efficient manner (Table 1).

Ali Karabulut et al. [8] proposed **OEC-MAC** which combines the features of orthogonal frequency division multiplexing (OFDMA) with cooperative communication. OFDMA helps in achieving higher data rates and reduced latency which is a prior requirement in VANETs, whereas cooperative communication boosts the communication's quality and reliability by reducing channel impairments due to high mobility rate in VANETs.

**CB\_MAC** [9] is a cluster-based MAC protocol proposed to limit the channel contention by clustering the VANETs into small groups. It eliminates the hidden terminal problem by using RTS/CTS mechanism for non-safety data delivery. It introduces novel control packets for the process of cluster registration, formation and cluster merging which helps in achieving higher throughput and better resource utilization. The limitation of the protocol is that it cannot efficiently handle abrupt increase in number of vehicles, also a drastic increase in traffic leads to higher collision rate and thus decreased throughput.

Li et al. [10] proposed **SC-MAC**, a slotted contention-based multichannel protocol which divides the medium into CCH and multiple SCHs. It overcomes the problem of changing traffic densities efficiently and provides better packet delivery ratio as compared to many other back-off schemes. The protocol provides: (1) reservation

**Table 1** MAC protocols for VANET

Protocol	Type	Technique used	Simulator	Pros
1. OEC-MAC [2020]	OFDMA, cooperative, SINR and distributed	OFDMA, FSM with UML and SINR	MATLAB	High throughput, Lesser PDR
2. CB-MAC [2018]	Clustering, handshake (RTS/CTS Pkts)	Clustering, RTS/CTS for NSM, CH and CM handshake for SM	MATLAB	Higher resource utility and Reduced PDR
3. SC-MAC [2020]	Contention-based (CSMA/CA) + TDMA, hybrid, distributed	Black burst-based slot reservation Spatio-temporal coordination, periodic random back-off scheme	Analysis	Alleviates collision, scalable
4. SAFE-MAC[2019]	Contention-Based (CSMA/CA), synchronous	Batch selection and residence time calculation	Numerical analysis	Reduced latency, better fairness Index
5. RECV-MAC [2021]	Random access, CSMA/CA, cooperative	Markov model analysis, optimal helper selection	MATLAB	Reliability, better PDR

period (RP) and (2) transmission period (TP) for every time slot. A reservation technique for slot assignment is designed in order to ensure that only one node succeeds in broadcasting during any transmitting phase.

A CSMA/CA-based single channel protocol which enhances overall network fairness and adjusts according to speed is designed in [11]. **SAFE-MAC** uses an adaptive back-off algorithm which effectively utilizes the contention window by readjusting it according to the network traffic. The size of the contention window is adjusted on the basis of the speed of the vehicles or their residence time. It aims at achieving proportional fairness by ensuring that every vehicle gets at least one chance to adjust MAC parameters of higher precedence. The residence time of a vehicle is calculated on the basis of certain mobility metrics like speed, location and direction. In addition, the vehicles are grouped into three batches according to the time they spend in a service area. Each batch has its own MAC parameters.

Shah et al. [12] proposed **RECV-MAC** protocol: a reliable, energy-efficient and cooperative protocol. The instability during communication is quite a prevalent problem in VANET due to their highly dynamic behaviour. RECV handles the instability issue by ensuring cooperation amongst the mobile nodes. This is done by exchanging control packets which helps in reducing latency and pkt. drop probability for safety data packets. In order to transmit the data packets securely, a novel algorithm is proposed. The working of the protocol has not been analysed when the number of vehicles increases to a great extent which makes it unsuitable for actual practical scenario. Also, the energy consumed by the whole network has not been considered which is a very crucial aspect for sensors as they have limited energy.

### **3.2 MAC Protocol for Underwater Acoustic Sensor Network**

Underwater sensor network has turned out to be a powerful technology as it has a diverse area of application, for example underwater pipeline monitoring, smart ocean border surveillance, route discovery, navigation assistance, underwater ecosystem data collection for research purpose. But there are several challenges that are related to underwater deployment and working of sensors such as propagation delay, limited bandwidth, node's dynamic behaviour and attenuation.

**GC-MAC** [13] is an energy-conserving, contention-free protocol designed for UWSNs. Based on TDMA approach, it employs graph colouring technique to improve the fairness index and provide optimal number of conflict-free transmissions. Reservation is done by performing distributed clustering up to two-hop neighbour nodes such that unique time slot is allotted to each colour. Nodes assigned similar colour can transmit concurrently without interference/collision thus supporting spatial reuse.

Alak Roy et al. [14] proposed **MOC-MAC**, a multichannel ordered contention-based protocol which is effective even in high traffic density scenarios, limiting collision by using control packets and multichannel communication. On the basis of physical proximity of the horizontal nodes, cluster formation is done. The inter-cluster communication is carried out by cluster heads of each cluster. The access mechanism for nodes of a cluster is handled by using a multiple reservation strategy. The nodes of a cluster are synchronized as they share same schedule. Dual channel communication has been used in which one channel transmits data packets and the other transmits control packets. Transmission is done using synchronized fixed length cycles, where each synchronous cycle has two stages: (a) reservation stage and (b) data transmission stage.

**DADC-MAC** [15] improves network coverage and limits unnecessary interference from adjoining nodes by utilizing a directional antenna instead of an omnidirectional antenna. Similar to MOC-MAC, it also divides the channel into two sub-channels. One of the sub-channels is used for sending data and control messages and the other is used for sharing the current communication status with the neighbours. This helps in reducing the collision rate. The hidden terminal problem and deafness issue which is quite prevalent in UWSNs with directional antennas is handled by using neighbour discovery mechanism and “directional network allocation vector”. The proposed protocol doesn't take into account the switch time between the two different modes of antenna, which cannot be overlooked in real-world scenarios (Table 2).

**UW-ALOHA-QM** [16] improves channel utilization and flexibility by using a topology agnostic method where node trajectories are unpredictable. Reinforcement learning is introduced to allow nodes to cope up with the dynamically changing environment through trial-and-error interaction, and this approach improves overall network resilience. Q-learning technique is used to assist the nodes for choosing unique slots to transfer the data packets. Moreover, a novel uniform random back-off (URB) algorithm to manipulate or adjust start time of each frame is used.

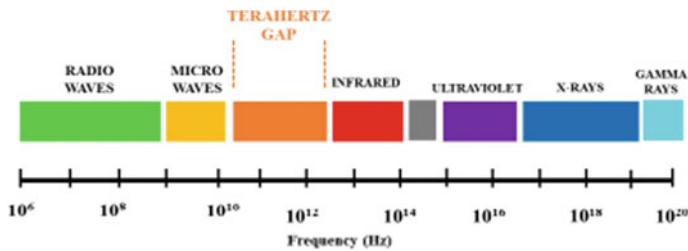
**Table 2** MAC protocols for UWSN

Protocol	Type	Technique used	Simulator	Pros
1. MOC-MAC [2020]	Contention-based, reservation	Multichannel reservation, clustering	NS-2	Collision-free, High data transmission rate
2. GC-MAC [2018]	TDMA-based, contention-free	Graph colouring, Distributed Clustering	Aqua-Sim (NS2-Based)	Better fairness index, less energy consumption
3.DADC-MAC [2020]	Contention-based	Directional antenna, multichannel transmission	NS-3	Less interference, better network coverage
4.UW-ALOHA-QM[2021]	Contention-based, asynchronous	Reinforcement learning, novel back-off scheme	–	Collision-free scheduling, high channel utilization
5.MLQ- MAC [2021]	Hybrid	Quorum system algorithm	JAVA-8, MATLAB	Reduced latency, reduced energy consumption

Sun et al. [17] proposed a **TDMA-based MAC** protocol which improvises the reliability of network by allowing multiple transmissions by different sensors at the same time without increasing collision probability. The goal is achieved by introducing a novel quorum system approach in which different sets of nodes are allocated time slots in a manner that increases system throughput without compromising network reliability. It follows concept similar to [15, 16] where transmission process is divided into two phases. The first phase carries out three operations (a) information gathering of neighbour nodes, (b) distribution of slots and (c) building quorum system. The data packets are transmitted during the next phase.

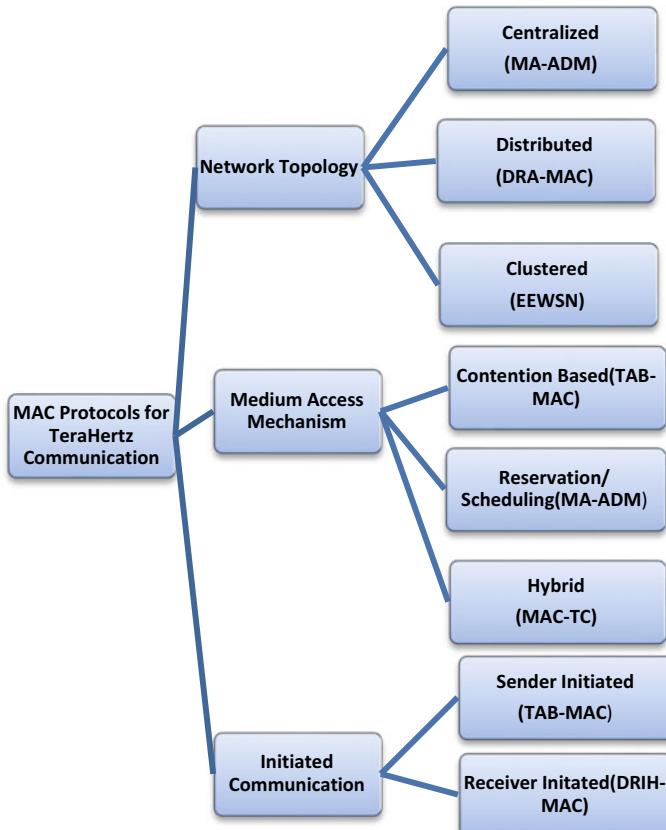
### 3.3 **MAC Protocol for Terahertz Communication**

Terahertz (one trillion cycles per second or 1012 Hz) communication is a powerful evolving technology to support the needs of upcoming ultra-high bandwidth network systems. They are electromagnetic waves (0.1–10 THz) appearing between the microwave and infrared bands within ITU designated band of frequencies (Fig. 2). The terahertz (THz) band commonly known as terahertz gap has the prospective to provide up to Tbps link speed which can adequately fulfil the necessities of fifth-generation (5G)-based network system [18]. The traditional MAC protocols are not



**Fig. 2** ITU designated frequency band

suitable, as they don't take into account the specificity related to terahertz band like multipath, reflection, refraction, path and molecular loss and scattering. Considering the features of terahertz bands, it is important to design MAC protocols fulfilling the requirements of THz network (Fig. 3).



**Fig. 3** Classification of MAC for terahertz network

A memory-guided “angular division multiplexing” MAC protocol (**MA-ADM**) is proposed in [19] for centralized terahertz communication network. The Access point service area is distributed into several angular slots using the 3D quantized angular space. The transmission of data packets is coordinated by the access points in order to improve the efficiency of the network. The data packets are sent with the help of directional antenna of node. Omnidirectional antennas are used to overcome beam alignment issues during the service discovery phase. The introduction of memory guidance in ADM helps in reducing latency and improvising throughput in comparison with other protocols not utilizing memory guidance.

An assisted beamforming access mechanism for THz communication networks (**TAB-MAC**) is proposed in [20] for distributed architecture in which communication is divided into two phases. In phase one, omnidirectional channel is used for controlling and coordinating data transmission, whereas the actual data transfer takes place in phase two only after the nodes have aligned their beams.

Zhang et al. presented **DRA-MAC** [21], which makes use of two radios for supporting simultaneous transmission of packets, thus improving overall network efficiency. Control signals are transmitted by using omnidirectional while transmission of data packets is done using beamforming technique. Control packets such as RTS/TTS are used for initially setting up the connection. Moreover, the network table along with RNAV, AoD and AoA data, is employed for line-of-sight blockage mitigation, throughput maximization and parallel transmission.

A special type of network that performs transmission by utilizing terahertz band is nano-sensor network. Miniature sensor devices often called nano-sensors are densely placed to gather information of physical quantities such as pressure, temperature at nanoscale level. **EEWNSN-MAC** is a light MAC protocol designed to perform efficiently for nano-sensor networks (WNSNs) [22]. All the nano-nodes of the network are divided into different groups representing clusters in order to reduce the collision rate probability. Each cluster has a cluster head also known as nano-router which carries out communication with other cluster's head. The intra-cluster communication is done using TDMA technique, and data packets are multi-hopped to the cluster head for inter-cluster communication. EEWSN effectively reduces the energy consumption of the whole network but does not take into account acknowledgement and retransmission aspect which makes it unpractical for real-world scenario (Table 3).

### 3.4 *MAC Protocol for Unmanned Aerial Vehicle*

Unmanned aerial vehicles (UAVs) are flying object that operates without any pilot. They are usually controlled with the help of a remote base station on ground. UAVs are of paramount importance as they support vast area of applications, for example border surveillance, disaster management, agriculture, aerial photograph and videography, traffic offloading using optimized UAV's trajectory [23]. As UAVs are of great use for real-time sensitive application, therefore it is important to address issues such high

**Table 3** MAC protocol terahertz network

Protocol	Type	Technique used	Simulator	Pros
1. MA-MDM [2017]	Centralized, reservation	Angular division multiplexing, omni + directional antennas	Monte Carlo simulations	Improved throughput, reduced delay
2. TAB-MAC [2016]	Random access	Beam forming	Numerical analysis	Maximize channel utilization
3.DRA-MAC [2021]	Random access	Beamforming, control signalling	Monte Carlo	LoS blockage alleviation, concurrent transmission support
4.EEWSN-MAC[2017]	TDMA	Clustering	NS 3	Scalability, improved PLR
5. MRA-MAC [2017]	Random access	Pulse wave modulation	Monte Carlo	Reduced delay, outage probability

mobility rate, energy efficiency, latency, packet drop ratio while designing MAC protocol.

A collision-free, hybrid CSMA/TDMA protocol **CF-MAC** is proposed in [24] which exploits a region marking scheme in order to alleviate the collision probability, thus improving overall system throughput. The communication process is performed in two stages using half-duplex transceiver and omnidirectional antenna. The sensors have limited processing capacity and employing RTS/CTS control packets to alleviate hidden terminal problem further increases network overhead. CF-MAC efficiently handles the hidden terminal problem using a distributed region marking scheme without transmitting control packets.

**ACD-MAC** proposed in [25] is a hybrid MAC protocol for higher altitude, designed to cope up with the extremely mobile behaviour of UAV network. This is done by dynamically setting the relative length of the two sub-channels interval according to the network density and traffic congestion levels. The arrival rate of date packets is calculated at Poisson rate which is appropriate for UAVs real-world applications. The channel resources distribution is done considering the UAVs density and data traffic within the high altitude platform resulting in reduced end-to-end delay and improved throughput.

**UD-MAC** [26] is a hybrid MAC protocol which employs priority-based scheduling along with a flexible TDMA technique. CNPC links are used by UD-MAC for providing critical security features such as “crash avoidance, real-time control”. Also, the UAVs arriving back to the ground are tracked down with the help of CNPC links containing sensing information. In UD-MAC protocol, a state bit is added in CNPC packets and controlling packets to describe the state of UAV and

**Table 4** MAC protocol for UAV

Protocol	Type	Technique used	Simulator	Pros
1. CF-MAC [2016]	TDMA-based, reservation	Half-duplex radio, region marking scheme	MATLAB	Zero probability of collision, high data transmission rate
2. ACD-MAC [2020]	TDMA + contention-based	CCHI, SCHI	Analysis	Better end-to-end delay, improved throughput
3. UUD-MAC [2020]	Reservation-based	CNPC links	NS-3	Higher access efficiency
4. EF-MAC [2020]	TDMA + CSMA (hybrid)	Clustering, priority access mechanism	MATLAB	Reduced latency, energy-efficient
5. HP-MAC [2019]	CSMA/CA, reservation	Beaconing,	MATLAB	Enhanced fairness, improved PDR

discover returning. Moreover, an adaptive access time slots assignment scheme is designed in order to achieve orderly and reliable data transmission (Table 4).

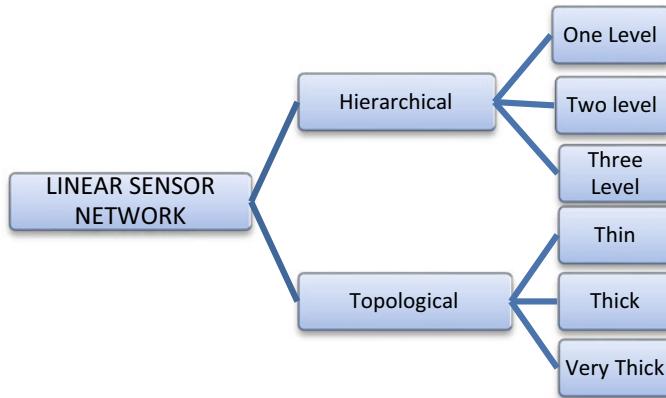
A hybrid MAC **EF-MAC** protocol for time-sensitive applications is proposed in [27] which utilizes both TDMA and CSMA for increasing network overall efficiency and minimizing delay. The initial process of registration is carried out using CSMA while data transmission is done through slot reservation using TDMA. A novel priority access mechanism is designed to enhance fairness of the network in which priority is assigned to the cluster head on the basis of data buffer size, remaining energy and CDT.

A MAC protocol **HP-MAC** [28] which is also a hybrid protocol utilizing CSMA/CA and TDMA for fairness, also enhancing network performance in terms of packet delivery ratio and throughput. During the first phase, the UAVs are detected and the registration process of unidentified nodes is carried out. In the next period, the scheduling process information is forwarded to every registered node and thereafter time slots are allocated to the registered nodes in last phase.

### 3.5 MAC Protocol for Linear Sensor Network

A linear sensor network is a special category of wireless sensor network in which the sensor nodes are either placed in a straight line or randomly deployed between two parallel lines. LSNs can be used for border monitoring and surveillance, offshore underwater gas or fuel pipeline monitoring, road accident or high traffic alert, etc. The MAC protocol designing needs to consider the specificity related to LSNs such as energy-hole problem and linear topology in order to avoid premature termination of network. The general taxonomy of linear sensor is presented in Fig. 4 [29].

Sokullu et al. [30] proposed **LINE-MAC**, a MAC protocol modified specially for LWSN to overcome challenges faced due to the linear topology of a sensor



**Fig. 4** Classification of linear sensor network

network. It is specifically suitable for time-bound WSN applications having linearity in structure. The two major challenges faced in LSN are: successful source-to-sink data delivery and reducing end-to-end delay due to multihop communication. As the number of possible routes to reach the destination is very limited in LSN, due to this the chances of data packets delivery failure are much higher than traditional WSNs. Also, the bottleneck problem in the proximity of sink node is quite common as the data gets accumulated over the nodes nearer to sink resulting in “relay burden problem”. This leads to deterioration of the communication system due to increased energy consumption and increased delay. To improve the end-to-end delay, the proposed protocol set the number of successive packets to be sent as per the traffic conditions by means of a parameter called “packetLimit” which can be dynamically changed.

**SA-MAC** protocol designed in [31] is a synchronous duty cycling MAC protocol which uses staggered or pipelined scheduling to ensure that the data packet reaches sink node in a single cycle. The whole network has been divided into several grades in order to efficiently perform pipelining. This is a multi-hop communication network in which nodes at higher grade relay data to the nodes at the lower grades. SA-MAC further optimizes the duty cycling process by selectively awakening the nodes in a grade according to traffic load, thus stabilizing the contending nodes density. The awakening probability for contending nodes is calculated dynamically in such a manner that the throughput function is maximized. Also, the nodes with saturated buffers are kept in sleeping mode at the time of receiving data from higher grades. A mathematical model of a node’s buffer state based on DTMC is designed for evaluating the performance of SA-MAC. The proposed protocol provides better throughput than PRI-MAC in high node density condition. However, SA-MAC does not take into account the heterogeneous behaviour exhibited by different grades of LSNs.

Hashing-based medium access protocol (**H-MAC**) is a contention-free, higher throughput yielding MAC protocol [32] for LSN. Similar to the scheme proposed

in [31], the LSN is divided into grades based on nodes located almost at a similar distance from the sink or destination node. The proposed protocol uses hashing technique to improve the fairness index of the network. For this purpose, hash function is used to perform a distributed election with the help of which a collision-free access mechanism takes place every time. The priority number for each contending node is calculated at the starting of transmission process. The additional overhead of control packets (RTS/CTS) is eliminated, at the cost of little extra local processing.

**LC-MAC** has been proposed in [33] for networks in which sensors are deployed linearly over a longer distance. It is a synchronous, duty cycling MAC protocol which support burst data transmission, thereby reducing source-to-sink delay, the most common issue in linear sensor network. The multiple data packets transmission is done by introducing a novel frame “superSYNC” which multihops through all grades and shares scheduling information of upcoming packets from higher grades. This way the channel can be kept free by sharing data transmission information prior sending, thus assuring continuous data forwarding as the downstream nodes adjust their sleep/wake-up schedule according to the data arrival time from the upstream nodes. Extensive simulation has displayed that LC-MAC outperforms S-MAC in terms of end-to-end delay without sacrificing energy efficiency or network throughput (Table 5).

The energy-hole problem which is quite prevalent in linear sensor network leads to decreased network life span. **RDCPF** is proposed in [34] is a duty cycling, pipelining-based protocol which effectively handles the bottleneck problem arising due to excessive energy consumption near sink node. In order to even out the energy consumption of the network, redundant nodes are deployed which only act as relay nodes. To avoid the complexity due to redundancy, a topology optimization algorithm is proposed which assures stability even after increased number of grades.

**Table 5** MAC Protocol for LSNs

Protocol	Type	Technique used	Simulator	Pros
1. LINE-MAC [2014]	Random access	Preambling (PreAck, Packetlimit)	Castalia	Improved PDR, reduced latency
2. SA-MAC [2019]	Synchronous, contention-based	Pipelined scheduling, selective awakening	MATLAB	Reduced collision, improved throughput
3.H-MAC [2020]	Synchronous, contention-based	Hashing, pipelined scheduling	MATLAB	Collision-free, uniform performance
4.LC-MAC [2011]	Staggered scheduling	Location detection packet, message passing	NS-2	Better traffic delivery, energy efficient
5. RDCPF-MAC [2020]	Synchronous, staggered scheduling	Topology optimization, pipelining	OMNET++	Improved AEC, reduced source to sink delay

## 4 Conclusion

The advanced technologies-based system be it UAV, VANET, LSNs or terahertz N/W will certainly transform the world and its economy in the coming years. Integrating such systems into day-to-day activities will undoubtedly create a sustainable business scenario. The study that we have proposed is relevant from the perspective that energy conservation is one of the most challenging issues in current scenarios of wireless sensor network. The medium access control (MAC) protocol is seen as an important part as it handles the coordination and communication amongst the sensor nodes by controlling access to the channel. Therefore, for a WSN to provide high throughput, designing an efficient access mechanism is of paramount importance. The analysis of various MAC protocols reflects the varying performance output based on application area. Therefore, a single MAC protocol cannot be considered as an ideal solution for all UWSN-based real-world applications. Existing literature covered the work done for WSN optimization by using energy-efficient MAC protocol but still there is a lot of possibility of further research for LSN optimization. Research needs to be done keeping in mind the linearity of LSNs as it may lead to several new areas of interest. The protocols designed so far are mostly application-specific and do not provide a general solution to energy-hole problem frequently faced in LSNs. Therefore, several objective functions like packet drop probability, source-to-sink delay, energy consumption and channel utilization can be exploited in such a way that the desired parameters performance is further improved leading to the optimization of linear sensor network.

## References

1. Ye, W., Heidemann, J., Estrin, D.: An energy-efficient MAC protocol for wireless sensor networks. In: Proceedings of the IEEE INFOCOM, pp. 1567–1576 (2002)
2. Demirkol, I., Ersoy, C., Alagoz, F.: MAC protocols for wireless sensor networks: a survey. *IEEE Commun. Mag.* **44**(4), 115–121 (2006)
3. Mishra, B., Mohindru, V., Singh, Y.: Comparative analysis of energy efficient secure MAC protocols for wireless sensor network. *J. Basic Appl. Eng. Res.* **1**(6), 13–18 (2014)
4. Djimli, A., Merniz, S., Harous, S.: Energy-efficient MAC protocols for wireless sensor networks: a survey. *TELKOMNIKA Telecommun. Comput. Electron Control* **17**(5), 2301–2312 (2019)
5. Huang, P., Xiao, L.: The evolution of MAC protocols in wireless sensor networks: a survey. *IEEE Commun. Surv. Tutorials* **15**(1) (2013)
6. Yang, O., Heinzelman, W.: Modeling and performance analysis for duty-cycled MAC protocols with applications to S-MAC and X-MAC. *IEEE Trans. Mob. Comput.* **11**(6), 905–920 (2012)
7. Hota, L., Nayak, B.P., Kumar, A.: An analysis on contemporary MAC layer protocols in vehicular networks: state-of-the-art and future directions. *Future Internet* (2021)
8. Karabulut, A.M., Shahen Shah, M.F.A., Ilhan, H.: OEC-MAC: a novel OFDMA based efficient cooperative MAC protocol for VANETS. *IEEE Access* **8**, 94665–94677 (2020)
9. Karabulut, A.M., Shahen Shah, M.F.A., Ilhan, H.: CB-MAC: A novel cluster-based MAC protocol for VANETs. *IET Intell. Transp. Syst.* **13**, 587–595 (2019)

10. Li, S., Liu, Y., Wang, J., Sun, Z.: SCMAC: a slotted-contention-based media access control protocol for cooperative safety in VANETs. *IEEE Internet Things J.* **7**, 3812–3821 (2020)
11. Siddik, M.A., Moni, S.S., Alam, M.S., Johnson, W.A.: SAFE-MAC: speed aware fairness enabled MAC protocol for vehicular ad-hoc network. *Sensors* **19** (2019)
12. Shahen Shah, M.F.A., Ilhan, H., Tureli, U.: RECV-MAC: a novel reliable and efficient cooperative MAC protocol for VANETs. *IET Commun.* **13**, 2541–2549 (2019)
13. Alfonzanz, F., Shahabi, A., Ghoreyshi, S.M., Boutaleb, T.: Graph colouring MAC protocol for underwater sensor networks. In: Proceedings of the 2018 IEEE 32nd international conference on advanced information networking and applications (AINA), pp. 120–127 (2018)
14. Roy, A., Sarma, N.: Multichannel ordered contention MAC protocol for underwater wireless sensor networks. *Comput. J.* (2020)
15. Yang, J., Qiao, G., Hu, Q., Zhang J.: A dual channel medium access control (MAC) protocol for underwater acoustic sensor networks based on directional antenna. *Symmetry* (2020)
16. Park, S.H., Mitchell, P.D., Grace, D.: Reinforcement learning based MAC protocol (UW-ALOHA-Q) for underwater acoustic sensor networks. *IEEE Access* **7**, 165531–165542 (2019). <https://doi.org/10.1109/ACCESS.2019.2953801>
17. Sun, N., Wang, X., Han, G., Peng, Y., Jiang, J.: Collision-free and low delay MAC protocol based on multi-level quorum system in underwater wireless sensor networks. *Comput. Commun.* **173**, 56–69 (2021)
18. Paul A. et al.: A neuro-Fuzzy based IDS for internet-integrated WSN. In: Bansal J.C., Paprzycki M., Bianchini M., Das S. (eds.) Computationally Intelligent Systems and their Applications. Studies in Computational Intelligence, vol. 950. Springer, Singapore (2021). [https://doi.org/10.1007/978-981-16-0407-2\\_6](https://doi.org/10.1007/978-981-16-0407-2_6)
19. Han, C., Tong, W., Yao, X.-W.: MA-ADM: a memory-assisted angular-division multiplexing MAC protocol in terahertz communication networks. *Nano Commun. Network* **13**, 51–59 (2017)
20. Yao, X.-W., Jornet, J.M.: TAB-MAC: assisted beamforming MAC protocol for terahertz communication networks. *Nano Commun. Netw.* **9**, 36–42 (2016)
21. Zhang, X., Han, C., Wang, X.: Dual-radio-assisted (DRA) MAC protocols for distributed terahertz networks. *IEEE Open J. Veh. Technol.* **2**, 111–124 (2021)
22. Rikhtegar, N., Keshtgari, M., Ronaghi, Z.: EEWNSN: energy efficient wireless nano sensor network MAC protocol for communications in the terahertz band. *Wireless Pers. Commun.* **97**(1), 521–537 (2017)
23. Poudel, S., Moh, S.: Medium access control protocols for unmanned aerial vehicle-aided wireless sensor networks: a survey. *IEEE Access* **7**, 65728–65744 (2019). <https://doi.org/10.1109/ACCESS.2019.2917948>
24. Jiang, A., Mi, Z., Dong, C., Wang, H.: CF-MAC: a collision-free MAC protocol for UAVs ad-hoc networks. In: Proceeding of IEEE Wireless Communications and Networking Conference (2016)
25. Ruan, Y., Zhang, Y., Li, Y., Zhang, R., Hang, R.: An adaptive channel division MAC protocol for high dynamic UAV networks. *IEEE Sens. J.* **20**(16), 9528–9539 (15 Aug 2020). <https://doi.org/10.1109/JSEN.2020.2987525>
26. Liu, X., Wei, Z., Feng, Z., Ning, F.: UD-MAC: delay tolerant multiple access control protocol for unmanned aerial vehicle networks. In: 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), pp. 1–6 (2017). <https://doi.org/10.1109/PIMRC.2017.8292602>
27. Banerjee, A., et al.: Construction of effective wireless sensor network for smart communication using modified ant colony optimization technique. In: Bianchini M., Piuri V., Das S., Shaw R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_22](https://doi.org/10.1007/978-981-16-2164-2_22)
28. Ramli, M.R., Lee, J.-M., Kim, D.-S.: Hybrid mac protocol for UAV-assisted data gathering in a wireless sensor network. *Internet of Things* **100088** (2019)

29. Jawhar, I., Mohamed, N., Agrawal, D.P.: Linear wireless sensor networks: classification and applications. *J. Netw. Comput. Appl.* **34**, 1671–1682 (2011)
30. Sokullu, R., Demir, E.: Investigating energy efficiency and timeliness for linear wireless sensor networks with dense topologies. *Procedia Computer Science Elsevier* **37**(1), 24–31 (2014)
31. Villordo-Jimenez, I., Cruz, N.T., Carvalho, M.: A selective-awakening MAC protocol for energy-efficient data forwarding in linear sensor networks. *Wirel. Commun. Mob. Comput.* **2018**, 1 (2018)
32. Villordo, I., Menchaca, R., Rivero, M.E., Torres, N.: An energy-efficient hash-based MAC protocol for LSN. In: IEEE 10th Latin-American Conference on Communications (LATINCOM) (2018)
33. Fang, C., Liu, H., Qian, L.: Lc-mac: An efficient mac protocol for the long-chain wireless sensor networks” In Proceedings of the Third International Conference on Communications and Mobile Computing (CMC), pp. 495–500 (Apr 2011)
34. Zhang, Q., Li, D., Fei, Y.: RDCPF: A Redundancy Based Duty Cycling Pipelined—Forwarding MaC for Linear Sensor Networks, vol. 20, no. 19 (2020)

# A Framework for Lung Cancer Detection Using Machine Learning



Aakash Nakarmi, Anil Kumar Sagar, Seerat Musharaf, and Hadya Jahangir

**Abstract** Cancer is a lethal disease that has been ruling the world for the death of millions of people every year. It causes abnormal and uncontrollable growth of cells and destroys the body tissues and so similarly, in lung cancer, cells in the lungs grow uncontrollably. So, the detection of lung cancer is a must as we also know that the earlier the detection, the survival rate of the victim also increases accordingly. In many cases, cancer is detected lately and the chances to save the patient's life also diminishes. The figure of persons diagnosed with lung cancer is directly proportionate to the figure of chain smokers and other hazardous addictions. The prediction of cancer and its prognosis has been a tedious task for doctors and medical workers since cancer's discovery. Among the various applied technologies, supervised machine learning is a very relevant option in which a particular model is trained to work on the complex datasets, containing different features, and then the prediction is made accordingly. There will be the usage of different techniques and algorithms (like K-nearest neighbors, decision trees, naïve Bayes, kernelized support vector machines, and logistic regression) to work on the labeled datasets. Adding up, the model basically focuses on two foci: cancer susceptibility and model accuracy.

**Keywords** Cancer detection · Supervised machine learning · Labeled datasets · KNN · Logistic regression · Naïve Bayes · Support vector machine

---

A. Nakarmi (✉) · A. K. Sagar · S. Musharaf · H. Jahangir

School of Engineering and Technology, Sharda University, Greater Noida, India  
e-mail: [aakasnakarmee@gmail.com](mailto:aakasnakarmee@gmail.com)

A. K. Sagar  
e-mail: [anil.sagar@sharda.ac.in](mailto:anil.sagar@sharda.ac.in)

S. Musharaf  
e-mail: [sheikhdayima@gmail.com](mailto:sheikhdayima@gmail.com)

H. Jahangir  
e-mail: [hadyajehangir066@gmail.com](mailto:hadyajehangir066@gmail.com)

## 1 Introduction

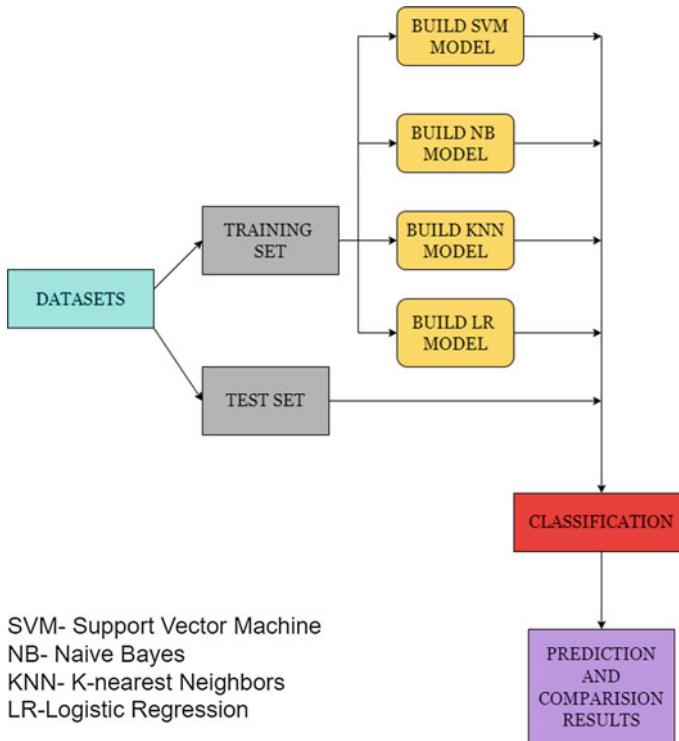
In most of the cancer-death cases, the lung cancer is the major case as this cancer directly affects the respiratory system of our body. Lung cancer mostly affects to the people who consume hazardous stuffs like tobacco and cigarette. Including that, it is also affected by the intake of the excessive polluted air, different chemicals and it is obvious that the people with some lung disease like emphysema are likely to be the victim. The stage of the cancer depends on the spread of the size of the abnormal growth of the cancer cells. When the cancer/malignant cells start expanding and affect the other organs, then it is cautious and critical [1]. That is why, it is very important for prognosis of the cancer at an early stage for the cure of cancer; otherwise, it is unfortunate that life span of the patient is shortened and limited.

Different techniques and methods have been developed over the past decades. Supervised machine learning is one of them. Machine learning is not a new domain in this field of cancer detection and prognosis. Including cancer detection, this technology has been applied in various fields of medical sector like X-ray and CRT image identification [2]. Machine learning has evolved by a huge margin in the past few decades and so it is playing a very important role in cancer prediction and the researchers are continuously exploring the algorithms to make the model more and more accurate. In this paper, different classification machine learning algorithms are discussed for lung cancer detection. The main aim is the detection of the presence of lung cancer and the stage of it, if it is present so that the radiologist can effectively and efficiently proceed for the diagnosis of the patient [3]. This will contribute to early enhancement of detection (Fig. 1).

The diagram depicts the overall structure of the flowchart of the model. The datasets of the lung cancer required for this study can be taken from Kaggle Data Science Bowl and the UCI Machine Learning Repository and Data World [4]. Initially, the provided labeled datasets are split into training datasets and test datasets using the K-fold cross-validation method. Then by the usage of different techniques and algorithms like SVM, naïve Bayes, logistic regression, and decision tree, the model is trained using the retrieved datasets under some parameters, respectively. Then, by using the test data, the trained models are evaluated to determine the accuracy [5]. The better the accuracy, the convenient to detect the stage of the lung cancer.

## 2 Literature Review

Supervised machine learning makes the usage of the AI software to train the machine/model from the previously labeled data and predict the outcome on the test datasets. The AI technology has evolved in a vast margin, and till now many research works have been done for the lung cancer detection using the supervised



**Fig. 1** Overall architecture

machine learning algorithms. A summarized explanation of the previous research works is discussed below:

Günaydin et al. [6] focus on the comparison of the different classification algorithms on the basis of their accuracy and variation. It makes use of the prognosis of the life expectancy of the patients of the lung cancer using different algorithms like artificial neural network, naïve Bayes, and decision trees. For every single classifier, the precision was tested. Abdullah and Ahmed et al. [7] deal with the implementation of different classification algorithms on the lung cancer datasets and make the comparison among them with respect to the accuracy of that particular algorithm. The classifiers were KNN, logistic regression, NN, and SVM. SVM had the highest accuracy among those classifiers. This method aided the doctors in predicting the case of lung cancer with better precision. Sharma et al. [8] discussed which algorithms are the most effective at detecting brain tumor. Using volumes and locations, there is calculation of the overall precision rate by using two classifier classes, logistic regression and decision tree, and three classifier classes, linear SVM, Gaussian SVM, cosine KNN, and complex and median trees. Alam et al. [9] discussed the various segmentation algorithms like naïve Bayes, hidden Markov model, and some more.

It describes the implementation of these segmentation models' role in the detection of lung tumor. It shows a different approach for the prognosis of cancer. Patra et al. [10] made a comparative study of supervised machine learning models which involved KNN, radial basis function network, etc. Among those proposed models, RBF model showed up with the highest accuracy and was considered as the most effective classifier algorithm. Ismail et al. [11] depicted how we can generate a system flow diagram for an algorithm which can be utilized to detect the lung tumor. There are various classification methods and other methods regarding the detection of lung cancer which are listed below:

- Classifiers: logistic regression, SVM
- Compression of data techniques: decision tree, artificial neural network
- Description about different sets of data like PLCO dataset

### 3 Classification Algorithms

#### 3.1 *K-Nearest Neighbor*

It is a machine learning technique which is used for the classification of different classes according to features of the classes. In this particular algorithm, the model finds similarity among the features of the dataset and then, a category is established. In this technique, there are no certain parameters for affecting the classification, but the number of k-neighbors affects the classification [11]. Also, the fitting of data matters the classification. When there is under fitting or overfitting of dataset, there may be deflection in the classification.

In this model, the Euclidean distance determines the distance between the test data point and the k-nearest neighbor (closest class instance). And the closest class instance determines the classification. If  $A(x_1, y_1)$  and  $B(x_2, y_2)$  are two data points, then Euclidean distance (ED) is calculated by:

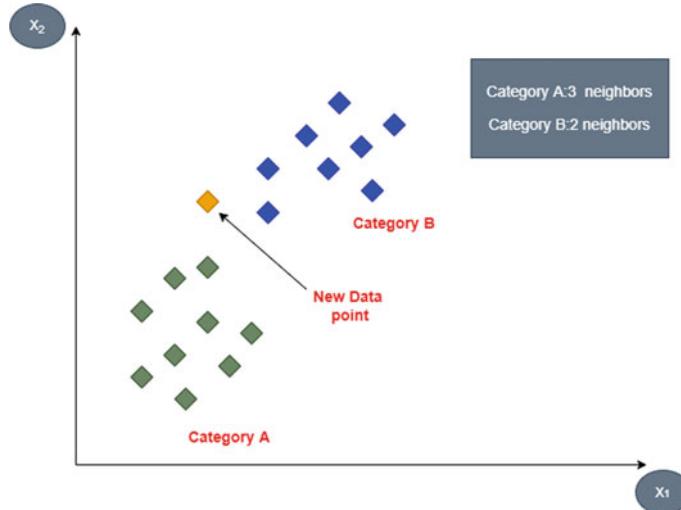
$$ED = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Formula 1. Euclidean distance calculation.

Figure 2 depicts a typical model of K-nearest neighbor. When there are 5 number of k-neighbors, 3 instances of A category are closest to test data point and so the test data is classified to category A.

#### 3.2 *Naïve Bayes*

Naïve Bayes is a probabilistic classifier which is used in the sector of machine learning. In this algorithm, we use different equations of probabilities to classify the



**Fig. 2** K-nearest neighbors diagram

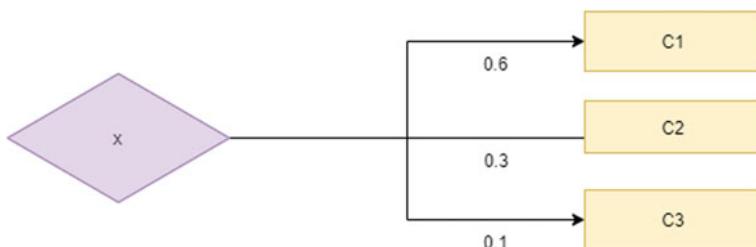
dataset. Some equations of probabilities used in naïve Bayes are discussed below:

$$\frac{P\left(\frac{K1}{X}\right)}{P\left(\frac{K2}{X}\right)} > 1$$

$$\frac{P\left(\frac{K1}{X}\right)}{P\left(\frac{K2}{X}\right)} = \frac{\frac{P\left(\frac{X}{K1}\right)*P(K1)}{P(x)}}{\frac{P\left(\frac{X}{K2}\right)*P(K2)}{P(x)}}$$

Formula 2. Bayes' theorem for comparison of probabilistic values.

Firstly, the probabilistic value of each instance is found out and then the result/final outcome is decided by the instance having the highest probabilistic value [9]. In the diagram below, it is clear that for the incoming instance X, D1 has the greatest value of probability and so the final class for X will be D1 (Fig. 3).



**Fig. 3** A demo data and its probability on different classes

### 3.3 Logistic Regression

Logistic regression is a linear model for classification in supervised machine learning which works on a logistic function. This mathematical model works for the analysis of datasets related to epidemiology, mostly in the field of machine learning.

The logistic function used in this model is implemented in the following way:

1. Firstly, evaluate using logistic function
2. Get to know about the coefficients in the regression model
3. Eventually, the predictions are made with the help of the linear regression model.

$$f(x) = \frac{L}{1 + e^{-k(x-x_0)}}$$

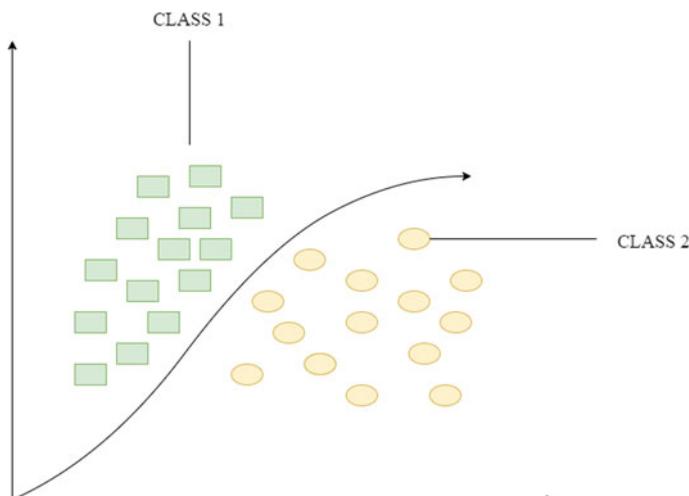
Formula 3. Logistic function equation.

- $E$  = Euler's number
- $x_0$  = middle  $x$ -value of sigmoid function
- $L$  = The maximum value of the curve
- $K$  = Abruptness of curve

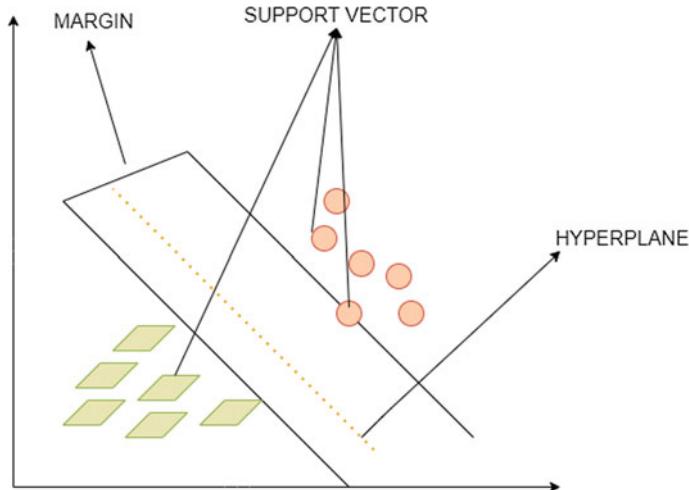
$$y = (e^{b_0+b_1+x}) / (1 + e^{b_0+b_1+x})$$

Formula 4. Equation for LR Model.

Different parameters also affect the logistic model. The data fit in the linear regression model is regularized by the parameters which controls the overfitting, underfitting, and perfect fitting of linear model (Fig. 4).



**Fig. 4** Logistic regression to discuss two classes



**Fig. 5** Structure of support vector machine

### 3.4 Support Vector Machine (SVM)

Support vector machine is a supervised learning technique which is used for classification purpose. Mainly, it is flexible for the nonlinear datasets and large datasets. The aim in this model is to calculate the least and maximum distanced position [12]. The diagram Fig. 5 depicts the support vector machine Model. In the figure, two variant classes represent two different features. The hyperplane divides the classes and along with that, the support vectors and margin are also properly labeled.

## 4 Evaluation and Accuracy of Algorithms

The labeled datasets can be retrieved from the Kaggle data bowl. In the repository, we have used, 59 instances are present in the dataset, and there are various features like alcohol consumption and smoking. The description of the features of dataset contains that the features are the main factors responsible for proper classification of the stage of lung cancer. There are various features responsible for this like age, gender, air pollution, lung disease, proper sleep and diet, obesity, habit of smoking, severe chest pain, and many more are provided as aspects for the prognosis of lung cancer. If the lung cancer is at a malignant stage, the intensive value in label is ‘2’, benign stage—‘1’ and healthy person (no tumor)—‘0’.

The accuracies of the different classification algorithms are mentioned below (Table 1):

From the table, we can observe that for K-nearest neighbors, the accuracy is very poor. However, the accuracy of other 3 applied models has an accuracy of 100%.

**Table 1** For applied model and its accuracy

Applied model	Accuracy
Logistic regression	100
K-nearest neighbors	67
Naïve Bayes	100
SVM	100

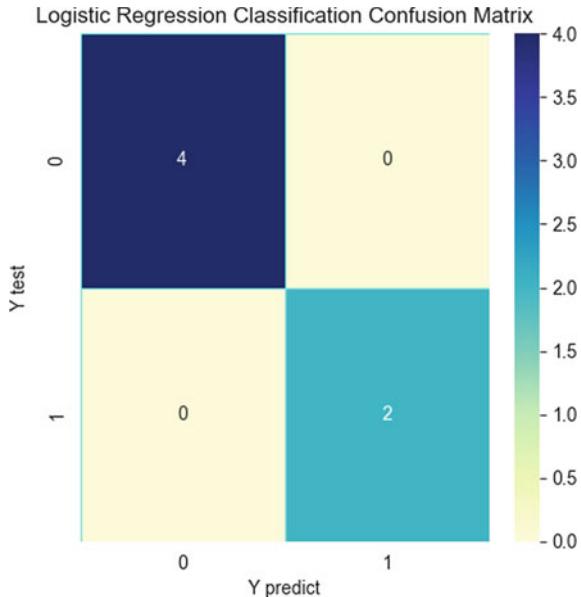
The precision and accuracy of the applied models can be calculated with the help of the confusion matrices. For particular models, there are respective confusion matrices. From the confusion matrices, we can also observe the prediction for lung cancer detection.

Figure 6 depicts the confusion matrix for the logistic regression. From the matrix, we can observe that the model is correctly predicting 4 patients as lung cancer patients and 2 as healthy persons. There is no case of any false prediction.

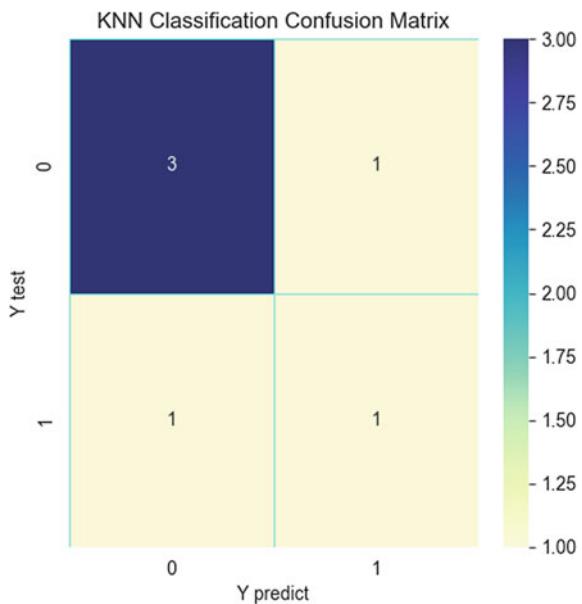
Among the 4 applied algorithms, the KNN model is the poorest in accuracy as it made false predictions for healthy and cancer patients. We can clearly observe from Fig. 7 of confusion matrix. This model predicted one cancer patient as healthy and a healthy person as cancer patient.

Figure 8 clearly describes that naïve Bayes model has a very good prediction status for that particular dataset as the prediction also depends on the dataset's variance. For all the 6 test data instances, the predictions are made correctly. There is no any false prediction.

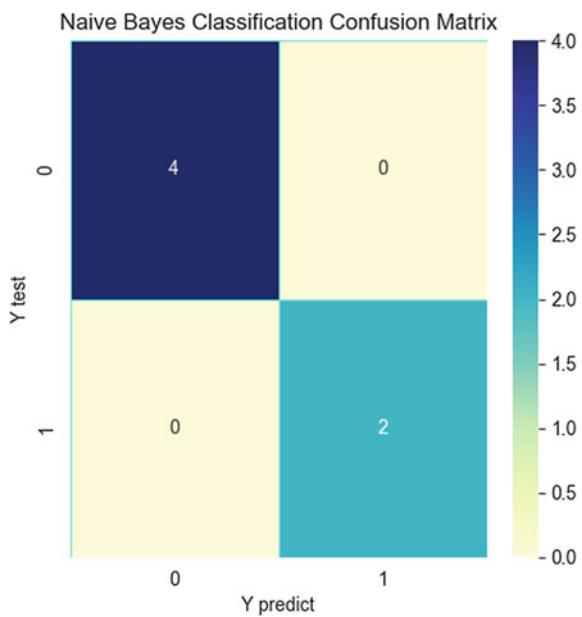
**Fig. 6** Confusion matrix for logistic regression model



**Fig. 7** Confusion matrix for K-nearest neighbor



**Fig. 8** Confusion matrix for Naïve Bayes



**Fig. 9** Confusion matrix for support vector machine

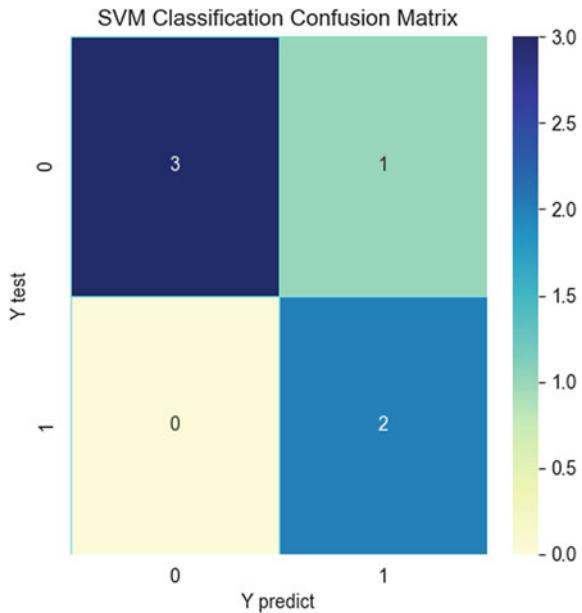


Figure 9 depicts that SVM is full accurate for this particular dataset. By the means of dimensionality, support vector machine has an extra edge over the other algorithms. Clearly, it can be seen that it did not make any false prediction for the 6-test data.

Also, from the table of accuracy also, we can clearly sum up that the KNN is the least accurate among all the other algorithms. However, for this particular dataset, we can observe that the performance of all other algorithm was perfect as they did not make any false prediction and made exact prediction as cancer patient which is the most important aspect.

## 5 Conclusion

Cancer is known to be one of the deadliest diseases. Earlier, the patient would have to go through multiple tests to detect the presence of cancer cells in the lung. The patient has to go through various unnecessary check-ups during the process of diagnosis. Reduction of these inessential steps and process is possible by the use of machine learning. Use of machine learning algorithms such as K-nearest neighbor, logistic regression, support vector machine, and naïve Bayes to train the model for the detection of lung cancer helps us to reduce the effort and improve the accuracy and precision. A comparative analysis is done among the classification algorithms. From the accuracy table among the classification algorithms, support vector machine, logistic regression, and naïve Bayes were very effective algorithms with the maximum accuracy. However, K-nearest algorithm showed a disappointing

accuracy. All in all, machine learning has a very effective result for lung cancer detection.

## References

1. Cruz, J.A., Wishart, D.S.: Applications of machine learning in cancer prediction and prognosis. *Cancer Informatics*. (2006). <https://doi.org/10.1177/117693510600200030>
2. Radhika, P.R., Nair, R.A.S., Venna. G.: A comparative study of lung cancer detection using machine learning algorithms. In: 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), pp. 1–4 (2019). <https://doi.org/10.1109/ICECCT.2019.8869001>
3. Tekade, R., Rajeswari, K.: Lung cancer detection and classification using deep learning. In: 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA) (2018), pp. 1–5. <https://doi.org/10.1109/ICCUBEA.2018.8697352> (Rado, Suhi, H. (eds.) Academic, New York, pp. 271–350) (1963)
4. Kadir, T., Gleeson, F.: Lung cancer prediction using machine learning and advanced imaging techniques. *Transl. Lung Cancer Res.* **7**(3), 304 (2018)
5. Faisal, M.I. et al.: An evaluation of machine learning classifiers and ensembles for early stage prediction of lung cancer. In: 2018 3rd International Conference on Emerging Trends in Engineering, Sciences and Technology (ICEEST). IEEE (2018)
6. Günaydin, Ö., Günay, M., Şengel, Ö.: Comparison of lung cancer detection algorithms. *Sci. Meeting Electr. Electron. Biomed. Eng. Computer Sci. (EBBT)* **2019**, 1–4 (2019). <https://doi.org/10.1109/EBBT.2019.8741826>
7. Abdullah, D.M., Ahmed, S.W. (2021): A review of most recent lung cancer detection techniques using machine learning. In: International Journal of Science and Business, IJSAB International, vol. 5(3), pp. 159–173. M. Young, The Technical Writer’s Handbook. University Science, Mill Valley, CA (1989)
8. Mridha, K. et al.: Web based brain tumor detection using neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 137–143 (2021). <https://doi.org/10.1109/ICCCA52192.2021.9666248>
9. Alam, J., Alam, S., Hossan, A.: Multi-stage lung cancer detection and prediction using multi-class SVM classifier. In: 2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2). IEEE (2018)
10. Patra, R.: Prediction of lung cancer using machine learning classifier. In: International Conference on Computing Science, Communication and Security. Springer, Singapore (2020)
11. Ismail, M.B.S.: Lung cancer detection and classification using machine learning algorithm. *Turkish J. Computer Math. Educ. (TURCOMAT)* **12**(13), 7048–705 (2021)
12. Wu, Q., Zhao, W.: Small-cell lung cancer detection using a supervised machine learning algorithm. In: 2017 International Symposium on Computer Science and Intelligent Controls (ISCSIC). IEEE (2017)

# Efficient Diagnosis of Alzheimer's Disease Using EfficientNet in Neuroimaging



H. Sharen, B. Dhanush, P. Rukmani, and D. Dhanya

**Abstract** Alzheimer's disease is a progressive disease that weakens mind's memory and overall functioning. In the early identification of Alzheimer's disease, neuroimaging is increasingly being utilized to support clinical examinations (AD). One of the most commonly utilized and promising modalities for detecting brain abnormalities in persons who may be at risk for AD but have not yet exhibited symptoms is structural magnetic resonance imaging (MRI). In this study, a transfer learning model called EfficientNetB7 architecture is analyzed to enhance the prediction with pre-trained weights in a benchmark dataset of neuroimages. The network is further fine-tuned via layer-wise tuning, which involves training a pre-defined set of layers using MRI images. The performance of the proposed system is evaluated over the Kaggle brain MRI dataset that includes four classes such as mild demented, moderately demented, non-demented, and very mildly demented. For AD classification, the proposed trained model achieves enhanced accuracy and F1-score as 89.7% and 0.91, respectively.

**Keywords** Deep learning · Transfer learning · MRI · Neuroimaging · Artificial intelligence · Alzheimer's disease · EfficientNet

---

H. Sharen (✉) · B. Dhanush · P. Rukmani

School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India  
e-mail: [sharen.h2020@vitstudent.ac.in](mailto:sharen.h2020@vitstudent.ac.in)

B. Dhanush

e-mail: [dhanush.b2020@vitstudent.ac.in](mailto:dhanush.b2020@vitstudent.ac.in)

P. Rukmani

e-mail: [rukmani.p@vit.ac.in](mailto:rukmani.p@vit.ac.in)

D. Dhanya

Mar Ephraem College of Engineering and Technology, Marthandam, India

## 1 Introduction

Alzheimer's disease is a neurologic disorder that causes the brain to shrink (atrophy) and brain cells to die. Alzheimer's disease is the most prevalent type of dementia, considered as a gradual loss of cognitive, behavioral, and social abilities that make it difficult for a person to work independently. Most illnesses attack the body, while Alzheimer's disease attacks the mind, obliterating the self in the process. Alzheimer's disease affects people of all ages and backgrounds. It is a chronic, irreversible brain condition for which there is currently no effective treatment. Medicines that are now available, on the other hand, may be able to slow down the progression of the disease. As a result, early detection of Alzheimer's disease is critical for preventing its progression. The major goal is to provide an end-to-end framework for early Alzheimer's disease detection and medical image classification for various stages of the disease.

Deep learning algorithms have attained state-of-the-art performance in a wide range of areas, including medical image processing, where magnetic resonance imaging (MRI) is analyzed to diagnose Alzheimer's disease accurately and precisely. Convolutional neural networks are used for image data processing because of their ability to handle the implementation of large of unstructured data and extract essential features automatically [1–3].

Traditional approaches rely on manual feature extraction, which looks to be time-consuming and subjective and relies largely on technical skill and repeated processing. As a result, deep learning, particularly convolutional neural networks (CNN), is a powerful tool for solving these issues [4]. CNN can improve efficiency even further and has demonstrated excellent performance in Alzheimer's disease diagnosis [5].

The collecting of various types of image modalities such as MRI images, functional magnetic resonance imaging (fMRI), and electroencephalography (EEG) is the focus of neuroimaging research. These approaches provide strong neuroimaging tools for researching the human brain's anatomy and function. The neuropathology and clinical processes of Alzheimer's disease are studied using magnetic resonance imaging which reveals the typical cerebral abnormalities as well as brain shrinkage. It allows researchers to distinguish between the patterns of brain damage that distinguish Alzheimer's disease from other brain diseases and abnormalities.

In neuroimaging research, machine learning approaches have led to extraordinary success and a surge in interest. Recent research has shown that deep learning models outperform traditional machine learning techniques. Deep learning's application to healthcare challenges like brain disease detection and prediction has sparked a lot of attention.

The performance of the convolution neural network (CNN) in detecting Alzheimer's disease has been exceptional. The availability of large-scale datasets for model training is often required for convolutional network success. To this goal, researchers use huge samples of imaging data to collect data. The use of convolutional neural networks is fraught with difficulties. The scarcity of medical datasets

that have been labeled. Furthermore, the data unbalance problem causes network performance to suffer.

Alzheimer's disease has four stages: mild demented, moderately demented, non-demented, and very mildly demented. Many complex networks that are frequently implemented as backbones in studies handled two-dimensional images as input. To put it another way, using MRI scan images like 2D images may be advantageous.

One of these convolutional neural network variations is EfficientNet. EfficientNet uses the compound scaling approach to scale the network's dimensions. Under a certain resource limitation, the applied grid search method was used to identify the relationship between the different scaling parameters of the baseline network. The scaling factors in each of the scaled-up variables might be determined using this approach. These factors were used to scale the baseline network to a suitable size. The compound scaling strategy enhanced the model's accuracy and efficiency more than the previous CNN method [1, 6].

## 2 Related Work

The use of a convolution neural network on brain MRI for Alzheimer's disease (AD) classification has sparked a lot of attention. Pan et al. [7] obtained a 90.47% accuracy rate on 2D convolutional neural networks and ensemble learning which they combined and implemented. Fanar et al. [8] worked on MRI brain abnormalities which indicated Alzheimer's disease and MCI on the ADINI dataset. They implemented data augmentation techniques to solve the problem of an unbalanced dataset. Jaina et al. [9] used the OASIS dataset to collect data. The models were built from the ground up, which is a laborious and time-consuming process. Several researches have used transfer learning to tackle this problem. For an early diagnosis of Alzheimer's disease transfer learning based on deep convolutional neural networks has been implemented. Nawaz et al. [10] employed the VGG16 feature extractor, which has been pre-trained on the ImageNet dataset. In addition, for AD classification a pre-trained Alex Net model was used to train their model. Lin et al. [11] used magnetic resonance imaging data and transfer learning, accurately to estimate MCT-to-AD conversion [12] attaining accuracy of 79.9%. This approach achieves a 79.9% accuracy rate on their model. Islam et al. [13] used MRI data to automatically segment and classify the hippocampal region in Alzheimer's disease. These proposed deep learning models were able to achieve an accuracy of 88.9%. A group of convolutional models was proposed for simultaneous hippocampal segmentation and illness classification.

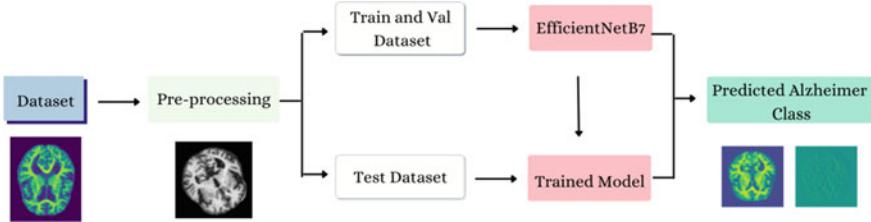
Abrol et al. [14] used structural MRI data to predict the subset of MCI patients who will develop Alzheimer's disease within three years and the subset of MCI patients who will not develop Alzheimer's disease within this time frame. Gorji et al. [15] based on MRI images, distinguished between healthy people and the two types of MCI groups and achieved an accuracy of 94.54% using CNN. Furthermore, their models EMCI/LMCT and CN/EMCL resulted in an accuracy of 93.96 and 93.0%.

Aderghal et al. [16] proposed an approach for Alzheimer's disease classification that integrates MRI and MD imaging modalities. To address the shortage of data, Nannie et al. [17] suggested a cross-modal transfer learning strategy on the basis of an ensemble of pre-trained CNNs, compared traditional machine learning with transfer learning on AD detection utilizing MRI modality. Using the ADNI dataset, they analyzed the possible application of the transfer learning approach. They discovered that transfer learning outperforms typical machine learning methods in terms of accuracy. They also demonstrated transfer learning models that perform equally as custom CNN models. Ashraf et al. [18] also emphasize the positive outcomes of transfer learning. They use the ADNI dataset to compare different CNN models utilizing transfer learning. DenseNet model performed better in their tests. Some research has focused on the 3D CNN as a way to counteract the information loss that the 2D CNN causes. For example, Yagis et al. [19] used a 3D VGG16 to avoid the information loss. On the ADNI dataset, they achieved 73.4% classification accuracy and 69.9% on the OASIS dataset. Raju et al. [20] presented a three-way classification of AD-MCI-NC for AD utilizing a cascaded three-dimensional-convolutional neural network. They were able to obtain a high level accuracy, with a score of 97.77%. 3D CNN, on the other hand, takes a long time to train.

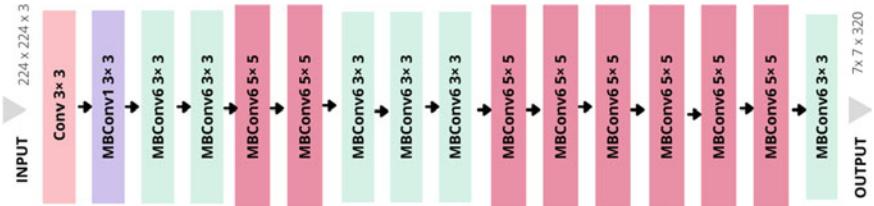
There are numerous features in brain scans that can be utilized to diagnose Alzheimer's disease, and healthy persons. One of the characteristics that distinguish AD patients from typical aging persons is the rate of brain atrophy [13, 14]. Other typical characteristics used by scientists include ventricular enlargement, hippocampal atrophy, and the rate at which brain atrophy changes [16, 17]. Alterations in the gray matter, white matter, and cerebrospinal fluid are three additional significant features of MCI and AD patients. The EfficientNet network addresses the major flaws in current convolutional neural networks. Some preprocessing operations, such as image scaling, are included in data preparation. Islam et al. [13] worked on MRI image dataset and resulted in 79% accuracy with VGG16, 92% accuracy with ResNet, 93% accuracy with DenseNet, and 96% accuracy with the EfficientNet model.

### 3 Proposed Methodology

The proposed method for the efficient classification of Alzheimer's disease is shown in Fig. 1. This section details the EfficientNet transfer learning model used for diagnosing the four-class Alzheimer's MRI images. The proposed model resulted with optimized classification achieving high values in all performance evaluation metrics like accuracy, F1-score, recall, and precision.



**Fig. 1** Workflow of proposed model



**Fig. 2** Architecture of EfficientNet [22]

### 3.1 EfficientNet

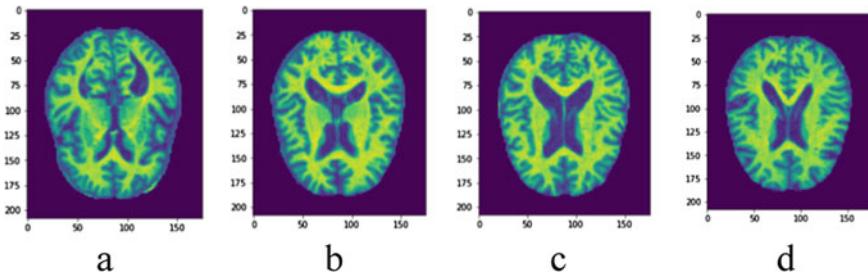
EfficientNet is a deep convolutional neural network design and scaling method that evenly scales all depth, breadth, and resolution dimensions using a compound coefficient. Google released EfficientNet [21], a lightweight a neural architecture search (NAS)-based network, in 2019. The EfficientNet scaling approach consistently increases network width, depth, and resolution with a set of predefined scaling coefficients, unlike conventional practice, which adjusts these parameters randomly. The authors used NAS to create their base network, which optimizes for both accuracy and FLOPS. Since it was discovered using a comparable search space, the design is similar to M-NASNet. To enhance performance, a new baseline network was constructed by utilizing the AutoML MNAS framework to perform a neural network models search, which maximizes both accuracy and efficiency (FLOPS). The resultant architecture is similar to MobileNetV2 and MNASNet where it employs mobile inverted bottleneck convolution (MBConv), although it is significantly bigger owing to an increased FLOP budget. The basic network is then scaled up to create the EfficientNets family of models [21] (Fig. 2).

### 3.2 Dataset and Preprocessing

The dataset used in this work is taken from Kaggle Alzheimer dataset by Yasir Hussein Shakir [23]. The dataset consisted of 6418 MRI scan images which were

**Table 1** Alzheimer's MRI image count used for processing

Class	Train	Test
Mild demented MRI image	717	179
Moderately demented MRI image	52	12
Non-demented MRI image	2580	640
Very mildly demented MRI image	1792	448

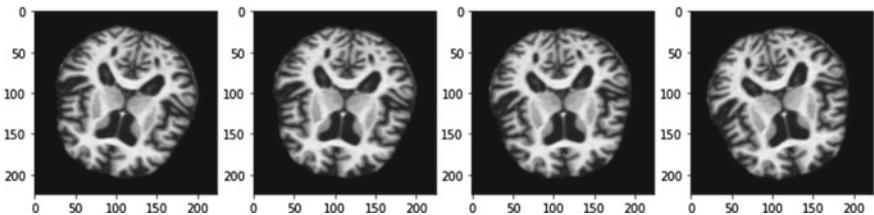
**Fig. 3** Sample images of **a** mild demented **b** moderately demented **c** non-demented **d** very mildly demented

then separated into training and testing datasets. The data was split into four classes which included mild demented, moderately demented, non-demented or normal, and very mildly demented. The training dataset consisted of 717 mildly demented images, 52 moderately demented images, 2580 non-demented images, and 1792 very mildly demented images. The test dataset consisted of 179 mildly demented, 12 moderately demented, 640 non-demented images, and 448 very mildly demented images. Table 1 shows the number of images in the dataset for all four classes.

Figure 3 shows the sample MRI images used for analysis. The train data was preprocessed and split into train and test which included 4138 train and 1035 test data after augmentation. The image from the dataset is of size  $176 \times 208 \times 3$ , which was then resized to  $224 \times 224 \times 3$  in preprocessing for faster computation.

### 3.3 Data Augmentation

Data Augmentation was performed on the four-class MRI images after splitting the data to train and test. The augmentation techniques were used on four classes with fewer images, and the techniques used included horizontal flipping, random zooming, and random contrast enhancement. A negative  $30^\circ$  flip was used in place of horizontal flipping. This study also uses a lossless augmentation technique to increase and preserve the sample count. The rotation range is  $15^\circ$ , the shear range used is 0.2, the height shift range implemented is 0.2, the zoom range of 0.3 was used, the width shift range is 0.2 was utilized, and the fill mode is nearest (Fig. 4).



**Fig. 4** Sample augmented MRI Alzheimer's images

## 4 Results and Discussion

The experimental setup, model training, and testing of the EfficientNetB7 model for the four-class classification are detailed in this section. The performance comparison of the recommended approach with past research works is analyzed to show the upper hand of the model in analyzing Alzheimer's.

### 4.1 Experimental Setup

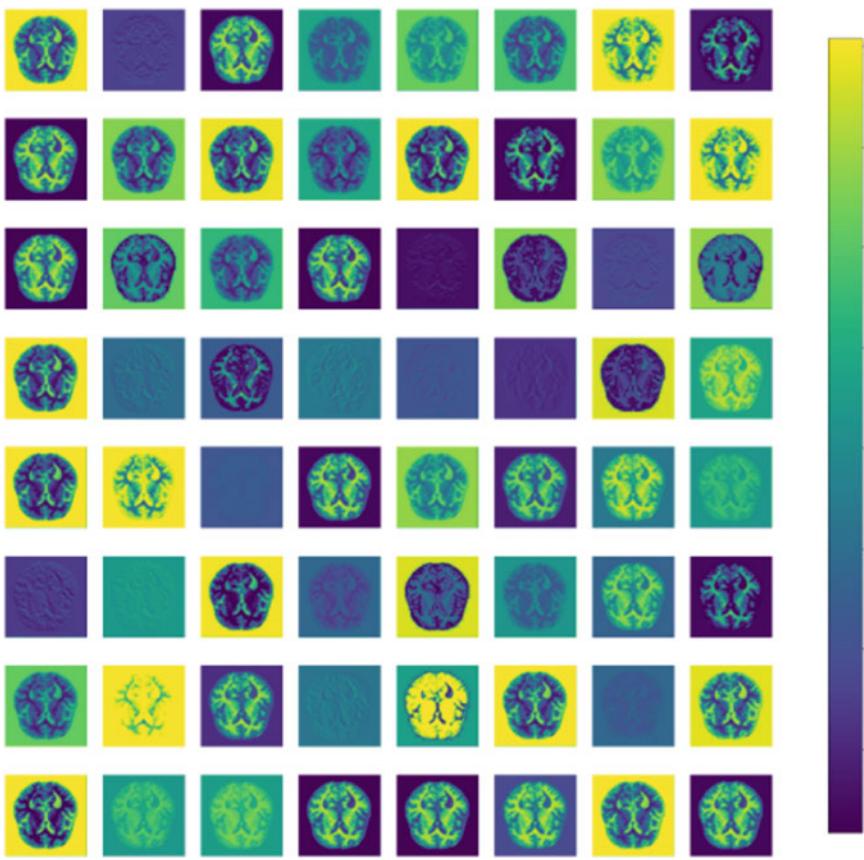
Kaggle, a cloud computing environment, was used to implement the proposed work. Kaggle offers a free graphics processing unit (GPU) for deep learning model development. The EfficientNetB7 model was trained and tested using Kaggle Notebook. Data Augmentation was performed using OpenCV library and the model is implemented using TensorFlow libraries in Python programming.

### 4.2 Hyper parameter Tuning

The hyper parameters used in this model to improve the performance are learning rate, dropout, batch size, and a number of epochs. The number of epochs ranging from 30 to 150 was analyzed where 140 was selected with a batch size of 40 for the enhanced performance. The dropout factor of 20% was tested for its efficiency. Instead of using the usual stochastic gradient descent approach, Adam is an optimization technique that can be used to update weights in the network repeatedly depending on training data where Adam uses the value of AdaGrad and RMSProp. Adams gradient optimizer was included as the optimizing function which outperformed resulting in enhanced accuracy.

### 4.3 Visually Interpreted Features of the Trained Model

Figure 5 depicts the visualization of features generated by the EfficientNetB7's inner layers. This image assists in understanding how a model processes an image on the internal layers. The figure also includes the data visualization of EfficietNetB7's activation layer clearly depicting the precise features selected for attaining good accuracy.



**Fig. 5** Visualization of features generated in the internal and activation layer

#### 4.4 Evaluation Metrics

Evaluation metrics are very essential in computing and analyzing the effectiveness and performance of the proposed model. The EfficientNetB7 model for the four-class classification of Alzheimer's MRI images is evaluated using the performance metrics like accuracy, F1 score, recall, and precision. The receiver operating characteristic curve (ROC) and the confusion matrix were also utilized to assess how well the model proceeded in each layer. As an output, the confusion matrix generates a matrix that summarizes the model's overall performance. True Positives(A) are cases where predicted as YES and the actual result was also YES. True Negatives(B) are situations where the expected is NO but resulted is also NO. False Positives (C) are circumstances in which where YES was expected but predicted to be NO. False Negatives (D), or times when a NO was anticipated but the prediction was YES instead. Equations (1) to (4) show the formulas to compute the evaluation metrics like accuracy, precision, recall, and F1-Score.

$$\text{Accuracy} = \frac{A}{A + B + C + D} \quad (1)$$

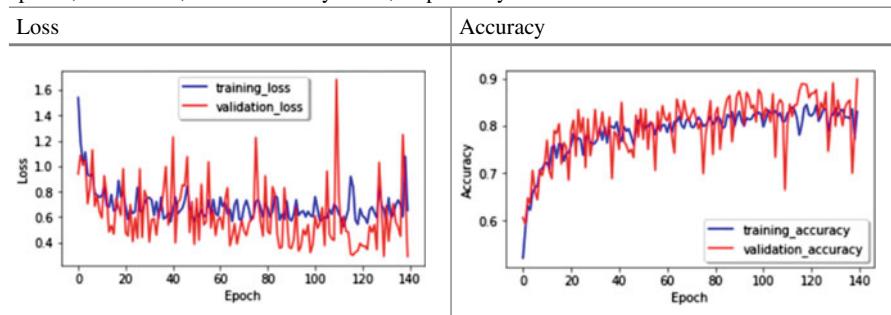
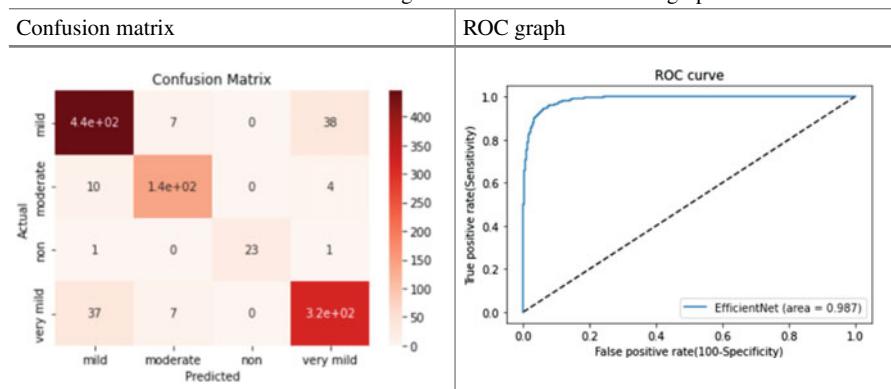
$$\text{Precision}(P) = \frac{A}{A + C} \quad (2)$$

$$\text{Recall}(R) = \frac{A}{A + D} \quad (3)$$

$$\text{F1Score} = \frac{2 * P * R}{P + R} \quad (4)$$

##### 4.4.1 Performance Analysis

By adopting the clustered block instead of single layers, or a custom CNN architecture's model complexity and the number of trainable parameters were considerably reduced. EfficientNetB7 attained an accuracy of 89.7% on the four-class test dataset. Table 2 shows the enhanced learning process resulting by the EfficientNetB7 model for four-class classification with respect to the number of epochs, loss, accuracy curve, respectively. The model attained an accuracy of 89.7% for a hyper tuned 140 epochs. Table 3 shows the model's confusion matrix as well as the ROC generated for the four-class Alzheimer's MRI images. Confusion matrix shows the correctly classified imaged for each class is given as follows, 440—mild demented, 140—moderately demented, 23—non-demented or normal, and 320—very mildly demented. ROC curve in Table 3 clearly shows the performance of the proposed EfficientNetB7 four-class classification of Alzheimer's at different thresholds. In Table 4, the model's evaluation metric is detailed showing the results obtained for accuracy, precision,

**Table 2** Learning process recorded by the EfficientNetB7 model with respect to the number of epochs, model loss, model accuracy curve, respectively**Table 3** Evaluation of EfficientNetB7 using confusion matrix and ROC graph**Table 4** Evaluation metrics of the four-class classification using EfficientNetB7 model

Class	Recall	Precision	F1-score	Support
Mild demented	0.91	0.90	0.91	490
Moderately demented	0.91	0.91	0.91	158
Non-demented	0.92	1.00	0.96	25
Very mildly demented	0.88	0.88	0.88	362
	Parameters	64,189,851	Accuracy	89.7
	Sensitivity	0.98	Specificity	0.93

recall, and F1-Score. Table 5 shows a comparison of several research works along with EfficientNetB7 showing an improved accuracy.

**Table 5** Comparison of several models and works with EfficientNetB7

S.No	Source	Method	Accuracy (%)
1	Stoeckel et al. [24]	CSVM	72.5
2	Abdulkadir et al. [25]	SVM (Linear)	87
3	Moller et al. [26]	SVM	86.2
4	Fulton et al. [27]	GBM	91.3
5	Lui et al. [28]	SAE-Zero mask	53.8
6	Sorensen et al. [29]	Combined biomarkers	62.7
7	<b>Ours</b>	<b>EfficientNetB7</b>	<b>89.7</b>

## 5 Conclusion

Alzheimer's a disease which is a kind of dementia is one of the common disease in old-aged people with about 1 in 14 people over the age 65 needs proper detection. The classification and detection of Alzheimer's are a difficult task in medicine, and hence, computer-aided detection makes the task easier. Several deep learning models have been used for Alzheimer's detection, and there is still chance of improvement. In this study, a convolutional neural network-based pre-trained model named EfficientNetB7 was proposed for Alzheimer's four-class classification in MRI images to diagnose Alzheimer's. The effectiveness of our proposed work has been evaluated on the MRI images obtained from Kaggle consisting of 6418 subjects. The dataset was augmented for generating more images for better accuracy. The model was fine-tuned with various parameters resulting in enhanced performance. The future work includes exploring more deep learning methods to detect different stages of Alzheimer's with optimized precision and accuracy.

## References

1. Hwang, E.J., Park, S., Jin, K.N., Im Kim, J., Choi, S.Y., Lee, J.H., Goo, J.M., Aum, J., Yim, J.J., Cohen, J.G., Ferretti, G.R.: Development and validation of a deep learning-based automated detection algorithm for major thoracic diseases on chest radiographs. *JAMA* **2**(3), e191095 (2019)
2. Jawahar, M., Anbarasi, L.J., Jasmine, S.G., Narendra, M: Diabetic foot ulcer segmentation using color space models. In 2020 5th International Conference on Communication and Electronics Systems (ICCES), pp. 742–747. IEEE, (2020)
3. Sharon, J.J., Anbarasi, L.L.: Diagnosis of DCM and HCM heart diseases using neural network function. *Int. J. Appl. Eng. Res.* **13**(10), 8664–8668 (2018)
4. Prajith SenthilKumar, A.L., Narendra, M., Jani Anbarasi, L., Raj, B.E.: Breast cancer analysis and detection in histopathological images using CNN approach. In Proceedings of International Conference on Intelligent Computing, Information and Control Systems, pp. 335–343. Springer, Singapore, (2021)
5. Li, F., Tran, L., Thung, K.-H., Ji, S., Shen, D., Li, J.: A robust deep model for improved classification of AD/MCI patients. *IEEE J. Biomed. Heal informatics.* **19**(5), 2168–2194 (2015)

6. <https://analyticsindiamag.com/implementing-efficientnet-a-powerful-convolutional-neural-network>
7. Pan, D., Zeng, A., Jia, L., Huang, Y., Frizzell, T., Song, X.: Early detection of Alzheimer's disease using magnetic resonance imaging: a novel approach combining convolutional neural networks and ensemble learning. *Front. Neurosci.* **14** (May 2020)
8. Fanar E.K., Al-Khuzaie, Bayat, O., Duru, A.D.: Diagnosis of Alzheimer disease using 2d MRI slices by convolutional neural network. *Appl. Bionics Biomech.* **1**(9), 6690539 (2021)
9. Jain, R., Jain, N., Aggarwal, A., Hemanth, D.J.: Convolutional neural network based Alzheimer's disease classification from magnetic resonance brain images. *Cogn. Syst. Res.* **57**, 147–159 (2019)
10. Nawaz, H., Maqsood, M., Afzal, S., Aadil, F., Mehmood, I., Rho, S.: A deep feature-based real-time system for Alzheimer disease stage detection. *Multimedia Tools Appl.* **1**, 19 (2020)
11. Lin, W., Tong, T., Gao, Q., Guo, D., Du, X., Yang, Y., Guo, G., Xiao, M., Du, M., Qu, X. and Alzheimer's Disease Neuroimaging Initiative: Convolutional neural networks-based MRI image analysis for the Alzheimer's disease prediction from mild cognitive impairment. *Front. Neurosci.* **12** (Nov 2018)
12. Liu, M., Li, F., Yan, H., Wang, K., Ma, Y., Alzheimer's Disease Neuroimaging Initiative, Shen, L., Xu, M.: A multi-model deep convolutional neural network for automatic hippocampus segmentation and classification in Alzheimer's disease. *NeuroImage* **208** (2020)
13. Islam, J., Zhang, Y.: Early diagnosis of alzheimer's disease: a neuroimaging study with deep learning architectures. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE (June 2018)
14. Abrol, A., Fu, Z., Du, Y., Calhoun, V.D.: Multimodal data fusion of deep learning and dynamic functional connectivity features to predict Alzheimer's disease progression. In: 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE (July 2019)
15. Taheri Gorji, H., Kaabouch, N.: A deep learning approach for diagnosis of mild cognitive impairment based on MRI images. *Brain Sci.* **9**(9), 217 (2019)
16. Aderghal, K., Khvostikov, A., Krylov, A., Benois-Pineau, J., Afdel, K., Catheline, G.: Classification of Alzheimer disease on imaging modalities with deep CNNs using cross-modal transfer learning. In: 2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS), IEEE (June 2018)
17. Nanni, L., Interlenghi, M., Brahnam, S., Salvatore, C., Papa, S., Nemni, R., Castiglioni, I., Alzheimer's Disease Neuroimaging Initiative: Comparison of transfer learning and conventional machine learning applied to structural brain MRI for the early diagnosis and prognosis of Alzheimer's disease. *Front. Neurol.* **11** (Nov 2020)
18. Ashraf, A., Naz, S., Shirazi, S.H., Razzak, I., Parsad, M.: Deep transfer learning for Alzheimer neurological disorder detection. *Multimedia Tools Appl.* **1**(26) (2021)
19. Yagis, E., Citi, L., Diciotti, S., Marzi, C., Atnafu, S.W., De Herrera, A.G.S.: 3d convolutional neural networks for diagnosis of Alzheimer's disease via structural MRI. In: 2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS), IEEE (July 2020)
20. Raju, M., Gopi, V.P., Anitha, V.S., Wahid, K.A.: Wahid: multi-class diagnosis of Alzheimer's disease using cascaded three dimensional-convolutional neural network. *Phys. Eng. Sci. Med.* **1**(10), 1219–1228 (2020)
21. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning, pp. 6105–6114. PMLR, (2019)
22. <https://ai.googleblog.com/2019/05/efficientnet-improving-accuracy-and.html>
23. <https://www.kaggle.com/yasserhessein/dataset-alzheimer>
24. Stoeckel, J., Fung, G.: SVM feature selection for classification of SPECT images of Alzheimer's disease using spatial information. In: Proceedings of the 5th IEEE International Conference on Data Mining (ICDM'05). IEEE, 8 p (2005)
25. Abdulkadir A, Mortamet B, Vemuri P, Jack Jr C.R., Krueger G, Klöppel, S.: Alzheimer's disease neuroimaging initiative. Effects of hardware heterogeneity on the performance of SVM Alzheimer's disease classifier. *Neuroimage* **58**(3), 785–792 (2011)

26. Moller, C., Pijnenburg, Y.A., van der Flier, W.M., Versteeg, A., Tijms, B., de Munck, J.C., Hafkemeijer, A., Rombouts, S.A., van der Grond, J., van Swieten, J., Dopper, E., et al.: Alzheimer disease and behavioral variant frontotemporal dementia: automatic classification based on cortical atrophy for single-subject diagnosis. *Radiology* **279**(3), 838–848 (2015)
27. Fulton, L.V., Dolezel, D., Harrop, J., Yan, Y., Fulton, C.P.: Classification of Alzheimer's disease with and without imagery using gradient boosted machines and ResNet-50. *Brain Sci.* **9**(9), 212 (2019)
28. Liu, S., Liu, S., Cai, W., Che, H., Pujol, S., Kikinis, R., et al.: Multimodal neuroimaging feature learning for multiclass diagnosis of Alzheimer's disease. *IEEE Trans. Biomed. Eng.* **62**(4), 1132–1140 (2014)
29. Sørensen, L., Igel, C., Pai, A., Balas, I., Anker, C., Lillholm, M., et al.: Differential diagnosis of mild cognitive impairment and Alzheimer's disease using structural MRI cortical thickness, hippocampal shape, hippocampal texture, and volumetry. *NeuroImage: Clinical* **13**, 470–482 (2017)

# LiDAR Integration with ROS for SLAM Mediated Autonomous Path Exploration



Rapti Chaudhuri and Suman Deb

**Abstract** Path exploration is a significant problem domain of point-to-point robot navigation. Even the availability of robust sensors has created evolving computational challenges for mobile robot navigation (MRN) specially in GPS denied indoor environments (IE). This paper presents the theoretic and experimental analysis of path exploration challenges and possibilities in an indoor environment by systematic integration of LiDAR with ROS platform for constructing assessable SLAM. The algorithmic solutions have been tested with customized differential drive structure robot platform (CUBOT). This also studies the mapping of the concerned trajectory and environment localization. Consumer grade 2D RP LiDAR is used here as the percept for the edging obstacle periphery, and the associated tangential data is used to construct the localization and mapping. Significant analysis is done on visual representation of LiDAR data in ROS platform for performance evaluation of the path exploration and planning algorithms. This evaluated result would clearly serve as reference to researchers for selecting appropriate SLAM algorithm with an associated memory representation of already explored path. It would also facilitate further combination with optimized global and local path planning techniques to achieve a desired goal by autonomous mobile robot in constrained indoor environment.

**Keywords** Path exploration · Hector SLAM · Differential drive · Global and local path planning · Autonomous mobile robot

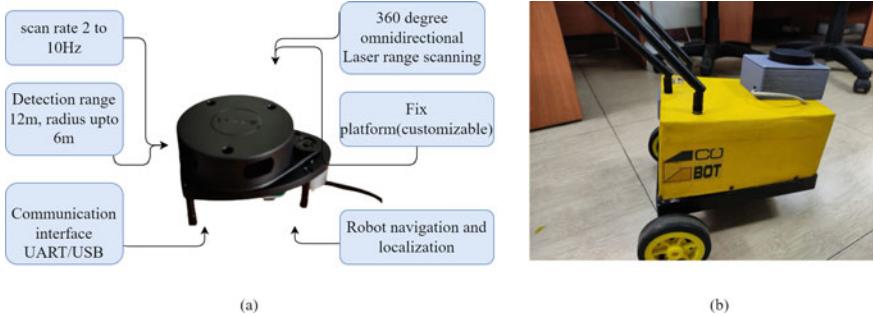
## 1 Introduction

The domain of mobile robot navigation (MRN) is fortified with number of environmental, computational as well as sensor-oriented challenges. Environmental complexities involve unknown cluttered environment congested with noise, chances

---

R. Chaudhuri (✉) · S. Deb  
National Institute of Technology Agartala, Agartala, India  
e-mail: [rapti.ai@nita.ac.in](mailto:rapti.ai@nita.ac.in)

S. Deb  
e-mail: [sumandeb.cse@nita.ac.in](mailto:sumandeb.cse@nita.ac.in)



**Fig. 1** **a** A1M8 series RP LiDAR with its associated customized features. **b** Customized robot platform used for experimentation equipped with two active wheel differential drive design and with RP LiDAR sensor mounted on its top

of facing on-route obstacles and limited reachability. Computational complexities mainly acquire time taken to reach a goal and the extent of optimal path chosen to reach the specific target. Robot navigation [1, 2] has been in fame in case of solving various issues including rescue [3], exploration of unknown path and automated logistic [4]. Simultaneous localization and mapping (SLAM) [5–7] serves as one of the best techniques to obtain map of the whole environment and trajectory of the path explored by the concerned mobile robot. Hector SLAM algorithm provides simultaneous position of the robot with respect to the sensor and mapping mainly of 2D environment, and 2D LiDAR acts as visual input source. Most of the reviewed works proposed various sensor-based graph theoretic path planning and exploring techniques by a mobile robot accompanied with production of real-time 2D map construction of the robot trajectory using SLAM [8]. The presented work on this paper uses the SLAM algorithm accompanied with various techniques to obtain more accurate and precise map of the path traversed by the customized mobile robot. GPS denied indoor environment (IE) [9] is supported with the primary challenges handled by LiDAR-based sensor [10] for indoor robot navigation (Fig. 1). In this work, A1M8 360 degree 2D laser scanner model of RP LiDAR is used to scan the concerned environment and detect the periphery of the obstacles situated in same plane as the sensor. An indoor laboratory environment is taken for accomplishing the whole experiment.

The traversable movement of trajectory is noted for preparing the SLAM environment visuals as well as path exploration data for the purpose of finding the optimized path out of multiple options which can be finally selected by the mobile robot for movement from point x to point y. The primary objective in this work was to integrate the sensor input for which the robot operating system (ROS) [11, 12] platform is explored and successfully integrated to simulate with Hector SLAM algorithm for obtaining the trajectory of the traversable path explored by the mobile robot. The computational process is carried out using single board-based computer, 1.3 GHz Jetson Nano, provided by NVIDIA [13]. In the proposed methodology for inte-

grating the sensors along with the ROS, an experimental platform is illustrated in subsequent sections where the algorithm used here is analysed for benchmarking the desired results for finding out the optimal path.

## 2 Related Work

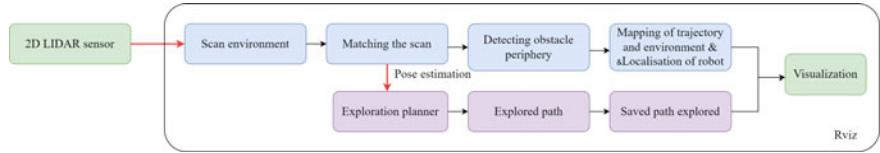
2D mapping construction using LiDAR in unknown environments and localization of robot based on detected landmarks is presented on [14]. Its application is mainly depicted on static environment. Reference [11] simulates Hector SLAM in ROS environment in a custom-built L-shaped environment. The real-time map [15] is created and saved for various applications. Paper [16] presents a proposal of safe navigation in uncertain congested environment taking input from RGB depth sensor using Visual SLAM. Primarily, Visual SLAM is used with deep reinforcement learning-based policy ensuring precise navigation in cluttered environment. In the works of [17], run-time mapping and localization results are presented based on measurement data acquired. Reference [17] has also evaluated the acquired data in terms of accuracy of the map by comparing landmark position estimates with a ground truth map which is up to the millimeter accuracy. In the further extended work of [18], they presented an ensemble approach which is based on a two-layer control architecture, and an automatic algorithm is proposed for defining roadmap. In the same work, the coordination of the whole system in order to increase the flexibility and the global efficiency is aimed to maintain. In all the mentioned works, noticeable gap is obtained in systematic filtration, smoothing operation for achieving accurate and precise solution of robot trajectory. The mentioned techniques in this paper try to tackle these gaps by systematic analysis with the sensor data in ROS environment been presented at optimized solution.

## 3 Methodology

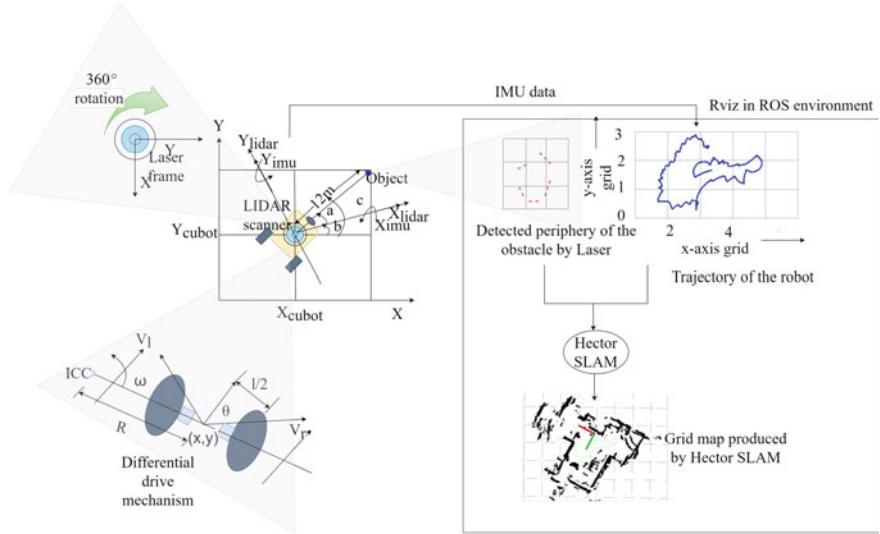
Process of getting LiDAR data and formation of map of the trajectory of robot using Hector SLAM are presented briefly into subsequent two sections. The first section represents the incorporation of ROS environment with LiDAR sensor, and the second section presents the working of Hector SLAM in formation of the map.

*Integration of RP LiDAR with ROS.* This work is carried out by integrating RP LiDAR with ROS melodic for capturing live sensor data to build a map using Hector SLAM (Fig. 2).

ROS melodic is compatible with Ubuntu version 18.04. As Jetson nano is used, ROS melodic is installed to carry out LIDAR data processing. As a single board-based portation, Jetson Nano is incorporated for native ROS processing where ROS melodic is installed to process LIDAR data. For live visualization of sensor data in



**Fig. 2** Working architecture of the model proposed including all works (mapping and exploration) carried out in Rviz visualizer of ROS environment



**Fig. 3** Geometrical description of getting LiDAR input and processing of LiDAR data and Hector SLAM for map generation with differential drive mechanism explained taking Instantaneous Center of Curvature (ICC) as reference

Rviz GUI, Laserscan object is selected to view sensor measurements. Rigid body transformation to map is illustrated through Hector SLAM [19] which starts the hector\_mapping, hector\_trajectory\_server and hector\_geotiff nodes. Map view gets availed on Rviz. Hector\_geotiff saves the map as well as the robot trajectory (Fig. 3).

**Hector SLAM.** The laser triangulation system has been developed by SLAM-TEC having high quality overall performance. Hector SLAM is applied for scanning cross section of single plane with 2D LiDAR. It is considered for particle filter-based mapping. Hector SLAM does not use odometer for data production. This algorithm is used on platform exhibiting roll or pitch motion of the sensor [14]. Main motive of this algorithm is to estimate optimally the pose of the robot as represented by rigid body transformation ( $\epsilon$ ) =  $[P_x, P_y, \theta]$  from the robot to the prior map [3].  $M(R_i(\epsilon))$  is the value of the map at  $R_i(\epsilon)$ , world coordinate of scan end points  $R_i = (S_{i,x}, X_{i,y})^T$ .  $M(R_i(\epsilon))$  obeys the function:

$$R_i(\epsilon) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} S_{i,x} \\ S_{i,y} \end{bmatrix} + \begin{bmatrix} P_x \\ P_y \end{bmatrix} \quad (1)$$

Optimal pose estimation is obtained after adding minimum error to the original pose estimation. The full Hector SLAM algorithm is presented in Algorithm 1.

---

**Algorithm 1** Hector SLAM

---

**Input:** - Laser data and a priori map,  $map_{a-priori}$

**Output:** - Optimal Pose estimation of the robot  $\epsilon^*$

- 1: Estimate probability of a grid being occupied.
  - 2: Divide the grid map.
  - 3: **for**  $P_m \in map_{a-priori}$  **do**
  - 4:      $\nabla M(P_m) = \left( \frac{\delta M}{\delta x}(P_m), \frac{\delta M}{\delta y}(P_m) \right)$
  - 5: Approximate gradient and derivatives using four closest integer coordinates  $P_{00}$ ,  $P_{01}$ ,  $P_{10}$  and  $P_{11}$ .
$$M(P_m) \approx \frac{y-y_0}{y_1-y_0} \left( \frac{x-x_0}{x_1-x_0} M(P_{11}) - \frac{x_1-x}{x_1-x_0} M(P_{01}) \right)$$

$$+ \frac{y_1-y}{y_1-y_0} \left( \frac{x-x_0}{x_1-x_0} M(P_{10}) - \frac{x_1-x}{x_1-x_0} M(P_{00}) \right)$$

$$\frac{\partial M}{\partial x}(P_m) \approx \frac{y-y_0}{y_1-y_0} M((P_{11}) - M(P_{01}))$$

$$+ \frac{y_1-y}{y_1-y_0} M((P_{10}) - M(P_{00}))$$

$$\frac{\partial M}{\partial y}(P_m) \approx \frac{x-x_0}{x_1-x_0} M((P_{11}) - M(P_{01}))$$

$$+ \frac{x_1-x}{x_1-x_0} M((P_{01}) - M(P_{00}))$$
  - 6: Pose expression in 2D Environment.  
 $\epsilon = (P_x, P_y, \chi)^T$ .
  - 7: Optimize the pose expression by matching the laser data and the map.
  - 8: **for**  $\epsilon = (P_x, P_y, \chi)^T, map_{output} \leftarrow M(R_i(\epsilon))$  at  $R_i(\epsilon)$  **do**
  - 9:      $\epsilon^* = \arg \min_{\epsilon} \sum_{i=1}^N [1 - M(R_i(\epsilon))]^2$
  - 10:    **for**  $\epsilon \leftarrow \epsilon + \Delta \epsilon$  **do**
  - 11:      $\sum_{i=1}^N [1 - M(R_i(\epsilon))]^2 \rightarrow 0$
  - 12: Use iterative and gradient descent for solving minimization problem of  $\Delta \epsilon$ .
  - 13:  $\epsilon^* \leftarrow \epsilon$
- 

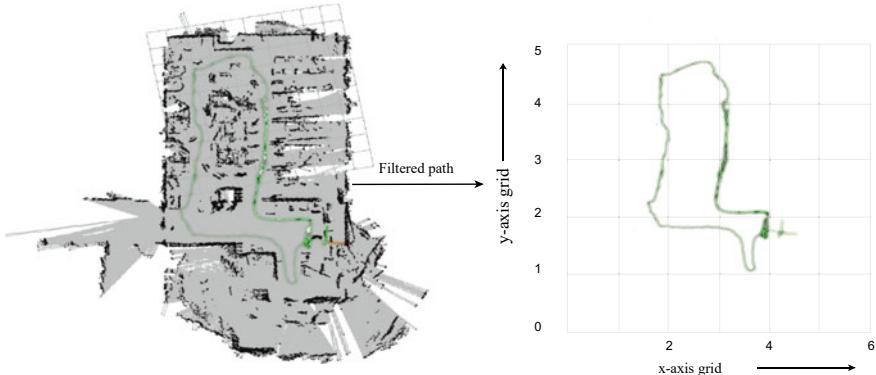
## 4 Experimental Analysis

The LiDAR input data is obtained by calling the listener through rostopic list on Rviz visualizer of ROS and is exported as csv file. Table 1 represents the data format generated in continuous manner with the scanning of environment during movement of the robot with LiDAR scanner mounted on its top. The main contributions in accomplishing the path explored by the mobile robot include:

*Filtration of LiDAR Data for inference.* The experimental environment is converted to virtual environment in Rviz where SLAM is performed for achieving the raw

**Table 1** Sequential data obtained by operating RP LiDAR on the indoor laboratory environment showing the respective fields obtained and their brief description

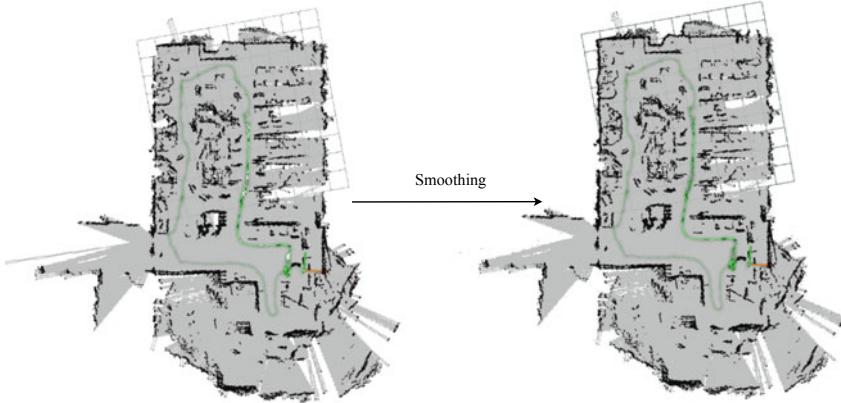
Fields obtained from LiDAR data	Remarks
seq	Sequence of continuous scanning by RP LiDAR
secs	Time required to scan nearest environment
angle_min	Least angle lower than which LiDAR is unable to scan
angle_max	Angle beyond which the LiDAR cannot scan
angle_increment	With movement of LiDAR, angle increases
time_increment	With increase in angle of scan, time increases
scan_time	Time taken to scan the nearest possible obstacles
range_min	Minimum range or distance up to which the LiDAR is able to scan
range_max	Maximum range or distance beyond which the LiDAR is able to scan
Ranges	Range up to nearest obstacles from LiDAR
Intensities	Intensity of scanning the environment per unit time by LiDAR



**Fig. 4** Filtered out explored path from raw map of the indoor environment produced by Hector SLAM

map of the path explored. After obtaining raw map of the trajectory in a specific environment and periphery of the on-route obstacles, finally the filtered map of the circumference is achieved removing all the extra environment and noise in the map (Fig. 4).

*Smoothness Operation on percept inputs.* Some left parts of the map are again scanned with the help of Luna LiDAR [20] to find the range up to the nearest obstacle if there is any. Smoothing of the achieved map is done for getting more efficiency and accuracy in obstacle detection and avoidance in case of achieving collision-free path. Black borders on the map denote the detected periphery of the obstacles. Some overshooted areas also got reflected in the unrefined diagram. The refined diagram clears all detected difficulties removing the overshooted areas and constructing the

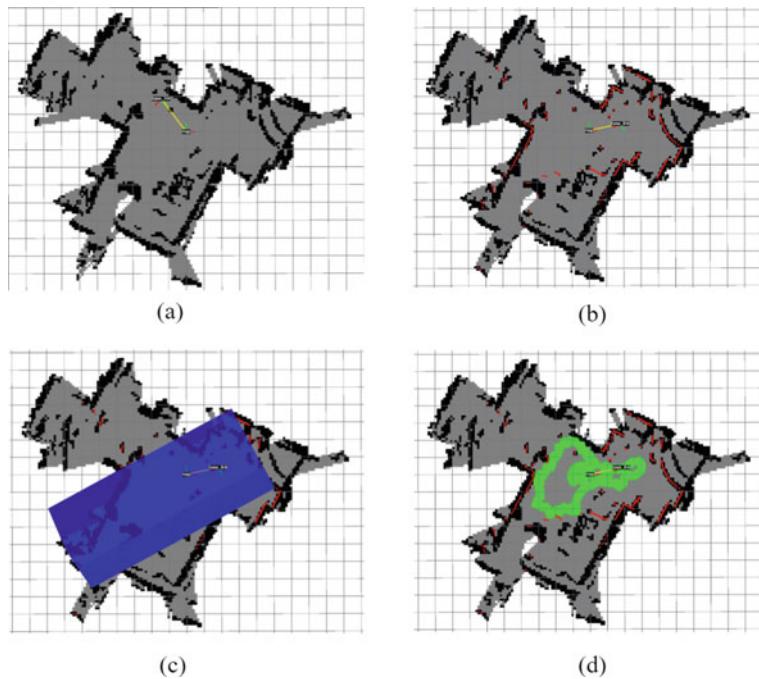


**Fig. 5** Smoothing operation is done on some overshooted areas and some gaps in identifying obstacles

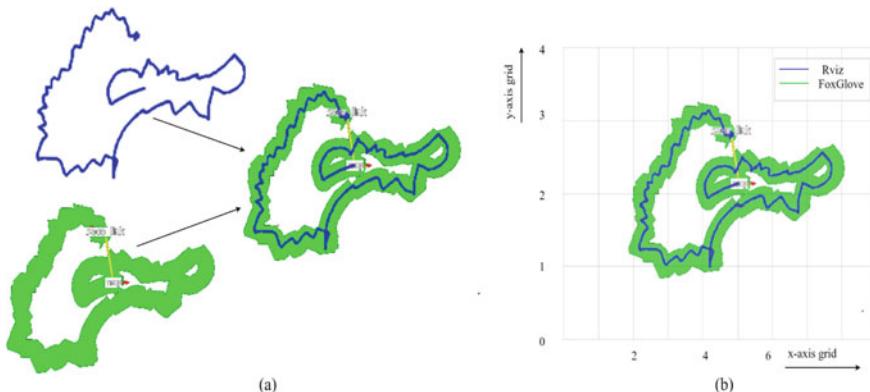
proper map with all possible nearest obstacles relative to the sensor. Figure 5 depicts the whole smoothing operation.

*Overlap of path obtained by SLAM mapping.* Precise map of the explored path is created on ROS visualizer [21], Rviz, and on another different visualizer, FoxGlove, and the resultant paths are overlapped with each other. The live data is visualized on FoxGlove to explain the comparison between two visual representations to make a cross-check in sake of research validation of the obtained data. This process reflects the absence of any kind of over or under scaling and confirms the unified direction followed by the mobile robot for selecting error-free path. The environment and path taken is same as earlier. Figure 7 shows the full overlapping operation of the map of the trajectory obtained (Fig. 6).

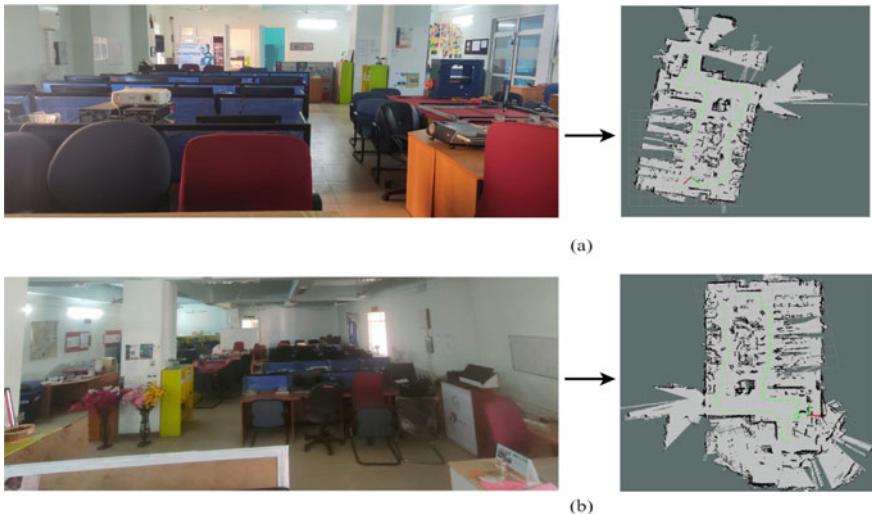
*Realization of Optical Reference.* View of original environment where the experiment is executed and attached with the map obtained from the exploration of that environment by mobile robot is done depicting visual inference. Figure 8 depicts a laboratory environment where part (a) denotes the visual reference of the actual environment with obstacles attached with the map produced by Hector SLAM in that particular environment. The green line on the obtained map denotes the path explored by the robot on that particular area. Part (b) similarly denotes the optical reference of the full indoor environment attached with the hector map produced with green line denoting the path explored by that time in that specific environment. Each optical reference is taken in between one round exploration of the whole laboratory environment with ongoing live data of Hector SLAM map. Some overshooted areas are reflected from the map formed in Fig. 5 which are smoothed using Luna



**Fig. 6** **a** Mapping of the indoor environment. **b** Periphery of the obstacles identified during mapping. **c** Path obtained by the LiDAR sensor with pointer pointing towards direction of sensor. **d** Final path obtained by the sensor



**Fig. 7** **a** Overlapping of same path obtained on exploring an indoor home environment from two different visualizers. **b** Overlapped path in graphical format formed in different visualizer to check the accuracy of the SLAM algorithm



**Fig. 8** Optical reference presenting the true congested environment on left side and Hector SLAM map of the specified environment formed on right side

LiDAR. All possible obstacles are nearly correctly identified situated on same plane as the sensor, as RP LiDAR detects only the periphery of the obstacles situated to its nearly possible closest range.

## 5 Conclusion and Future Scope

LiDAR data is used as visual source input in case of concerned indoor laboratory environment, and Hector SLAM is carried out which resulted in obtaining explored path along with perfect obstacle periphery detection on same plane as the sensor. Though limitations exist in detecting hanging obstacles or obstacles in other planes by RP LiDAR, final map of the path obtained is filtered out by removing the extra overshooted areas and other noise associated within. Smoothing and overlapping of path obtained from different visualizers have been done which validate the function of SLAM algorithm. Map of path and environment has been included along with original view of the particularly used laboratory environment serving as reference tag to the trajectory obtained. Experimental results would serve as one of the best references for future researchers to adopt the aforesaid algorithm and combine with the optimal path planning techniques to maximize the benefit of robot navigation.

## References

- Chow, J.F., Kocer, B.B., Henawy, J., Seet, G., Li, Z., Yau, W.Y., Pratama, M.: Toward underground localization: Lidar inertial odometry enabled aerial robot navigation. arXiv preprint [arXiv:1910.13085](https://arxiv.org/abs/1910.13085) (2019)
- Pfrunder, A., Borges, P.V., Romero, A.R., Catt, G., Elfes, A.: Real-time autonomous ground vehicle navigation in heterogeneous environments using a 3d LiDAR. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2601–2608. IEEE (2017)
- Zhang, X., Lai, J., Xu, D., Li, H., Fu, M.: 2d LiDAR-based slam and path planning for indoor rescue using mobile robots. *J. Adv. Transp.* (2020)
- Wang, M., Long, X., Chang, P., Padlr, T.: Autonomous robot navigation with rich information mapping in nuclear storage environments. In: 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), pp. 1–6. IEEE (2018)
- Taketomi, T., Uchiyama, H., Ikeda, S.: Visual slam algorithms: a survey from 2010 to 2016. *IPSJ Trans. Comput. Vis. Appl.* **1**, 1–11 (2017)
- D’Alfonso, L., Griffi, A., Muraca, P., Pugliese, P.: A slam algorithm for indoor mobile robot localization using an extended Kalman filter and a segment based environment mapping. In: 2013 16th International Conference on Advanced Robotics (ICAR), pp. 1–6. IEEE (2013)
- Russo, L., Rosa, S., Bona, B., Matteucci, M.: A ROS implementation of the mono-slam algorithm. *Int. J. Comput. Sci. Inf. Technol.* **6**(1), 339–351 (2014)
- Ratasich, D., Frömel, B., Höftberger, O., Grosu, R.: Generic sensor fusion package for ROS. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 286–291. IEEE (2015)
- Harik, E.H.C., Korsaeth, A., et al.: Combining hector slam and artificial potential field for autonomous navigation inside a greenhouse. *Robotics* **7**(2), 22 (2018)
- Zingg, S., Scaramuzza, D., Weiss, S., Siegwart, R.: MAV navigation through indoor corridors using optical flow. In: 2010 IEEE International Conference on Robotics and Automation, pp. 3361–3368. IEEE (2010)
- Nagla, S.: 2d hector slam of indoor mobile robot using 2d LiDAR. In: 2020 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS), pp. 1–4 (2020)
- Olalekan, A.F., Sagor, J.A., Hasan, M.H., Oluwatobi, A.S.: Comparison of two slam algorithms provided by ROS (robot operating system). In: 2021 2nd International Conference for Emerging Technology (INCET), pp. 1–5. IEEE (2021)
- Sebastián Valladares, Mayerly Toscano, Rodrigo Tufiño, Morillo, P., Vallejo-Huanga, D.: Performance evaluation of the Nvidia Jetson nano through a real-time machine learning application. In: International Conference on Intelligent Human Systems Integration, pp 343–349. Springer (2021)
- Saat, S., Abd Rashid, W.N., Tumari, M.Z.M., Saealal, M.S.: Hectorslam 2d mapping for simultaneous localization and mapping (SLAM). *J. Phys: Conf. Ser.* **1529**, 042032 (2020)
- De Gregorio, D., Cavallari, T., Di Stefano, L.: SkiMap++: real-time mapping and object recognition for robotics. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 660–668 (2017)
- Jin, S., Meng, Q., Dai, X., Hou, H.: Safe-Nav: learning to prevent PointGoal navigation failure in unknown environments. *Complex Intell. Syst.* 1–18 (2022)
- Beinschob, P., Reinke, C.: Graph slam based mapping for AGV localization in large-scale warehouses. In: 2015 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), pp. 245–248. IEEE (2015)
- Digani, V., Sabattini, L., Secchi, C., Fantuzzi, C.: Ensemble coordination approach in multi-AGV systems applied to industrial warehouses. *IEEE Trans. Autom. Sci. Eng.* **12**(3), 922–934 (2015)
- Megalingam, R.K., Teja, C.R., Sreekanth, S., Raj, A.: ROS based autonomous indoor navigation simulation using slam algorithm. *Int. J. Pure Appl. Math.* **118**(7), 199–205 (2018)

20. Hussein, M.W., Tripp, J.W.: 3d imaging LiDAR for lunar robotic exploration. In: Space Exploration Technologies II, vol. 7331, p. 73310H. International Society for Optics and Photonics (2009)
21. Hoang, K.C., Chan, W.P., Lay, S., Cosgun, A., Croft, E.A.: Arviz—an augmented reality-enabled visualization platform for ROS applications. arXiv preprint [arXiv:2110.15521](https://arxiv.org/abs/2110.15521) (2021)

# Textual Inference Identification in the Malayalam Language Using Convolutional Neural Network



Sara Renjit and Sumam Mary Idicula

**Abstract** Natural language inference (NLI), earlier known as textual entailment, is an important task related to the semantic matching of natural language sentences. Systems and methodologies that can identify inferences are helpful for most language processing tasks like document summarization and question answering systems. There is active research in NLI for English and other foreign languages. Considering Indian languages like Malayalam, there are very few works done. Here, we focus on identifying inferences in the Malayalam language using one-dimensional convolutional neural network (CNN), multichannel CNN, and CNN architecture on matching sentences over interaction space with fastText-based embeddings. This work is an attempt to apply convolutional neural networks for NLI in Malayalam without any hand-engineered features. This approach contributes with a recall of 0.66% for binary and 0.51% for multiclass classification. This work also contributes to the language resources community.

**Keywords** Malayalam · Natural language inference · Text entailment · Deep learning · CNN

## 1 Introduction

Natural language inference or textual entailment is a reasoning relationship between two sentences. It is defined as a directional relationship between a pair of sentences, namely the premise and hypothesis. The two significant sub-areas of textual entailment are as follows: recognizing textual entailment (RTE) and generating textual entailment (GTE). In the case of RTE, the system identifies inferences from given sentence pairs. RTE can be considered a binary classification problem or a multiclass classification problem. The premise entails the hypothesis when the meaning of the hypothesis can be inferred from the meaning of the premise. Hypothesis contradicts

---

S. Renjit (✉) · S. M. Idicula

Department of Computer Science, Cochin University of Science and Technology, Kochi, India  
e-mail: [sararenjit.g@gmail.com](mailto:sararenjit.g@gmail.com)

the premise if the meaning conveyed in the hypothesis contradicts that of a premise; otherwise, it is neutral. The generation of textual entailment deals with the automatic generation of inferred sentences from a given sentence. It may require systems to model commonsense knowledge as well. There are rule-based and LSTM models proposed for generating textual entailments [12].

There are two sub-areas for NLI, i.e., recognizing inferences and generating inferences. NLI is a common subtask in the English language-based NLP techniques. NLI extended to other low-resource languages would help NLP systems in that particular language to perform better. Our effort is a novel contribution in that aspect to the Malayalam language, where efforts are progressing in summarization and anaphora resolution. Systems that identify entailments for Malayalam would help improve other NLP systems, such as summarization systems.

Malayalam is a Dravidian language used in verbal and written form in the Indian state of Kerala. It has an official language status in Kerala. It is agglutinative and has a highly inflectional nature with free word order. This language has incorporated many elements from Sanskrit and later from English. Loan words from Sanskrit are seldom used, but those from English are prevalent nowadays.

The contributions in this work include the application and analysis of:

- one-dimensional convolutional neural networks
- combined 1D and 2D convolutional layers
- multiple channels of 1D convolutional layers.

for solving inference identification problems of Malayalam language texts.

The rest of the article is as follows: Sect. 2 mentions the related works in English and other languages. We present the CNN-based inference along with dataset details in Sect. 3. Experimental results obtained for binary and multiclass classification are shown in Sect. 4. Section 5 has the discussions on the results. A brief note concludes the work in Sect. 6.

## 2 Related Works

The works related to NLI started initially as RTE challenges using RTE challenge datasets, followed by independent contributions with publicly available datasets like SICK, SNLI, MNLI, and XNLI. This section discusses the state-of-the-art methods and datasets used in English and other languages.

### 2.1 Works in English

There are numerous works reported in English and other languages since the first PASCAL recognizing textual entailment challenge happened in 2005 [7]. The growth

in size and veracity of the dataset has led to many research contributions every year. Since then, most of the works reported use syntactic similarity, tree edit distance, machine learning based on lexical, syntactic, semantic features, and probabilistic models. Stanford's creation of the SNLI dataset in 2015 has shifted the approaches used from lexical matching and syntactic analysis to deep learning-based techniques. The SNLI dataset thus helped the research community to use deep learning techniques in NLI and use those models to other tasks through transfer learning.

The Stanford Natural Language Inference (SNLI) dataset in 2015 [4] compared lexicalized and unlexicalized feature-based models with LSTM. As there is an increased data size, LSTM has the upper hand to understand sentence meaning without any handcrafted features provided to the system. LSTM with word by word attention mechanism has improved performance according to [20].

Convolutional neural network-based architecture for matching sentences in [13] details two kinds of CNN architectures: a Siamese architecture using one-dimensional CNN and extended by multiple two-dimensional CNN layers to study the local information and high-level information representations. Tree-based CNN encoders in Siamese architecture are used in [16] to predict inferences. BiLSTM with average pooling and attention mechanisms on the same sentence produce attention-weighted sentence representations [23]. Most of the works used a Siamese architecture, with the element-wise product, element-wise difference, and concatenation of sentence representations to model the relationship between text and hypothesis. Learning sentence representations using NLI tasks can help in transfer learning for other tasks, and its suitability is discussed [5]. Recent work [18] states the use of bidirectional encoder representations from the transformer that derives sentence representations for classification and regression tasks. They have experimented with the MultiNLI dataset and achieved an accuracy of 86.7%. The methods discussed here have performances varying from 77 to 89% in terms of accuracy.

**Datasets in English** The different datasets developed for NLI systems in English are RTE challenge datasets, SICK, SNLI, and MNLI. Reference [10] details the RTE challenge datasets (1–6) created since 2005, and each year dataset size and scope are increased. Sentence Involving Compositional Knowledge (SICK) dataset consists of 10,000 sentence pairs, collected from 8k ImageFlickr dataset and SemEval 2014 Video Description dataset [15].

Stanford Natural Language Inference dataset has 570k sentence pairs annotated by human annotators with labels entailment, contradiction, and neutral [4]. Multi-Genre NLI (MultiNLI) Corpus has 433k sentence pairs collected from spoken and written text sources. It is also suitable as an additional test set for cross-genre transfer evaluation [24].

## 2.2 Works in Other Languages

NLI in other languages includes a few challenging tasks associated with conferences and workshops like SemEval and NTCIR (NII Testbeds and Community for Informa-

tion access Research). Reference [17] performed binary and multiclass classification for Japanese and Chinese textual datasets provided as a challenge dataset at NTCIR. This system translated sentence pairs into English and classified them using SVM with lexical similarity, lexical distance, and syntactic similarity as features.

Partial textual entailment problem was defined and explored in [8]. It is mentioned that in practical systems, partial entailments are more prevalent than strict entailment. They created a corpus and proposed a baseline architecture using lexical features—unigram, bigram, longest common subsequence, and skip-gram match. Reference [22] discusses entailment recognition as a subtask in the question answering system. This work is part of COLIEE, 2017, Fourth Competition on Legal Information Extraction/Entailment. They have done a binary classification of articles and questions into entailed pairs or non-entailed pairs using sentence encoding-based recurrent neural networks. Decomposable attention model-based sentence encoding is also a part of their work.

For the Malayalam language, there are very few attempts to identify inferences. There are works such as detecting paraphrases in sentences, carried out as part of the DPIL challenge at Forum for Information Retrieval, 2016. Reference [1] overviews the different features like POS tags, stopwords, stem, lemma, and similarity measures used in combination to identify paraphrases. Reference [21] used a lexical feature-based classifier for paraphrase detection. There is less prior research exclusively to identify inferences in Malayalam. One reason was the absence of a dataset, which has been solved with the MaNLI dataset [19]. Different sentence embedding models, namely paragraph vectors, fastText, multilingual bidirectional encoder representations from transformers (mBERT), and Language Agnostic Sentence Representations (LASER), embed Malayalam sentences. The embedded sentences, namely text and hypothesis, are then classified using softmax/sigmoid-based DenseNet classifier [19].

**Datasets in Other Languages** The Cross-lingual Natural Language Inference (XNLI) is a collection of 112k annotated pairs of sentences, translated from 14 different languages French, Spanish, German, Greek, Bulgarian, Russian, Turkish, Arabic, Vietnamese, Thai, Chinese, Hindi, Swahili, and Urdu [6]. Evaluation of NLP and Speech Tools for Italian (EVALITA) in 2009 has RTE task, and it provides a complete Italian RTE gold corpus [3].

### 3 Inference Using CNNs

Inference frameworks implemented to recognize entailments in Malayalam texts consist of binary and multiclass classification systems (3-class). The binary classes are entailment and contradiction. The multiclass includes a third class named neutral. If the semantics in one sentence is inferred from the other, the two sentences are entailed. Sentence pairs are contradictory if the meaning of the sentences is opposite

Text in English	Hypothesis in English	Text in Malayalam	Hypothesis in Malayalam	Label
A group of men are standing in a field with papers.	Men are working.	ഒരു കൂട്ട് പുറപ്പെട്ടവർക്ക് കൂടലാസ്തുകളുമായി നയവിൽ നിൽക്കുന്നു.	പുറപ്പെട്ടവർക്ക് കൂടലാസ്തുകളുമായി നയവിൽ നിൽക്കുന്നു.	neutral
A group of men are standing in a field with papers.	Men are standing.	ഒരു കൂട്ട് പുറപ്പെട്ടവർക്ക് കൂടലാസ്തുകളുമായി നയവിൽ നിൽക്കുന്നു.	പുറപ്പെട്ടവർക്ക് നിൽക്കുന്നു.	entailment
A group of men are standing in a field with papers.	Men are riding a cat.	ഒരു കൂട്ട് പുറപ്പെട്ടവർക്ക് കൂടലാസ്തുകളുമായി നയവിൽ നിൽക്കുന്നു.	പുറപ്പെട്ടവർക്ക് പുറപ്പെട്ടവർക്ക് നിൽക്കുന്നു.	contradiction

**Fig. 1** Sample data in English and Malayalam

to each other. The neutral class conveys that the two sentences remain neutral to each other.

### 3.1 Corpus Information

We have used the Malayalam Natural Language Inference (MaNLI) dataset in this work. It is created by human and automatic translations of sentence pairs from the English language SNLI dataset. The semantic aspect of the translated pairs converges with the semantics of the reference SNLI dataset; hence, the annotated labels are the same. Cohen's kappa score-based evaluation resulted in a 0.96 score showing almost good agreement on the labels annotated by two annotators. Cohen's kappa score is a statistical measure that quantitatively evaluates the inter-annotator agreement. The sample data from the dataset is provided in Fig. 1. Also, its English translation is provided for better understanding and readability.

### 3.2 Convolutional Neural Networks

Convolutional neural networks are developed mainly for use in image processing applications. Recently, they have shown good performance for text processing as well. CNN has mainly two operations: convolution and pooling for feature extraction. Convolution is a simple mathematical matrix multiplication between the filters and different regions of the input matrix representation. The convolution operation for an input  $i$  with kernel  $k$  is represented as

$$C[m, n] = (i * k)[m, n] = \sum_p \sum_q k[p, q] \cdot i[m - p, n - q] \quad (1)$$

where  $m, n$  are the dimensions of the output matrix. Pooling is a form of downsampling which helps to reduce the dimensions of the input. Max pooling replaces the output result of the convolution with the maximum value of the intermediate dot

products. It is represented as

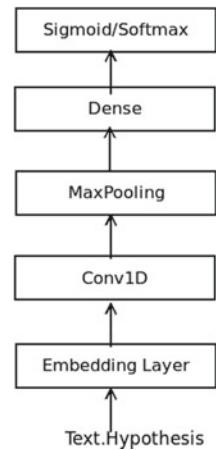
$$f(X, Y) = \max_{a,b=0}^1 C_{2X+a, 2Y+b} \quad (2)$$

where  $C$  is the matrix obtained after convolution operation and  $f$  is the resultant matrix after max pooling.

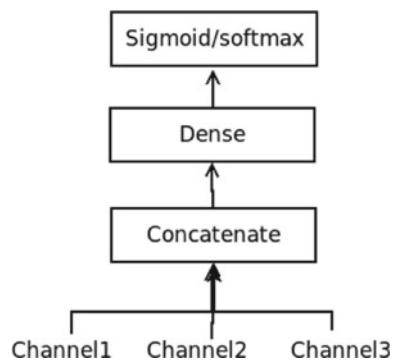
For NLP applications with CNN, sentences are represented by 1D vector representations. We employ text to sequence and fastText pre-trained word vectors for deriving sentence representations. Text to sequence method consists of obtaining sentence representation as a collection of word indices from the word index dictionary. We use pre-trained word vectors available for Malayalam in one of the systems. fastText [11] has a collection of pre-trained word vectors in 157 languages, where each language word collection is from Wikipedia and common crawl. Such collections provide a good start for language processing tasks in low-resource languages like Malayalam. Recently, BERT and sentence-BERT architectures also provided pre-trained models for NLP tasks in different languages [9]. In this work, we focus on the applicability and performance of convolutional neural networks for recognizing entailments in the Malayalam language. We use the methods discussed below to implement various models discussed in this work. It acts as a binary classifier with entailment and contradiction classes or a multiclass classifier with entailment, contradiction, and neutral classes for NLI. The different systems implemented are as follows:

- 1D convolutional neural network (1DCNN) by [14] implements different combinations of CNN. They are CNN-rand, with random weights for words, CNN-static having pre-trained weights, CNN-non-static with fine-tuning, and a CNN-multichannel, having multiple channel CNN. CNN-rand is a simple model in which the neural network learns embeddings from scratch. The sentences are tokenized and form a sequence or collection of words. The concatenated form of text and hypothesis sentence pairs are input to the embedding layer, and then one-dimensional convolutions and global max pooling are performed. It is then classified through a dense layer with softmax/sigmoid. The system design is shown in Fig. 2.
- Multichannel CNN (MCNN), also based on [14], uses multiple input channels having 1D convolutional layer with different sized kernels to learn different feature representations independently. A number of channels can be used. Each channel has an embedding layer, followed by a one-dimensional convolution layer, a dropout layer, and a max pooling layer. The outputs from each channel are flattened and concatenated. It is then fed to a DenseNet with sigmoid/softmax for classification. The system design is shown in Fig. 3, where each channel has 1DCNN layers.
- INT is based on matching sentences over their interaction spaces. As [13] proposed, this model captures rich semantic information. It adapts the CNN strategy followed in computer vision and speech. We implement this system with the MaNLI dataset.

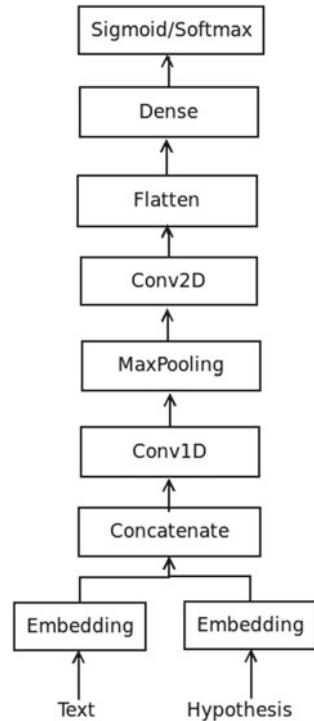
**Fig. 2** Layers of 1DCNN-based entailment classification, input in the form of concatenated text hypothesis pair



**Fig. 3** Layers of multichannel CNN-based classification system, where each channel consists of 1DCNN shown in Fig. 2.



fastText pre-trained word vectors available for the Malayalam language are used to initialize the embedding layer. Separate embeddings are derived for text and hypothesis. Then the embedded text and hypothesis are concatenated. All possible combinations of sentence pairs are modeled here using a 1D convolution layer, which represents a low-level interaction between sentence pairs, followed by max pooling, and then we have layers of 2D convolutions, which is a high-level representation of information from both sentences. The output from 2D convolutions are then flattened, batch normalized and passed to dense layers with sigmoid/softmax for binary and multiclass classification. The diagram of the system is shown in Fig. 4.

**Fig. 4** Layers of INT

## 4 Experiments and Results

Here, we discuss the experimental setup used to classify various approaches mentioned above. The MaNLI data is split into 20% for testing, the rest for training, and 10% data for validation. ReLU activation and filter sizes of (2, 3, 5) are used in convolution layers. Two 2D convolution layers are used in the model based on INT. These systems have a dense layer with sigmoid activation and binary cross-entropy loss for binary classification and softmax activation with categorical cross-entropy loss for multiclass classification. fastText-based model used 300-dimensional embeddings; other models used 50-dimensional embeddings.

### 4.1 Evaluation Metrics

These systems are evaluated based on classification metrics: precision, recall, and F1 score metrics. True positives (TP) and false positives (FP) are the numbers of pairs that have correctly or incorrectly, respectively, been classified as positive entailment, contradictory or neutral pairs. True negatives (TN) and false negatives (FN) are the

**Table 1** Binary classification report using 1DCNN

1DCNN	Precision	Recall	F1 score	Support
Contradiction	0.70	0.57	0.63	798
Entailment	0.64	0.76	0.69	800
Macro average	0.67	0.66	0.66	1598
Weighted average	0.67	0.66	0.66	1598

**Table 2** Binary classification report using MCNN

MCNN	Precision	Recall	F1 score	Support
Contradiction	0.64	0.60	0.62	798
Entailment	0.62	0.67	0.64	800
Macro average	0.63	0.63	0.63	1598
Weighted average	0.63	0.63	0.63	1598

numbers of pairs that have correctly or incorrectly, respectively, been classified as negative entailment, contradictory or neutral pairs.

Precision defines how precise the model is out of the total predicted positive.

$$\text{Precision} = \text{TP} \div (\text{TP} + \text{FP}) \quad (3)$$

The recall is a measure of the model predicted positive to actual positives.

$$\text{Recall} = \text{TP} \div (\text{TP} + \text{FN}) \quad (4)$$

F1 score is a function of precision and recall, and it provides a balance between precision and recall.

$$\text{F1 Score} = (2 * \text{Precision} * \text{Recall}) \div (\text{Precision} + \text{Recall}) \quad (5)$$

## 4.2 Results

Experimental results for the binary classification based on the above-mentioned methods are given in Tables 1, 2, and 3. The results of various models for multiclass classification are given in Tables 4, 5, and 6.

It is observed that the weighted average precision and recall are good for simpler architecture of 1D convolutional layers as compared with multiple channel CNNs and combination of 1D and 2D convolutional layers. Also binary classification has more accurate predictions as compared with multiclass classification. With respect to

**Table 3** Binary classification report using INT

INT	Precision	Recall	F1 score	Support
Contradiction	0.57	0.81	0.67	798
Entailment	0.68	0.39	0.50	800
Macro average	0.62	0.60	0.58	1598
Weighted average	0.62	0.60	0.58	1598

**Table 4** Multiclass classification report using 1DCNN

1DCNN	Precision	Recall	F1 score	Support
Contradiction	0.55	0.45	0.49	800
Entailment	0.54	0.49	0.51	803
Neutral	0.46	0.59	0.52	797
Macro average	0.52	0.51	0.51	2400
Weighted average	0.52	0.51	0.51	2400

**Table 5** Multiclass classification report using MCNN

MCNN	Precision	Recall	F1 score	Support
Contradiction	0.38	0.53	0.44	800
Entailment	0.55	0.28	0.37	803
Neutral	0.42	0.46	0.44	797
Macro average	0.45	0.42	0.42	2400
Weighted average	0.45	0.42	0.42	2400

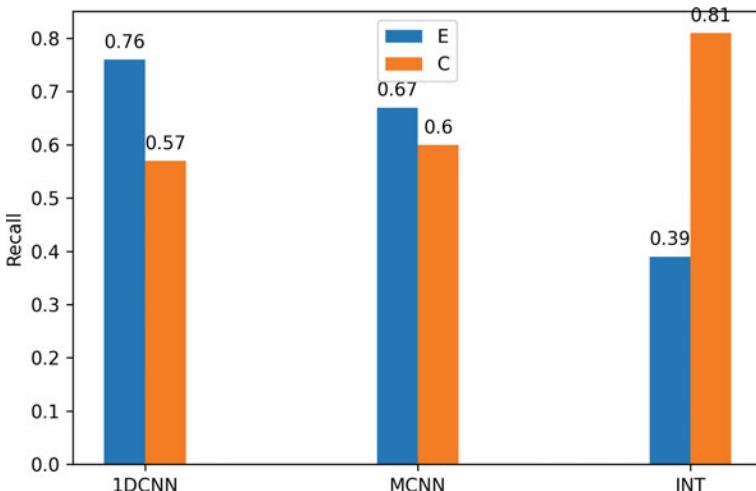
**Table 6** Multiclass classification report using INT

INT	Precision	Recall	F1 score	Support
Contradiction	0.52	0.25	0.34	800
Entailment	0.53	0.45	0.48	803
Neutral	0.43	0.71	0.53	797
Macro average	0.49	0.47	0.45	2400
Weighted average	0.49	0.47	0.45	2400

accuracy measures, Table 7 shows the performance of these CNN models for binary and multiclass classification.

**Table 7** Accuracy of CNN-based classification

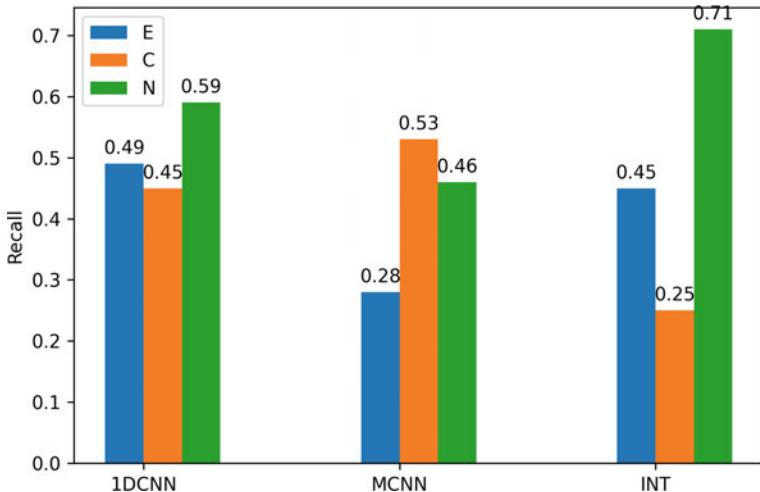
Model	Binary	Multiclass
1DCNN	0.66	0.51
MCNN	0.63	0.42
INT	0.59	0.46

**Fig. 5** Recall for binary classification

## 5 Discussion

It can be inferred from the recall comparison graph in Fig. 5 that for binary classification, 1DCNN and MCNN are the models that performed better than INT. 1DCNN has 0.76% recall when considering entailment classes. The models 1DCNN and MCNN have recall values above 50% and are equally suitable for binary classification.

When neutral classes are also included for classification, performance is less as models find it difficult to distinguish entailment and neutral separately. In the case of multiclass classification, with respect to entailment, 1DCNN has more recall. INT model has recall for neutral class, but it is because INT model regarded most of cases as neutral. When comparing recall of each class, 1DCNN and MCNN give an average performance as compared to other models as shown in Fig. 6. It is also noted that simple models can perform adequately well in this task. All the above systems used CNNs for the NLI task in Malayalam.



**Fig. 6** Recall for multiclass classification

### 5.1 Comparison with Other Methods

This section compares the CNN-based systems and the previous methods implemented for the Malayalam language.

Reference [19] implemented sentence embedding techniques using pre-trained multilingual models, and classification is done through DenseNet. Their work mainly focused on applying different embedding techniques to represent Malayalam language texts for entailment recognition. Different embedding techniques are used, namely Doc2Vec, fastText, BERT, and LASER. Language Agnostic Sentence Representations (LASER) [2] embedded sentence classifier showed the best results in their classification. In this work, we evaluate the applicability of CNNs for Malayalam language textual entailment recognition. The performance comparison is given in Table 8.

## 6 Conclusion

In this paper, we have presented the utilization of convolutional neural networks for inference in the Malayalam language. As an attempt to recognize inferences in Malayalam, systems developed as part of this work can be regarded as CNN-based NLI systems. It is also found that 1DCNNs perform much better than other complex designs involving multichannel and combined 1D and 2D CNNs. Since the dataset is created as per SNLI dataset, the diversity in context information is also retained in this dataset. Hence, it would be challenging and exciting to understand

**Table 8** Comparison of different methods

Method	Binary	Multiclass
<i>CNNs</i>		
1DCNN	0.66	0.51
MCNN	0.63	0.42
INT	0.59	0.46
<i>Embeddings</i>		
Doc2Vec	0.58	0.49
fastText	0.68	0.52
BERT	0.66	0.50
LASER	0.77	0.64

how deep learning systems capture the contextual information from sentence pairs of low-resource languages. As we have not used any domain-specific sentences, this can be considered as an attempt that uses generic models with generic datasets in low-resource language settings.

## References

1. Anand Kumar, M., Singh, S., Kavirajan, B., Soman, K.: DPIL@FIRE 2016: overview of shared task on detecting paraphrases in Indian languages (DPIL). In: CEUR Workshop Proceedings, vol. 1737, pp. 233–238 (2016)
2. Artetxe, M., Schwenk, H.: Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Trans. Assoc. Comput. Linguist.* **7**, 597–610 (2019)
3. Bos, J., Zanzotto, F.M., Pennacchiotti, M.: Textual entailment at EVALITA 2009. *Proc. EVALITA* **2009**(6.4), 2 (2009)
4. Bowman, S.R., Angeli, G., Potts, C., Manning, C.D.: A large annotated corpus for learning natural language inference. arXiv preprint [arXiv:1508.05326](https://arxiv.org/abs/1508.05326) (2015)
5. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A.: Supervised learning of universal sentence representations from natural language inference data. arXiv preprint [arXiv:1705.02364](https://arxiv.org/abs/1705.02364) (2017)
6. Conneau, A., Rinott, R., Lample, G., Williams, A., Bowman, S.R., Schwenk, H., Stoyanov, V.: XNLI: evaluating cross-lingual sentence representations. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics (2018)
7. Dagan, I., Glickman, O., Magnini, B.: The pascal recognising textual entailment challenge. In: Machine Learning Challenges Workshop, pp. 177–190. Springer (2005)
8. Das, A., Pal, D.R.: Exploring the partial textual entailment problem for Bengali news texts. *Res. Comput. Sci.* **86**, 43–52 (2014)
9. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR [abs/1810.04805](https://arxiv.org/abs/1810.04805) (2018). <http://arxiv.org/abs/1810.04805>
10. Ghuge, S., Bhattacharya, A.: Survey in textual entailment (2014)

11. Grave, E., Bojanowski, P., Gupta, P., Joulin, A., Mikolov, T.: Learning word vectors for 157 languages. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018) (2018)
12. Guo, M., Zhang, Y., Zhao, D., Liu, T.: Generating textual entailment using residual LSTMs. In: Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, pp. 263–272. Springer (2017)
13. Hu, B., Lu, Z., Li, H., Chen, Q.: Convolutional neural network architectures for matching natural language sentences. In: Advances in Neural Information Processing Systems, pp. 2042–2050 (2014)
14. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751 (2014)
15. Marelli, M., Menini, S., Baroni, M., Bentivogli, L., Bernardi, R., Zamparelli, R., et al.: A sick cure for the evaluation of compositional distributional semantic models. In: LREC, pp. 216–223 (2014)
16. Mou, L., Men, R., Li, G., Xu, Y., Zhang, L., Yan, R., Jin, Z.: Natural language inference by tree-based convolution and heuristic matching. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Vol. 2: Short Papers), pp. 130–136 (2016)
17. Pakray, P., Bandyopadhyay, S., Gelbukh, A.F.: Binary-class and multi-class based textual entailment system. In: NTCIR (2013)
18. Reimers, N., Gurevych, I.: Sentence-BERT: sentence embeddings using Siamese BERT-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3973–3983 (2019)
19. Renjit, S., Idicula, S.: Natural language inference for Malayalam language using language agnostic sentence representation. PeerJ Comput. Sci. **7**, e508 (2021)
20. Rocktäschel, T., Grefenstette, E., Hermann, K.M., Kocišký, T., Blunsom, P.: Reasoning about entailment with neural attention. Corr abs/1509.06664 (2015)
21. Sarkar, K.: Ks\_ju@ dpil-fire2016: detecting paraphrases in Indian languages using multinomial logistic regression model. arXiv preprint [arXiv:1612.08171](https://arxiv.org/abs/1612.08171) (2016)
22. Son, N.T., Phan, V.A., Nguyen, L.M.: Recognizing entailments in legal texts using sentence encoding-based and decomposable attention models. In: COLIEE@ ICAIL, pp. 31–42 (2017)
23. Sun, C., Liu, Y., Liu, B., Lin, L., et al.: Recognizing text entailment via bidirectional LSTM model with inner-attention. In: International Conference on Intelligent Computing, pp. 448–457. Springer (2017)
24. Williams, A., Nangia, N., Bowman, S.: A broad-coverage challenge corpus for sentence understanding through inference. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, vol. 1 (Long Papers), pp. 1112–1122. Association for Computational Linguistics (2018). <http://aclweb.org/anthology/N18-1101>

# Automated Crack Severity Level Detection and Classification for Surface Crack Using Deep Convolutional Neural Networks



**Harshad B. Nehate, Parth K. Kharkar, Pratiksha S. Bhat, Varad S. Rane, and Kavita Tewari**

**Abstract** Surface wall cracks could indicate that a building's structural integrity is compromised. The evaluation of these defects can be used to define the building's state. Currently, the inspection of surface cracks is done manually. This leads to subjective judgement on cataloguing the severity of crack which ultimately poses risk to everyone. It is critical to inspect and monitor the structure for surface cracks regularly to ensure the buildings' structural health and reliability. As a better approach to visual human inspection, use of deep learning is suggested in this research paper. Hence, a convolutional neural network is proposed based on the solution to detect surface concrete cracks. The Middle East Technical University (METU) dataset contains overall 40,000 images of crack and non-crack types. This is used to optimize and train our proposed CNN model for image classification. After classifying cracks, our model can predict the severity of the crack. A well-known pre-trained object detection model, 'YOLOv5', is also implemented for comparison study. This was trained and tested on a self-annotated dataset with labels based on severity. An android app is also developed to deploy the YOLOv5 model.

**Keywords** Convolutional neural network · Computer vision · Crack detection · Crack width calculation

## 1 Introduction

It is seen that various types of surface cracks develop on every building. The detection of crack damages in industrial and civil structures has been a long-overdue problem. Due to a variety of internal and external factors, bridges, tall buildings and highways,

---

H. B. Nehate · P. K. Kharkar · P. S. Bhat · V. S. Rane (✉) · K. Tewari

Department of Electronics, Vivekananda Education Society's Institute of Technology, Mumbai, India

e-mail: [2018.varad.rane@ves.ac.in](mailto:2018.varad.rane@ves.ac.in)

H. B. Nehate

e-mail: [2018.harshad.nehate@ves.ac.in](mailto:2018.harshad.nehate@ves.ac.in)

for example, they deteriorate over time causing structural damage to the health of the infrastructure because of which structural safety remains a challenging part of most of the infrastructures. There are two types of methods that were used in traditional crack detection technology, i.e. manual detection and machine detection. Manual detection is widely regarded as being less accurate and time consuming. Machine detection technologies based on an ultrasonic, microwave and other signals have advanced rapidly in recent years [1–6].

The importance of ensuring infrastructure structural safety is directly proportional to the infrastructure's age [7, 8]. It is dangerous to assess buildings for damages in order to keep the occupants safe. Insufficient utilization of current technologies for structural health monitoring (SHM) jeopardizes the infrastructure's safety and durability. So to reduce the loss of human lives and property, inspection procedures for conditional evaluation of infrastructure must be improved [9].

There are several crack detection methods available, each of which can be classified into one of the several categories. The first category includes an excitation device on one end of support and a receiving device on the other end. The location and depth of cracks can be determined without causing any structural damage by analysing the characteristics of the waveform amplitude and frequency. However, the popularity accuracy of this method is low.

The second category is based on the minimum path algorithm [10–12], which takes into account the image light and geometric properties. The path is made up of a series of adjacent pixel values, the intensity is determined by the sum of adjacent pixel values, and the crack is the shortest path. This category necessitates a significant amount of computational effort. The third category relies on image processing [13–18], which entails applying filters to the collected crack images, such as median and mean filtering, and then recognizing the cracks using the Hough transform, binarization and tensor voting. The fourth category relies on traditional machine learning algorithms [19], in which crack images are preprocessed using algorithms like random forest and AdaBoost [20]. Deep learning is another method for crack detection and recognition. This method is distinguished by the use of convolutional neural networks and segmentation algorithms to recognize, segment and extract cracks, as well as the augmentation of the dataset. This method, which is used in this article, is more accurate and robust than the others.

In this paper, a CNN model is proposed that can recognize the cracked surface. After determining the crack location in the image, an algorithm is also developed to determine the width of the detected crack and classified it according to severity. An accuracy of 99.69% of CNN using the said algorithm with appropriate image preprocessing is achieved.

A real-time detection-based system using the YOLOv5 architecture was also tested. A real-time mean average precision (mAP) of 96.7% was achieved.

## 2 Data Preparation

For CNN, a lot of datasets were searched, and a custom dataset was considered. A lot of testing was done before finally deciding to use a dataset provided by Middle East Technical University (METU) [21]. Forty thousand images were found, separated into two classes, positive and negative. Essentially, either the image had a surface crack in it, or it did not.

### 2.1 Dataset Building for CNN

Eight thousand images were used per class to train the model to differentiate between crack and non-crack images, 2000 images per class for validation and 1000 images for testing. Each image was of the dimension  $227 \times 227$ . It is also stated in one of the research papers that image preprocessing is necessary when training CNN models, as it improves the performance recognition of one of the models. One of the essential parts of preprocessing is image enhancement. Hence, it is important to determine the relationship between image enhancement and the CNN model. An image enhancement algorithm can enlarge the difference between the features of different objects in the images [22].

Preprocessing of the images was completed before to amplify the crack area. Then the images were converted to greyscale after reading them, and then the binary invert thresholding function from the cv2 library [23] was applied.

Since the CNN model was designed completely from scratch, different thresholding techniques were experimented, and to save time and resources, the preprocessed images were stored first and then read again for experimenting with the CNN model (Fig. 1).

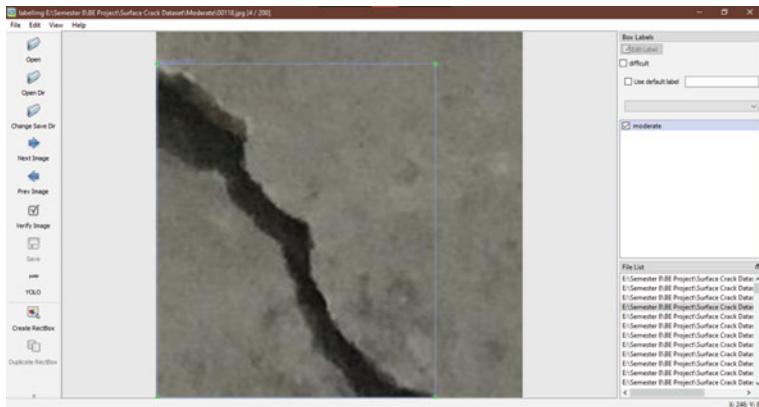


(a) Original Image



(b) Binary Inverted Image

**Fig. 1** Difference between original image and binary inverted image



**Fig. 2** Annotating images in LabelIMG application

## 2.2 Dataset Building for YOLOv5

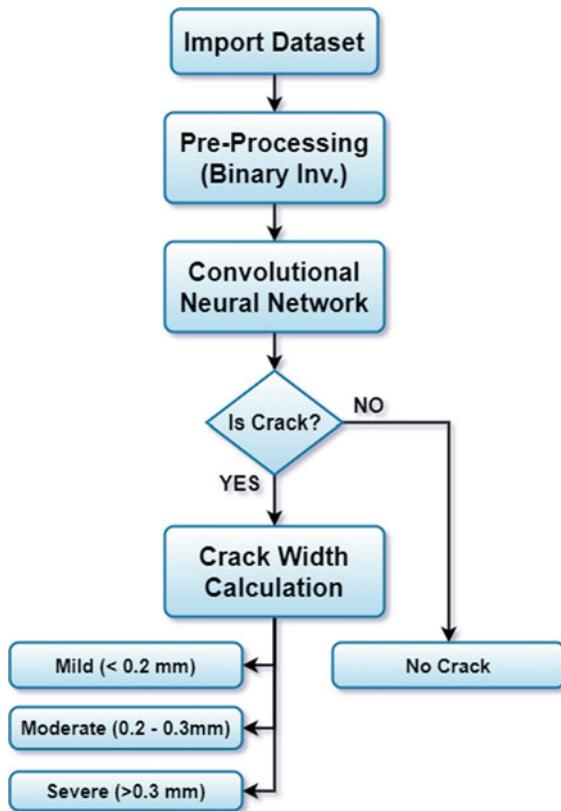
For real-time detection of surface cracks, classification was done into three distinct classes based on severity, namely mild, moderate and severe. To achieve that a width calculation algorithm was utilized that pre-sorted the images for us according to the above categories, 600 distinct images were manually annotated using the LabelImg programme [24]. This tool is used to generate a .txt file containing the crack location and classification needed for YOLOv5 to function correctly (Fig. 2).

Using Roboflow and its augmentation techniques (flip, rotate and shear) [25], a full generated dataset consisted of 1260 images in the training section (Fig. 3).

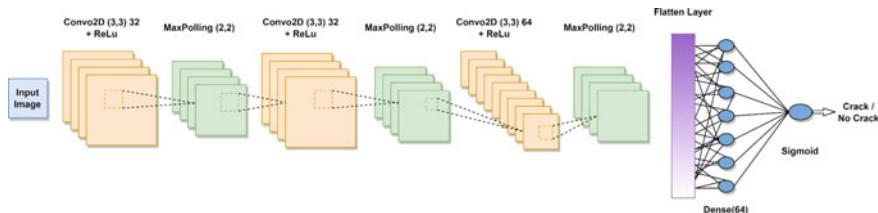
## 3 Crack Recognition Using Convolutional Neural Networks and Crack Width Calculation

### 3.1 Overview of the Proposed System

After importing the METU dataset, the images need to be preprocessed to increase the performance of CNN. The CNN model is built and train with the preprocessed dataset. If the model predicts that the surface contains a crack, it gave that image as an input to an algorithm that calculated the width of the crack using each pixel of the crack. According to the safety guidelines, a threshold to 0.3 mm needs to be set, and if the width of the crack is greater than 0.3 mm, then it is categorized as a severe crack.



**Fig. 3** Flow chart of the proposed system



**Fig. 4** CNN architecture

### 3.2 CNN Architecture

Binary inverted image, when saved, gets saved in regular RGB format, so the model has to train with  $227 \times 227 \times 3$  image size and save in the form of an array. The dataset is normalized by dividing the entire array by 255 (the highest value for RGB colour space) as it is a requirement for CNN (Fig. 4).

With our data now reshaped and ready to use, data is augmented using Keras Library [26] and three similar images from a single image in the dataset. The augmented images are slightly different in terms of orientation (Mirrored), zoom and angle. This gives us a robust dataset and improves accuracy as more data is available. It also helps in preventing overfitting. The many layers present in the convolutional network model are shown in Figure A. The Keras sequential model framework is used entirely in the model. The model uses an image input that has been transformed into a  $227 \times 227 \times 3$  pixel format. Because there are just two classes in the database, the picture data generator is set to ‘binary’ class mode with shuffle mode enabled. The conv2d layer, with 32 filters, kernel size 3, 2 strides, and activation layer ‘relu,’ is the first neural layer. Maxpooling layer is the second neural layer. Two more times, the combination of these two layers is employed to lower the size. The 3D layer is now transformed to 1D using the flatten layer after two more convolution layers. The final 1D matrix is compressed by the dense layers. The last sigmoid layer is now utilized to make final predictions about whether or not there is a crack in the image. The final output of the model is the class with the highest probability ( $>0.5$ ).

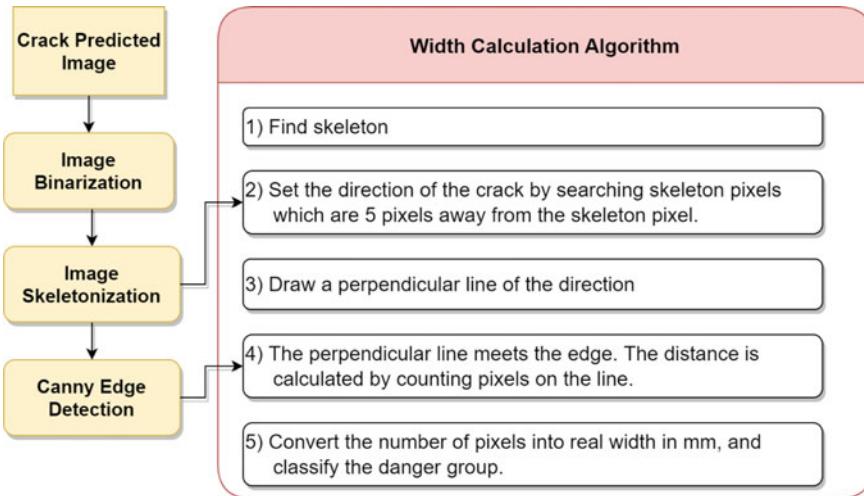
### 3.3 Width Calculation

As shown in Fig. 5, the first step is to convert the predicted crack image into a binary image and extract the crack’s skeleton information; the second step is to extract crack edge pixels using Canny edge operators, which track and match edge pixels in the counterclockwise direction and eight neighbouring pixels. The whole crack contour is obtained by matching all edge pixels, as illustrated in Fig. 5; the third step is to estimate the orientation of the crack skeleton based on the recovered crack skeleton information [27].

**Image Binarization:** Image is binarized yet again to remove any unnecessary data for edge detection and width calculation. It is the operation of dividing the image into black/white pixels to separate the cracks and non-cracks within the image.

**Image Skeletonization:** Image skeletonization is a process in which foreground regions are reduced in a binary image to a state where skeletal remnants largely preserve the connectivity of the original region by throwing/removing away most of the original pixels [28]. Extract the central skeleton of the crack. You can see the progress of the crack. You can find the crack width by drawing a line perpendicular to the crack propagation direction at the pixel on the skeleton.

**Canny Edge Detection:** Canny edge detection is a multi-step process to reach from any regular image to just the edges. First, a Gaussian filter is applied. This helps to sort out the image histogram and eliminate any noise. Then the intensity gradient is found in the image. This is followed by thresholding to the gradient magnitude. This is done to remove spurious responses to edge detection. Double threshold is used to detect edges and then track the edges to form a continuous edge.

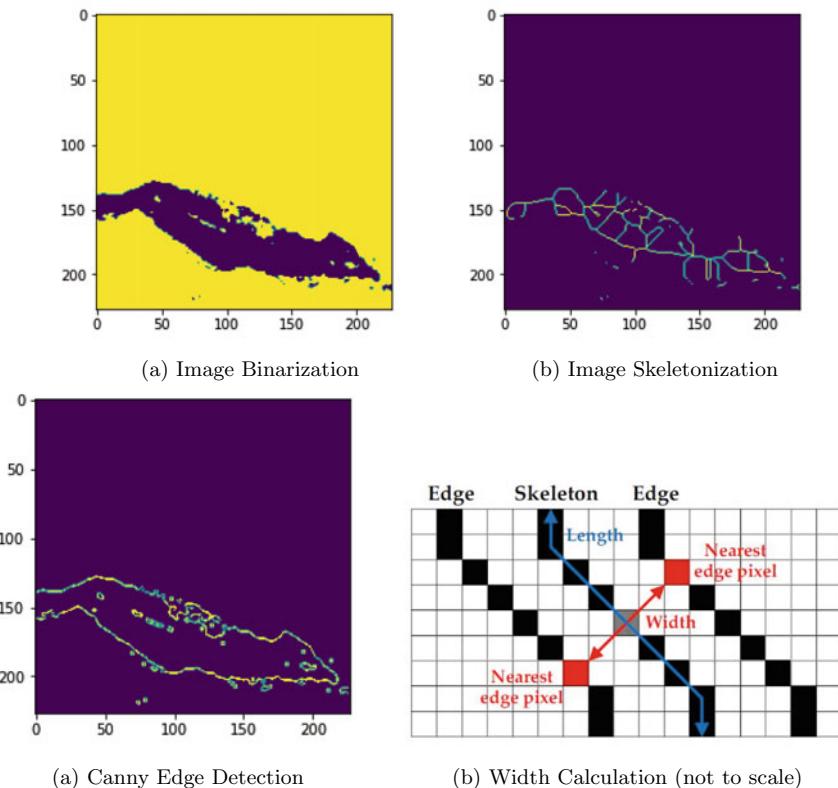


**Fig. 5** Width calculation steps

Extract the outline of the crack. In skeleton, a line perpendicular to the crack propagation direction and the crack outline are used together to find the crack width.

**Width Calculation Algorithm:** The image skeleton is saved in the form of an array. To search for the skeleton pixels from the image, breadth first searching algorithm is used. Then the direction of the crack from the skeleton pixel is found. From there on out, the direction of the crack is estimated and the direction of the crack would be the direction in which the next five pixels from the skeleton pixel are moving. The direction of the crack is saved as an angle theta using the ACoS formula. Once the direction of the crack is found, the direction perpendicular to it would be the width of the crack and the direction parallel to it would be the direction of the crack.

The perpendicular distance is calculated by counting the pixels on the perpendicular line from the skeleton until we meet the edge of the crack. This is repeated to hence find the width of the entire crack in the image and saving these values in the form of a width array. This width array is used to average out the width and then convert this pixel width into the real width.

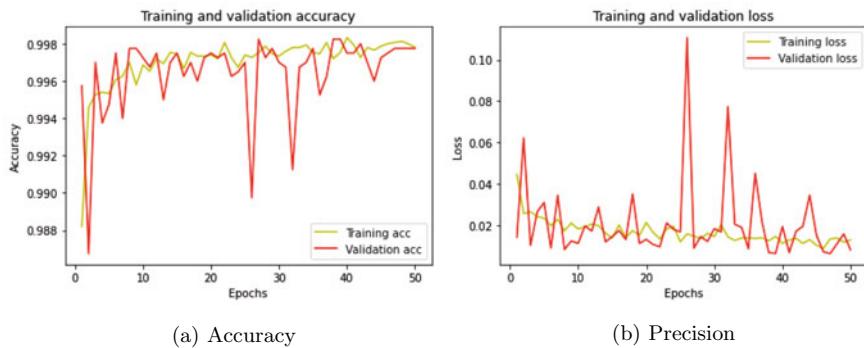
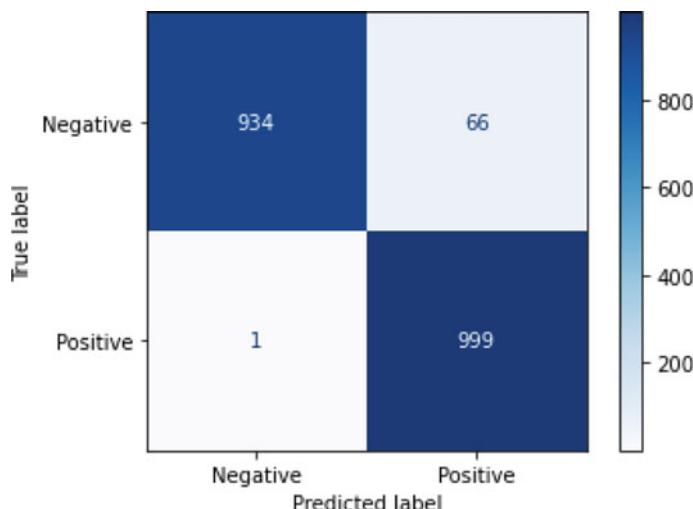


### 3.4 Result and Analysis

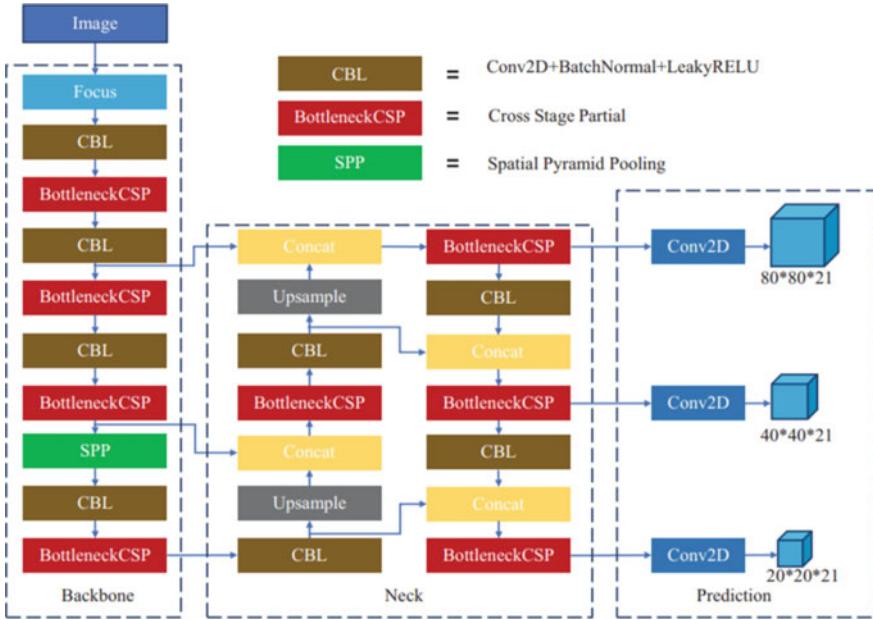
Sixteen thousand images were used to train our custom CNN model and used 4000 images to validate the said model, 2000 more images to test and evaluate our CNN model and draw a detailed analysis of the performance of the model.

An accuracy of 99.78% was achieved and a loss of 0.013 after training our custom CNN model for 50 epochs. The results of training accuracy are seen in Figs. 6 and 7.

The figure above shows the confusion matrix. Out of 2000 images tested, 934 of them were true negative, 1 was a false negative, 66 were false positives, and 999 were true positives. Using the confusion matrix precision, recall, accuracy and F1-scores are calculated and given in Table 1.

**Fig. 6** Accuracy and precision of our custom CNN model**Fig. 7** Confusion matrix of CNN model**Table 1** Calculation of precision, recall and F1-score for CNN model

	Precision	Recall	F1-score	Support
Negative	0.93	1.00	0.97	935
Positive	1.00	0.94	0.97	1065
Accuracy			0.97	2000
Macro average	0.97	0.97	0.97	2000
Weighted average	0.97	0.97	0.97	2000



**Fig. 8** YOLOv5 architecture

## 4 Crack Detection Using YOLOv5

### 4.1 Architecture

Yolov5’s network is divided into three sections: backbone, neck and output. The input image with a resolution of  $256 \times 256 \times 3$  passes via the focus structure in the backbone. It first becomes a  $128 \times 128 \times 12$  feature map via the slicing process and then a  $128 \times 128 \times 32$  feature map after a convolution operation with 32 convolution kernels. A basic convolution module is the CBL module. Conv2D + BatchNormal + LeakyRELU is represented by a CBL module.

Four variants with varying parameters can be obtained by altering the width and depth of the BottleneckCSP module, namely YOLOv5s, YOLOv5m, YOLOv5l and YOLOv5x. The SPP module primarily improves the network’s receptive field and acquires features of various scales (Fig. 8).

In addition, based on the FPN structure, YOLOv5 adds a bottom-up feature pyramid structure. The FPN layer transfers strong semantic features from top to bottom, whereas the feature pyramid represents robust positional features from the bottom-up. To improve the network’s ability to detect targets at different scales, use feature aggregation from several feature layers [29]. Output the categorization results and object coordinates at the bottom of the figure.

**Table 2** Calculation of precision, recall and mAP for YOLOv5s

Classes	Images	Labels	Precision	Recall	mAP@.5	mAP@.95
All	120	120	0.96	0.977	0.965	0.665
Mild	120	44	1	0.932	0.983	0.625
Moderate	120	39	0.843	1	0.919	0.652
Severe	120	37	0.876	1	0.994	0.718

## 4.2 Modification of Classifier

There are 80 item categories in the COCO dataset, and the output tensor has a dimension of  $3(5 + 80) = 255$ , where 3 denotes the three template boxes for each grid prediction. The coordinates  $(x, y, w, h)$  and confidence of each prediction box are represented by the number 5 (confidence,  $c$ ). The YOLOv5 classifier needs to be altered because we have three classes of objects in terms of crack detecting severity.  $3(5 + 3) = 24$  becomes the output dimension. Adapt to the situation of fracture severity detection. The number of network parameters can be lowered, lower computing overhead and increase detection accuracy and speed by altering them [29].

## 4.3 Result and Analysis

In our experiment, all of the images in the training set were cropped at random to  $256 \times 256$  pixels. And they all use data augmentation techniques, including random flipping, geometric distortion, lighting distortion, image occlusion, random erase, cut-off and mix-up, among others. Our images were scaled to  $256 \times 256$  throughout the testing phase, which made it easier for us to evaluate the network (Table 2).

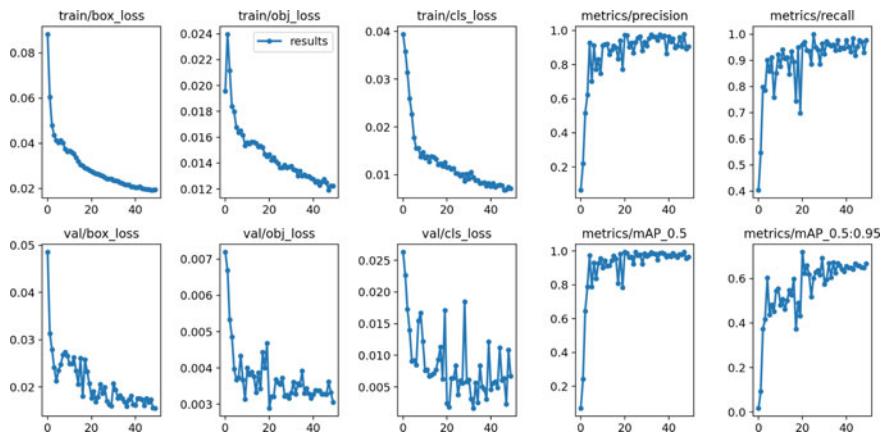
$$P = \frac{TP}{TP+FP} \quad (1)$$

$$R = \frac{TP}{TP+FN} \quad (2)$$

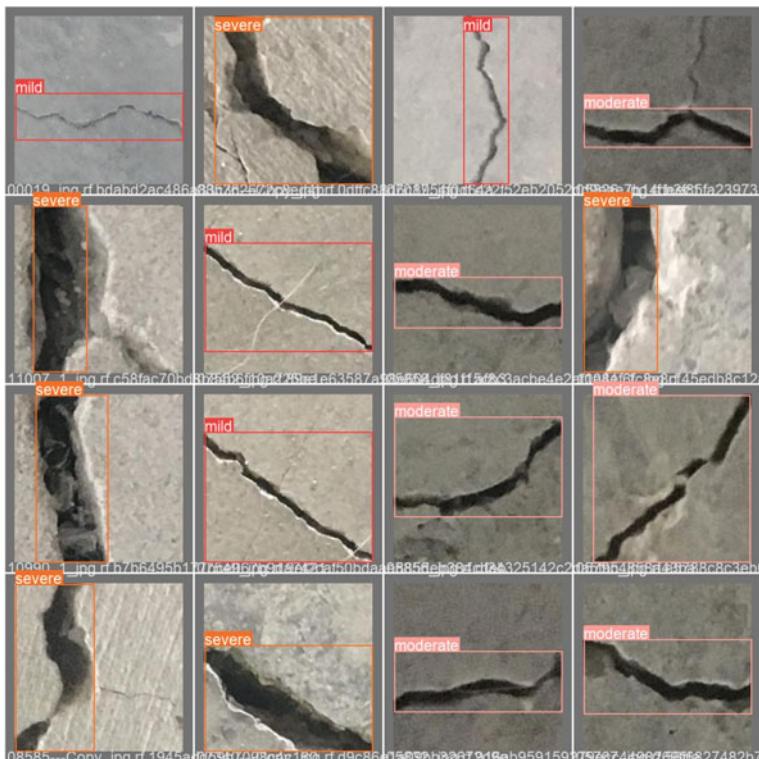
$$AP = \frac{1}{11} \sum_{recall=0}^1 \text{Precision}(recall) \quad (3)$$

The YOLOv5s model was trained with 1260 training images, 600 validation images and 120 test images for 50 epochs. The accuracy (mAP) achieved is seen in Figs. 9 and 10.

The experimental results adopt the average precision and mean mAP. For each category of object, calculate the precision rate and recall rate.



**Fig. 9** Graphs for accuracy, loss, precision and mAP for YOLOv5s



**Fig. 10** Examples of detected cracks from validation dataset



**Fig. 11** TensorFlow mobile app for object detection using our trained model

The above table shows the precision of all classes individually. The average precision (all) is 0.96 and mAP at 0.5 as well as 0.95, which are both consistent with the precision value.

As time and epochs increase, the mAP first rapidly increases and then settles at around 0.8–0.95 overtime. This is ideal behaviour for YOLOv5 architecture and with the losses that box and object loss have reduced substantially from the beginning. These are the indications that our model is working as intended, and there is no sign of overfitting. And that can be verified with the test samples that are supplied to the model.

## 5 Mobile App

To implement and test our YOLOv5s model, an app is used, deploy our now trained model in it and take some real-world input. An already-compiled app from TensorFlow Lite was used [30] to save time on making an app with less functionality and forked the repository. Android Studio was used for building the app. The app to detect and show correct severity, the phone can be held 50–60 cm away from the surface. This makes it ideal for indoor use, on walls and ceiling. Below are the screenshots from within the app (Fig. 11).

## 6 Conclusion

This study proposes a CNN-based method that is used to detect surface cracks. A CNN-based model is employed, and for training the model, the METU dataset is used. The CNN built is validated with both crack and non-crack images obtained from the METU dataset. After carrying out the whole procedure, very high accuracy is achieved. The model predicts if the image which is given as the input contains a crack or not and a special algorithm is designed which calculate the width of the crack.

According to the safety guidelines, a threshold limit is mentioned that will help categorize the images into types based on their severity. The threshold limit works as an indicating measure used to categorize the images according to the severity of the crack. In this way, a CNN model is built which helps identify the category of the crack based on the image which is provided to the model.

While working on the CNN model, another thought of using a pre-trained model was realized, i.e. YOLOv5 which can help in real-time object detection. The dataset has to be annotated priorly without using it directly as an input to the model. The YOLOv5 pre-trained model also uses the algorithm used for calculating the width of the crack. By using YOLOv5, a remarkable mean average precision was found.

Therefore, throughout the project, two different models to predict if the image contains a crack or not were used. The categorization of the dataset is solely dependent on the severity of the dataset. In this way, a project with remarkable accuracy indicates that the model is complete.

## References

1. Albishi, A.M., Ramahi, O.M.: Surface crack detection in metallic materials using sensitive microwave-based sensors. In: Proceedings of the 2016 IEEE Annual Wireless and Microwave Technology Conference, Clearwater, FL, USA, 11–13 April 2016
2. Lacidogna, G., Piana, G., Accornero, F., Carpinteri, A.: Multi-technique damage monitoring of concrete beams: acoustic emission, digital image correlation, dynamic identification. *Constr. Build. Mater.* **242**, 118114 (2020)
3. Liu, P., Lim, H., Yang, S., Sohn, H.: Development of a “stick-and-detect” wireless sensor node for fatigue crack detection. *Struct. Health Monit.* **16**, 153–163 (2016)
4. Zhao, S., Sun, L., Gao, J., Wang, J.: Uniaxial ACFM detection system for metal crack size estimation using magnetic signature waveform analysis. *Measurement* **164**, 108090 (2020)
5. Yang, X., Zhou, Z.: Design of crack detection system. In: Proceedings of the 2017 International Conference on Network and Information Systems for Computers, Shanghai, China, 14–16 April 2017
6. Zhang, X., Wang, K., Wang, Y., Shen, Y., Hu, H.: Rail crack detection using acoustic emission technique by joint optimization noise clustering and time window feature detection. *Appl. Acoust.* **160**, 107141 (2020)
7. Gibb, S., La, H.M., Louis, S.: A genetic algorithm for convolutional network structure optimization for concrete crack detection. In: IEEE Congress on Evolutionary Computation (2018). <https://doi.org/10.1109/CEC.2018.8477790>

8. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**, 2278–2324 (1998)
9. National Transportation Safety Board: Collapse of I-35W Highway Bridge, Minneapolis, Minnesota (2007). <http://www.dot.state.mn.us/i35wbridge/ntsb/finalreport.pdf>
10. Amhaz, R., Chambon, S., Jerome, I.: Automatic crack detection on two-dimensional pavement images: an algorithm based on minimal path selection. *Trans. Intell. Transp. Syst.* **17**, 2718–2729 (2016)
11. Amhaz, R., Chambon, S., Jerome, I., Baltazart, V.: A new minimal path selection algorithm for automatic crack detection on pavement images. In: Proceedings of the 2014 International Conference on Image Processing, Paris, France, 27–30 Jan 2014
12. Yang, L.C., Vincent, B., Rabih, A., Peilin, J.: A new A-star algorithm adapted to the semi-automatic detection of cracks within grey level pavement images. In: Proceedings of the 2016 International Conference on Digital Image Processing, Chengdu, China, 20–22 May 2016
13. Cheon, M.H., Hong, D.G., Lee, D.H.: Surface crack detection in concrete structures using image processing. In: Proceedings of the 2017 International Conference on Robot Intelligence Technology and Applications, Daejeon, Korea, 14–15 Dec 2017
14. Tedeschi, A., Benedetto, F.: A real-time automatic pavement crack and pothole recognition system for mobile Android-based devices. *Adv. Eng. Inform.* **32**, 11–25 (2017). [CrossRef]
15. Xiao, Y., Zhang, H.: Research on surface crack detection technology based on digital image processing. In: Proceedings of the 2019 International Workshop on Advanced Algorithms and Control Engineering, Shenzhen, China, 21–22 Feb 2020
16. Sun, H., Liu, Q., Fang, L.: Research on fatigue crack growth detection of M(T) specimen based on image processing technology. *J. Fail. Anal. Prev.* **18**, 1010–1016 (2018). [CrossRef]
17. Wang, Y., Huang, Y., Huang, W.: Crack junction detection in pavement image using correlation structure analysis and iterative tensor voting. *IEEE Access* **7**, 138094–138109 (2019). [CrossRef]
18. Li, W., Ju, H., Susan, L., Ren, Q.: Three-dimensional pavement crack detection algorithm based on two-dimensional empirical mode decomposition. *J. Transp. Eng. Part B: Pavements* **143**, 2573–5438 (2017). [CrossRef]
19. Wang, S., Yang, F., Cheng, Y., Yang, Y., Wang, Y.: Adaboost-based crack detection method for pavement. In: Proceedings of the 2018 International Conference on Civil and Hydraulic Engineering, Qingdao, China, 23–25 Nov 2018
20. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **55**, 119–139 (1996). [CrossRef]
21. Özgenel, Ç.F.: Concrete Crack Images for Classification. Mendeley Data V2 (2019). <https://doi.org/10.17632/5y9wdsg2zt.2>
22. Image enhancement effect on the performance of convolutional neural networks by Xiaoran Chen. <http://www.diva-portal.org/smash/get/diva2:1341096/FULLTEXT02.pdf>
23. [https://docs.opencv.org/4.x/d7/d1b/group\\_\\_imgproc\\_\\_misc.html](https://docs.opencv.org/4.x/d7/d1b/group__imgproc__misc.html)
24. <https://github.com/tzutalin/labelImg>
25. Roboflow—Image Augmentation. <https://docs.roboflow.com/image-transformations/image-augmentation>
26. Tensorflow Tutorials—Data Augmentation. [https://www.tensorflow.org/tutorials/images/data\\_augmentation](https://www.tensorflow.org/tutorials/images/data_augmentation)
27. Wang, W., Hu, W., Wang, W., Xu, X., Wang, M., Shi, Y., Qiu, S., Tutumluer, E.: Automated crack severity level detection and classification for ballastless track slab using deep convolutional neural network. *Autom. Constr.* (2021)
28. Kusmakar, S., Muthuganapathy, R.: Skeletal approach to mandible reconstruction represented as an image. *Comput.-Aided Des. Appl.* (2015)
29. Zhou, F., Zhao, H., Nie, Z.: Safety helmet detection based on YOLOv5. In: 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA) (2021)
30. Github—Tensorflow Examples. <https://github.com/tensorflow/examples/>
31. Kim, B., Yuvaraj, N., Sri Preethaa, K.R., Arun Pandian, R.: Surface crack detection using deep learning with shallow CNN architecture for enhanced computation. *Neural Comput. Appl.* **33**(15) (2021)

32. Rui, X., Jiazhao, S., Guangteng, L., Ruichao, H., Xufei, Y.: Appearance detection of HPLC communication module based on Res-DU-Net. In: 2021 IEEE 9th International Conference on Computer Science and Network Technology (ICCSNT) (2021)

# Haze Elimination Strategy for Indoor Robot Vision with Dark Channel Prior and Encoded Blur Kernel Space



Shubham and Suman Deb

**Abstract** In an indoor environment, extracting information from an image becomes strenuous when humidity gets on the camera glass, and resulting image becomes hazy. A lot of times due to improper handling and indifference toward camera cleanliness, the camera lens screen becomes dirty which also causes haziness in image. Moreover, the shakiness of the robot vehicle which is inevitable for all mobile robot also results in shaky and blurred images. Haze and blur removal is an ill-posed problem and very difficult task. Over the years many methods have come which uses various constraints to achieve reasonable results and also fail to properly remove blur. All these methods fail to address the hazy and blurry image simultaneously in the context of indoor robot vision. In this paper, we propose a method combining dark channel prior and encoded blur kernel space which will remove this haziness and shaky blur in real time without extra overhead. This paper focuses on vision of mobile robots in hazy and shaky conditions.

**Keywords** Haze · Blur · Dark channel prior · Encoded blur kernel space · GAN

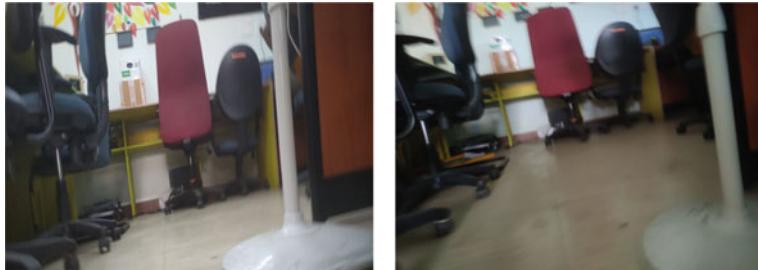
## 1 Introduction

In indoor environment, mobile robots are getting popular. Their navigation mostly depends on the camera feed. This makes images and videos coming from the camera of paramount importance. Any problem in the camera feed received by mobile robot hinders the accuracy of the action done by it. It is quite evident that mobile robots have some kind of shakiness in them. This causes the image to be little bit smoky and blurred. Moreover, humidity is also responsible for haziness, when it gets on the foremost glass lens of camera. This causes disruption in the light rays which are

---

Shubham (✉) · S. Deb  
National Institute of Technology Agartala, Agartala, India  
e-mail: [shubhamkhy@gmail.com](mailto:shubhamkhy@gmail.com)

S. Deb  
e-mail: [sumandeb.cse@nita.ac.in](mailto:sumandeb.cse@nita.ac.in)



**Fig. 1** Smokiness and blurriness due to shaking of mobile robot camera

coming toward the camera lens. This ruins the whole image and subsequently hinders the decision-making process of the robot. One more factor comes into play which is human error. When someone mishandles the camera, the lens get dirty, and again, it causes scattering in the incoming rays which causes smokiness and haziness in the image feed. All these above factors are hurdles in proper functioning of mobile robots (Fig. 1).

This haze is not some new problem we are encountering. It has always been present, but mostly it was an outdoor photography problem. Now, this haze problem appeared in indoor environment with its own set of challenges. Varying lightning condition is very common occurrence in indoor moving camera. The light reaching to camera after multiple reflection and with every different subsequent location that lightning condition change. It makes the dehazing task more difficult. In this paper, we are dealing with haze problem in varying lightning condition.

The obstruction of clarity in atmosphere due to suspended speck, fog and dust is widely known as haze. Light needs to penetrate the atmosphere substantially to get clear sharp images, but due to scattering, it becomes impossible. This is where many intelligent machines fail to extract information from images. An automatic driving model will not be able to identify road. Face recognition algorithms will fail to detect humans. In wildlife conservation, animals detection will become difficult. Almost all vision task which depends on clear photographs become challenging. This is the sole reason for the requirement of haze removal for computer vision application connected to mobile robots.

An unknown depth which fluctuates at different positions makes haze removal a hectic and challenging problem. Since haze transmission depends on depth in image, it increases the complexity to de-haze the image. Copious techniques have been created over the years to ameliorate the hazy image. Techniques like histogram-based, contrast-based [22] and saturation based [3] have been widely used to remove haze from single image. Furthermore, copious techniques have been introduced which has numerous images consisting details about depth. In the paper [21], polarization technique has been used. Here, for a single image, numerous images with different values of polarization have been used. In another method [16], many constraints have been used, and exact same visual has been captured multiple times under different

atmospheric conditions. Depth-based methods [16] are there which demand depth information from user inputs or take data from familiar 3D models. Although these methods are quite promising still in reality depth data is unavailable most of the times. Neither do we have multiple hazy images to do the estimation.

A remarkable progress have been seen in single image haze removal recently. This is because of better assumptions and priors. Specially, a local contrast maximizing method [23] based on Markov random field (MRF) which takes an assumption that for an image without haze its neighborhood contrast is quite numerically more than a regular hazy image. Splendid results have been shown by this contrast maximizing technique, but it gravitates toward creating oversaturated images. Another method [4], independent component analysis (ICA) removes haze from color images. It takes very minimal input, although the results are acceptable but it is highly time consuming. In real life, this method fails to perform, and it is not responsive to images containing heavy haze. After comparing results of many experiments on images with no haze, it has been verified that in a pixel there is a color channel whose value is very low an even in some cases close to zero. It was inspired by dark object subtraction technique, which led to discovery of dark channel prior (DCP) [6]. The thickness of haze is estimated with dark channel prior, and the haze is removed by atmospheric scattering model. Still there are shortcomings in DCP, it fails to remove haze from sky images, and it is computationally expensive. To overcome the limitations, few improved algorithms for dehazing are introduced. The image is modeled using factorial MRF [18] to find out the scene radiance accurately.

Another problem faced by mobile robots is blurriness due to shake. Mostly, camera is mounted on the robot, so shake becomes an innate part of image capturing process. Whenever robots try to turn or find an uneven platform, a shaky blurry image or video is captured. This defeats the purpose a vision equipment on a mobile robot. Most of the vision algorithms are made to work on clear sharp images, and a blurry image therefore is an obstruction in applying any vision algorithm.

The task of image deblurring is to recover a clear sharp image  $x$  when a blurry image  $y$  is presented. The below model can be assumed to formulate this idea that relates  $x$  and  $y$  mathematically:

$$y = \hat{\mathcal{F}}(x, k) + \eta \approx \hat{\mathcal{F}}(x, k) \quad (1)$$

In the above equation  $\hat{\mathcal{F}}(., k)$  is the blur operator, and  $k$  represents the blur kernel. Also,  $\eta$  represents noise in the above equation. It is assumed that  $\hat{\mathcal{F}}(., k)$  is a convolution function in its very straightforward form and  $k$  is its convolution kernel. Here,  $\eta$  is considered as white Gaussian noise. The task of deblurring states that, when a blurry image  $y$  is given, a sharp image  $x$  without blur is to be recovered with its blur operator  $\hat{\mathcal{F}}(., k)$ . However, in many cases, recovering the blur operator is optional. To recover a sharp image, a well-liked approach maximum a posterior (MAP) estimate is used. Still this is an ill-posed problem, and it causes concurrency of infinitely many pairs of  $(k, x)$  which will lead to same probability. There are many deblurring methods which focus either on designing self-made priors for  $x$  and  $k$  [2, 8, 13, 19] or they tried to make the models learn those deep image priors [20, 25]. The problematic part

is, all these works assume that the blur operator is a convolutional operator. In pragmatic scenarios, this assumption does not hold always affirmative. When tested on real-world blurry images, these MAP-based methods produce unacceptable results. These methods miserably fail to handle complex in-the-wild blur operators.

A different approach is to create deep convolutional network which will take the blurry image as input and will train on corresponding clean image as output. It will directly learn the function and the parameters of that convolutional network that maps blurry image to non-blurry image. By using this paired blurry and non-blurry images [10, 11, 15, 24], the learned function will deblurr our image. This method works differently than MAP-based approach, and this method learns the inverse function of the blur operator directly. It does not even use blur operator and the distribution of the blur kernel. Due to lack of proper representation for the space of the blur kernels, this method becomes confined to individual blur kernels. Any generalization of this approach does not give any promising results. This approach produces unacceptable results when they are tested on blur operators which are different than training datasets. In our experiments, it is observed that when we tested it on an blur operator which is different from the training, and then, the received image is nearly similar to input image. This issue is called ‘the trivial solution’. By putting prior distributions on the sharp images and the blur kernel, this problem is tackled by MAP-based approaches. Still, these priors cannot be directly applied to current deep learning models. Lack of explicit representation for blur kernels restricts its applicability to deep learning models.

In this paper, we are tackling two paramount problems of a mobile robot in vision task. First, removing a haze and second is removing the blur from an image. The proposed solution is a combination of dark channel prior (DCP) and encoded blur kernel space (EBKS). DCP uses basic image processing algorithms, and EBKS uses state-of-the-art GAN.

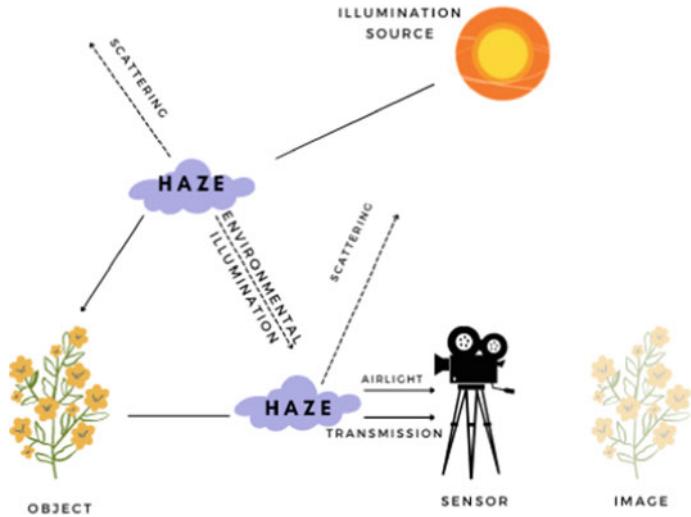
## 2 Related Work

### 2.1 Dehazing

Numerous methods for image dehazing have been proposed over the years. We will laconically review some of the important literature which has used atmospheric scattering model, which is the core principle of image dehazing, and also those methods which have assumed important and useful haze relavent features.

**Atmospheric Scattering Model** The atmospheric scattering model was first proposed by McCartney [14] to describe the creation of hazy image. It was further improved by Narsimhan and Nayar [16, 17]. Mathematically, the atmospheric scattering model is written as

$$E(x) = G(x)t(x) + A(1 - t(x)) \quad (2)$$



**Fig. 2** Atmospheric scattering

In this equation,  $E(x)$  is the obtained hazy image, and here,  $E$  is also called observed intensity.  $G(x)$  is the real image that has to be recovered back, and  $t(x)$  is called medium transmission.  $A$  represents global atmospheric light, and  $x$  represents the indexes pixels int the observed hazy image.

The objective of haze removal operation is to recover  $G, A$  and  $t$  from  $E$ . There are  $3N$  constraints and  $4N + 3$  unknowns for a  $N$ -pixel color image. This makes the haze removal process a convoluted problem (Fig. 2).

In (2), the term in the beginning of the equation  $G(x)t(x)$  is known as direct attenuation [23], and the last term  $A(1 - t(x))$  is known as airlight [7]. In this equation, direct attenuation depicts the scene radiance and its decay in the given medium. Also, a shift in scene colors can be seen as a result of airlight from previously scattered lights. Major difference between these two is direct attenuation and is a multiplicative distortion of the scene radiance while the airlight is an additive distortion. In an homogeneous atmospheric condition, the mathematical formulation of the transmission  $t$  can be expressed as

$$t(x) = e^{-\beta d(x)} \quad (3)$$

Here, the medium transmission map  $t(x)$  actually explains that part of light is not scattered and reaches to the camera. Moreover,  $d(x)$  represents the distance between the object in the image and the camera the haze. Furthermore,  $\beta$  represents scattering coefficient of the space. In Equation (3),  $d(x)$  and  $t(x)$  are inversely proportional, means when  $d(x)$  value goes near infinity the value of  $t(x)$  starts reaching to zero. Combining this with Eq. (2), we have

$$A = I(x), d(x) \rightarrow \infty \quad (4)$$

In real-life situations, distance view  $d(x)$  cannot be infinity. It can be considered a long distance such that it provides a very low transmission  $t_0$ . To find the global atmospheric light  $A$ , Eq. (3) is not used, instead of that it is properly estimated using the following formula

$$A = \max_{y \in (x | t(x) \leq t_0)} I(y) \quad (5)$$

This atmospheric scattering model straightforwardly indicates that that estimation accurate medium transmission map is of paramount importance in recovering a clean haze free image .

**Haze-relevant features** Haze removal is an ambiguous problem which is ill posed. Existing methods propose various assumptions on the basis of empirical observations which are used to compute intermediate haze-relevant features. Complete solution of hazy image is achieved using these haze-relevant features.

**Dark Channel Prior** The method of dark channel prior comes from an unique observation of outdoor images that are haze free. The color patches where there is no intrusion of haze have at least one color channel whose intensity value is very low, in some cases close to zero. For an arbitrary image  $I$ , its dark channel  $I^{dark}$  is given by

$$I^{dark}(i) = \min_{j \in \Omega(i)} (\min_{c \in r,g,b} I^c(j)) \quad (6)$$

Here,  $I^c$  represents a color channel in  $I$ , and  $\Omega(i)$  is a neighborhood patch gathered around  $i$ .

Dark channel is a result of two minimum operators. For each and every pixel  $\min_{c \in r,g,b}$  is performed. Subsequently,  $\min_{j \in \Omega(i)}$  is also performed which is also known as minimum filter. These minimum operators are commutative in nature.

Applying the concept of dark channel, it is observed that if  $I$  is an outdoor image without haze then except the sky region the intensity of  $I$ 's dark channel is low and in some cases zero.

$$I^{dark} \rightarrow 0 \quad (7)$$

This observation is known as dark channel prior. The amount of haze present in and image and dark channel are highly correlated.

This important information is the core observation which is used to find the medium transmission map [6]

**Maximum Contrast** Contrast of an image and transmission in hazy image is inversely proportional. The contrast is reduced by

$$\sum_i \|\nabla G(i)\| = t \sum_i \|\nabla H(i)\| \leq \sum_i \|\nabla J(i)\|$$

Noting this important information in a  $p \times p$  neighborhood patch considering the middle pixel as center  $\Omega_s$  the variance of neighborhood contrast [23] and also considering highest value of the local contrast in a  $q \times q$  region  $\Omega_r$  is put in a formula as:

$$C(x) = \max_{y \in \Omega_r(x)} \sqrt{\frac{1}{|\Omega_s(y)|} \sum_{z \in \Omega_n(y)} \|I(z) - I(y)\|^2} \quad (8)$$

where  $|\Omega_s(y)|$  is the local neighborhood's cardinality.

Contrast feature and medium transmission  $t$  has correlation between them. It can be seen visually, so the visibility of the image is enriched when local contrast is maximized which is depicted in above equation.

**Color Attenuation** A sharp descent is seen in the saturation  $I^s(x)$  of patch. Moreover, under the influence of haze and brightness, the color of scenes decreases. Simultaneously, brightness value increases  $I^v(x)$ , and this causes the difference to become high.

The gap between brightness and the saturation is a paramount piece information when used with color attenuation [4]. This is used to find density of haze.

$$G(x) = H^v(x) - H^s(x) \quad (9)$$

Here,  $H^v(i)$  and  $H^s(i)$  expressed in the HSV color space as  $H^v(i) = \max_{c \in \{r, b, g\}} H^c(i)$  and  $H^s(i) = (\max_{c \in \{r, b, g\}} H^c(i) - \min_{c \in \{r, b, g\}} H^c(i)) / \max_{c \in \{r, b, g\}} H^c(i)$ . When the scene depth increases, color attenuation also increases, and similarly both decreases simultaneously which implies that they are directly proportional  $d(i) \propto A(i)$ . This information helps to estimate transmission easily. The color attenuation feature is proportional to the scene depth  $d(i) \propto A(i)$  and is used for transmission estimation easily.

**Hue Disparity** In this method, original image  $I(x)$  and its semi-inverse image,  $I_{si}(x) = \max[I^c(x), 1 - I^c(x)]$  with  $c \in \{r, g, b\}$ , are used to detect haze. Difference between both these images is estimated which is an indication of haze. The semi-inverse image of pixel without haze is observed that the pixel values do flip in the channels. This causes a huge difference in the findings between  $I_{si}(x)$  and  $I(x)$ .

The property of hue disparity can be written in formula [1] as:

$$H(x) = |I_{si}^h(x) - I^h(x)| \quad (10)$$

Here, in HSV color channel, 'h' represents the hue in the image. Medium transmission  $t(x)$  propagates inversely  $H(x)$  according to (10).

## 2.2 Deblurring

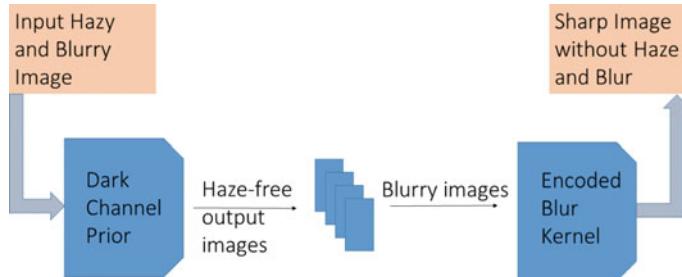
Image deblurring algorithms are broadly structured into two types. First is map-based, and second is learning-based method. **MAP-based blind image deblurring** The two important aspects of focus in MAP-based methods are first finding proper priors  $P(x)$  and second blur kernels  $P(k)$ . In case of sharp images, the gradient of natural images is highly scattered that is why gradient-based prior is used. In particular, Chan and Wong [2] proposed a total variation (TV) penalty that encouraged the sparsity of the image gradient. The image gradient follows hyper-Laplacian distribution as showed by Krishnan and Fergus [8]. On the other hand, Levin et al. [12] suggested that these gradient-based priors favor blurry images over sharp images and it leads to the trivial solution, e.g.,  $x = y$  and  $k$  is the identity operator. Using ‘ $1/2$ ’ regularization can give sharp image quite low penalty [9]. Also, the dark channel of a sharp image is scattered most of the times than the dark channel of the corresponding blurry image [19]. These priors are not enough to state the blur problem properly. They are neither enough nor domain-invariant.

Ulyanov et al. [25] introduced deep image prior (DIP) for restoration of images. In this method,  $G$  was learnt in which each image  $I$  was represented by a fixed vector. Ren et al. [20] proposed SelfDeblur method using two DIPs for  $x$  and  $k$ .  $X$  and  $k$  are used combined by a gradient-based optimizer. All aforementioned methods assumed that the blur kernel was linear and uniform; i.e., it can be represented as a convolution kernel. But this assumption is false for blur in practical scenarios. In reality, original blur is very complex than synthesized blur. Nonlinear camera response functions can cause nonlinear blur kernels, while non-uniform blur kernels appear when only a small part of the image moves.

**Learning-based deblurring** A substantial number of deep deblurring models have been introduced in recent years. A multi-scale network for end-to-end image deblurring has been introduced [15]. It was a three-step model. Output of every stage was input of subsequent stage. In the same way, scale recurrent structure [24] has been introduced for image deblurring. Recently, GANs [5] are widely used in deblurring task. It was first used in [10], and high quality image was generated as output. Kupyn et al. [11] introduced DeblurGANv2, which used feature dynamic networks [11] to extract image features and two discriminators for global and patch levels. An impressive result was shown by DeblurGANv2, and it also followed some benchmarks. However, deep learning models fail to show prominent results in cross-domain task, so deep deblurring models have not been used in real-world applications.

## 3 Methodology

First module of the mentioned working architecture includes giving hazy and blurry images as input to dark channel prior(DCP) [6] module. This module will remove the haziness or smokiness from the image and give us new proper image for processing to next module (Fig. 3).



**Fig. 3** Removal of haze and blur

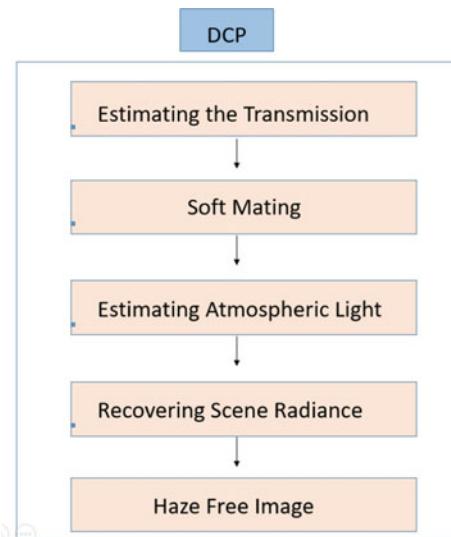
This output image from first module is passed to module second which is actually a GAN. This output image contains blur but no haze. With these images, the model will learn blur operator family [encoded blur kernel space] that models the blur function between blurred image and its corresponding non-blur image between the sharp image and blurry image. After training the model, it will start giving sharp images as output. These both modules jointly solve the problem.

## 4 Algorithms

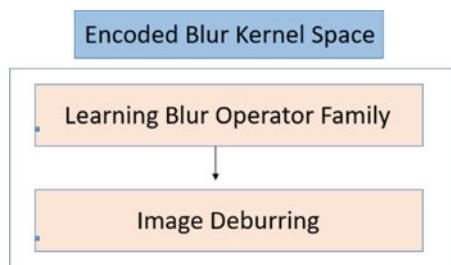
This proposed solution works in two modules. The first one is to remove haze, and the other one is to remove blur. DCP module takes input a hazy and blurry image. Its output will be non-hazy image which will act as input of encoded blur kernel space. Dark channel prior has five steps, and encoded blur kernel space has two steps. Initially, the output of dark channel prior is collected, and it is used to train encoded blur kernel space model. This model will learn directly from blurry image and its corresponding clean image. Once the blur kernel is learnt, then the model is ready to perform. Then, the output of DCP will directly act as input of encoded blur kernel space. Its output will be clean and sharp image.

Here, we are using OHAZY dataset to test our results for the first module of removing haze. The original image of dataset has been converted to  $300 \times 300$  size image for experimentation purpose. Using dark channel prior(DCP), haze has been removed, and the output is compared on PSNR and SSIM values. In second module, the output from first module is used, and encoded blur kernel space is applied to remove blurriness from the image. Then, the output is compared on PSNR and SSIM values. If the image does contain unseen blur, then it can also handled as the encoded kernel is very robust. If there is no blur and haze in the image, still it is handled properly when going with this process. Since it is divided into two modules which act one after another, it will not fail to show results (Figs. 4 and 5).

**Fig. 4** Removing haze from hazy and blur image



**Fig. 5** Removing blur from the image



## 5 Experiment Results

In this section, results of the experiment are discussed. Currently, first module of the algorithm is experimented, and its result is discussed. Then later, second module will be implemented, and results will be published.

### 5.1 Results

Dark channel prior(DCP) has been implemented and tested on OHAZY dataset. Here, the images are converted to the size of  $300 * 300$  for convenience in experiment. Figure 6 is showing the result for clearing the haze through dark channel prior and encoded blur Kernel Space. The result has a little more contrast than the ground truth, but it is giving promising result. DCP is clearing haze in the image, and EBKS is removing blur in the image. Table 1 is showing peak signal-to-noise ratio (PSNR)



**Fig. 6** First hazy image from OHAZY dataset. Then its dark channel after that the scene radiance then at fourth the output after dark channel prior and at the end ground truth

**Table 1** PSNR and SSIM values after Dehazing and Deblurring

Figure reference	PSNR value	SSIM value
Fig. 6a	27.61130569753935	0.6654022062071318
Fig. 6b	27.914224145783898	0.737012323438237
Fig. 6c	27.401768101526134	0.6772983558826853
Fig. 6d	27.460594993241948	0.663189491477003

and structural similarity index measure (SSIM) values for given input images and output images. In images, PSNR compares the quality of reconstruction of image with respect of original image. The average PSNR and SSIM value of the output is at the level of state-of-art work. The SSIM values clearly express that structural similarity has been properly maintained in the output dehazed image.

## 6 Conclusion

In this paper, we present a technique to eliminate haze for indoor robot vision. A very robust method of dark channel prior has been used. The results are very promising visually and empirically both. Encoded blur kernel space is used to remove blur normally, but here, it is used to remove the haze which resembles the properties of blurriness. There are many conditions where mobile robot vehicles are used, and the algorithms they used for recognition fail due to bad quality of light or haze and also due to improper handling of camera lens. In these conditions, our method fills the research gap and makes the indoor robot robust enough to handle these situations.

## References

1. Ancuti, C.O., Ancuti, C., Hermans, C., Bekaert, P.: A fast semi-inverse approach to detect and remove the haze from a single image. In: Asian Conference on Computer Vision, pp. 501–514. Springer (2010)
2. Chan, T., Wong, C.K.: Total variation blind deconvolution. *IEEE Trans. Image Process.* **7**(3), 370–375 (1998). <https://doi.org/10.1109/83.661187>
3. Eschbach, R., Kolpatzik, B.W.: Image-dependent color saturation correction in a natural scene pictorial image (1995). US Patent 5,450,217
4. Fattal, R.: Single image dehazing. *ACM Trans. Graphics (TOG)* **27**(3), 1–9 (2008)
5. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. *Adv. Neural inform. Process. Syst.* **27** (2014)
6. He, K., Sun, J., Tang, X.: Single image haze removal using dark channel prior. *IEEE Tran. Pattern Anal. Mach. Intell.* **33**(12), 2341–2353 (2010)
7. Koschmieder, H.: Theorie der horizontalen sichtweite. *Beitrage zur Physik der freien Atmosphare*, pp. 33–53 (1924)
8. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-Laplacian priors. *Adv. Neural Inform. Process. Syst.* **22**, 1033–1041 (2009)
9. Krishnan, D., Tay, T., Fergus, R.: Blind deconvolution using a normalized sparsity measure. In: CVPR 2011, pp. 233–240. IEEE (2011)
10. Kupyn, O., Budzan, V., Mykhailych, M., Mishkin, D., Matas, J.: Deblurgan: blind motion deblurring using conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8183–8192 (2018)
11. Kupyn, O., Martyniuk, T., Wu, J., Wang, Z.: Deblurgan-v2: deblurring (orders-of-magnitude) faster and better. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8878–8887 (2019)
12. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Understanding and evaluating blind deconvolution algorithms. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1964–1971. IEEE (2009)
13. Liu, G., Chang, S., Ma, Y.: Blind image deblurring using spectral properties of convolution operators. *IEEE Trans. Image Process.* **23**(12), 5047–5056 (2014)
14. McCartney, E.J.: Optics of the atmosphere: scattering by molecules and particles. New York (1976)
15. Nah, S., Hyun Kim, T., Mu Lee, K.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3883–3891 (2017)
16. Narasimhan, S.G., Nayar, S.K.: Contrast restoration of weather degraded images. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(6), 713–724 (2003)

17. Nayar, S.K., Narasimhan, S.G.: Vision in bad weather. In: Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 2, pp. 820–827. IEEE (1999)
18. Nishino, K., Kratz, L., Lombardi, S.: Bayesian defogging. *Int. J. Computer Vis.* **98**(3), 263–278 (2012)
19. Pan, J., Sun, D., Pfister, H., Yang, M.H.: Blind image deblurring using dark channel prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1628–1636 (2016)
20. Ren, D., Zhang, K., Wang, Q., Hu, Q., Zuo, W.: Neural blind deconvolution using deep priors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3341–3350 (2020)
21. Schechner, Y.Y., Narasimhan, S.G., Nayar, S.K.: Instant dehazing of images using polarization. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), vol. 1, pp. I–I. IEEE (2001)
22. Stark, J.: Adaptive image contrast enhancement using generalizations of histogram equalization. *IEEE Trans. Image Process.* **9**(5), 889–896 (2000). <https://doi.org/10.1109/83.841534>
23. Tan, R.T.: Visibility in bad weather from a single image. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE (2008)
24. Tao, X., Gao, H., Shen, X., Wang, J., Jia, J.: Scale-recurrent network for deep image deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8174–8182 (2018)
25. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Deep image prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9446–9454 (2018)

# Financial Modeling Using Deep Learning



Gunpreet Singh Walia, Nikita Sinha, Nalin Kashyap, Divakar Kumar,  
Subrata Sahana, and Sanjoy Das

**Abstract** With the advent of artificial intelligence, programs and procedures have streamlined the automated routine tasks and improved the customer service experience which helped businesses in their bottom line. The artificial intelligence world started exploding around 2000s, and it seemed that no industry or sector could resist its impact and prevalence by remaining untouched in these recent times. The world of quant finance and hedge funds is among those which did necessary ways to leverage the power of this ever-changing technology. The reinforcement recurrent learning (RRL) type of technique is being used to optimize different assets in the trading system and has reached outstanding results. Here, we use shares' capabilities to locate basics which include ee-e book value, dividends, or sales. We attempt to research those hazard and go back traits of fundamentally weighted and market cap-weighted indexes and hire numerous hazard-adjusted techniques to make certain that those returns' variations have been now no longer pushed with the aid of using hazard. To deal with this assignment of continuation of movement and operating in a multi-dimensional kingdom space, we proposed this stacked deep dynamic recurrent learning (SDDRL) structure to assemble a real-time ideal portfolio.

**Keywords** Quantitative finance · Statistical analysis · Portfolio optimization · GJR Garch model · Time series modeling · CAPM model · Risk factor

## 1 Introduction

These developing intelligent trading agents have attracted numerous investors. It gave investors and quants with the recent discovering in the technological sector. An opportunity manner to trade called computerized records pushed investment that is absolutely unique from conventional buying and selling techniques evolved primarily

---

G. S. Walia (✉) · N. Sinha · N. Kashyap · D. Kumar · S. Sahana

School of Engineering and Technology, Sharda University, Greater Noida, Uttar Pradesh, India  
e-mail: [2019539411.gunpreet@ug.sharda.ac.in](mailto:2019539411.gunpreet@ug.sharda.ac.in)

S. Das

Indira Gandhi National Tribal University, Amarkantak, India

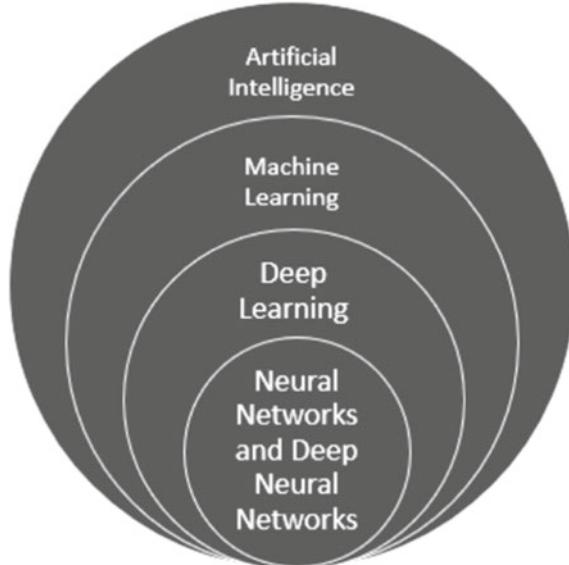
based totally on microeconomics theories [1]. The wise dealers are skilled via way of means of the usage of ancient records and loads of machine learning (ML) strategies that have been carried out to execute those trades for the education process [2]. A preferred instance encompasses reinforcement mastering methods which have been evolved to clear up those hard Markov selection problems.

Some of the works that can be performed in the financial sectors are:

- Portfolio management (CAPM model)
- Time series models
- Loan application evaluation
- Customer credit research
- Autoregressive integrated moving average (ARIMA)
- Seasonal autoregressive integrated moving average (SARIMA)
- Neural network models (LSTM, GRU cell).

In Fig. 1, artificial intelligence is the major field which simulates human intelligence processes by machines, where machine learning is sub-domain of AI which deals with practical segment of AI which does the automation of regular tasks and computation, whereas deep learning is broader type of machine learning which tends to perform well on complex-level task, where traditional machine learning fails [3].

**Fig. 1** Phases of artificial intelligence



## 2 Literature Survey

Frerk Saxen, Philipp Werner, Sebastian, Laslo Kredov, Su suggested that this data referred to the metrics and comparisons measuring the financial features of the economical space of these companies derived from these public filings [4]. There have been numerous attempts to use a value-based recurrent reinforcement learning approach in the financial market; a  $\text{TD}(\lambda)$  approach had been applied for financial modeling segment [5].

Use Q-learning to prepare decisions based on package allocation. However, such value methods are less suitable for online algorithmic trading due to the provision of naturally delayed response [6] and also because they simply mean having more dimensional action space. In addition, the Q-learning approach is much more stable than comparing RRL methods such as their income statements and balance sheets [5]. Examples of items taken from these documents include market margin, total revenue growth, and cash flow. Baseline models try to determine the factors that affect the risk and return of the asset [7]. You need to calculate the exposure return of the asset stream for each item. This is similar to the calculation of the risk premie [8].

Vinitha V, Valeguina T, in their work, they proposed a statistical analysis. This analysis is based on historical data, and risk exposures estimated on historical data may or may not affect the exposures going forward [9]. As such, computing the risk exposure of to a factor is not enough. You must put confidence bounds on that risk exposure and determine whether the risk exposure can even be modeled reasonably [10]. In risk analysis, we often model active returns (returns relative to a benchmark) and active risk (standard deviation of active returns, also known as tracking error or tracking risk) [11].

Handrich, Ehsan Othman describe about statistical inference. Since these make predictions about a large group based on smaller samples, it is traditionally broken into two segments, estimation and hypothesis testing [12]. Estimation provides values for specific things that you may be interested in, such as mean or variance, with a provided confidence interval. Confidence interval provides a region within which you can expect to find the true value of the parameter you estimated, as an estimation will almost never be exact. Confidence intervals use a set confidence level to choose how wide the interval should be; to achieve a higher confidence, we must report a wider interval [5].

## 3 Previous Models

Theoretical market impact models attempt to estimate transaction costs of trading by utilizing order attributes. There are many published market impact models. Here are some of the works:

- Quantopian Volume Slippage Model 2018

- Almgren et al. 2009
- Kissell et al. 2010
- J. P. Morgan Model 2012.

## 4 Problem Statement

We seek to increase the return and reduce the risk and transaction costs involved with that financial asset and how these constraints impact algorithm performance. By the end of this paper, you should be able to:

- Understand the attributes that influence risk and costs for any asset based on published market impact model research and our own derived model experience
- Understand the impact of turnover rate, transaction costs, and leverage on the strategy performance of that individual stock/asset
- Become familiar with how large institutional quant trading firms think about these constraints and manage them in a time-bound manner.

## 5 Technology Used

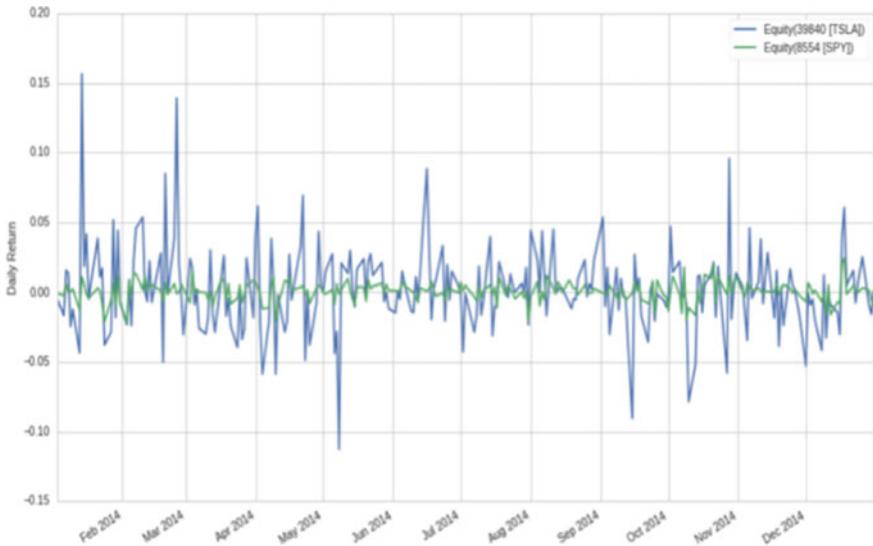
See Table 1.

## 6 Proposed Work

Factor models are a way of explaining the returns of one asset via a linear combination of the returns of other assets. The general form of a factor model looks familiar, as it is exactly the model type that a linear regression fits [13]. The w's can also be indicators rather than assets. An example might be an analyst estimation. List of technology used is described in Table 1.

**Table 1** List of technology used

S. No.	Name
1	Statistical analysis methods
2	CAPM model
3	ARIMA forecasting
4	Time series models
5	SARIMA model
6	Portfolio selection algorithms
7	TensorFlow and Scikit-learn
8	Deep learning (LSTM and GRU cell)



**Fig. 2** Spikes of beta returns with the normal returns

## 6.1 Beta

This is a measure of how an individual asset moves when the overall stock market increases or decreases. Thus, beta is a useful measure of the contribution of an individual asset to the risk of the market portfolio when it is added in small quantities [14].

Here, we are plotting betas of two different equities TSLA and SPY and comparing the extremeness of risk in both these equities. In the  $X$  axis, we are taking the time period from Feb 2014 to Jan 2015 distributing on a monthly basis, while on the  $Y$  axis, there are all the daily returns between ranges  $-0.15$  and  $0.20$ . In the graph, we are observing uneven spikes in TSLA stock, while a SPY is concerned in a mid-range of  $-0.05$  to  $0.05$  in Fig. 2.

## 6.2 Risk Exposure

More generally, this beta gets at the concept of how much risk exposure you take on by holding an asset. If an asset has a high beta exposure to the S&P 500, then while it will do very well while the market is rising, it will do very poorly when the market falls. A high beta corresponds to high speculative risk. You are taking out a more volatile bet.

Here, we value strategies that have negligible beta exposure to as many factors as possible. What this means is that all of the returns in a strategy lie in the portion of

the model and are independent of other factors. This is highly desirable, as it means that the strategy is agnostic to market conditions [15]. It will make money equally well in a crash as it will during a bull market. These strategies are the most attractive to individuals with huge cash pools such as endowments and sovereign wealth funds [16].

In our model, risk management is critical for constructing portfolios and building algorithms. Its main function is to improve the quality and consistency of returns by adequately accounting for risk. Any returns obtained by unexpected risks, which are always lurking within our portfolio, can usually not be relied upon to produce profits over a long time [17]. By limiting the impact of or eliminating these unexpected risks, the portfolio should ideally only have exposure to the alpha we are pursuing [20]. In this lecture, we will focus on how to use factor model in risk management.

These risk factors can be:

- Classical fundamental factors, such as those in the CAPM (market risk) or the Fama–French 3-Factor Model (price-to-book (P/B) ratio, volatility)
- Sector or industry exposure
- Macroeconomic factors, such as inflation or interest rates
- Statistical factors that are based on historical returns and derived from principal component analysis [21].

### ***6.3 The Fundamental Law of Asset Management***

The essential driver of the worth of any system is whether or not it gives a convincing danger changed return, i.e., the Sharpe ratio. As communicated in *The Foundation of Algo Success and “The Fundamental Law of Active Management”*, by Richard Grinold, Sharpe ratio can be decayed into two parts, ability and broadness, as:

Technically, this is the meaning of the information ratio (IR), yet for our motivations, it is identical to the Sharpe ratio. The IR is the proportion of the overabundance return of a portfolio over its benchmark per unit dynamic danger, i.e., the abundance return of a long-just portfolio less its benchmark per unit following blunder. In the hour of Grinold’s distribution, be that as it may, long/short contributing was an extraordinariness [22]. Presently, in the realm of mutual funds and long/short contributing, there is no benchmark. We look for outright returns; thus, for this situation, the IR is comparable to the Sharpe ratio.

In this condition, ability is estimated by information coefficient (IC), determined with Alphalens. The IC is basically the Spearman rank connection, used to correspond your forecast and its acknowledgment. Expansiveness is estimated as the quantity of free wagers in the period.

If the wagers are totally autonomous, then, at that point, broadness is the all-out number of wagers we have made for each individual resource, the quantity of resources times the quantity of periods. Assuming the wagers are not autonomous, then the successful broadness can be a whole lot not exactly the quantity of resources.



**Fig. 3** Factor analysis for market (MKT), small-minus big (SMB), high-minus low (HML)

We should see exactly how beta openness and area openness deal with compelling broadness.

#### 6.4 Factor Returns and Exposures

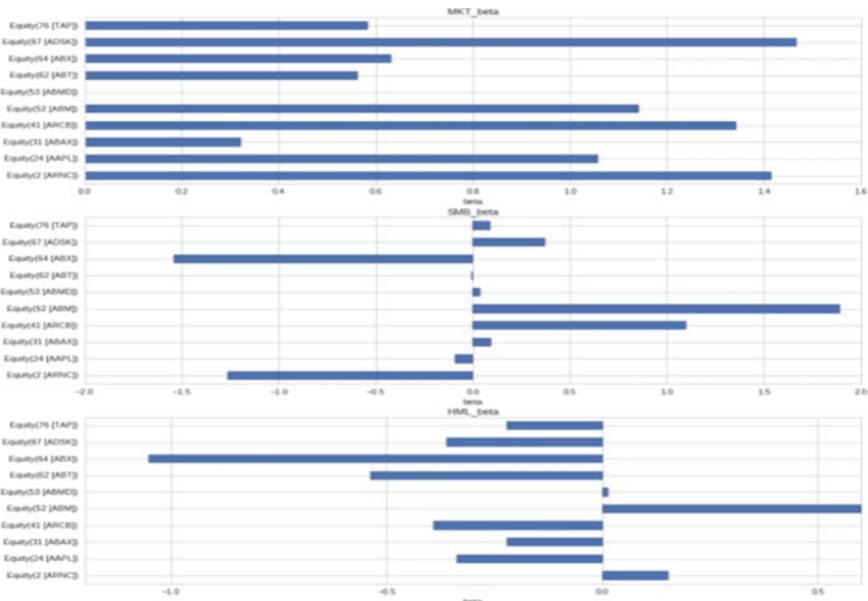
We will start with the classic Fama–French factors. The Fama–French factors are the market, company size, and company price-to-book (PB) ratio. We compute each asset’s exposures to these factors, computing the factors themselves using pipeline code borrowed from the fundamental factor models lecture in Fig. 3.

Here, we run our pipeline and create the return streams for high-minus low and small-minus big.

#### 6.5 Calculating the Exposures

Running a multiple linear regression on the fundamental factors for each asset in our universe, we can obtain the corresponding factor exposure for each asset. Here, we express beta for each asset. This shows us how much of each individual security’s return is made up of these risk factors.

We calculate the risk exposures on an asset-by-asset basis in order to get a more granular view of the risk of our portfolio. This approach requires that we know the



**Fig. 4** Different equity positioning on the Fama–French factors

holdings of the portfolio itself, on any given day, and is computationally expensive in Fig. 4.

## 6.6 Risk-Constrained Optimization

Currently, we are operating with an equal-weighted portfolio. However, we can reapportion those weights in such a way that we minimize the common factor risk illustrated by our common factor exposures. This is a portfolio optimization problem to find the optimal weights. We define the objective function as whichever business goal we value the highest. This can be something such as maximizing the Sharpe ratio or minimizing the volatility. Ultimately, we want to solve for in this optimization problem is the weights.

## 6.7 The Data Pipeline

Since every individual element has been added, it is currently an ideal opportunity to get every one of the important information immediately. In the calculation, this will occur once every day.

Later all the while, we will require a component to make a surmised S&P 500, so we will likewise incorporate another variable called SPY\_proxy (SPY is an ETF that tracks the S&P 500). The S&P 500 is an assortment of 500 of the biggest organizations exchanged on the financial exchange. Our translation of the S&P 500 is a gathering of 500 organizations with the best market capitalizations; be that as it may, the genuine S&P 500 will be marginally unique as Standard and Poor's.

## 6.8 Aggregation

Since we have every one of our information, we really want to control this to make a solitary positioning of the protections. Coming up next is the calculation for giving positions to each stock:

- Extract the S&P 500 from the arrangement of values and find the mean and standard deviation of each element for this dataset (standard\_frame\_compute)
- Use these figured qualities to normalize each variable (standard\_frame\_compute)
- Replace esteems that are more prominent than 10 or less than -10 with 10 and -10 separately to restrict the impact of exceptions (filter\_fn)
- Sum these qualities for every value and gap by the quantity of elements to give a worth between -10 and 10 (composite score).

## 6.9 Stock Choice

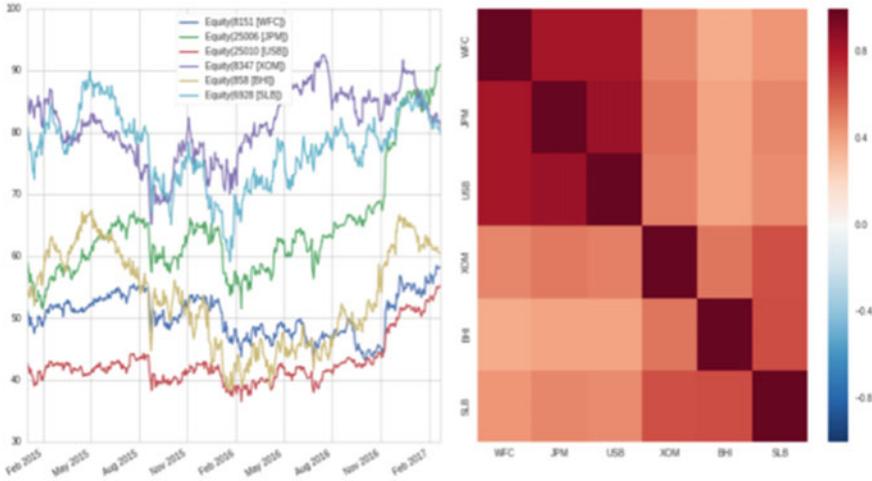
Since we have positioned our protections, we really want to pick a long bin and a short container. Since we really want to keep the proportion 130/30 among years and shorts, why not have 26 aches and 6 shorts (in the calculation, we will gage each of these similarly, giving us our ideal influence and openness).

## 6.10 Forecasts and Bet Correlation

We characterize a bet as the estimate of the remaining of a security return. This estimate can be understood—i.e., we purchase a stock and hence implicitly we gage that the stock will go up. What do we mean by lingering? With no extravagant math, this just method returns a less support. How about we work through three models. We utilize the Ledoit-Wolf covariance assessor to evaluate our covariance taking all things together [23].

### Case 1: No Hedge!

In the event that we go long on a bunch of protections, yet do not stand firm on any short situations, there is no support! So, the leftover is simply the stock return.



**Fig. 5** Bet correlation plot of equities in no hedge

$$r_{\text{resid},i} = r_i \quad (1)$$

Here is the correlation of our bets in this case as in Fig. 5.

Average pairwise correlation: 0.5644.

The result here is that we have six bets, and they are all very highly correlated.

### Case 2: Beta Hedge

In this case, we will assume that each bet is hedged against the market (SPY). In this case, the residual is calculated as:

$$r_{\text{resid},i} = r_i - \beta_i r_i \quad (2)$$

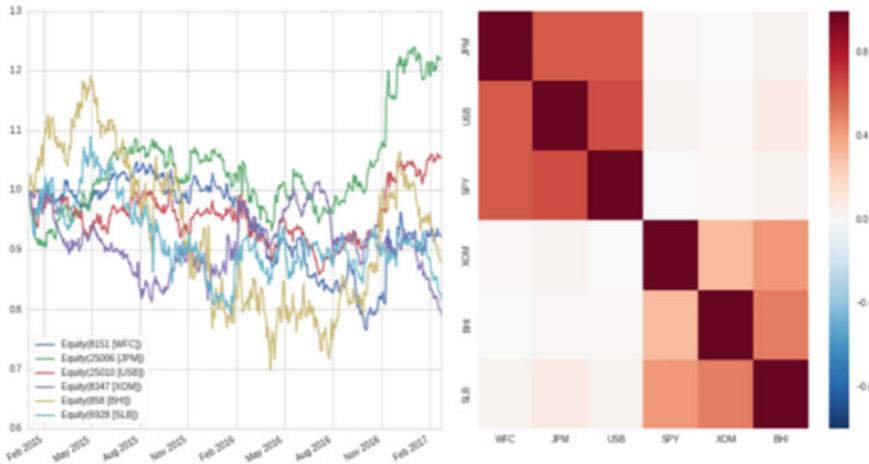
where BETA is the beta to the market of security is calculated with the CAPM and is the return of security.

Average pairwise correlation: 0.2256.

The beta fence has cut down the normal relationship significantly. Hypothetically, this ought to work on our expansiveness. Clearly, we are left with two profoundly connected bunches nonetheless. How about we see what happens when we fence the sector risk in Fig. 6.

### Case 3: Sector Hedge

The area return and the market return are themselves profoundly corresponded. Thus, you cannot do a multivariate relapse because of multicollinearity, an exemplary infringement of relapse presumptions. To support against both the market and a given security's area, you first gage the market beta residuals and afterward compute the area beta on those residuals



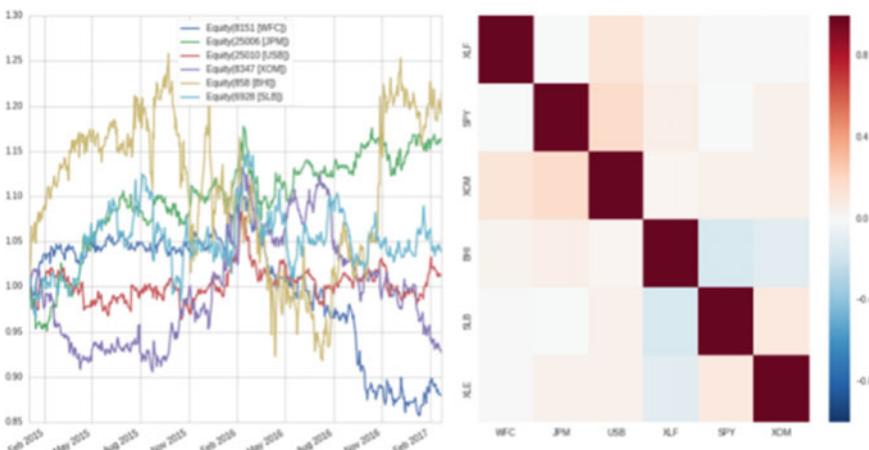
**Fig. 6** Bet correlation of equities in beta hedge

$$r_{\text{resid},i} = r_i - \beta_{i,\text{SECTOR},i} r_{\text{resid},i} \quad (3)$$

Here  $r$  is the residual between the security return and a market beta hedge and is the residual between that residual and a hedge of that residual against the relevant sector in Fig. 7.

Average pairwise correlation: 0.0281.

There we go! The sector hedge brought down the correlation between our bets to close to zero.



**Fig. 7** Bet correlation of equities in sector hedge

### 6.11 Calculating Effective Breadth

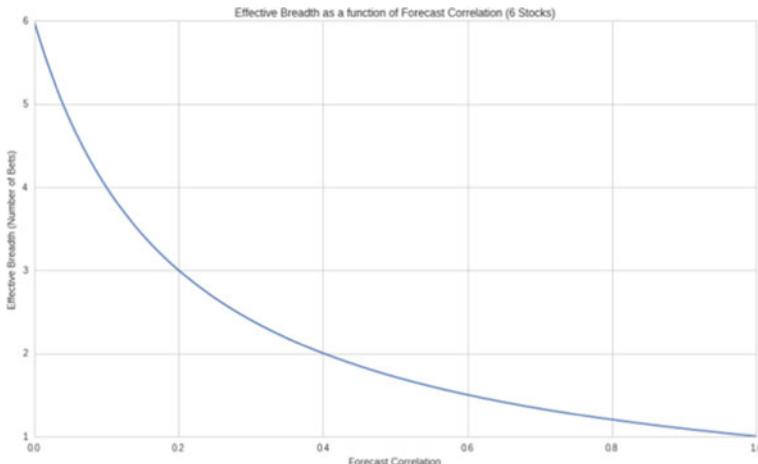
Buckle derives the “semi-generalized fundamental law of active management” under several weak assumptions. The key result of this paper (for us) is a closed-form calculation of effective breadth as a function of the correlation between bets. Buckle shows that breadth can be modeled as:

$$\text{BR} = \frac{N}{1 + \varphi(N - 1)} \quad (4)$$

where  $N$  is the number of stocks in the portfolio and  $\varphi$  is the assumed single correlation of the expected variation around the forecast.

Our experimentation, as shown in Fig. 8, has shown that on account of the long-just portfolio, where the normal connection is 0.56, we are successfully making just around 2 wagers. At the point when we support beta, with a subsequent normal relationship of 0.22, things improve three powerful wagers. At the point when we add the area support, we draw near to zero connection, and for this situation, the quantity of wagers approaches the quantity of resources, 6.

More autonomous wagers with a similar IC prompt higher Sharpe proportion.



**Fig. 8** Effective breadth as a function of forecast correlation

## 7 Conclusion

Currently, we are operating with an equal-weighted portfolio. However, we can reapportion those weights in such a way that we minimize the common factor risk illustrated by our common factor exposures. This is a portfolio optimization problem to find the optimal weights.

We define the objective function as whichever business goal we value the highest. This can be something such as maximizing the Sharpe ratio or minimizing the volatility. Ultimately, we want to solve for in this optimization problem is the weights.

## References

1. AIRMIC, ALARM: “IRM, a risk management standard,” The Institute of Risk Management. In: The National Forum for Risk Management in the Public Sector, The Association of Insurance and Risk Managers, London, UK (2002)
2. Arrington, C.E., Hillison, W., Jensen, R.E.: An application of analytical hierarchy process to model expert judgments on analytical review procedures. *J. Acc. Res.* 298–312 (1984)
3. Baccarini, D., Archer, R.: The risk ranking of projects: a methodology. *Int. J. Project Manag.* **19**(3), 139–145 (2001)
4. Bagranoff, N.A.: Using an analytic hierarchy approach to design internal control system. *J. Acc. EDP* **4**(4), 37–41 (1989)
5. Zacharias, O., Mylonakis, J., Askounis, D.Th.: RASM: a risk-based projects auditing selection methodology for large scale programs. *Int. Res. J. Finance Econ.* **11**, 181–194 (2007)
6. Chapman, C., Ward, S.: Project Risk Management: Processes, Techniques and Insights. Wiley (1996)
7. European Commission: Commission Regulation (EC) No. 1881/2006 of 19 December 2006 setting maximum levels for certain contaminants in foodstuffs. *Off. J. Eur. Union* **364**, 5–24 (2006)
8. Committee of Sponsoring Organizations of the Treadway Commission: Internal Control, Integrated Framework: Evaluation Tools, vol. 4. Committee of Sponsoring Organizations of the Treadway Commission (1992)
9. Coso, I.I.: Enterprise Risk Management-Integrated Framework. Committee of Sponsoring Organizations of the Treadway Commission 2 (2004)
10. Cooper, D.F., et al.: Project Risk Management Guidelines: Managing Risk in Large Projects and Complex Procurements. Wiley (2021)
11. Deumes, R., Knechel, W.R.: Economic incentives for voluntary reporting on internal risk management and control systems. *Audit.: J. Pract. Theory* **27**(1), 35–66 (2008)
12. Eilifsen, A., Knechel, W.R., Wallace, P.: Application of the business risk audit model: a field study. *Acc. Horizons* **15**(3), 193–207 (2001)
13. Almgren, R., et al.: Direct estimation of equity market impact. *Risk* **18**(7), 58–62 (2005)
14. Garman, M.B., Klass, M.J.: On the estimation of security price volatilities from historical data. *J. Bus.*, 67–78 (1980)
15. Kissell, R., Glantz, M., Malamut, R.: A practical framework for estimating transaction costs and developing optimal trading strategies to achieve best execution. *Financ. Res. Lett.* **1**(1), 35–46 (2004)

16. Rajesh, P., et al.: A Real Time Stock tendency prognostication using Quantopian. In: 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE (2020)
17. Chung, C., Kotlova, A., Zimmerman, G.: Market-neutral strategy for quantitative trading: long positive beta and shorting negative beta stocks. Demos: J Polit. Int. Relat. Econ. Bus. Finance Lake Forest College 1(1), 7 (2018)

# Application of Hybrid ANFIS-CSA Model in Suspended Sediment Load Prediction



Abinash Sahoo, Nihar R. Mohanta, Sandeep Samantaray,  
and Deba Prakash Satapathy

**Abstract** Accurate suspended sediment load (SSL) prediction and modelling in rivers play an essential part in designing engineering structures, environmental science and watershed management. Recently, data-driven models have been majorly employed for estimating SSL, and these models have generated effective outcomes. This study compares the prediction capability of adaptive neuro-fuzzy inference systems (ANFIS) and hybrid ANFIS-CSA (crow search algorithm) techniques for SSL prediction. For developing the models, monthly average SSL data of Brahmani river basin for 30 years were obtained. Reliability of applied techniques is evaluated based on statistical criteria, which include correlation coefficient ( $r$ ), root-mean-square-error (RMSE) and Nash–Sutcliffe efficiency (NSE). Obtained results revealed that ANFIS-CSA gave highest value of  $R$  and NSE, whereas lowest RMSE value than conventional ANFIS model. Results showed that combining optimisation algorithm and AI model improved the capability of models for SSL prediction.

**Keywords** Brahmani River · Sediment · ANFIS · ANFIS-CSA

## 1 Introduction

Along with their flow, rivers carry suspended sediments. Depending on course and discharge of river, these sediments get deposited at different places. However, these sediment deposits impact environmental health, portable water sources and agricultural activities. These deposits reduce the flow area, hence affecting transport of

---

A. Sahoo

Department of Civil Engineering, NIT Silchar, Silchar, Assam, India

N. R. Mohanta

Department of Civil Engineering, NIT Raipur, Raipur, Chhattisgarh, India

S. Samantaray (✉) · D. P. Satapathy

Department of Civil Engineering, OUTR, Bhubaneswar, Odisha, India

e-mail: [sandeep1139\\_rs@civil.nits.ac.in](mailto:sandeep1139_rs@civil.nits.ac.in)

D. P. Satapathy

e-mail: [dpsatapathy@cet.edu.in](mailto:dpsatapathy@cet.edu.in)

marine lives and eventually leading to a variation of river paths. Therefore, data of SSL and their change provides vital information to different experts [1, 2]. Several authorities need forecasted SSL data in rivers to function various hydraulic structures appropriately. Usually, SSL prediction is thought-provoking because of multiple aspects, comprising site-related modelling, site-related data, absence of numerous considered aspects utilised to predict and pattern difficulty. Lately, artificial intelligence (AI) techniques have been utilised for forecasting different hydrological phenomena and SSL estimation [3–9].

Azamathulla et al. [10] presented gene expression programming (GEP), regression model and ANFIS to model SSL relationship for three Malaysian rivers. Performance of GEP model was satisfactory compared to ANFIS and regression model. Vafakhah [11] applied ANN, ANFIS, ordinary kriging (OK) and cokriging CK to forecast SSL using stream flow and rainfall data from Kojor forest watershed near Caspian Sea. Olyaie et al. [12] compared precision of ANN, ANFIS, wavelet-based neural network (WNN) and traditional sediment rating curve (SRC) methods for estimating daily SSL in two gauge sites in the USA. Results indicated that WNN model computed SSL values closer to observed data than other models. Rezaei et al. [13] compared predictive capability of ANN, ANFIS, and least-square support vector machines and group method of data handling (GMDH) models for predicting SSL. Results indicated that LS-SVM produced better outcomes than other applied models. Babanezhad et al. [14] concentrated on sensitivity analysis of setting parameters of ANFIS on accuracy of numerical outcomes for finding best ANFIS model due to rapid oscillation in dataset. Results revealed that ANFIS provided fast computational time, and addition of more nodes for prediction did not change overall calculation time. Regardless of approval from researchers, there are certain limitations suffered by ANFIS that halts its application in problems with significant inputs, like computational expense and curse of dimensionality.

Liu et al. [15] constructed ELM-CSA model for analysing groundwater quality to resolve incompatibility of water parameters and fuzziness of water quality. Stability of proposed hybrid model was assessed against standard ELM and BP algorithm and found that ELM-CSA model proved to be a reliable and stable evaluation technique. Rezaie-Balf et al. [16] evaluated multivariate adaptive regression splines l(MARS), MARS-CSA (crow search algorithm) and gene expression programing (GEP), for forecasting daily solar radiation. Hybrid models combining ANFIS with various optimization algorithms have been successfully used to model different hydrological parameters [17–20]. Panahi et al. [21] used hybrid ANFIS-BWOA (black widow optimization algorithm) and SVM-BWOA to estimate SSL of two main tributaries of River Telar located in Northern part of Iran. Their performances were benchmarked against SVM-BA (bat algorithm), ANFIS-BA, ANFIS-PSO (particle swarm optimization), SVM-PSO, standalone ANFIS and SVM. Results showed highest value of  $R^2$  and lowest MAE for ANFIS-BWOA algorithm indicating it as the best performing model. Huangpeng et al. [22] predicted future hydropower generation from Jinanqiao Dam situated on Jinsha River considering effect of climate change using ANN-CSA model.

In present research, CSA is used to train ANFIS model for predicting monthly SSL of Brahmani River basin located in Odisha, India. Furthermore, performance of ANFIS-CSA is assessed against standalone ANFIS in sediment load prediction. Application of ANFIS-CSA in SSL prediction is the novelty of this study.

## 2 Study Area

River Brahmani lies between Latitude  $20^{\circ} 30' 10''$  to  $23^{\circ} 36' 42''$  N and Longitude  $83^{\circ} 52' 55''$  to  $87^{\circ} 00' 38''$  E in Eastern region of India. It is located amid Mahanadi basin (on left) and Baitarani basin (on right). The total catchment area of the basin is  $39,313 \text{ km}^2$ , having four distinctive subbasins i.e. Jaraikela, Tilga, Jenapur and Gomlai. The basin receives a mean annual precipitation of 1305 mm, with most rain happening during four months of south-west monsoon period (June–October). The minimum temperature drops to  $4^{\circ}\text{C}$  in winter, and in summer, maximum goes as high as  $47^{\circ}\text{C}$ . Brahmani River basin consists of surplus water and is an important source of water supply for different industries and towns and irrigation in Orissa, India (Fig. 1).

## 3 Material and Methodology

### 3.1 ANFIS

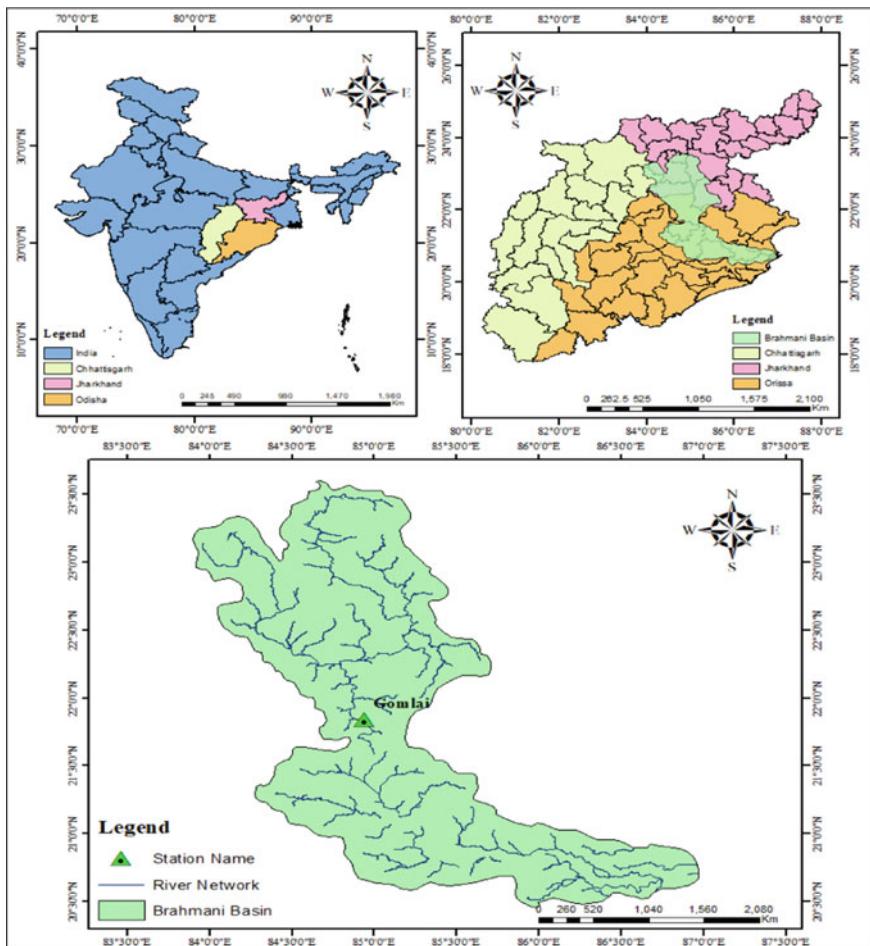
ANFIS is a combination of ANN and fuzzy inference system (FIS) that takes advantage of both FIS and ANN [23, 24]. In an ANFIS network, ANN extracts fuzzy rules from provided input, and parameters of fuzzy membership function (MF) were flexibly utilised during the hybrid learning process. ANFIS creates a relationship between input and output depending on human knowledge utilising input-output data sets and employing an algorithm on basis of hybrid learning. Utilising a pattern of input and output dataset, ANFIS yields resulting if-then rules:

$$\text{if}(x = A_1) \text{ and } (y = B_1) \rightarrow f = p_1x + q_1x + r_1 \quad (1)$$

$$\text{if}(x = A_2) \text{ and } (x = B_2) \rightarrow f = p_2x + q_2x + r_2 \quad (2)$$

where  $A_1$ ,  $B_1$ ,  $A_2$  and  $B_2$ —associated MFs,  $x$  and  $y$ —inputs,  $p_1$ ,  $q_1$ ,  $r_1$  and  $r_2$ —consequent parameters (Fig. 2).

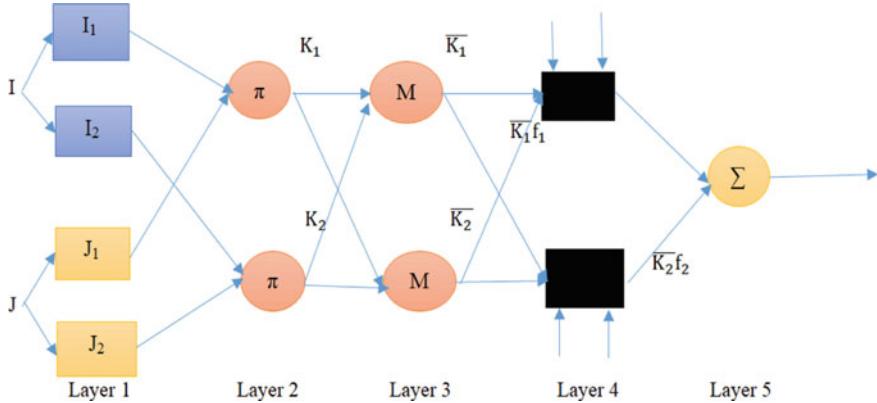
For enhancing the predictive precision of ANFIS and refraining from subsiding into local optima, parameter learning is executed by CSA. CSA is utilised for training parameters that comprise a reverse channel and a forward channel.



**Fig. 1** Location of Brahmani River basin

### 3.2 CSA

Among birds and animals, crows are one of the cleverest creatures. Relative to their body size, they have the largest brain, and because of this, their brains are marginally inferior to that possessed by humans. Crows have shown self-consciousness in mirror tests and can also create tools. They can use tools, memorise faces, connect sophisticatedly and hide and recover food in different seasons. This behaviour helps them determine where food is hidden by others and rob it once the owner departs. If a crow recognises that one more crow is behind it, it flies to a different location to deceive the follower. Based on this approach, Askarzadeh [25] recommended CSA



**Fig. 2** General structure of ANFIS

as an evolutionary algorithm for solving a broad variety of problems on basis of subsequent parts:

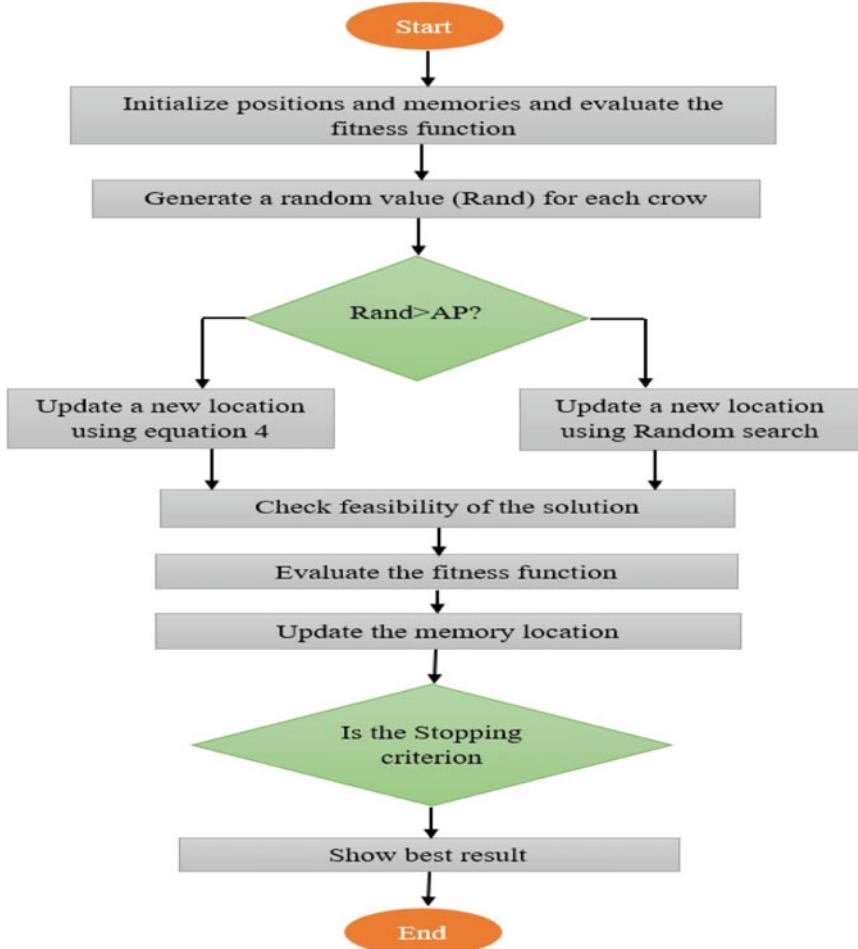
- (1) Every single crow is part of entire group;
- (2) Crows can reminisce location of their food (hidden place);
- (3) They chase each other to steal their hidden food; and
- (4) Crow uses a probability for protecting their hidden food from theft

In standard CSA, with the perception of other crows, position of each crow varies. For example, supposing  $i$ th crow pursues  $j$ th crow for stealing hidden food from  $j$ th crow, the  $i$ th crow updates its position utilising following formula to steal hidden food from  $j$ th crow:

$$X^{i,\text{iter}+1} = \begin{cases} X^{i,\text{iter}} + r_i f_l^{i,\text{iter}} (M^{j,\text{iter}} - X^{i,\text{iter}}); r_i \geq AP_i^{\text{iter}}, \\ X^{i,\text{iter}+1} = \text{a random position in search space} \end{cases} \quad (3)$$

where  $f_l^{i,\text{iter}}$  and  $AP_i^{\text{iter}}$ —flight capability and awareness probability of  $i$ th crow in  $\text{iter}$ th iteration, respectively.  $r_i$ —arbitrary number for  $i$ th crow and  $M^{j,\text{iter}}$ —memory position of  $j$ th crow for  $\text{iter}$ th iteration. Again,  $M^{i,\text{iter}+1}$  is updated using equation given below (Fig. 3):

$$M^{i,\text{iter}+1} = \begin{cases} X^{i,\text{iter}}; f(X^{i,\text{iter}+1}) \geq f(M^{i,\text{iter}}) \\ M^{i,\text{iter}} \end{cases} \quad (4)$$



**Fig. 3** Flowchart of CSA algorithm

### 3.3 Evaluating Constraint

Suspended sediment load data of 30 years (1990–2019) was collected. About 70% of collected data i.e. 1990–2010, were taken for training while the rest 30% i.e. 2011–2019 for testing proposed models. RMSE,  $r$  and NSE statistical indices are used for assessing effectiveness of proposed models:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n [T_{\text{obs}} - T_{\text{pre}}]^2} \quad (5)$$

$$r = \left( \frac{\frac{1}{n} \sum_{i=1}^n [T_{\text{obs}} - \bar{T}_{\text{obs}}] (T_{\text{pre}} - \bar{T}_{\text{pre}})}{\sqrt{\frac{1}{n} \sum_{i=1}^n [T_{\text{obs}} - \bar{T}_{\text{obs}}]^2} \sqrt{\frac{1}{n} [T_{\text{pre}} - \bar{T}_{\text{pre}}]^2}} \right)^2 \quad (6)$$

$$\text{NSE} = \frac{\sum_{k=1}^N [T_{\text{obs}} - T_{\text{pre}}]^2}{\sqrt{\sum_{k=1}^N [T_{\text{obs}} - \bar{T}_{\text{obs}}]^2}} \quad (7)$$

where  $T_{\text{pre}}$  and  $T_{\text{obs}}$  = predicted and observed sediment values;  $\bar{T}_{\text{pre}}$  and  $\bar{T}_{\text{obs}}$  = average of predicted and observed sediment values;  $N$  = data length.

## 4 Results and Discussions

In the present study, a simple and efficient model is proposed for SSL prediction. The assessment results of hybrid and standalone models during training and testing periods are provided in Table 1. In both stages, ANFIS-CSA provided superior predictions than ANFIS. Different MFs, number of inputs, and number of iterations are utilised for finding best prediction outcomes. First, ANFIS network is trained with collected data, and secondly, CSA is used to find optimal parameters of ANFIS.

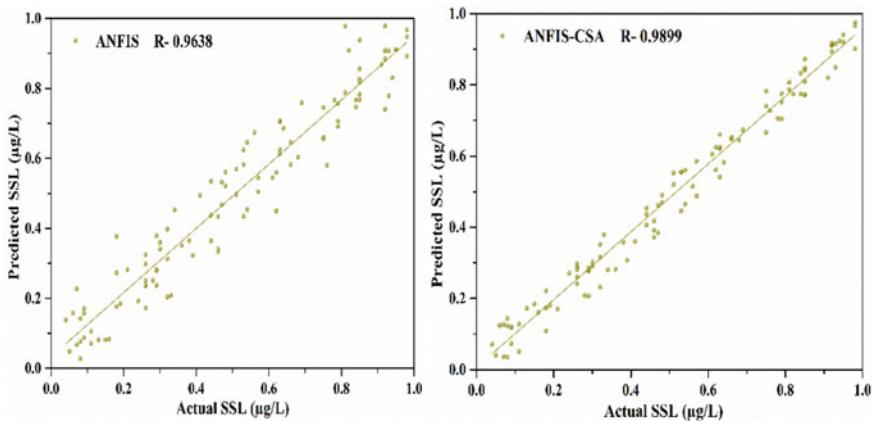
Scatter plots of observed versus predicted sediment load for ANFIS and ANFIS-CSA prediction models during testing phase are shown in Fig. 4. It signifies a correlation coefficient ( $r$ ) between predicted and actual data for best input combination. Based on the figures, best model is attained when ANFIS-CSA is taken as algorithm. In this case,  $r$  is more than 0.9899 for testing data. Correspondingly for ANFIS, prominent value of R is 0.9638 in testing set.

Figure 5a and b show the time series plot of observed and predicted values of daily SSL for both training and testing periods; it can be said that the daily SSL is predicted much better by ANFIS-CSA than the conventional ANFIS model.

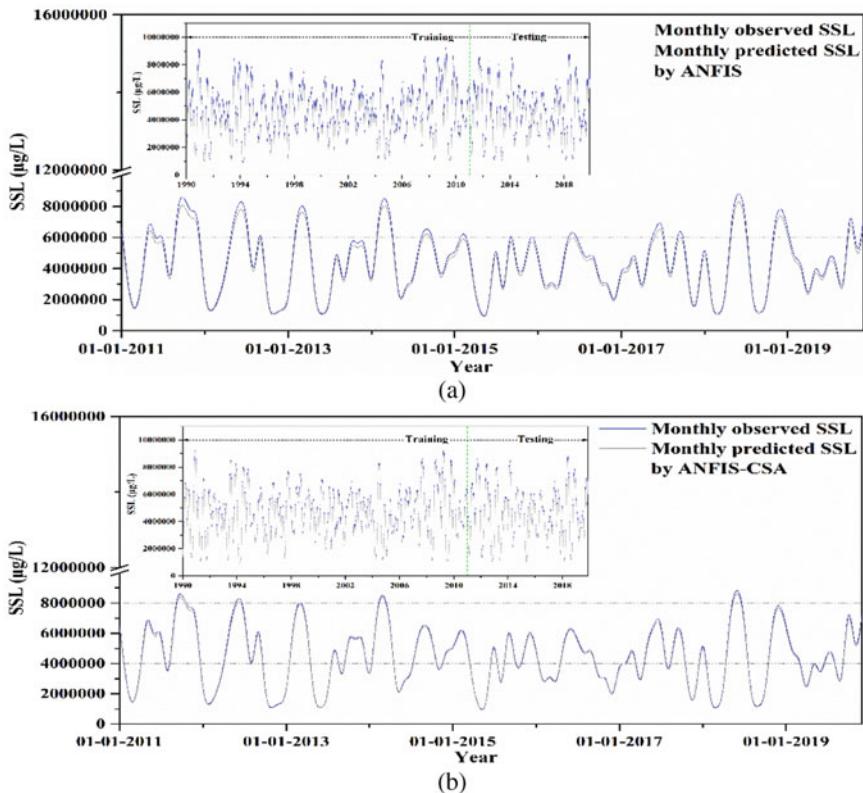
A clear graphical illustration of how ANFIS and ANFIS-CSA models have forecasted SSL magnitude is presented in Fig. 6. Figure 6 shows performance of proposed

**Table 1** Performance of model

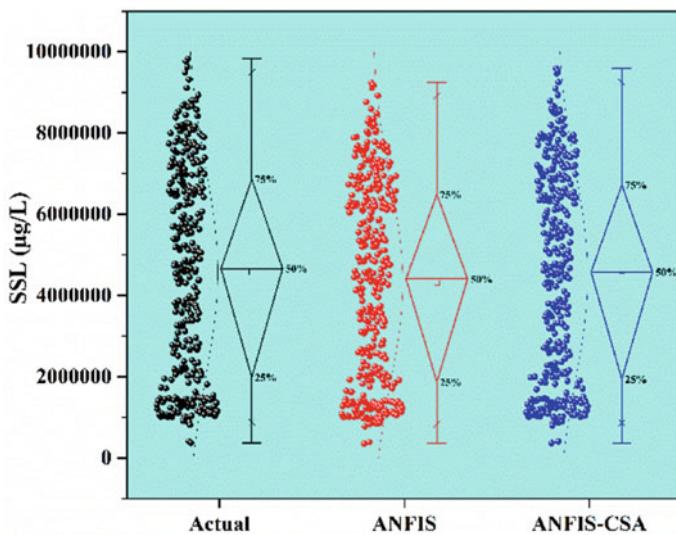
Technique	Input	Membership function	CC	RMSE	NSE	CC	RMSE	NSE
			Training phase			Testing phase		
ANFIS	ANFIS#1	Gbell	0.9627	6.8264	0.9571	0.9586	8.2689	0.9482
	ANFIS#2	Gauss 2	0.965	5.998	0.9617	0.9615	7.0549	0.9557
	ANFIS#3	Gbell	0.9682	4.6825	0.977	0.9638	5.2397	0.9702
ANFIS-CSA	ANFIS-CSA#1	Gauss	0.9879	1.5236	0.9905	0.9829	2.995	0.9764
	ANFIS-CSA#2	Tri	0.9905	0.3548	0.9931	0.9874	2.1974	0.9804
	ANFIS-CSA#3	Gbell	0.9927	0.0019	0.9964	0.9899	0.8637	0.9913



**Fig. 4** Scatter plot showing actual versus predicted sediment load during testing period



**Fig. 5** SSL prediction results of sediment load at Gomlai station using **a** ANFIS and **b** ANFIS-CSA



**Fig. 6** Boxplot of applied models for SSL prediction

ANFIS-CSA and ANFIS models in the form of a box plot. It is observed that ANFIS-CSA model performed very well as their prediction values are closely related to actual dataset of the basin. It represents the forecasting capability of two models that need to be further investigated, taking more case studies. It is concluded that ANFIS-CSA's forecasting stability and accuracy are higher than conventional ANFIS.

## 5 Conclusion

To understand variations in the movement of nonpoint source pollution, accurate SSL prediction is a vital factor. Concerning the significance of SSL, as a complicated phenomenon, the use of data-driven models helps in accurate SSL prediction. Because of the high prediction capability of optimised models, a robust ANFIS-CSA model was utilised for predicting SSL in one subbasin of Odisha, India. The input dataset was comprised of observed monthly sediment load. The results showed that ANFIS-CSA model was an efficient tool than ANFIS to predict the SSL with low RMSE and high R and NSE. The RMSE of ANFIS-CSA during training period was 0.0019, whereas R and NSE values were 0.9809 and 0.9964, respectively. For future studies, alternative meteorological and hydrological factors can be considered for modelling SSL process. Present research can be a preliminary study for future works and other areas (other data-driven models or optimisation algorithms) pursuing SSL prediction in a watershed scale utilising improved models.

## References

- Samantaray, S., Ghose, D.K.: Assessment of suspended sediment load with neural networks in arid watershed. *J. Inst. Eng. (India) Ser. A* **101**(2), 371–380 (2020)
- Samantaray, S., Ghose, D.K.: Sediment assessment for a watershed in arid region via neural networks. *Sādhanā* **44**(10), 1–11 (2019)
- Mohanta, N.R., Panda, S.K., Singh, U.K., Sahoo, A., Samantaray, S.: MLP-WOA is a successful algorithm for estimating sediment load in Kalahandi Gauge Station, India. In: Proceedings of International Conference on Data Science and Applications 2022, LNNS, vol. 288, pp. 319–329. Springer, Singapore (2022)
- Samantaray, S., Sahoo, A.: Prediction of suspended sediment concentration using hybrid SVM-WOA approaches. *Geocarto Int.* 1–27 (2021)
- Samantaray, S., Sahoo, A., Ghose, D.K.: Assessment of sediment load concentration using SVM, SVM-FFA and PSR-SVM-FFA in arid watershed, India: a case study. *KSCE J. Civ. Eng.* **24**(6), 1944–1957 (2020)
- Samantaray, S., Sahoo, A.: Assessment of sediment concentration through RBNN and SVM-FFA in Arid Watershed, India. In: Smart Intelligent Computing and Applications 2020, SIST, vol. 159 pp. 701–709. Springer, Singapore (2020)
- Talebi, A., Mahjoobi, J., Dastorani, M.T., Moosavi, V.: Estimation of suspended sediment load using regression trees and model trees approaches (Case study: Hyderabad drainage basin in Iran). *ISH Journal of Hydraulic Engineering* **23**(2), 212–219 (2017)
- Zounemat-Kermani, M., Kişi, Ö., Adamowski, J., Ramezani-Charmahineh, A.: Evaluation of data driven models for river suspended sediment concentration modeling. *J. Hydrol.* **535**, 457–472 (2016)
- Nourani, V., Mogaddam, A.A. Nadiri, A.O.: An ANN-based model for spatiotemporal groundwater level forecasting. *Hydrol. Process. Int. J.* **22**(26), 5054–5066 (2008)
- Azamathulla, H.M., Cuan, Y.C., Ghani, A.A., Chang, C.K.: Suspended sediment load prediction of river systems: GEP approach. *Arab. J. Geosci.* **6**(9), 3469–3480 (2013)
- Garg, C., et al.: Adaptive fuzzy logic models for the prediction of compressive strength of sustainable concrete. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_47](https://doi.org/10.1007/978-981-16-2164-2_47)
- Olyaei, E., Banjerd, H., Chau, K.W., Melesse, A.M.: A comparison of various artificial intelligence approaches performance for estimating suspended sediment load of river systems: a case study in United States. *Environ. Monit. Assess.* **187**(4), 1–22 (2015)
- Rezaei, K., Pradhan, B., Vadiati, M., Nadiri, A.A.: Suspended sediment load prediction using artificial intelligence techniques: comparison between four state-of-the-art artificial neural network techniques. *Arab. J. Geosci.* **14**(3), 1–13 (2021)
- Babanezhad, M., Behroyan, I., Marjani, A., Shirazian, S.: Artificial intelligence simulation of suspended sediment load with different membership functions of ANFIS. *Neural Comput. Appl.* **33**(12), 6819–6833 (2021)
- Sharma, P., et al.: Vibration-based diagnosis of defect embedded in inner raceway of ball bearing using 1D convolutional neural network. *Artif. Intell. Future Gener. Robot.* 25–36 (2021). <https://doi.org/10.1016/B978-0-323-85498-6.00011-3>
- Rezaie-Balf, M., Maleki, N., Kim, S., Ashrafiyan, A., Babaie-Miri, F., Kim, N.W., Chung, I.M., Alaghmand, S.: Forecasting daily solar radiation using CEEMDAN decomposition-based MARS model trained by crow search algorithm. *Energies* **12**(8), 1416 (2019)
- Samantaray, S., Sahoo, A.: A comparative study on prediction of monthly streamflow using hybrid ANFIS-PSO approaches. *KSCE J. Civ. Eng.* **25**(10), 4032–4043 (2021)
- Sahoo, A., Samantaray, S., Paul, S.: Efficacy of ANFIS-GOA technique in flood prediction: a case study of Mahanadi river basin in India. *H<sub>2</sub>Open J.* **4**(1), 137–156 (2021)
- Sridharam, S., Sahoo, A., Samantaray, S., Ghose, D.K.: Estimation of water table depth using Wavelet-ANFIS: a case study. In: Communication Software and Networks 2021, LNNS, vol. 134, pp. 747–754. Springer, Singapore (2021)

20. Samantaray, S., Sumaan, P., Surin, P., Mohanta, N.R., Sahoo, A.: Prophecy of groundwater level using hybrid ANFIS-BBO approach. In: Proceedings of International Conference on Data Science and Applications 2022, LNNS, vol. 288, pp. 273–283. Springer, Singapore (2022)
21. Panahi, F., Ehteram, M., Emami, M.: Suspended sediment load prediction based on soft computing models and black widow optimization algorithm using an enhanced gamma test. Environ. Sci. Pollut. Res. 1–21 (2021)
22. Singh, P., et al. Development of prediction models to determine compressive strength and workability of sustainable concrete with ANN. In: Mekhilef S., Favorskaya, M., Pandey, R.K., Shaw, R.N. (eds.) Innovations in Electrical and Electronic Engineering. Lecture Notes in Electrical Engineering, vol. 756. Springer, Singapore (2021). [https://doi.org/10.1007/978-981-16-0749-3\\_59](https://doi.org/10.1007/978-981-16-0749-3_59)
23. Samantaray, S., Sahoo, A.: Modelling response of infiltration loss toward water table depth using RBFN, RNN, ANFIS techniques. Int. J. Knowl.-Based Intell. Eng. Syst. **25**(2), 227–234 (2021)
24. Agnihotri, A., Sahoo, A., Diwakar, M.K.: Flood prediction using hybrid ANFIS-ACO model: a case study. In: Inventive Computation and Information Technologies 2022, LNNS, vol. 336, pp. 169–180. Springer, Singapore (2022)
25. Askarzadeh, A.: A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. Comput. Struct. **169**, 1–12 (2016)

# An Enhanced Technique to Improve the Performance of Multi-image Compression Technique



Dibyendu Barman, Abul Hasnat, and Bandana Barman

**Abstract** In the field of multi-image compression technique, for  $n$  number of channels of more than images a single common code vector is formed by adjusting some tuning parameters. The size of the common code vector is very much less compared to the total size of generated code vector of  $n$  number of channels of multiple images that reduces amount of memory require to store it, but at the same time a huge amount of data loss occurs which degrade the visual quality of the images. In this work, a new method is proposed that reduces the amount of data loss occur and improve the visual quality of the decompressed images. This proposed method generates two common code vectors instead of one. Those code vectors of  $n$  number of channels of multiple images that play vital role in the formation of decompressed images are put as it is in first common code vector pool, whereas the code vectors that are less important are put in the second code vector pool by adjusting some tuning parameters. In these processes, very less amount of data loss occurs compared to the multi-image compression technique found in literature. This proposed method is applied on many standard images found in literature and images from UCID v.2 database. The experimental results are analyzed in terms of SSIM, PSNR, and the compression ratio achieved. From the experimental results, it is analyzed that the visual quality of the decompress images is improved significantly in terms of PSNR and SSIM by scarifying some amount of compression ratio.

**Keywords** Compression ratio · Image compression · Multi-image compression · PSNR · SSIM

---

D. Barman (✉) · A. Hasnat

Government College of Engineering & Textile Technology Berhampore, Berhampore, West Bengal, India

e-mail: [dibyendu.barman@gcettb.ac.in](mailto:dibyendu.barman@gcettb.ac.in)

B. Barman

Kalyani Government Engineering College, Nadia, West Bengal, India

## 1 Introduction

In the field of data compression, image compression technique plays a vital role to represent the image, transfer the image over a communication medium. Images are compressed based on the three redundancies: i. coding redundancy, ii. inter pixel redundancy, and iii. psychovisual redundancy [1–11]. Basically, an image is compressed by two steps: i. compression reduces the amount of memory required to store the image and ii. decompression- reverse process of compression that reform the image. In the literature, two types of image compression technique are found [1–6]. i. lossless image compression—Visual quality of the decompressed image is almost same as original, but compression ratio achieved is not up to the mark. This lossless [1–11] image compression is much suitable where visual quality of the image is major issue rather than compression ratio. Arithmetic encoding, huffman encoding, run length encoding (RLE), and LZW [4–6] are well-known lossless image compression technique that are found in the literature. (ii) lossy image compression [4–6]. In case of lossy image compression, the visual quality of the decompressed image is not up to the mark, but it achieved very high amount of compression ratio. Vector quantization, color image quantization, JPEG, and JPEG2000 [4–6] are some well-known lossy image compression technique that found in the literature.

## 2 Literature Review

In literature a very less amount works are found on multi-image compression techniques and the works that improve the performance of the algorithm.

In 2019, Hasnat et al. [4] proposed a method where chrominance channels of multiple images in YCBCR color model can be compressed together and form only one 3D common code vector. Size of this 3D code vector matrix is much lesser than total size of individual code vectors. To maintain the visual quality of the decompressed images, luminance channels are compressed individually by normal vector quantization algorithm. To retain better quality of the decompressed images, the 3D common code vector matrix is optimized by PSO optimization technique. The compression ratio achieved by this method is significantly better than normal vector quantization algorithm, but at the same time, it also decreases the visual quality of the images.

Barman et al. [6] in 2021 proposed a new technique to improve the compression ratio of the algorithm by representing each value of code vector matrix by five bits instead of eight bits. It reduces the size of code vectors by 37.5% that improve the compression ratio of the overall algorithm by 2–3%. But at the same time, some amount of data loss occurs that reduces the visual quality of the decompressed images.

In 2022, Barman et al. [5] proposed a method that generates a common code vector from each channel of multiple color images by adjusting some tuning parameters. The size of the common code vector is much lesser than the total size of the individual

code vectors that improves the performance of the overall algorithm in terms of compression ratio, but at the same time degrade the visual quality of the image due to data loss.

Hasnat et al. [9] in 2017 proposed modified vector quantization method which improves the visual quality of images by removing blocking artefacts that appear in the decompressed images of the clustering algorithms. But, the limitation of this method is that it decreases the performance of the algorithm in terms of compression ratio.

### 3 Proposed Method

In case of multi-image compression technique [4, 5], multiple images can be compressed together by forming a single common code vector from image channels of multiple images to achieve better compression ratio. But at the same time, large amount of data loss occur that degrade the visual quality of the images. In this proposed work, a new method is introduced where two common code vectors are formed instead of one. First code vector matrix holds those code words same as it is those play vital roles in the formation of decompressed images. The code words that play fewer roles in the formation of decompressed images are put in second common code vector matrix by adjusting some tuning parameters. The steps of the proposed method are discussed below.

**Step 1:** Apply normal vector quantization algorithm on each channel of multiple color images. That generates  $n * 3$  number of index matrix  $idx_{i,j}$  and code vector matrix  $cv_{i,j}$  where  $i = 1, 2, \dots, n$   $j = 1$  to  $3n =$  no of images. The size of each code vector will be  $256 * \text{block\_size} * \text{block\_size}$  where 256 is the number of clusters. The index matrix size will be  $h * w / \text{block\_size} * \text{block\_size}$  where  $h =$  height of the image and  $w =$  width of the image.

**Step 2:** Each cell of the index matrix  $idx_{i,j}$  refers the index value of the respective code vector matrix which it refers to.

Figures 1 and 2 shows an index matrix and its corresponding code vector matrix.

**Step 3:** Count, number of times an index value presents in index matrix and store the result in count\_index matrix.

Figure 3 shows the count\_index matrix of the index matrix shown in Fig. 1.

**Step 4:** Set a tuning parameter  $t_p$ . Put each code word of all  $n * 3$  channels whose count value in count\_index matrix greater or equal to  $t_p$  into first common code vector matrix  $ccv1$ . Rest code vectors of  $n * 3$  channels will be placed in second common code vector matrix  $ccv2$  by adjusting the tuning parameters of the multi-image compression technique.

**Step 5:** Update the index matrixes as per the index value of the respective code words in two common code matrixes (Fig. 4).

11	2	10	2	5
6	21	2	2	1
2	11	5	11	2
9	51	70	74	99
11	2	15	19	37

**Fig. 1** Sample index matrix  $idx_{i,j}$ 

	1	2	3	15	16
1	15	21	13	---	15
2	70	71	15	---	85
3	15	32	33	---	34
---	---	---	---	---	---
---	---	---	---	---	---
255	75	71	12	---	32
256	11	1	10	---	16
					17

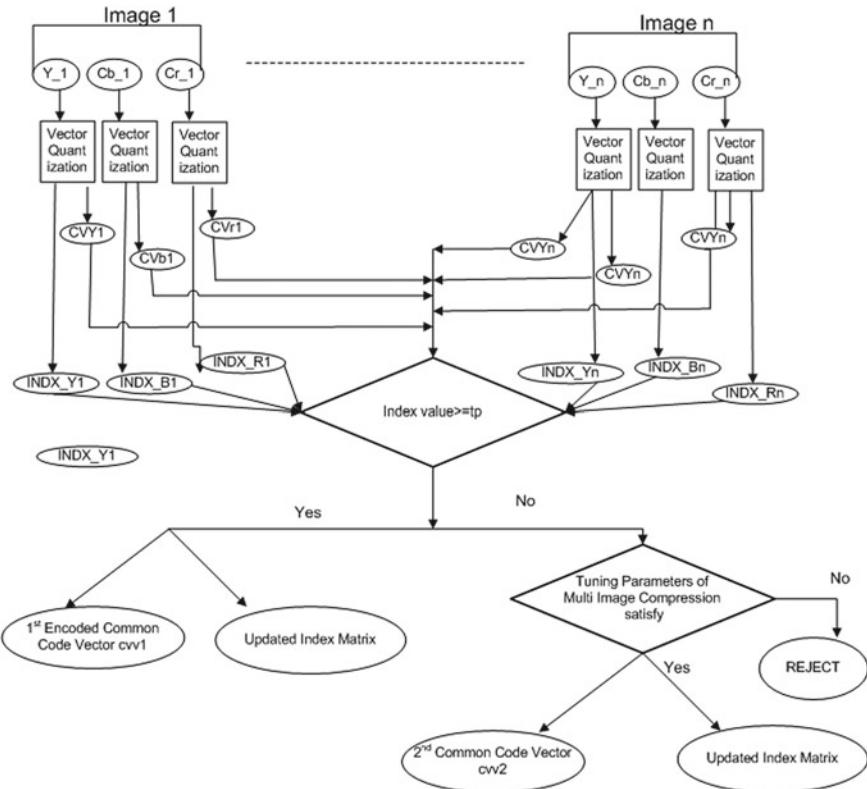
**Fig. 2** Code vector matrix  $cv_{i,j}$  of Index matrix in Fig. 1

Index	11	2	10	5	6	21	1	9	51	70	74	99	19	37
Count	4	7	1	2	1	1	1	1	1	1	1	1	1	1

**Fig. 3** Count\_index of index matrix  $idx_{i,j}$  of Fig. 1

## 4 Experimental Results

The proposed method is implemented in MATLAB 2018 and applied many standard images and image from ucid v.2 databases. The performance of the proposed method is measured by three parameters: SSIM, PSNR, and compression ratio [12, 13]. The experimental results show that the visual quality of the decompressed images



**Fig. 4** Entire proposed method is shown

increased significantly by sacrificing some amount of data loss which reduces the compression ratio.

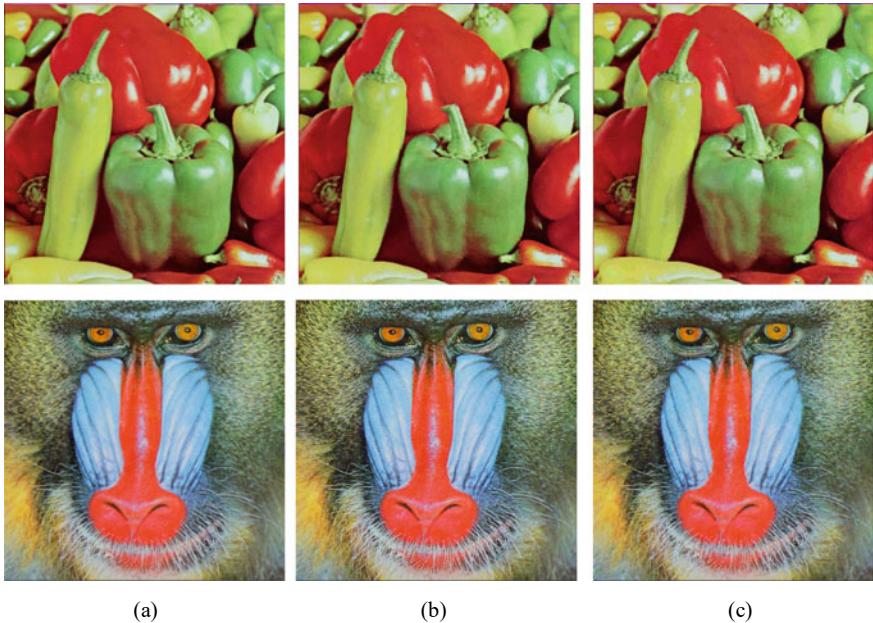
Original pepper and baboon images, decompressed images using multi-image compression technique, and decompressed images using the proposed method are shown in Fig. 5.

From Fig. 5, it can be easily seen that visual quality of the proposed method is much better than the decompressed images using multi-image compression technique [4, 5].

Comparative study of structural similarity index measure(SSIM) [12, 13] between decompressed images using multi-image compression and proposed method is shown in Table 1.

From Table 1, it can be analyzed that SSIM [12, 13] value of proposed method is much better than existing multi-image compression technique [4, 5].

Comparative result using PSNR between multi-image compression and proposed method is shown in Table 2.



**Fig. 5** **a** Original image and images using, **b** multi-image compression, and **c** proposed method

**Table 1** SSIM between multi-image compression technique and proposed method

Image name	Multi-image compression	Proposed method
4.1.01. tiff (Girl)	0.8722	0.8913
4.1.02. tiff (Couple)	0.8638	0.8869
4.1.05. tiff (House)	0.9508	0.9639
4.1.08.tiff (Jelly Beans)	0.9592	0.9734
4.2.02.tiff (Tiffany)	0.9591	0.9724
4.2.03.tiff (Baboon)	0.8032	0.8336
4.2.07.tiff (Pepper)	0.9635	0.9761
House.tiff	0.8974	0.9168

From Table 2, it can be seen that decompressed using proposed method has better visual quality than existing multi-image compression in terms of PSNR. Space reduction using multi-image compression and proposed method is shown in Table 3.

From Table 3, it can be seen that space reduction using proposed method is almost same or little bit lower than existing multi-image compression technique [4, 5].

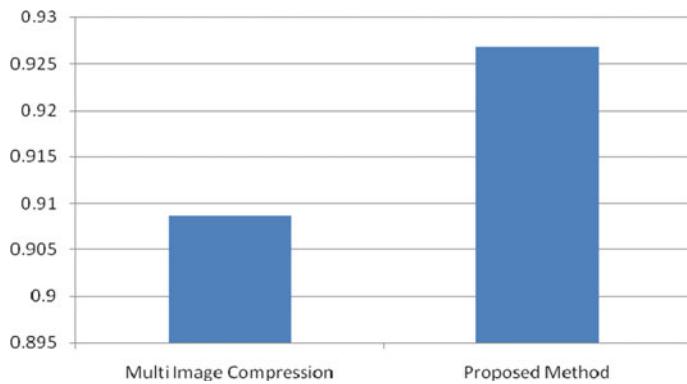
The graphical representation of comparative study of SSIM, PSNR, and compression ratio [12, 13] between multi-image compression [4, 5] and proposed method using bar chart is shown in Figs. 6, 7, and 8, respectively.

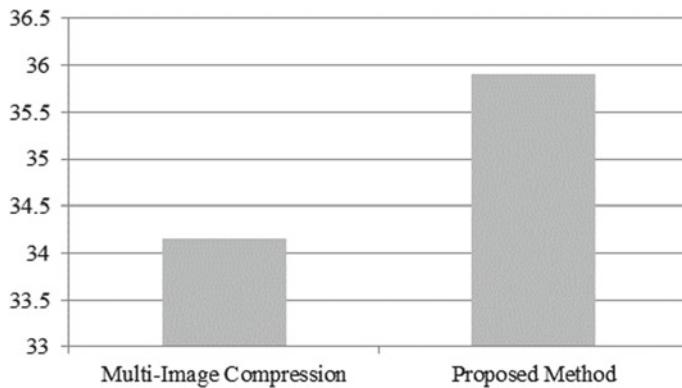
**Table 2** PSNR between multi-image compression and proposed method

Image name	Multi-image compression			Proposed method		
	Y	Cb	Cr	Y	Cb	Cr
Girl	33.50	37.43	35.98	35.13	39.50	37.93
Couple	34.13	38.58	37.24	36.88	40.10	38.89
House	34.24	36.38	34.70	36.27	38.10	35.95
Jelly Beans	35.25	36.08	35.78	37.04	38.90	36.99
Tiffany	33.98	33.97	36.09	35.65	35.84	37.78
Baboon	25.25	30.61	31.45	26.08	32.83	32.73
Pepper	32.62	34.18	33.88	34.10	35.89	35.61
House	30.17	35.29	32.84	31.98	36.88	34.41

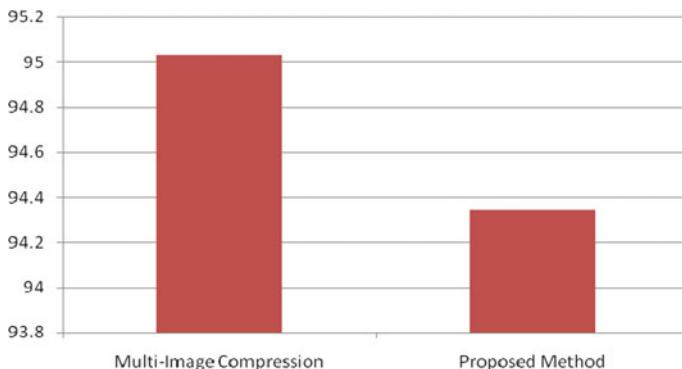
**Table 3** Space reduction using multi-image compression and proposed method

Image	Multi-image compression		Proposed method	
	Total	% of space reduction	Total	% of space reduction
4.1.01.tiff (Girl)	11,453	94.17	12,967	93.40
4.1.02.tiff (Couple)	11,424	94.18	12,997	93.38
4.1.05.tiff (House)	10,575	94.62	12,456	93.66
4.1.08.tiff (Jelly Beans)	9953	94.93	11,256	94.27
4.2.02.tiff (Tiffany)	31,940	95.93	36,898	95.30
4.2.03.tiff (Baboon)	44,057	94.39	49,752	93.67
4.2.07.tiff (Pepper)	32,585	95.85	36,498	95.35
House.tiff	29,957	96.19	33,625	95.72

**Fig. 6** Comparative result of SSIM between multi-image compression and proposed method



**Fig. 7** Comparative result of PSNR between multi-image compression and proposed method



**Fig. 8** Comparative result of compression ratio between multi-image compression and proposed method

## 5 Conclusion

This work proposed a new method that reduces the amount of data loss that occur during multi-image compression technique that improves the quality of the decompressed images by scarifying some amount of compression ratio. This proposed method generates two common code vectors instead of one. Those code vectors of multiple images that play vital role for formation of decompressed images are put as it is in first common code vector pool, whereas the code vectors that are less important are put in the second code vector pool by adjusting some tuning parameters. In this process, a very less amount of data loss occurs compared to the multi-image compression technique. This proposed method is applied on many standard images found in literature and images from UCID v.2 databases. The experimental results are analyzed in terms of SSIM, PSNR, and the compression ratio achieved. From the

experimental result, it is analyzed that the visual quality of the decompressed images is improved significantly by scarifying some amount of compression ratio.

## References

1. Gonzalez, R.C., Woods, R.E., Eddins, S.L.: Digital Image Processing Using MATLAB. Mc-Graw Hill (2011)
2. Gan, G., Ma, C., Wu, J.: Data clustering theory, algorithms and applications. SIAM (2007)
3. Leitao, H.A.S., Lopes, W.T.A., Madeiro, F.: PSO algorithm applied to codebook design for channel-optimized vector quantization. IEEE Latin Am. Trans. **13**(4), 961–967 (2015). <https://doi.org/10.1109/TLA.2015.7106343>
4. Hasnat, A., Barman, D.: A proposed multi-image compression technique. JIFS, IOS Press **36**(4):3177–3193 (2019). <https://doi.org/10.3233/JIFS-18360>
5. Hasnat, A., Barman, D., Barman, B.: Development of Multi-Image Compression Technique based on Common Code Vector. Springer, SN Computer Science (2022)
6. Hasnat, A., Barman, D., Barman, B.: A codebook modification method of vector quantization to enhance compression ratio. In: 2nd Springer International Conference on Computer Vision, High Performance Computing, Smart Devices and Networks (CHSN- 2021). <https://doi.org/10.1007/978-981-16-9885-9>
7. Mridha, K., et al.: Deep learning algorithms are used to automatically detection invasive ducal carcinoma in whole slide images. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 123–129 (2021). <https://doi.org/10.1109/ICCA52192.2021.9666302>
8. Shah, P.K., Pandey, R.P., Kumar, R.: Vector quantization with codebook and index compression In: IEEE International Conference System Modeling & Advancement in Research Trends, India (2016) <https://doi.org/10.1109/SYSMART.2016.7894488>
9. Hasnat, A., Barman, D., Halder, S., Bhattacharjee, D.: Modified vector quantization algorithm to overcome the blocking artefact problem of vector quantization algorithm. J. Intell. Fuzzy Syst. IOS Press **32**(5):3711–3727 (2017) <https://doi.org/10.3233/JIFS-169304>
10. Hasnat, A., Barman, D., Barman, B.: Luminance approximated vector quantization algorithm to retain better image quality of the decompressed image. Springer, vol. 80, pp. 11985–12007 (2021) <https://doi.org/10.1007/s11042-020-10403-9>
11. Barman, D., Hasnat, A., Sarkar, S., Rahaman, M.A.: Color image quantization using gaussian particle swarm optimization (CIQ-GPSO). In: IEEE International Conference on Inventive Computation Technologies, India (2016). <https://doi.org/10.1109/INVENTIVE.2016.7823295>
12. Sara, U., Akter, M.: Uddin MS: image quality assessment through FSIM, SSIM, MSE and PSNR—a comparative study. J Comput. Commun. **7**(3), 8–18 (2019). <https://doi.org/10.4236/jcc.2019.73002>
13. Mandal, J.K.: Reversible Steganography and Authentication via Transform Encoding. Springer (2020). ISBN: 9789811543975

# Diabetes Prediction Using Machine Learning Techniques



**Sourav Simanto, Krishna Mridha, Runa Saha, Milan Limbu, Ankush Ghosh, and Rabindra Nath Shaw**

**Abstract** “A metabolic disease that induces elevated blood sugar is diabetes mellitus, also known as diabetes” [1]. We all know that a hormone typically moves through the blood to our cells for energy to be collected. Any diabetes requires high blood that is unprocessed and can affect our nerves, lungs, kidneys, hearts, and other organs. For the fact why persons have to go to the diagnostic center, hospital, or pharmacy for testing, there are a lot of people afflicted or affected by diabetes at some periods. The management must, of course, store the appropriate test report and have an adequate diagnosis based on their report. But, the increase of approaches to machine learning addresses this crucial question. In hospitals, diagnostics, laboratories, or another healthcare sector, machine learning (MI), data science (DS), and artificial intelligence (AI) play a significant role. A lot of high-volume databases must behave in this sort of place. Using a data analysis methodology, a large dataset can be analyzed, and deeper details, deeper trends, deeper data signs, and the effect can be projected accordingly. This recitation aims to devise a specific model that can predict diabetes with full precision. The classification and estimation precision is not that high in the current system. We have suggested a new diabetes prediction model in this article, which model we have already implemented on the server. We add some extra

---

S. Simanto

Jatiya Kabi Kazi Nazrul Islam University, Mymensingh, Bangladesh

K. Mridha (✉) · M. Limbu

Marwadi University, Rajkot, Gujarat, India

e-mail: [krishna.mridha108735@marwadiuniversity.ac.in](mailto:krishna.mridha108735@marwadiuniversity.ac.in)

M. Limbu

e-mail: [milan.limbu106167@marwadiuniversity.ac.in](mailto:milan.limbu106167@marwadiuniversity.ac.in)

R. Saha

Gono Bishwabidyalay Mirzanagar, Savar, Dhaka 1344, Bangladesh

A. Ghosh · R. N. Shaw

University Center for Research & Development (UCRD), Chandigarh University, Mohali, Punjab, India

e-mail: [ankushghosh@ieee.org](mailto:ankushghosh@ieee.org)

R. N. Shaw

e-mail: [r.n.s@ieee.org](mailto:r.n.s@ieee.org)

inputs along with standard inputs such as age, glucose, BMI to get the best diabetes forecast. There are six classifications of machine learning (ML) algorithms used in this experiment to diagnose diabetes at an early level, such as gradient boosting classifier, logistic regression, decision tree, SVM, K-nearest neighbors, and naive Bayes. I used the “Pima Indian Diabetes Database (PIDD) obtained from the UCI machine learning library for this case study” [2]. With numerous techniques such as precision, F-measure, and recall, the accuracy of all six algorithms is measured. From this test, relative to all other algorithms used in this experiment, I got the highest accuracy of 84.20% from the gradient booster classification. The proper application of receiver operating characteristic curve (ROC) is corroborated by these accuracies.

**Keywords** Data analysis · Artificial intelligence · Classification · PIDD · Machine learning · Health care

## 1 Introduction

Diabetes is one of the most dangerous diseases on the planet. Obesity or a high blood glucose level, for example, can induce diabetes. It affects the hormone insulin, resulting in aberrant blood sugar levels. Crab metabolism is improved, as is the level of sugar in the blood. When the body does not produce enough insulin, diabetes develops insulin. The World Health Organization (WHO) says diabetes affects approximately 422 million people worldwide, with the majority of cases occurring in low- and middle-income nations, and this could be the case. By 2030, it will have risen to 490 billion dollars. However, diabetes is prevalent in a variety of countries such as Canada, China, and India, to name a few. India has a population of 1.3 billion people. To do so, this research looks into diabetes prediction using a variety of diabetes-related variables. We use the Pima Indian Diabetes Dataset for this, and we predict diabetes using several machine learning classification and ensemble techniques. Machine learning is a technique for explicitly training computers or machines. By constructing multiple classifications and ensemble models from acquired datasets, various machine learning techniques deliver efficient results for collecting knowledge. This type of information can be used to predict diabetes. Various machine learning approaches are capable of prediction; however, selecting the optimum methodology is difficult. As a result, we use common classification and ensemble algorithms on the dataset to make predictions.

With a population of more than 100 million people, India today has 40 million diabetes. Diabetes is a leading cause of death in the United States. Diabetes is nothing but a disorder that undermines the body’s capacity to regulate blood glucose, otherwise referred to as blood sugar. More than 35 million people are now afflicted by diabetes in India. The Blunt Prevalence Rate (CPR) is to be 9% in the Indian urban area, but it is roughly 3% of the total population in the rural area, where the total Indian population is more than 1000 million. The consequences are immense for the Indian healthcare sector. This typically happens when the body fails to produce



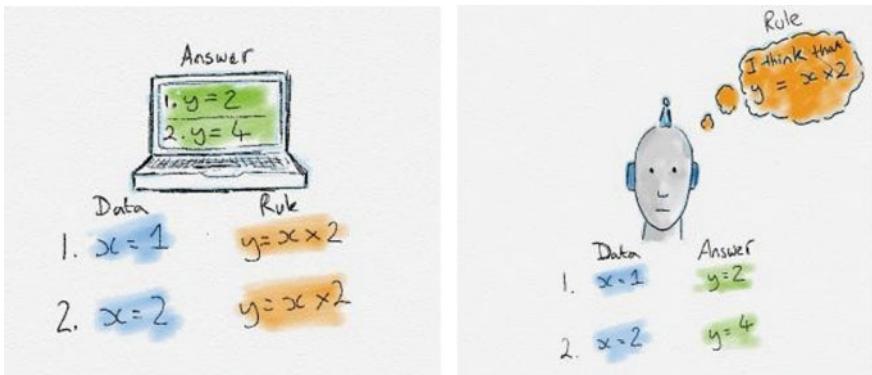
**Fig. 1** Diabetes by the numbers

insulin, also known as juvenile diabetes. I would like to mention three forms of diabetes in this report, based on (PIDD). Class 1: Type 1 diabetes is typically insulin dependent, which means that artificial insulin must be taken up regularly to survive. Type 2: Type 2 diabetes is a chronic condition. On the other hand, they are known to have insulin resistance in persons afflicted by type 2 diabetes. In all females, gestational diabetes does not exist and normally resolves after giving birth (Fig. 1).

## 2 Overview of Machine Learning

Machine learning is an automated learning process and changes from previous encounters with unconscious programming. ML is driven by the evolution of computer programs that can interpret and apply knowledge. Software engineering has historically mixed human-created rules and data to generate solutions to a dilemma.

Nowadays, machine learning leads a very important role in the world. Many of us are taking benefit from this noble technology, but we still do not know machine learning consistency in our entire life. Make a screenshot, continually learning, and improving with every experience, machine learning appears part of the engine behind the last. It is also behind world-changing trends, including diabetes forecasting. The remaining parts of this article are discussed by the structure as follows: Sect. 3 motivation of this article, Sect. 4 briefs linked work of different classification method of diabetes prediction, Sect. 5 methodology, Sect. 6 results and discussion, Sect. 7 future work, and Sect. 8 is for the conclusion (Fig. 2).



**Fig. 2** Traditional programming versus machine learning

### 3 Motivation

In the past decade, the number of people suffering from diabetes has risen significantly. The new human way of life is the key factor behind the development of diabetes. Three different forms of diagnosis can be found in the modern medical diagnosis process.

1. The false negative form in which a patient is already a diabetic patient in fact, but test results show that the individual does not have diabetes.
2. The sort of false positive. In this way, the patient is not necessarily diabetic, but test results suggest he/she is a diabetic patient.
3. The third type is an unclassifiable type that cannot be diagnosed by a device in a particular situation. This exists as a function of inadequate information extraction from past data that may be expected in an unclassified form by a given patient. However, the patient must predict that he or she is either in the diabetic or non-diabetic group. These mistakes in diagnosis can result in needless care or, if possible, no treatment at all. To prevent or decrease the number of affected people, machine learning takes an important role. The magnitude of such an effect involves the construction of a device using computers.

### 4 Related Work

Vijiya Kumar et al. [2] suggested a random forest diabetes estimation method to create a structure that can use the model proposed which includes the best outcomes for feature reduction, and the outcome proved that the prediction system is capable of successfully, accurately, and, most importantly, immediately forecasting the diabetes condition, and Ljumah et al. [3] created a methodology of predictive modeling using

the algorithm of the help support vector. Kavakiotis et al. proposed n-fold for validation test in [4] as an assessment tool for three different algorithms, including tenfold cross-validation where SVM provided 83% greater performance and precision than many of the other implementations. In [5], machine learning was added by Zheng et al. Regression of early stage estimation of diabetes mellitus is where filtering parameters can be enhanced. KNN, J48, ANN, Xero, and NB were added by Kavakiotis et al. [6] to separate diabetes datasets. In Delhi, India, Das et al. [7] researched dengue and malaria cases and conducted a predictive study of the data in [8]. Simi et al. addressed the significance of early detection of menstrual disorders in [9]. Results found that the random forest methodology performed the best accuracy, and the accuracy was 80%. They use another dataset with 26 parameters and 8 groups of female infertility.

For diabetes categorization, Qawqzeh et al. [10] suggested a logistic regression predicted on photoplethysmogram analysis. They used data from 459 patients to train the model and 128 statistics to test and validate it. Their suggested system properly classified 552 people as non-diabetic, with a 92% accuracy rate. The proposed technique, on the other hand, is not compared to state-of-the-art techniques. Pethunachiyar [11] introduced a machine learning-based diabetic mellitus classification method. He primarily was using a support vector machine with several kernel functions and diabetes data from the University of California, Irvine Machine Repository. He discovered that SVM with a linear function outperformed naive Bayes, decision trees, and neural networks. Nonetheless, there is no state-of-the-art assessment and no detailed parameter selection.

For diabetes categorization, Gupta et al. [12] used naive Bayes and support vector machine methods. The Pima Indian Diabetes Dataset was used. They also employed a feature selection strategy and k-fold cross-validation to increase the model's accuracy. The experimental results demonstrated that the support vector machine outperformed the nave Bayes model. However, along with attained accuracy, a state-of-the-art comparison is missing. A comparative analysis of diabetic classification approaches was published by Choubey et al. [13]. Pima Indian data from the UCI machine learning repository was combined with a local diabetes dataset. To categorize patients as diabetic or not, the researchers employed AdaBoost, K-nearest neighbor regression, and the radial basis function on both datasets.

To categorize and forecast diabetes, Maniruzzaman et al. [14] employed a machine learning methodology. For diabetes categorization, they used four machine learning algorithms: naive Bayes, decision tree, AdaBoost, and random forest. On the Pima dataset for diabetic categorization, Ahuja et al. [15] conducted a comparison examination of various machine learning approaches, such as NB, DT, and MLP. When compared to other classifications, researchers found MLP to be superior. Fine-tuning and efficient feature engineering, according to the authors, can improve MLP's performance. Recently, Mohapatra et al. [16] employed MLP to classify diabetes and reached a 77.5% accuracy on the Pima dataset, but they were unable to do state-of-the-art comparisons.

## 5 Methodology

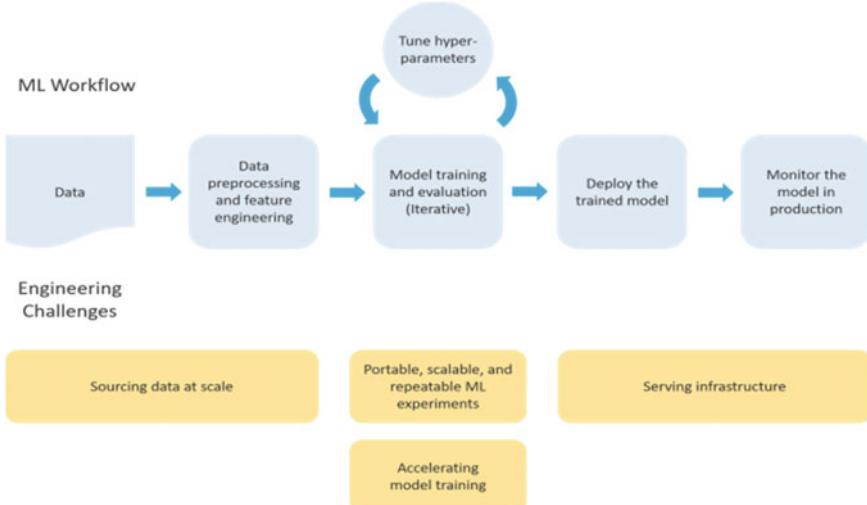
Because of the proposed solution alluded to in the introductory part, we propose a classifier model with greater accuracy to forecast the diabetic. Several classification algorithms, such as the support vector machine classification model, random forest, LR, KNN, and naive Bayes, have also been employed in this model. By using the parametric approach, the main aim is to boost the performance of a benchmark well-known diabetes dataset obtained from the Indian famous dataset, which involves several attributes. The architecture consists of the following significant stages: data collection (Pima Indian Diabetes Dataset) [17], data preprocessing, attribute selection by core principal components, etc. (Fig. 3).

The motive of this paper is to develop a significant model to predict diabetes. The accuracy, time complexity, precision, recall, and other side are good enough to predict diabetes.

### A. Dataset Description:

The total information that we have used in this paper is collected from one Indian famous dataset called Pima. This dataset contains 8 features and 1 target. The total samples are 768 (Table 1).

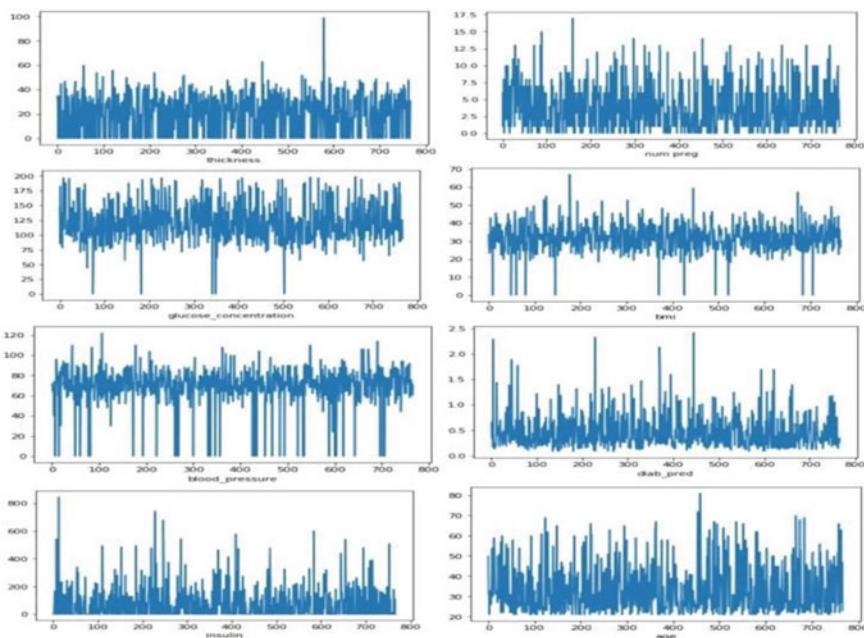
The ninth attribute of each data point is the class variable. For diabetics, this type of indicator indicates results 0 and 1 that mean positive or negative for diabetics (Fig. 4).



**Fig. 3** Proposed model

**Table 1** Data description

S. No.	Attributes	Mean	Std.
1	Pregnancy	3.8	3.4
2	Glucose	120.9	32.0
3	Blood pressure	69.1	19.4
4	Skin thickness	20.5	16.0
5	Insulin	79.8	115.2
6	Body mass index (BMI)	32.0	7.9
7	Diabetes pedigree function	0.5	0.3
8	Age	33.2	11.8

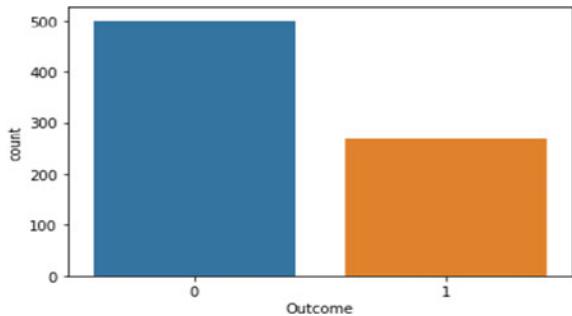
**Fig. 4** Statistical analysis for mean and standard deviation in Pima Indians Diabetics Dataset

*Distribution of Diabetic patients:* Our developed model counts the total number of positive cases as well as negative cases. This dataset includes 500 positive cases as class 1, and the rest 268 cases are negative as class 0 (Fig. 5).

#### B. Data Preprocessing:

The most critical method is data preprocessing. Most data relevant to health care has missing values and other impurities that may allow data to be effective. Data preprocessing is performed to maximize the accuracy and reliability obtained during the mining process. This step is essential for accurate and effective results to effectively

**Fig. 5** Ratio of diabetes and non-diabetes patients



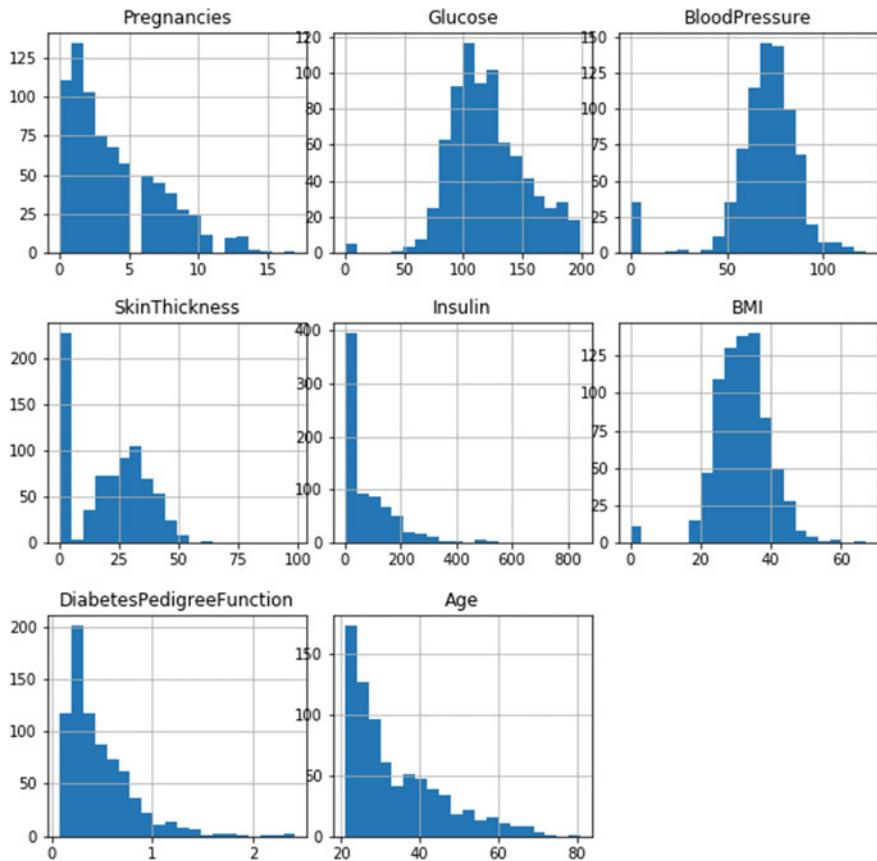
use machine learning approaches on the dataset. We need to do preprocessing in two phases for the Pima Indian Diabetes Dataset.

- (1) *Deletion of Incomplete Values:* Uninstall all instances with a value of zero (0). It is not feasible to have zero as a value. This scenario is then eliminated. We create a subset of features by deleting redundant features/instances, and this approach is called the collection of features subset, which decreases data dimensionality and helps to run faster.
- (2) *Data separating:* In training and checking the model, data is normalized after cleaning the data. The next step is splitting the dataset into two parts, one part is for training, and another part is for testing. The training data is used to acknowledge the machine, and the testing data is used for testing the machine knowledge. Machine trained by various logic, functions, and algorithms.
- (3) *Feature Selection:* Methods for feature selection will minimize the number of attributes, and this may prevent redundant characteristics. Many ways of feature selection exist. We used PCA and minimal redundancy maximum significance (mRMR) [4, 18] to decrease the dimensionality in this analysis. To perform the PCA algorithm, we used Mathematical Product and Service Solutions (SPSS). SPSS is a generic term for a series of IBM-launched software applications and associated services. For mathematical analysis, data processing, predictive analysis, and other functions, it is primarily used. The SPSS is easy to use and has a nice visual design (Figs. 6 and 7).

#### A. Machine Learning Algorithms:

There are six most common machine learning algorithms used to predict diabetes mellitus. We are also describing these approaches. Among all these algorithms, perform various accuracies with various parameters like precision, recall, and F1-scores.

- (1) *Support Vector Machine:* Support vector machine is one of the best classification algorithms among all supervised algorithms. The most common classification system is SVM [8]. SVM produces a hyperplane that divides two groups from each other. In high-dimensional space, it may generate a hyperplane or



**Fig. 6** Pair plot of each feature

a hyperplane sequence. In particular types, SVM differentiates instances and may even distinguish entities that are not supported over info.

- (2) *Naïve Bayes:* A naïve Bayes classifier is a system of probability machine learning used for classification. Equation no. 1 is showing naïve Bayes theorem.

Bayes Theorem:

$$P(A/B) = \frac{P(B/A)P(A)}{P(B)} \quad (1)$$

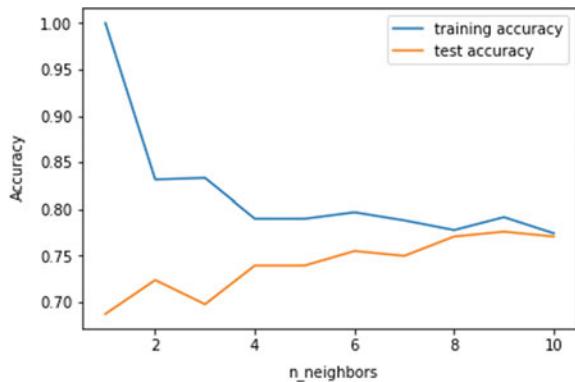
- (3) *K-nearest neighbor:* Another very important classification algorithm name is K-nearest neighbor. KNN has not solved only classification problems but also regression problems. KNN is a lazy strategy for forecasting. KNN believes that related objects are identical to one another. KNN allows organizing



**Fig. 7** Histogram of each feature

current projects depending on the measure of central tendency. KNN algorithm reports all documents and classifies them according to their measure of central tendency. The proposed algorithm classifies them as per their measure of central tendency. Using a tree-like structure to measure the path between some of the points, the algorithm considers the neighboring datasets in the training examples poised to guess at a new data point. It is the nearest.  $K$  = the number of neighbors here is always a positive integer. The value of a neighbor is selected from a class set. The primary definition is Euclidean distance. Normally,  $P$  and  $Q$  are the Euclidean intervals, i.e., two points. The following equation is used to describe  $P(p_1, p_2, P_n)$  and  $Q(q_1, q_2, \dots q_n)$ :

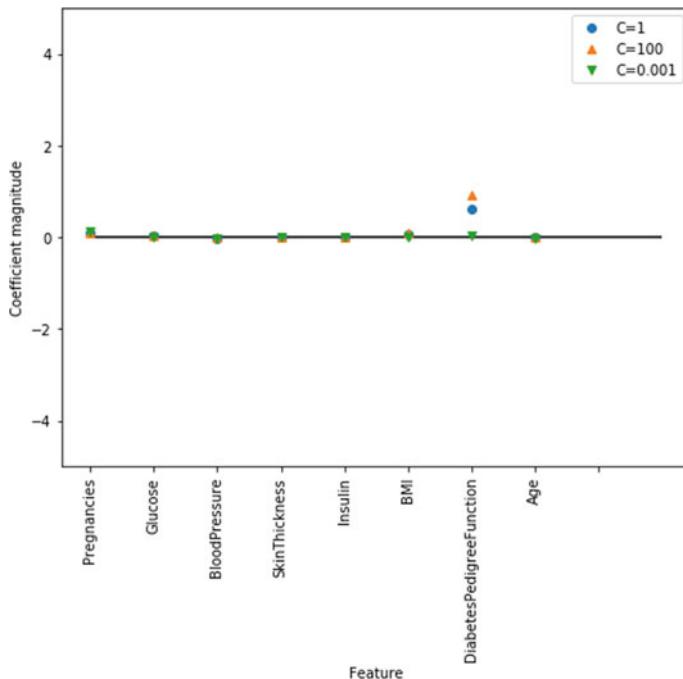
$$d(P, Q) = \sum_{i=1}^n (P_i - Q_i)^2 \quad (2)$$

**Fig. 8** Training and testing

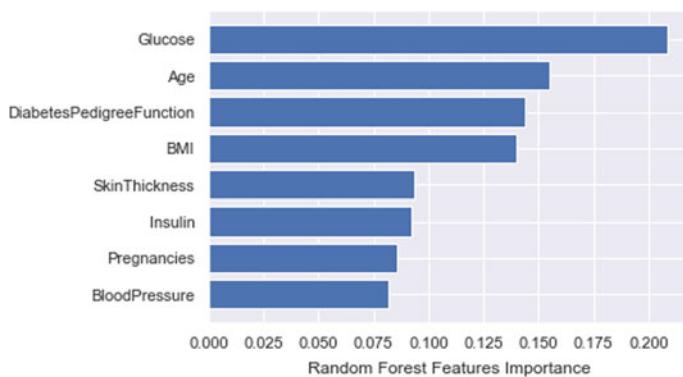
- (2) *Logistic Regression:* Another important algorithm is logistic regression for learning. It is also used to measure a categorical explanatory possibility dependent on one or even more predictor variables. They can be discrete or continuous. When we try to identify or distinguish certain data objects into groups, logistic regression is used. This article contains binary values 0 and 1. This paper is working for only two classes. For representing those two classes, binary values 0 and 1 are used. If the machine predicts class 1, that means she has diabetes, and if the machine predicts class 0, that means she does not have diabetes. A regression analysis system is based on logistic regression (Figs. 8 and 9).
- (4) *Random Forest:* It is a kind of method of ensemble learning. It is also often used for solving both classification and regression problems. Compared to others, the precision it offers is greater than any other algorithm. Huge datasets can be easily managed by this method. Leo Breiman created the random forest. It is a common learning method ensemble (Fig. 10).
- (5) *Gradient Booster Classification:* The most effective ensemble technique used for prediction is gradient boosting, and it is a method of classification. It integrates weak learners to create good prediction models for learners. It utilizes the model decision tree. It classifies difficult datasets and is a very effective and common technique. The efficiency of the model in gradient boosting increases over iterations (Fig. 11).

## 6 Results and Discussions

Using machine learning algorithms, the final result was successfully obtained. According to the Indian Pima Diabetes Dataset accessible in the UCI repository, various machine learning algorithms are such as random forest classifier, help vector machine, logistic regression, K-nearest neighbor, Gaussian naïve Bayes, and gradient boost classifier. After using various PIDD database machine learning algorithms, we



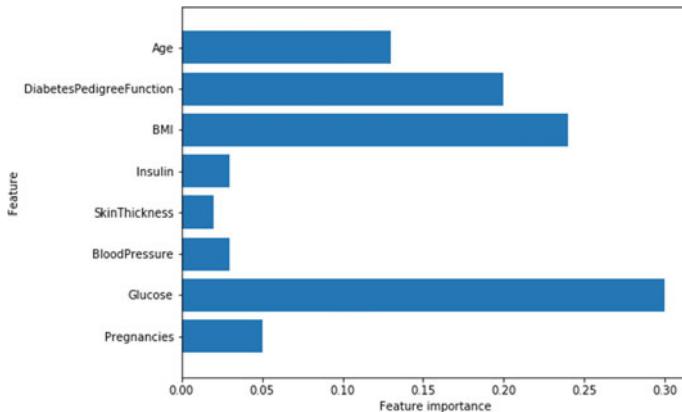
**Fig. 9** Logistic feature importance



**Fig. 10** Random forest feature importance

collected accuracies as described below. The gradient booster classification algorithm offers 84.20% best or highest efficiency or precision.

**Evaluation Metric:** We describe all the measures used in our research in this segment. Here, true positive means TP, false positive means FP, true negative means FN, false



**Fig. 11** Gradient booster classifier features importance

positive means FP. The precision, recall, and F1-scores equations are mentioned below [19].

In contrast to such metrics, the vacuity is often measured based on the machine learning techniques' appropriately classified instances. Accuracy has been officially determined as below

$$\text{Accuracy} = \frac{t_p + t_n}{t_p + t_n + f_p + f_n} \quad (3)$$

$$\text{Precision} = \frac{t_p}{t_p + f_p} \quad (4)$$

$$\text{Recall} = \frac{t_p}{t_p + f_n} \quad (5)$$

$$\text{F1 - score} = \frac{\text{precision}.\text{recall}}{\text{precision} + \text{recall}} \quad (6)$$

Let us see, how our model will perform (Table 2).

Machine learning models are built with the necessary training set described above. Tenfold cross-validation is implemented for improved tuning of the hyperparameters of algorithms and robustness in the model [20, 21]. The findings are shown in Table 3 in terms of the accuracy of one set of training and test sets, the average accuracy of the tenfold training and test set, and the standard deviation of the tenfold training and test set selection process.

According to the findings shown in Table 3, it is apparent that not only onefold, the gradient booster classifier algorithm performs better with this data, but also shows the least variation when conducting k-fold training and collection of test data and subsequent research.

**Table 2** How model will perform

ID	Actual sick?	Predicted sick?	Outcome
1	1	1	TP
2	0	0	TN
3	0	0	TN
4	1	1	TP
5	0	0	TN
6	0	0	TN
7	1	0	FP
8	0	1	FN
9	0	0	TN
10	1	0	FP
..	..	..	..
100	0	0	FN

**Table 3** Comparison between tenfold accuracy versus without cross-validation accuracy

Machine learning algorithms	K-fold mean Ac. (%)	Accuracy (%)
Gradient booster	85.34	84.20
Support vector machine	80.00	81.20
Logistic regression	84.67	83.54
K-nearest neighbors	82.20	80.34
Gaussian naïve bayes	80.10	79.23
Random forest	81.58	80.39

### Deployment of the Model on Server (Web Application)

- **Input:** All of the features showing on the user interface page.
- **Output:** Classify diabetes.
- When this concept is applied to development, we can create a web application that implies taking eight inputs from the user to result in diabetes.
- Here is what a web application created using Python, Stream lit, looks like (Figs 12 and 13).

## 7 Future Work

In this analysis, we focused only on diabetes disease; for the future, this can be expanded to apply this approach in other diseases. Small quantity of sample data



**Fig. 12** User interface (Page 1)



**Fig. 13** User interface (Page 2)

is to be used for this study. It could be used for future expansion in large volumes of data. In this research also, only an individual dataset was used; hence, numerous dataset can be used for forecasting for the future.

## 8 Conclusion

Finally, we have achieved our diabetes prediction results by widely applying machine learning algorithms. The entire procedure is nothing but a process of machine learning. With web or app development, we have completed a classification model task that can turn into the implementation of a large detection of wildfires all over the globe.

We want to develop an Android application for the proposed hypothetical diabetic monitoring system using the provided categorization and prediction algorithms in the future. For better monitoring, genetic algorithms can be combined with the proposed prediction mechanism.

## References

1. Kalyankar, G.D., Poojara, S.R., Dharwadkar, N.V.: Predictive analysis of diabetic patient data using machine learning and hadoop. In: International Conference On I- SMAC (2017). 978-1-5090-3243-3
2. VijiyaKumar, K., Lavanya, B., Nirmala, I., Sofia Caroline, S.: Random forest algorithm for the prediction of diabetes. In: Proceeding of International Conference on Systems Computation Automation and Networking (2019)
3. Aljumah, A.A., et al.: Application of data mining: diabetes health care in young and old patients. *J. King Saud Univ. Comput. Inf. Sci.* **25**(2), 127–136 (2013)
4. Mridha, K., Ranpariya, A., Kumbhani, S., Ghosh, A., Shaw, R.N.: Plant disease detection using web application by neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 130–136 (2021)
5. Mridha, K., et al.: Plant disease detection using web application by neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 130–136 (2021). <https://doi.org/10.1109/ICCCA52192.2021.9666354>
6. Kavakiotis, I., Tsav, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., Chouvarda, I.: Machine learning and data mining methods in diabetes research. *Comput. Struct. Biotechnol. J.* (2017)
7. Sinha, T., et al.: Analysis and prediction of COVID-19 confirmed cases using deep learning models: a comparative study. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_18](https://doi.org/10.1007/978-981-16-2164-2_18)
8. Chakraborty, R., Mridha, K., Shaw, R.N., Ghosh, A.: Study and prediction analysis of the employee turnover using machine learning approaches. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021)
9. Simi, M.S., Nayaki, K.S., Parameswaran, M., Sivadasan, S.: Exploring female infertility using predictive analytic. (2017)

10. Qawqzeh, Y.K., Bajahzar, A.S., Jemmali, M., Otoom, M.M., Aljaoui, A.: Classification of diabetes using photoplethysmogram (PPG) waveform analysis: logistic regression modeling. *BioMed Res. Int.* **2020**(6), (2020)
11. Pethunachiyar, G.A.: Classification of diabetes patients using kernel-based support vector machines. In: Proceeding of the 2020 International Conference on Computer Communication and Informatics (ICCCI), pp. 1–4. IEEE, Coimbatore, India, January (2020)
12. Palimkar, P., et al.: Machine learning technique to prognosis diabetes disease: random forest classifier approach. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies*. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_19](https://doi.org/10.1007/978-981-16-2164-2_19)
13. Choubey, D.K., Kumar, M., Shukla, V., Tripathi, S., Dhandhania, V.K.: Comparative analysis of classification methods with PCA and LDA for diabetes. *Curr. Diabetes Rev.* **16**(8), 833–850 (2020)
14. Maniruzzaman, M., Rahman, M.J., Ahammed, B., Abedin, M.M.: Classification and prediction of diabetes disease using machine learning paradigm. *Health Inf. Sci. Syst.* **8**(1), 7–14 (2020)
15. Ahuja, R., Sharma, S.C., Ali, M.: A diabetic disease prediction model based on classification algorithms. *Ann. Emerg. Technol. Comput.* **3**(3), 44–52 (2019)
16. Mohapatra, S.K., Swain, J.K., Mohanty, M.N.: Detection of diabetes using multilayer perceptron. In: Proceeding of the International Conference on Intelligent Computing and Applications, pp. 109–116. Springer, Ghaziabad, India, December (2019)
17. Muhammad Khan, D., Mohamudally, N.: An integration of K-means and decision tree (ID3) towards a more efficient data mining algorithm. *J. Comput.* **3**(12)
18. Mridha, K., Shaw, R.N., Ghosh, A.: Intelligent based waste management awareness developed by transfer learning. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–5 (2021)
19. Mridha, K.: Early prediction of breast cancer by using artificial neural network and machine learning techniques. In: 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT), pp. 582–587 (2021)
20. Rani, A.S., Jyothi, S.: Performance analysis of classification algorithms under different datasets. In: Computing for Sustainable Global Development (INDIACOM), 2016 3rd International Conference on, pp. 1584–1589. IEEE (2016)
21. Mridha, K., Kumar, D., Shukla, M., Jani, M.: Temporal features and machine learning approaches to study brain activity with EEG and ECG. In: 2021 International Conference on Advanced Computing and Innovative Technologies in Engineering (ICACITE), pp. 409–414 (2021)

# Machine Learning in Bioinformatics: New Technique for DNA Sequencing Classification



Shakil Sarkar, Krishna Mridha, Ankush Ghosh, and Rabindra Nath Shaw

**Abstract** The extraction of useful information from deoxyribonucleic acid (DNA) is a major component of bioinformatics research, and DNA sequence categorization has a variety of applications, including genomic and biomedical data processing. DNA sequence classification is a critical problem in a general computational framework for biomedical data processing, and numerous machine learning techniques have been used to complete this task in recent years. Machine learning is a data processing technique that uses training data to create judgments, predictions, classifications, and recognitions. To learn the functions of a new protein, genomic researchers classify DNA sequences into known categories. As a result, it is critical to discover and characterize those genes. We employ machine learning approaches to distinguish between infected and normal genes using classification methods. In this study, we used the multinomial Naive Bayes classifier, SVM, KNN, and others to classify DNA sequences using label and k-mer encoding. Different categorization metrics are used to evaluate the models. The multinomial Naive Bayes classifier, SVM, KNN, decision tree, random forest, and logistic regression with k-mer encoding all have good accuracy on testing data, with 93.16% and 93.13%, respectively.

**Keywords** DNA · Bioinformatics · Computational framework · Machine learning · Classification

---

S. Sarkar · K. Mridha (✉)  
Marwadi University, Rajkot, Gujarat, India  
e-mail: [krishna.mridha108735@marwadiuniversity.ac.in](mailto:krishna.mridha108735@marwadiuniversity.ac.in)

S. Sarkar  
e-mail: [shakil.sarkar111214@marwadiuniversity.ac.in](mailto:shakil.sarkar111214@marwadiuniversity.ac.in)

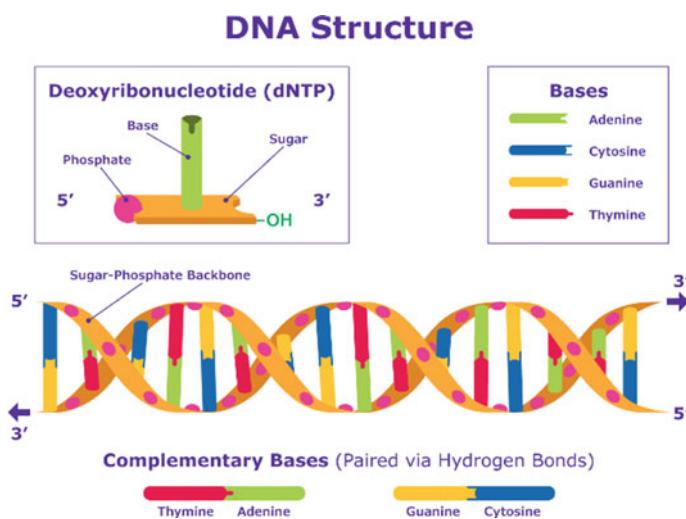
A. Ghosh · R. N. Shaw  
University Center for Research & Development (UCRD), Chandigarh University, Mohali, Punjab 140413, India  
e-mail: [ankushghosh@ieee.org](mailto:ankushghosh@ieee.org)

R. N. Shaw  
e-mail: [r.n.s@ieee.org](mailto:r.n.s@ieee.org)

## 1 Introduction

Friedrich Miescher's successful isolation of deoxyribonucleic acid (DNA) in 1869 was a watershed moment in biology, laying the framework for comprehending the blueprints of all organic life. DNA stands for deoxyribonucleic acid and is a hereditary molecule found in the cells of all humans and other living species. It contains the necessary information that determines our bodies' basic characteristics and functions as a genetic blueprint for an evolving organism. A character string consisting just of A, C, G, or T can be used to represent an isolated DNA sequence. DNA is a double-stranded molecule that is containing a phosphate group, a sugar group, and nitrogenous bases—thymine (T), guanine (G), cytosine (C), and adenine (A), and it is responsible for carrying and transmitting hereditary materials from parents to offspring. The base pairing is as follows: guanine pairs with cytosine (C-G) and thymine pairs with adenine (A-T). FASTA is the name of the format. DNA analysis is critical since it enables doctors to identify diseases, aids in the investigation of the spread of new infections, and can even be used to solve crimes or conduct paternity testing. As a result, DNA analysis is now a hot topic in computational biology [1].

Primers are fundamental tools for DNA analysis in conventional biology. Primers are human nucleotide sequences that are essential for all living organisms' DNA synthesis to begin. Synthetic primers are used in molecular biology for a variety of applications, including the detection of viruses [2], bacteria [3], and parasites [4]. These goals are served by primers, which are frequently found in human DNA sequences infected by a certain type of virus. The DNA fragment of an existing virus is amplified greatly using the polymerization chain reaction (PCR) technology, allowing researchers to detect the virus (Fig. 1).



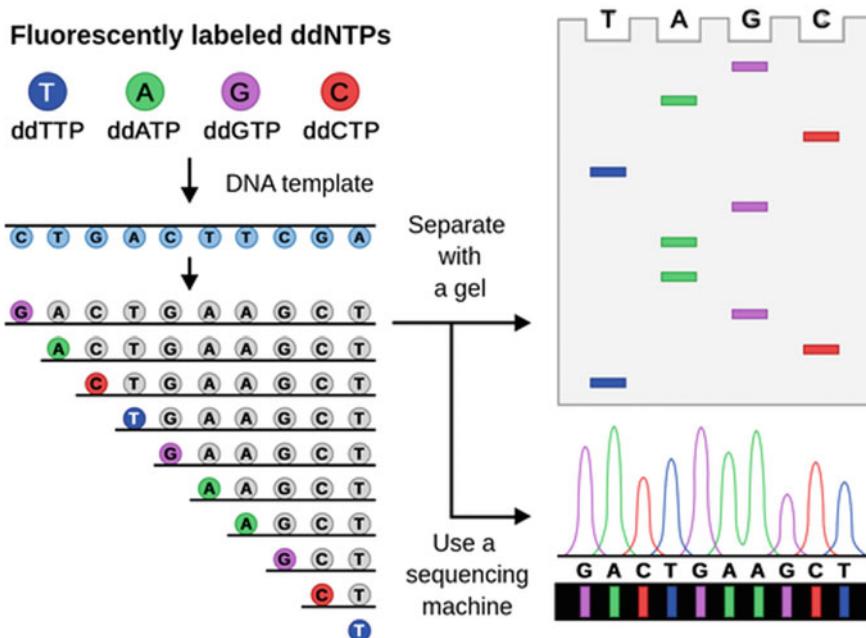
**Fig. 1** DNA sequencing

## 1.1 Deoxyribonucleic Acid (DNA) Sequencing

Biochemical similarities are frequently utilized to identify creatures that are closely related. The draft of one's life is frequently carried via DNA. The DNA sequence identifies the traits and types of species in simple words. At regular intervals throughout the cell, DNA includes the instructions for making proteins.

Figure 2 depicts the automation of DNA sequencing.

DNA sequencing is a technique for ensuring that nucleotides in a DNA segment are arranged correctly. A double strand of nucleotides can occasionally be seen in DNA. For DNA sequencing, a variety of approaches have been developed. The idea of utilizing DNA sequences to identify species is being investigated in a variety of fields. Many ways are recommended for detecting species using DNA sequences, including correspondence scores, phylogeny, population heritable information, and the discovery of species-specific sequence patterns. Shotgun cloning and walking are the two processes that DNA sequencing investigations usually take. Classification is a method of identifying a single character or a group of characters. To classify a supermolecule sequence into its specific division, secondary category, or family, a variety of classification techniques are used. These strategies try to eliminate a few alternatives, equalize the value of those options, and finally categorize



**Fig. 2** Automated DNA sequencing

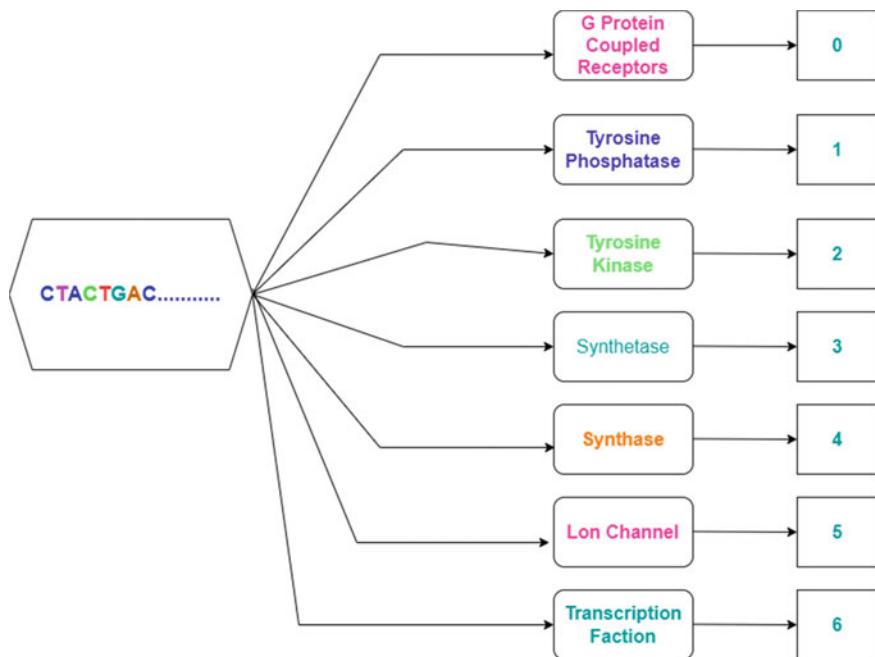
the super molecule sequence. Reference [5] makes it clear that DNA sequence analysis methodologies are still supported by PCA. [3] A common type of sliced inverse regression for sophistication prediction and sequence choice was planned using DNA microarray knowledge. Reference [6] has identified a way derived from biology for resolving difficult combinatorial problems. A directed Hamiltonian path downwards that is NP-complete was the subject of his experiment.

## ***1.2 Deoxyribonucleic Acid (DNA) Sequencing Classification***

The study of DNA sequence data is a major focus of bioinformatics. When we talk about DNA sequencing, we are talking about the process of determining the order of nucleotides in a nucleic acid sequence. The term categorization refers to the division of a nucleic acid or its combinations, referred to as a gene, into distinct regions. The selected-effect function genes are categorized into two primary groups in [7], which are functional DNA and rubbish DNA. Rubbish DNA does not have the unselected-effect function, but functional DNA does. Literal DNA and indifferent DNA are two types of functional DNA. Nucleotide order is selected in literal DNA, while merely the presence or lack of the sequence of indifferent DNA is selected in indifferent DNA. When it comes to rubbish DNA, it is classified into two categories: junk DNA and garbage DNA. Junk DNA contributes to the fitness of the organism, and as a result, it evolves under selection neutrality. Garbage DNA lowers the fitness of those who carry it. DNA in the above categories can be transcribed and translated, or transcribed but not translated, or transcribed but not translated. The assignment of a DNA region to a specific functional category can vary throughout time. Functional DNA, for example, can be transformed into junk DNA, junk DNA into garbage DNA, and so on [7] (Fig. 3).

## ***1.3 Machine Learning in Bioinformatics: DNA Sequencing Classification***

Many qualities of computational intelligence approaches, such as adaptability and fault tolerance, have made them appealing for bioinformatics research. For network classification, a machine learning approach is presented. Machine learning's goal is to explore, learn, and adapt to changing circumstances to improve the machine's performance. The reference input is utilized for machine learning algorithms in the field of bioinformatics so that they can "learn." Soft computing techniques are appealing to use in bioinformatics because of their ability to deal with unclear and partially true data. Here, machine learning techniques can be utilized to train the network for improved performance and system accuracy. Furthermore, machine learning methods are utilized to reduce the number of false positives.



**Fig. 3** DNA sequencing type and respected class

Machine learning is defined as a computing machine's ability to improve its performance based on previous outcomes. The image depicts the numerous machine learning tools that are useful in the field of bioinformatics. Classification has become a perilous undertaking in dealing with biological data in recent years, and it is not possible to do so using standard approaches. In bioinformatics, the artificial neural network is the most extensively used machine learning technology. One of the most significant components of soft computing is the neural network, which provides learning capabilities. There are two types of learning methods: supervised and unsupervised. In supervised learning, the network is taught by examples, but in unsupervised learning, the network is taught by itself. Certain class labels are not known. When the class labels are known, a model is created that can categorize new examples not previously seen by the learner. And N-cross-validation is used to evaluate this learning process. The learning algorithm is employed for 90% of the cases, while the remaining 10% is used to estimate the learned model's future accuracy. The neural network is used in a variety of biological applications [8].

- The difficulty of coding region recognition and gene identification.
- Identifying and analyzing signal binding sites and regulatory sites.
- Feature detection and sequence classification.

## 2 Literature Review

Individuals worked to identify some new items to apply after assessing the importance of this area. They wanted to complete the job more quickly and accurately. Other researchers have made some substantial and fascinating contributions to this study field in the past.

Liangyou Chen and Lois Boggess used the genomic signature to research gene classification. They investigated this issue using neural network technologies and its four back-propagation algorithm methods, radial-basis functions, review panel machine, and self-organizing maps [9]. After the experiment, the committee machine produced the best results, with an average inaccuracy of 16.88%.

Dr. P. Kiran Sree, Dr. P. S. V. Srinivasa Rao, and S. S. S. N. Usha Devi N proposed a novel concept for gene prediction and a new metagenomics classifier using a machine learning technique of transfer learning in their research [10]. To solve some bioinformatics problems, it combined hybrid cellular automata with a deep educational environment. With this classification, they were able to reach 98.7% accuracy in just 8 ns. According to Vrinda V. Nair, Karthika Vijayan, Deepa P. Gopinath, and Achuthsankar S. Nair's research, they employed ANN and chaos game representation to classify unknown genomic segments [11]. To test this proposed strategy, eight subsets from the taxonomic classification distribution of eukaryotic organisms were gathered. Qicheng Ma, Jason T. L. Wang, Dennis Shasha, and Cathy H. Wu used a neural network and an expectation–maximization technique to identify DNA sequences [12]. They introduced a novel method for classifying DNA sequences, to detect *E. coli* promoters in the bacteria's DNA. Determine whether a given DNA sequence is an *E. coli* promoter or not. The problem described above is known as a binary classification problem. Jun Miyake, Yuhei Kaneshita, Takashi Hirano, Satoshi Asatani, Seiichi Tagawa, and Hirohiko Niioka used deep learning to investigate a new approach for classifying DNA sequences [13]. The DNA of human leukocyte antigen alleles was employed in this categorization, which was done as a graphical classification. Data is gathered from the Database of Immune Polymorphism and compressed into a two-dimensional format.

Vrinda V. Nair examined a new approach for organism categorization based on a combination of FCGR and ANN [11], which was created by Karthika Vijayan, Deepa P. Gopinath, Achuthsankar S. Nair. They used the genetic sequence to solve the difficulty of categorizing organisms into distinct groups. For this, they used a few different species. For PNN-based classification, they achieved an accuracy of 86.8%. Jonathan Auerbach, Damian Gola, Elizabeth Held, Emily R. Holzinger, Marc-Andre Legault, Rui Sun, Nathan Tintle, Hsin-Chou Yang, and Inke R. König used machine learning and data mining to solve the problem of complex genomic data classification [14]. They have metagenome issues as well as a combination of several data structures. Another study looked into using discriminative k-mers to classify metagenomics and genomic sequences.

### 3 Methodology

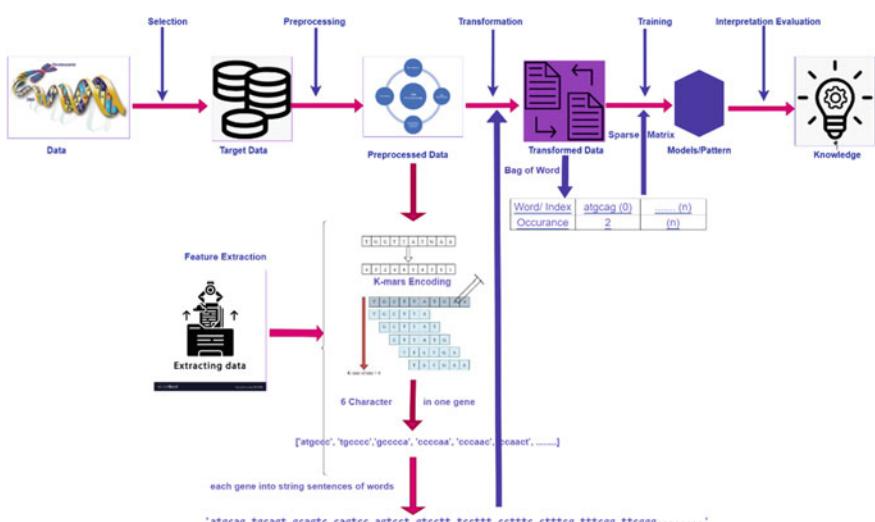
The research is divided into two phases: preprocessing and post-processing. The approach in the preprocessing phase focuses on data preprocessing stages, whereas the workflow in the article phase can be broken down into two subparts: model learning and framework evaluation. Figure 4 depicts a representation of a research study. The working follow of the study is related to machine learning (ML) and natural language processing (NLP). The NLP is used for processing the texting data and converting the data into a string then numerical values to fit the machine learning model. The architecture is following some steps that are discussed below.

#### A. Data Collection:

We have collected this dataset from the Kaggle repository. This dataset is available as public. The name of this dataset is “human\_data.txt”. The dataset contain two features one is deoxyribonucleic acid (DNA)sequencing and another one is class. The size of this dataset is (4380,2) where 4380 is the number of samples and 2 is the number of columns. Below, the dataset shown in Fig. 5.

Table 1 is showing about the gene family that is class and we have six classes in this dataset. We have also the number of occurrences per class as well the numeric values of the gene family class. This means we have converted the string class to a numeric class from 0 to 6 (7 class).

In Fig. 6, we have sketched one count plot to plot the occurrences per class have. The transcription factor (class 6) has the most data among all 7 classes and the number is 1343. The lowest class is the Lon channel which has 240 samples.

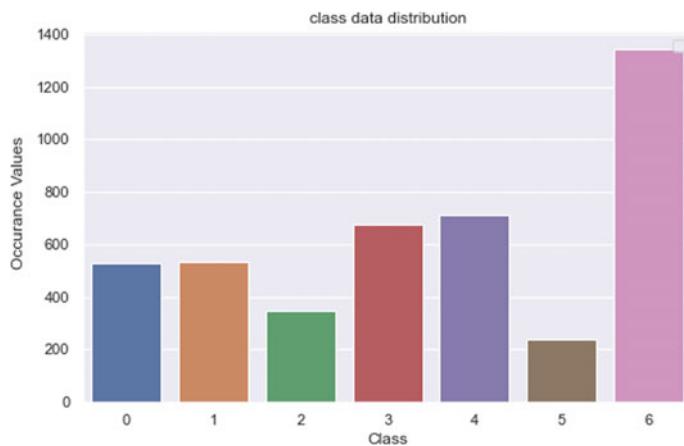


**Fig. 4** Proposed architecture

sequence	class
ATGCCCAACTAAACTACCGTATGGCCCACCATAATTACCCCCA...	4
ATGAACGAAAATCTGTCGCTTCATTGCCCCACAATCCTAG...	4
ATGTGTGGCATTTGGCGCTGTTGGCAGTGATGATTGCCTTCTG...	3
ATGTGTGGCATTTGGCGCTGTTGGCAGTGATGATTGCCTTCTG...	3
ATGCAACAGCATTGAATTGAATACCAGACCAAAGTGGATGGT...	3

**Fig. 5** Samples of dataset**Table 1** Gene family, number of occurrences, and their class label

Gene family	Number	Class label
G protein coupled receptors	531	0
Tyrosine kinase	534	1
Tyrosine phosphatase	349	2
Synthetase	672	3
Synthase	711	4
Lon channel	240	5
Transcription factor	1343	6

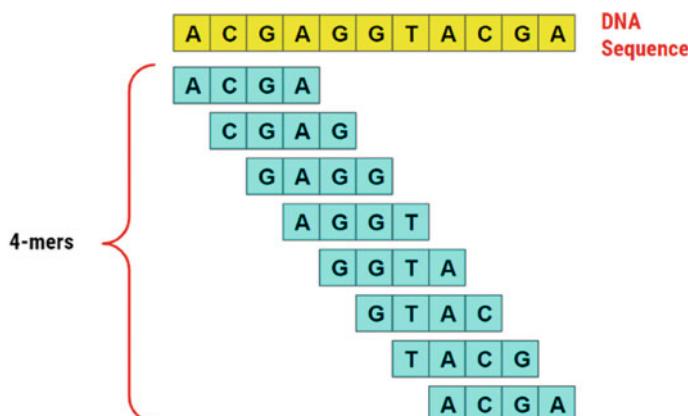
**Fig. 6** Class data distribution

### B. Data Preprocessing:

Data preprocessing is the procedure for preparing raw data for use in a machine learning algorithm. It is the first and most important stage in building training data. In data preprocessing, we have to do the different tasks for instance getting the dataset, importing libraries, importing datasets, finding missing data, encoding categorical data, splitting the dataset into training and test sets, and feature scaling. Due to the growing volume of huge datasets, datasets frequently contain missing or ambiguity. The extraction of information will be severely hampered by poor data quality.

### C. Feature Extraction:

After preprocessing the data, the next step is feature extraction from the processed data. Feature extraction is the most important task for the machine learning model. In feature learning extraction, we have to do some extra work on data to train the model. The model is receiving the features as input and produces the accepted output. For extracting the feature, we have so many techniques for the different datasets. In our case, we are using the k-mer counting algorithm for extracting the feature. From DNA sequences to amino acids, the k-mer technique simulates the process. A three-dimensional window is utilized to explore the entire DNA sequence, with a sliding unit of one at each step. The group of three nucleobases from the DNA sequence is obtained each time, and the associated amino acid is recorded. Stop codons are often overlooked. All different types of amino acids are counted after the entire DNA sequence has been traversed. After that, each amino acid's proportion is determined and plotted in a histogram. Consider the same DNA sequence as before. The DNA sequence “TTTGACTCGT” contains eight codons. “TTT,” “TTG,” “TGA,” “GAC,” “ACT,” “CTC,” “TCG,” and “CGT” are the acronyms for “TTT,” “TTG,” “TGA,” “GAC,” “ACT,” “CTC,” “TCG,” and “CGT.” (Fig. 7).



**Fig. 7** 4-mers in the sequence “ACGAGGTACGA”

**Algorithm 1:**  $k - mers$  counting algorithm

---

```

// Start with the Empty Dictionary
1. counts ← Empty Dictionary for Storing the Unique  $k - mers$  values
//Calculate how many kmers of length k there are
2. num_kmer ← len(DNA) – length_of_kmers + 1
3. for i in num_kmers do
4.   kmer ← DNA[i : i + k] // read i to i+k
    //Add the kmer to the dictionary if it is not there
5.   if kmer not in counts do
6.     counts[kmer] ← 0
7.   counts[kmer] ← += 1 // counts value increment by 1
// Return the final Values
8. return counts

```

---

**D. Transformation:**

Data transformation is an important task for machine learning. For transforming the data, we used natural language processing (NLP) technique that is the CountVectorizer method. The count is a Scikit-learn module. The vectorizer programmed converts a corpus of text into a vector of term/token frequencies. It also allows you to preprocess your text data before producing the word vectors, making it a very versatile text feature extraction module. To show how CountVectorizer is working, let's have an example that can be converted to a sparse matrix.

DNA = [atgcagtgcagtgcagtccagtcatgcaggcagtcatgcagcgacgacgacgag]

DNA	atgcag	tgcagt	gcagtc	cagtcc	cgacga	gacgag
Occurance	3	1	2	1	1	1

DNA	0	1	2	3	4	5
Occurance	3	1	2	1	1	1

We convert raw text to a numerical vector representation of words and n-grams using CountVectorizer. This makes it simple to use this format as signals (features) in machine learning tasks like text classification and clustering. The size of the sparse matrix is (4380, 232,414). We have 4380 (rows) documents and 232,414 unique words (columns).

**E. Machine Learning Model:**(i) *Random Forest:*

It is necessary to understand how a decision tree classifier [15] operates before discussing the random forest technique. A decision tree is a tree-like construct

that mimics human decision-making. Each node has a judgment, and the data is divided into separate child nodes. The final findings are displayed in the leaf nodes. The decision is evaluated using the impurity drop, and a good query should maximize the impurity drop. The decision tree is also a supervised learning model in which the model learns how to make queries and split data until specified criteria or threshold is reached using the training set. The random forest algorithm's underlying model is the decision tree [16].

A forest is made up of many trees, as its name suggests. The random forest model uses data to train several decision trees, with the average output of these trees being used as the final result. To construct different training datasets, the bootstrap aggregating approach is employed. This method takes some data from the original training dataset and replaces it with new data, which is then used to train a single tree. The random forest is a straightforward, easy-to-understand method that can handle difficult nonlinear classification problems. In our approach, two hyperparameters are required to be fine-tuned. The number of estimators is one of them. It determines the number of trees that should be planted during the trial. The other factor is each tree under certain. This number should be neither too high nor too low. Greater groups may result in the classifier, while smaller depths may result in parameters.

(ii) Logistic Regression:

A linear model used to do binary classification is known as logistic regression. The output unit, like the MLP model, was calculated using the sigmoid function. To avoid any overfitting, L2 regularization was also applied. The regularization term is added after the loss function in this method, as shown in formula (3). L2 regularization is the word for the additional term. The L2 regularization [17] degree is controlled by the hyperparameter  $C$  in this phrase. Greater regularization is indicated by smaller values.

$$\text{Loss} \leftarrow \text{Loss} + \frac{1}{C} \sum_{i=1}^n w_i^2 \quad (1)$$

The data points in the input  $X$  are usually considered to have a nonlinear model. The majority of the time, however, this assumption is incorrect. In such cases, logistic regression is a credible alternative model.

(iii) Support Vector Machine:

Another linear model for classification is the support vector machine (SVM) [18, 19]. An SVM has great generalization ability since it can handle a little amount of data and is less sensitive to noise in a dataset [20]. The goal of the SVM is to discover the hyperplane that maximizes the difference between the two classes. The Lagrange multiplier approach can be used to find the solution. If kernel functions are employed correctly, powerful nonlinear SVM models can be trained. Kernel functions generate new feature vectors, which are typically larger than the original input. In the new feature space, the SVM discovers the new hyperplane, which is linear.

## 4 Results and Discussions

The results and discussion part consist of some confusion matrix, learning curve, comparison graph, and more other information. We have used, six machine learning algorithms to classification DNA requesting. The approach we are using is nothing but a natural language processing technique. The k-mer technique and bag-of-words are used for preprocessing and transformation the data. For evaluated the result, we are used the confusion matrix, precision, recall, and F1-score. The evaluated matrix has four major to calculated accuracy for instance true positive (TP), false positive (FP), true negative (TN), and false negative (FN). Below the accuracy, precision, recall, and F1-score equations are written according to the confusion matrix.

$$\text{Accuracy} = \frac{t_p + t_n}{t_p + t_n + f_p + f_n} \quad (2)$$

$$\text{Precision} = \frac{t_p}{t_p + f_p} \quad (3)$$

$$\text{Recall} = \frac{t_p}{t_p + f_n} \quad (6)$$

$$F1 - \text{score} = \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (5)$$

Table 2 displays the accuracy of a single set of training and test sets, the mean accuracy of a tenfold training and test, and the sample variance in the tenfold process of selecting test cases.

As per the results in Table 2, the multinomial Naïve Bayes algorithm not only performs the best with this data onefold but also has the least variance while completing k-fold training and test data selection and subsequent training.

Other quantitative metrics comprehensive, like accuracy, recall, and F1-scores, are shown in Table 3 for all the algorithms used in both courses. The multinomial Naïve Bayes approach outperforms the others with a precision of 0.99%. On the other

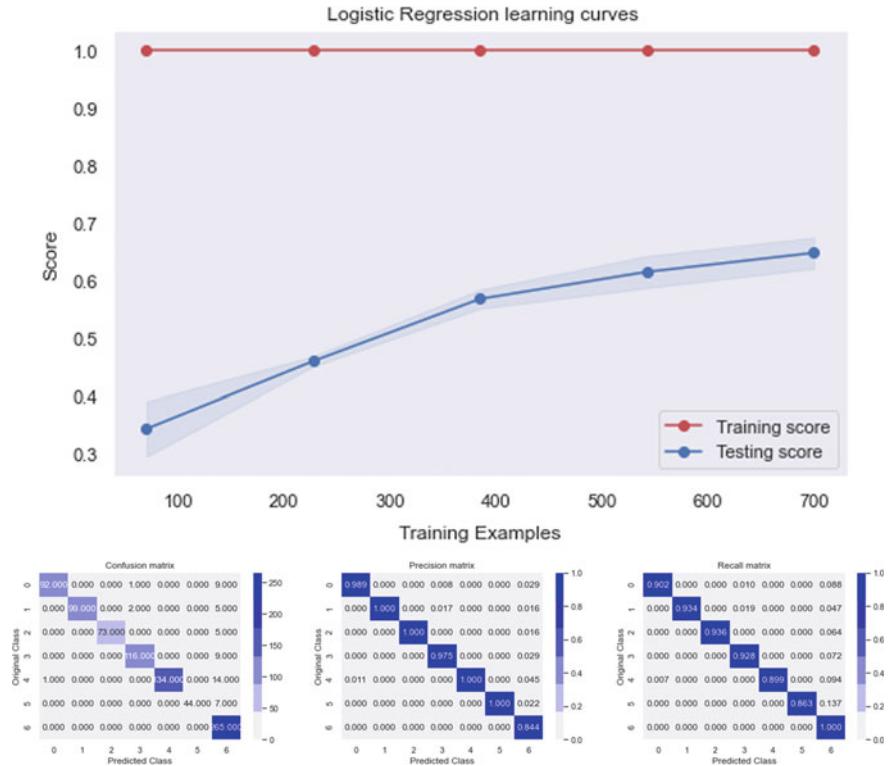
**Table 2** Comparison between 10-folder accuracy vs without cross-validation accuracy

Machine learning algorithms	K-fold Ac. (%)	Accuracy (%)
Decision tree	84.5	80.93
Support vector machine	87	80.59
Logistic regression	92.10	93.94
K-nearest neighbors	82.54	76.14
Random forest	90	91.55
Multinomial Naïve Bayes	<b>97.70</b>	<b>98.40</b>

**Table 3** Performance measures like **a** precision, **b** recall, and **c** F1-score

(a)							
	Classes	Logistic regression	KNN	Decision tree	Multinomial Naïve Bayes	Random forest	SVM
Precision	0	0.99	0.33	0.55	0.98	0.88	0.97
	1	1.00	1.00	0.85	1.00	1.00	1.00
	2	1.00	1.00	0.84	1.00	1.00	1.00
	3	0.97	1.00	0.83	0.99	0.79	1.00
	4	1.00	1.00	0.85	0.99	0.97	1.00
	5	1.00	1.00	0.84	1.00	1.00	1.00
	6	0.84	0.98	0.89	0.96	0.90	0.61
Averaged	0.97	0.90	0.81	<b>0.99</b>	0.94	0.94	
(b)							
	Classes	Logistic regression	KNN	Decision tree	Multinomial Naïve Bayes	Random forest	SVM
Recall	0	0.90	1.00	0.78	0.97	0.89	0.70
	1	0.93	0.72	0.80	0.98	0.91	0.69
	2	0.94	0.78	0.83	1.00	0.91	0.77
	3	0.93	0.85	0.81	0.99	0.93	0.74
	4	0.90	0.65	0.81	0.96	0.88	0.76
	5	0.86	0.67	0.73	1.00	0.84	0.61
	6	1.00	0.72	0.83	0.99	0.96	1.00
Averaged	0.92	0.77	0.80	<b>0.99</b>	0.90	0.75	
(c)							
	Classes	Logistic regression	KNN	Decision tree	Multinomial Naïve Bayes	Random forest	SVM
F1-score	0	0.94	0.50	0.65	0.98	0.89	0.81
	1	0.97	0.84	0.83	0.99	0.95	0.82
	2	0.97	0.88	0.84	1.00	0.95	0.87
	3	0.95	0.92	0.82	0.99	0.86	0.85
	4	0.95	0.79	0.83	0.98	0.92	0.86
	5	0.93	0.80	0.78	1.00	0.91	0.76
	6	0.92	0.83	0.86	0.98	0.93	0.76
Averaged	0.95	0.79	0.80	<b>0.99</b>	0.92	0.82	

hand, regularly outperforms all other algorithms in almost every table. This practically eliminates the potential of a type-1 error (i.e., a false positive indicating that class2 was mistakenly identified as class1) or a type-2 error (i.e., a false negative indicating that class2 was incorrectly detected as class1) (i.e., a false negative indicating



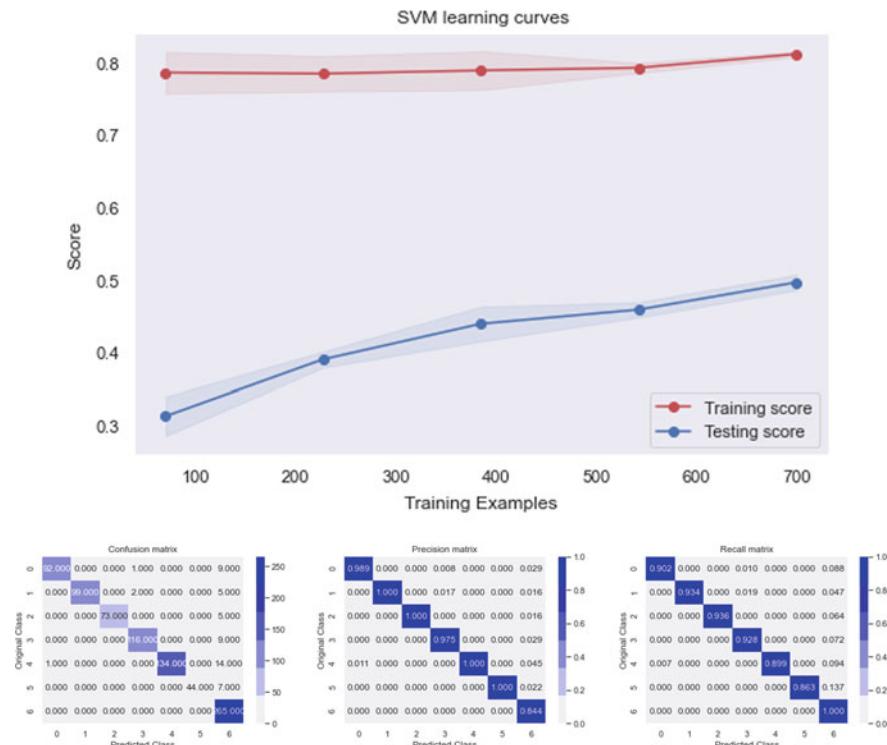
**Fig. 8** Logistic regression learning curve and confusion, precision, recall matrix

that class1 was mistakenly identified as class1). The F1-score is then calculated by averaging the best accuracy values [17, 21].

At this point, the learning curves of the models and the evaluation matrix-like confusion matrix, and precision-recall matrix are shown in Figs. 8, 9, 10, 11, 12, 13.

From the investigation of the learning curves, it is notified that the multinomial Naïve Bayes algorithm training and testing learning curves are almost closer to each other than other models do not have. On the other hand, the confusion matrix score is also better than other models. The confusion matrix value is almost 99%. The average score of precision and recall score is also 99%. According to the information, multinomial Naïve Bayes is the best model for this dataset (Fig. 14).

Depending on the comparison graph of all algorithms, it is being proved that the multinomial Naïve Bayes shows better performance than other models. The second most accuracy was performed by logistic regression (94%), the third-most accuracy performed by random forest (91%), and the lowest accuracy performed by KNN (76%).

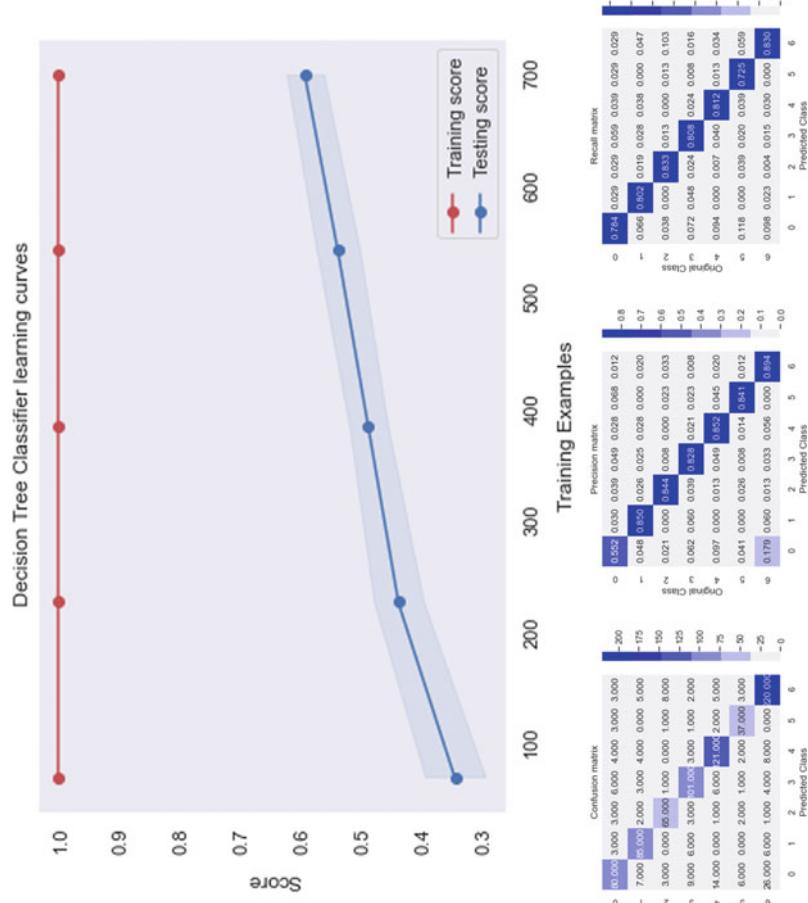


**Fig. 9** Logistic regression learning curve and confusion, precision, recall matrix

## 5 Conclusion and Future Scope

The DNA sequence database is still growing. This is because many new viruses, bacteria, or genomics sequences have been discovered as a result of research, and this has contributed to many unsolved difficulties affecting the DNA sequence of organisms. Deep learning is also evolving. As a result, deep learning appears to be much more promising in the future. It is reasonable to expect the discovery of a new invention or discovery, such as a new hybrid approach combining two procedures. A more elegant future will emerge when deep learning is discovered and can be used to design and integrate a system that can evolve and adapt to any environmental or contextual variation.

The purpose of this study was to assess the suggested model's ability to classify DNA sequences. The results were also compared to the prior best performance to see if the proposed model had improved. Furthermore, this study looked at different lengths of sequence to see how well the proposed model performed. The overall results of this study are within the expected range.



**Fig. 10** Decision tree learning curve and confusion, precision, recall matrix

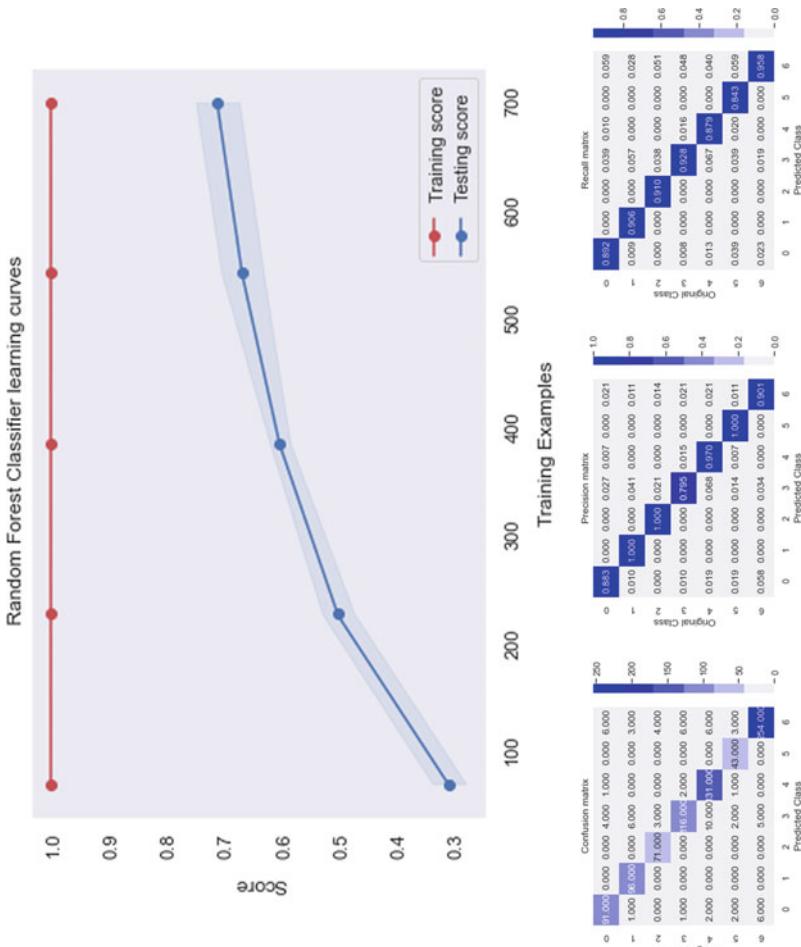
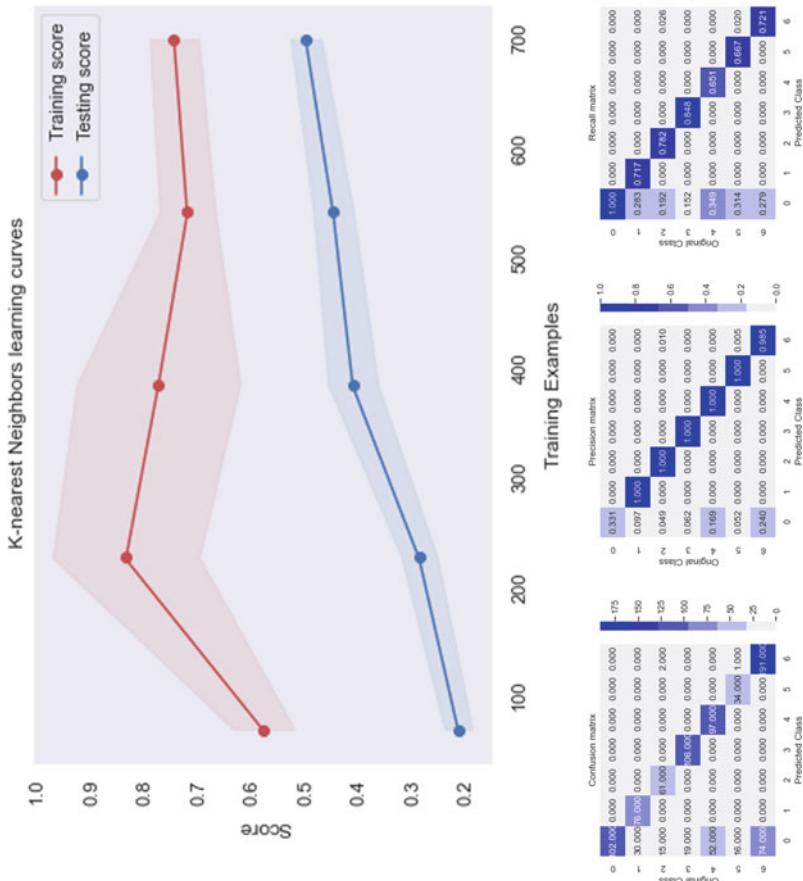
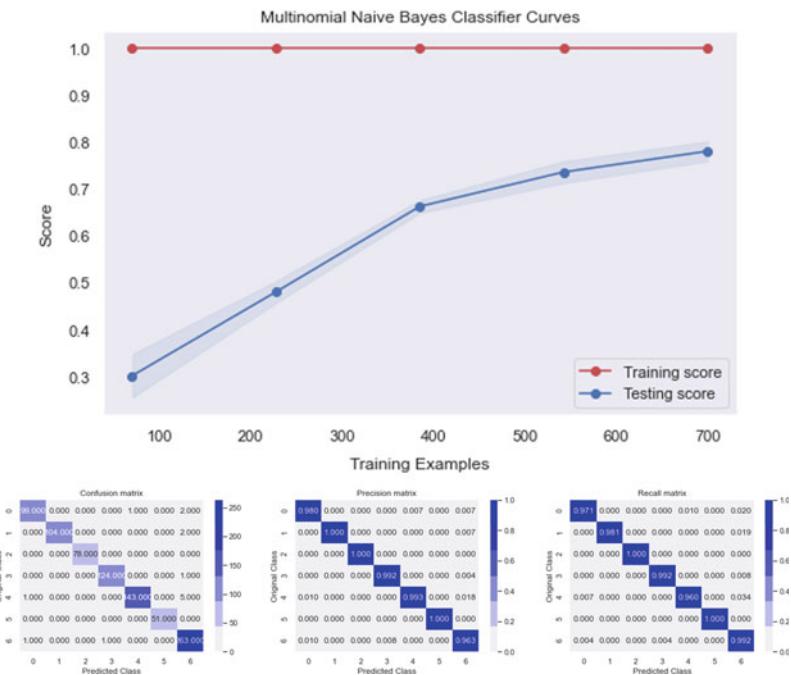


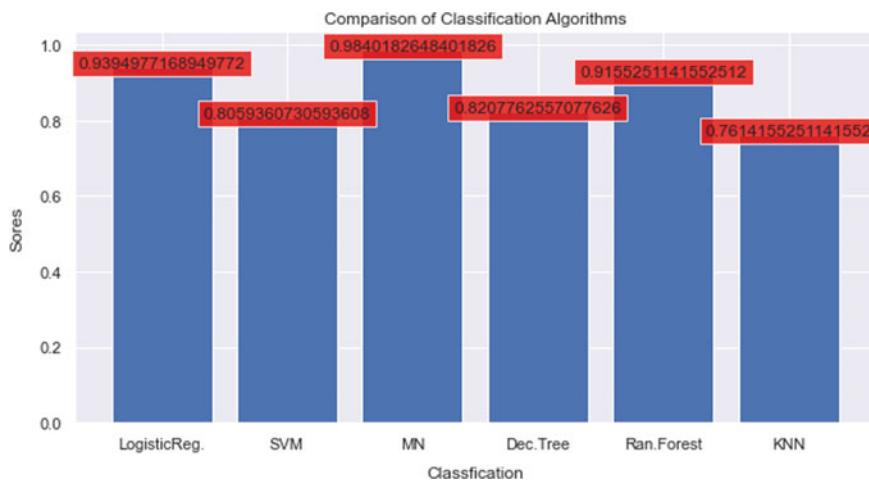
Fig. 11 Random forest learning curve and confusion, precision, recall matrix



**Fig. 12** KNN learning curve and confusion, precision, recall matrix



**Fig. 13** Multinomial Naive Bayes learning curve and confusion, precision, recall matrix



**Fig. 14** Comparison of algorithms accuracy

## References

1. Gelfand, M.S.: Prediction of function in DNA sequence analysis. *J. Comput. Biol.* **2**(1), 87–115 (1995)
2. Bukh, J., Purcell, R.H., Miller, R.H.: Importance of primer selection for the detection of hepatitis C virus RNA with the polymerase chain reaction assay. *Proc. Natl. Acad. Sci.* **89**(1), 187–191 (1992)
3. Dorn-In, S., Bassitta, R., Schwaiger, K., Bauer, J., Hözel, C.S.: Specific amplification of bacterial DNA by optimized so-called universal bacterial primers in samples rich in plant DNA. *J. Microbiol. Methods* **113**, 50–56 (2015)
4. Pacheco, M.A., Cepeda, A.S., Bernotienė, R., Lotta, I.A., Matta, N.E., Valkiūnas, G., Escalante, A.A.: Primers targeting mitochondrial genes of avian haemosporidian: PCR detection and differential DNA amplification of parasites belonging to different genera. *Int. J. Parasitol.* **48** (8), 657–670 (2018)
5. Mridha, K. et al.: Deep learning algorithms are used to automatically detection invasive ducal carcinoma in whole slide images. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 123–129 (2021). <https://doi.org/10.1109/ICCA52192.2021.9666302>
6. Mridha, K., et al.: Web based brain tumor detection using neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 137–143 (2021). <https://doi.org/10.1109/ICCA52192.2021.9666248>
7. Zheng, Y., Azevedo, R.B.R., Graur, D.: An Evolutionary Classification of Genomic Function, vol. 7, no. 3, p. 4 (2015)
8. Mridha, K., Pandey, A.P., Ranpariya, A., Ghosh, A., Shaw, R.N.: Web-based brain tumor detection using neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 137–143 (2021)
9. Boggess, L., Chen, L.: Neural networks for genome signature analysis. In: 9th International Conference on Neural Information Processing (ICONIP'0Z)
10. Srinivasa Rao, P.S.V., Usha Devi, N.S.S.S.N., Kiran Sree, P.: CDLGP: a novel unsupervised classifier using deep learning for gene prediction. In: IEEE International Conference on Power, Control, Signals, and Instrumentation Engineering (2017)
11. Vijayan, K., Gopinath, D.P., Nair, A.S., Nair, V.V.: ANN-based classification of unknown genome fragments using chaos game representation. In: Second International Conference on Machine Learning and Computing (2010)
12. Wang, J.T.L., Shasha, D., Wu, C.H., Ma, Q.: DNA sequence classification via an expectation–maximization algorithm and neural networks: a case study. In: IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews (2001)
13. Sinha, T., et al.: Analysis and prediction of COVID-19 confirmed cases using deep learning models: a comparative study. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_18](https://doi.org/10.1007/978-981-16-2164-2_18)
14. Auerbach, J., Gola, D., Held, E., Holzinger, E.R., Legault, M.-A., Sun, R., Tintle, N., Yang, H.-C., König, I.R.: Machine learning and data mining in complex genomic data—a review on the lessons learned in Genetic Analysis Workshop 19, p. 8 (2016)
15. Ross Quinlan, J.: Induction of decision trees. *Mach. Learn.* **1**(1), 81–106 (1986)
16. Breiman, L.: Random forest. *Mach. Learn.* **45**(1), 5–32 (2001)
17. Mridha, K., Kumar, D., Shukla, M., Jani, M.: Temporal features and machine learning approaches to study brain activity with EEG and ECG. *Int. Conf. Adv. Comput. Innov. Technol. Eng. (ICACITE)* **2021**, 409–414 (2021)
18. Mridha, K., Kumbhani, S., Jha, S., Joshi, D., Ghosh, A., Shaw, R.N.: Deep learning algorithms are used to automatically detection invasive ducal carcinoma in whole slide images. In: 2021 IEEE 6th International Conference on Computing, Communication, and Automation (ICCCA), pp. 123–129 (2021)

19. Palimkar, P. et al.: Machine learning technique to prognosis diabetes disease: random forest classifier approach. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_19](https://doi.org/10.1007/978-981-16-2164-2_19)
20. Hamidi, O., Poorolajal, J., Sadeghifar, M., Abbasi, H., Maryanaji, Z., Faridi, H.R., Tapak, L.: A comparative study of support vector machines and artificial neural networks for predicting precipitation in Iran. *Theoret. Appl. Climatol.* **119**(3–4), 723–731 (2015)
21. Mridha, K.: Early prediction of breast cancer by using artificial neural network and machine learning techniques. In: 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT), pp. 582–587 (2021)

# Hybrid Exponential Smoothing-LSTM-Based Univariate Stock Market Prediction for Financial Sectors in NIFTY50



C. Koushik, M. V. Pranav, R. K. Arjun, and S. Shridevi

**Abstract** Due to the highly volatile nature of stock market data, preprocessing is as important as fitting the prediction model. Most works fail to preprocess the data for removing fluctuations that directly impact the prediction capabilities of the system. This paper presents a hybrid system of a statistical-cum-deep learning model for univariate stock price forecasting. Exponential smoothing (ETS) is coupled with long short-term memory cells (LSTM), thereby obtaining a weighted average to predict a time series output. ETS helps in dynamically disassembling the key components of each individual time series and allows the model to learn how to represent them. Multi-layer LSTMs allow the model to better capture long-term seasonal interactions, thus ensuring a more efficient form of training. With a weighted pinball loss, a similar learning approach for LSTM and ETS leads to concurrent optimization of the model's prediction performance. Such an ensemble also advocates for a strong regularization and also promises minimal errors. These are reflected by a maximum RMSE of 4.27, MAE of 1.95, and MAPE of 0.12 from testing the model with financial sector datasets of the NIFTY50 dataset. The hybrid approach proves more efficient and superior to individual time series forecasting models.

**Keywords** Hybrid system · Stock price · Forecasting · Time series · Exponential smoothing (ETS) · Long short-term memory (LSTM)

---

C. Koushik · M. V. Pranav

School of Computer Science and Engineering, Vellore Institute of Technology,  
Vandalur-Kelambakkam Road, Chennai, Tamil Nadu 600127, India

R. K. Arjun

School of Electronics Engineering, Vellore Institute of Technology, Vandalur-Kelambakkam  
Road, Chennai, Tamil Nadu 600127, India

S. Shridevi (✉)

Centre for Advanced Data Science, Vellore Institute of Technology, Vandalur-Kelambakkam  
Road, Chennai, Tamil Nadu 600127, India

e-mail: [shridevi.s@vit.ac.in](mailto:shridevi.s@vit.ac.in)

## 1 Introduction

In today's economy, the stock market, often known as the equity market, has a major effect. They are places where individual and institutional investors meet to purchase and sell publicly owned shares, and currently exist as electronic markets. The rise or decline in the share price has a significant impact on the investor's profit. The stock market rate is dependent on many factors including but not limited to government policies, exchange rates, inflation, commodity price index, and others making it unpredictable.

All these aspects make share prices volatile and very difficult to forecast with high precision. Even geopolitical transformations can affect the stock market share trend, which was demonstrated by the impact of COVID-19 on stock price [1]. It is, therefore, difficult to carry out a reliable trend assessment on financial data.

The dated market hypothesis stresses that the nature of stock markets is completely random, and that prediction of such a trend is impossible. Nevertheless, advancements in computational techniques have demonstrated that most stock values are mirrored in past records, hence establishing that movement trends are critical to successfully predict such values.

Several robust machine learning and neural network models have been tested and implemented on these time series datasets to analyze the movement and predict stock values for near future. Machine learning and deep learning algorithms are the most efficient way to resolve this type of problem where a particular trend needs to be captured and visualized. ML techniques have the potential to make accurate predictions by discovering patterns or insight that cannot be comprehended on paper. Prediction provides knowledge of the current status of movement of stock prices. This can be beneficial in decision making for customers deliberating whether to purchase/sell the shares of a particular stock. The flexibility of machine learning models [2] has led to several researchers and enthusiasts considering it as the mainstream solution to predictive forecasting. These methodologies can be classified as AI models (deep learning techniques) or mathematical/statistical models.

ARIMA, ETS, and linear regression are typical examples of the latter. Although ARIMA and ETS can handle a standard time series problem, the use of additional operations such as decomposition or extension of the model are required by linear regression. These statistical models are relatively more simple, non-complex, and robust leading them to be popular choices among amateur/non-professional users. On the other hand, AI models manage to trade their simplicity for a faster learning curve and more accuracy. Neural networks (NNs) have been the most desired and researched deep learning techniques in recent times [3]. Their popularity can be attributed to their plethora of features such as learning capabilities, uniform estimation characteristics, ability to model nonlinear data, parallelism, etc. In specific, recurrent neural networks (RNNs) have been the go-to model for time series forecasting. They exhibit dynamic behavior while processing an input sequence by employing the use of a directed network along a concurrent sequence of inputs generated by the links between their nodes. Latest research demonstrates that RNNs (like LSTMs) are

extremely accurate in prediction and outmatch traditional methods such as ARIMA, SVMs, and ANNs. The main contribution of this paper is twofold:

1. Decomposing time series data using exponential smoothing (ETS) model to deal with seasonal fluctuations that contribute to errors.
2. Discussing how the hybrid ETS–LSTM model pipeline after deseasonalizing and smoothing the data for predicting stock prices outperforms existing state-of-the-art approaches to stock market predictions.

In specific, recurrent neural networks (RNNs) have been the go-to model for time series forecasting. They exhibit dynamic behavior while processing an input sequence by employing the use of a directed network along a concurrent sequence of inputs generated by the links between their nodes. Latest research demonstrates that RNNs (like LSTMs) are extremely accurate in prediction and outmatch traditional methods such as ARIMA, SVMs, and ANNs.

## 2 Related Works

Pang et al. [4] take an innovative neural network approach to enhance stock market predictions in real time. Deep learning models of long short-term memory (LSTM) with an embedded layer and encoders was proposed to vectorize the data, and the two models were compared in terms of prediction accuracies. Results show that the embedded approach produces better accuracy. Li et al. [5] analyze deep learning techniques for stock market analysis and forecasting. Studies showed that LSTM is widely implemented and used till today and looked for possible areas of research and development in this domain. Dudek et al. [6] project a hybrid deep learning framework for electric load forecasting. The proposed work integrates LSTM, ensembling and exponential smoothing to extract principal components from independent time series. Results showed that the proposed hybrid model is better than the existing state-of-the-art traditional ARIMA, ETS, and machine learning models.

EWT decomposition and LSTM models were used for stock closing price forecasting by Liu and Long [7]. The ORELM model was used for error correction to improve the accuracy of the overall system. The proposed hybrid model predicts the closing price precisely. A custom dataset was created by Yadav et al. [8] from Indian stock market data and analyzed. Since LSTM models are highly influenced by the selection of several parameters, the custom neural network architecture was adjusted by correlating stateless and stateful models and then tuning the model. A hybrid model consisting of ARIMA and XGBoost was used by Wang and Guo [9] for stock price forecasting, wherein simulation results show that the proposed system has good approximation and fitting abilities and predicts the opening price of the stock index accurately. Boosting is introduced to handle partial error data, whose results are combined using wavelet reconstruction.

Mehtab et al. [10, 11] built eight regression and eight classification models using the training data that consisted of NIFTY50 index records from 2014 to 2018. With

the models, the open values of NIFTY50 for 2018–2020 were predicted. Then, four models were built based on LSTM with four different architectures. Then, a comparative analysis between the different models showed that random forest yielded the best results on average when it came to classification-based models while LSTM-based univariate model with a mean absolute percentage error (MAPE) of 8.7% outperformed all the other machine learning models. After building and testing this framework, public sentiment was then taken into account. A sentiment augmented SOFNN algorithm based on natural language processing was implemented, and the results made it the most accurate model to forecast the stock prices. This study proved that public sentiment is very valuable input in stock price prediction.

Jain et al. [12] developed a model based on artificial neural networks by considering important factors that influence the banking and financial sectors. The model was then used to predict the stock prices of companies in the NIFTY50 index. A mix of gradient descent algorithm, data processing using sigmoid functions were used for the activation function and the network was configured. The prediction was performed on a total of ten companies, and results were achieved with error percentage ranging from 16.13 to 6.13.

Devi et al. [13] took the stock data of NIFTY50 midcap and built an ARIMA model for stock trend analysis. Akaike information criterion Bayesian information criterion was then applied to predict the accuracy of the model. A percentage error accuracy of 16.26 was achieved. Roondiwala et al. [14] proposed a recurrent neural network and LSTM method to approach the stock market prediction. The LSTM model was made up of a sequential input layer followed by two LSTM layers, a ReLU activation layer, and an output layer with a linear activation function. Efficiency was analyzed with RMSE, and a best result of 0.00859 was achieved.

Smyl, S [15] proposes a combination of exponential smoothing and RNNs for time series forecasting. The process involved deseasonalization and normalization of data, applying exponential smoothing formulas, preprocessing and a neural network model for forecasting. Certain criteria have to be met for a model to be good for time series forecasting. Siami-Namini et al. [16] have compared ARIMA and LSTM in time series forecasting. Monthly data of the NASDAQ index for a period of 33 years was taken as the dataset for the following comparison. ARIMA and LSTM models were applied to the dataset, and the results were tabulated. RMSE values upon prediction indicate that LSTM models outperform ARIMA significantly with 84–87 percentage reduction in error rates.

A hybrid model that combines the autoregressive moving reference (ARMR) model, exponential smoothing, and RNN was used by Rather et al. (2015) for prediction of stock returns. A sliding window approach was used, and a new prediction is calculated in every sliding window. MSE of 25 stocks for the hybrid model was observed as 0.0013 which outperforms the base RNN model. Moghar et al. [17] performed stock market prediction on GOOGL and NKE using LSTM models. 12, 25, 50, and 100 epochs were tried and tested on training data. The results showed that more epochs improved the training result of the model, and LSTM showed varied results as the training data was changed.

### 3 Materials and Methods

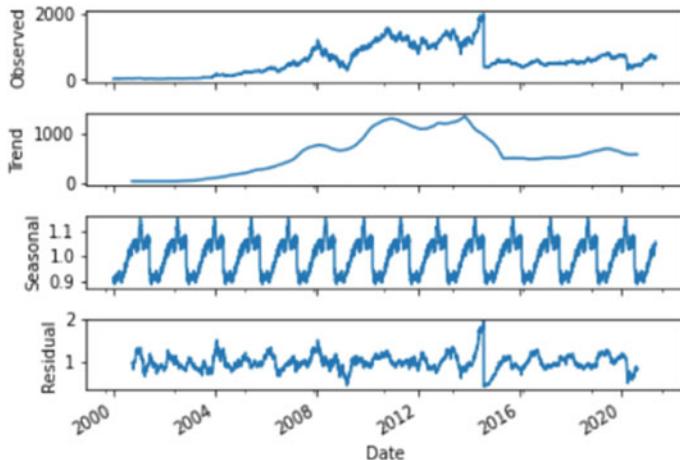
The NIFTY50 dataset is comprised of data derived from the National Stock Exchange (NSE) of India and has been made publicly available on Kaggle. It includes the price history and trading volumes of the fifty equities that form the NIFTY50 index which are recorded on a daily basis, with pricing and trade values spread across multiple months. Each stock data has a separate csv file and a metadata file that includes macro-information on the stock. At the time of usage, the dataset spans from January 2000 to April 2021 for each company.

Datasets of companies in the financial sector include:

1. Axis Bank Ltd. (AXISBANK)
2. Housing Development Finance Corporation Ltd. (HDFC)
3. HDFC Bank Ltd. (HDFCBANK)
4. ICICI Bank Ltd. (ICICIBANK)
5. Kotak Mahindra Bank Ltd. (KOTAKBANK)
6. IndusInd Bank Ltd. (INDUSINDBK)
7. State Bank of India (SBIN)

Motivated by existing proposals, this system also takes a similar approach but extends to daily forecasts for stock market data following a combination of two different types of forecasting onto a single system. The hybrid model creates a unique pipeline where the outputs from exponential smoothing (ETS) model are preprocessed and fed into the LSTM-based recurrent neural network to generate stock market predictions. This approach combines a mathematical prediction model with a neural network to work as a single framework, thereby increasing its accuracy and effectiveness. The proposed system was inspired by the submission to the M4 competition given by Smyl et al. (2020) for monthly forecasts. Dudek et al. [6] proposed a similar approach but with a modified architecture of the models for the purpose of forecasting electric load.

Due to the inherently unpredictable nature of stock market, the associated time series data may carry seasonal fluctuations, and movements, rendering a nonlinear trend that makes forecasting extremely difficult. As shown in Fig. 1, time series data is typically a combination of trend, seasonal, and error (residual) components. For such datasets, the predominant approach to reduce the complexity of forecasting would be to decompose the time series data, furthermore, dissect the seasonal and trend components from the original data. This decomposition can either be additive or multiplicative. In additive model, the time series data is a summation of individual components while multiplicative takes the product. The process of removal of seasonal/trend component is known as deseasonalization/detrending. They produce simpler data for unraveling patterns through modeling from which, seasonal and trend components are restored to produce final outputs.



**Fig. 1** Decomposition of time series data (AXISBANK) into components

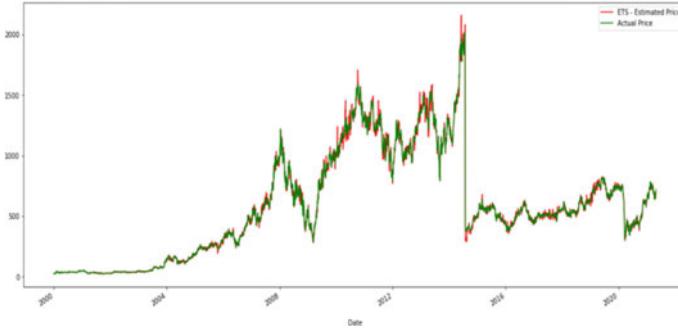
### 3.1 Exponential Smoothing (ETS)

ETS is a statistical predictive model that is operated on univariate time series data. It can be used as an extension to support features with a systemic trend or a seasonal component. It is an alternative to the popular autoregressive integrated moving average (ARIMA) method and used for forecasting use-cases where the data is non-stationary, i.e., having trend and seasonal components. Some combinations of these models are similar in working, but the two primarily differ in assignment of weights to past observations, or lag values. ETS explicitly uses an exponentially decreasing weight assignment for past observations. The ETS captures the error, the trend (or level), and the seasonal component. Existing models do not simplify this complex eight structure which makes the final solutions inaccurate and unoptimized. The ETS model has an incorporated system to deal with trends and seasonality in a better manner, wherein the individual components can either be added, multiplied, or simply left out in a smoothing calculation.

For the NIFTY50 datasets, we use the multiplicative Holt-Winters triple exponential smoothing model from the statsmodels Python module, and the parameters were decided with hyperparameter tuning using grid search. For the individual ETS model, we compare the estimated values from the model with the original stock prices from the test dataset. Figure 2 shows the original and estimated values from the ETS model in the form of a line plot for the AXISBANK dataset.

For the hybrid system, we use the ETS model to extract the level ( $L_i$ ) and seasonality ( $S_i$ ) in the original stock market data for every data point ( $Y_i$ ). Since ETS is used for univariate analysis, we only consider a single feature from all the datasets.

For modeling, we use the column VWAP, short for volume weighted average price, a general trading benchmark preferred by traders. To preprocess the data before fitting



**Fig. 2** Movement of estimated and actual stock prices for AXISBANK

into the LSTM, we simplify the series by deseasonalization and log normalization given by Eqs. (1) and (2). The resulting series lies in a very narrow range close to 1 which speeds up the entire training process and leads to a much efficient and faster approach to convergence. The resulting points are stored in vector ( $X$ ).

$$X_i = Y_i / (L_i * S_i) \quad (1)$$

$$X_i = \log\left(Y_i / (L_i * S_i)\right) \quad (2)$$

### 3.2 LSTM-RNN

LSTM is a specific type of recurrent neural network (RNN) that utilizes feedback connections making it apt for learning long-term attributes in sequential data. Just like other RNNs, each output of a training set is included in the next iteration of training, by shifting the batch by one step. As shown in Fig. 3, typical LSTM blocks consist of a memory cell, whose state remains static over time, coupled with three nonlinear gates that regulate the flow of information.

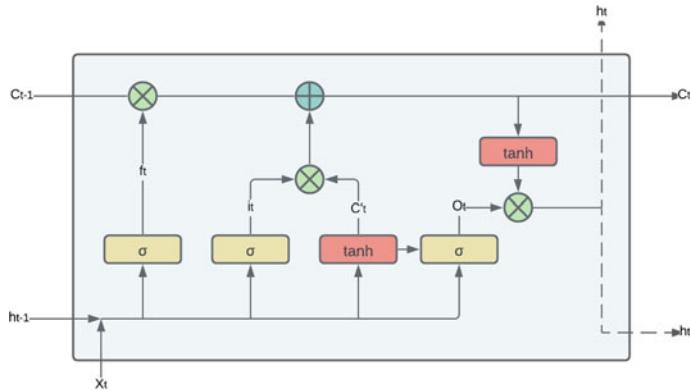
The mathematical equations [18] that describe the LSTM cells are defined below

$$i_t = \sigma(x_t U^i + h_{t-1} W^i) \quad (3)$$

$$f_t = \sigma(x_t U^f + h_{t-1} W^f) \quad (4)$$

$$o_t = \sigma(x_t U^o + h_{t-1} W^o) \quad (5)$$

$$C'_t = \tanh(x_t U^g + h_{t-1} W^g) \quad (6)$$



**Fig. 3** LSTM cell structure

$$C_t = \sigma(f_t * C_{t-1} + i_t * C'_t) \quad (7)$$

$$h_t = \tanh(C_t) * o_t \quad (8)$$

where  $i_t$  is the input gate,  $f_t$  is the forget gate, and  $o_t$  is the output gate.  $x_t$  is the input vector,  $h_{t-1}$  is previous cell's output,  $C_{t-1}$  is previous cell's memory, and  $h_t$  and  $C_t$  represent output and memory for the current cell.

Due to delays of inconsistent durations between key occurrences in a time series, LSTM networks are well-suited for categorizing, processing, and generating predictions. While LSTMs were originally created to solve the problem of vanishing gradients that occur while training time series data on a normal RNN, they have also been shown to perform better than most other sequence learning approaches due to its relative insensitivity to gap length.

In our system, the input to the LSTM module is the preprocessed time series data that is transformed into a sequence represented by an array. The data is first split into training and testing sets in the ratio 60:40. The window size is taken as 20 and the data is split into input and output vectors for training and testing sets. Output vector holds the forecasted sequence for the input. The vectors are then reshaped into a 3D vector containing the samples, time steps, and features to match the format of the LSTM's input layer. Each subset is then fitted into the LSTM model. The model is built using Keras's sequential API. The architecture contains two hidden LSTM layers with 50 units and activation function as 'ReLU' followed by a dense layer with 1 neuron to output the estimated value.

After fitting the model, the test data is used for prediction. The outputs from the LSTM model are then post-processed to retrieve the final stock values (VWAP) by restoring the trend and seasonal components for each output value after reversing log operation. The post-processing phase is defined by (9) as given.

$$\hat{Y}_i = \exp(X_i) * (L_i * S_i) \quad (9)$$

## 4 Results and Discussion

The post-processed values, i.e., predicted VWAP values are stored in a dataframe along with test set's VWAP values. Finally, the performance of the hybrid model is evaluated using error metrics such as the root mean square error (RMSE), mean absolute error (MAE), and mean absolute percent error (MAPE).

### 4.1 Error Calculations

The RMSE is the square root of the average between squared differences between actual and the forecasted values. RMSE is defined by (10) as shown.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N (Y_i - \hat{Y}_i)^2}{N}} \quad (10)$$

MAE is the mean of absolute differences between actual and forecasted values. MAE is defined by (11) as shown.

$$\text{MAE} = \frac{\sum_{i=1}^N |Y_i - \hat{Y}_i|}{N} \quad (11)$$

MAPE is the percentage of average of absolute differences between the actual and forecasted values. MAPE is defined by (12) as shown.

$$\text{MAPE} = \frac{\sum_{i=1}^N \left| \frac{Y_i - \hat{Y}_i}{Y_i} \right| \times 100}{N} \quad (12)$$

### 4.2 Results

We also considered individual ETS and LSTM models for a comparative analysis to draw a parallel between their performances. Table 1 shows different metrics for the

**Table 1** Results and comparisons among different datasets and models

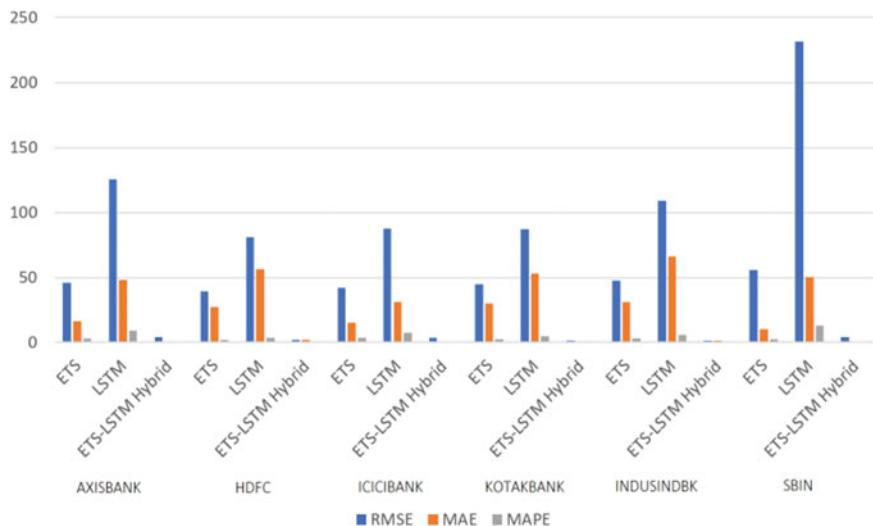
Dataset	Model	RMSE	MAE	MAPE
AXISBANK	ETS	45.77	16.49	2.88
	LSTM	125.73	48.34	8.94
	Proposed hybrid	4.27	0.26	0.05
HDFC	ETS	39.15	27.45	1.69
	LSTM	80.94	56.60	3.42
	Proposed hybrid	2.05	1.95	0.12
ICICIBANK	ETS	42.29	15.41	3.52
	LSTM	87.60	31.06	7.25
	Proposed hybrid	3.47	0.29	0.06
KOTAKBANK	ETS	44.60	29.86	2.65
	LSTM	86.97	53.25	4.62
	Proposed hybrid	1.64	0.85	0.11
INDUSINDBK	ETS	47.37	31.28	2.90
	LSTM	109.19	66.01	5.59
	Proposed hybrid	1.54	1.26	0.10
SBIN	ETS	55.80	10.45	2.53
	LSTM	231.50	50.19	13.07
	Proposed hybrid	3.97	0.25	0.07

aforementioned models to indicate their performance on each dataset. The comparative analysis carried out in Table 1 showcases the highly minuscule error rates on the NIFTY50 datasets proving the technical superiority of the proposed hybrid model.

Figure 4 exhibits a visual representation of the attained results through implementations of ETS, LSTM, and the ETS-LSTM hybrid models.

As the proposed system is completely novel in its approach and creates a new technological initiative to progress the domain of financial analytics, an inter-comparison has been carried out between the ETS, LSTM, and our proposed ETS-LSTM hybrid proposed methodology. Although there is a deficiency in extant works employing similar simulation parameters, we aimed to study some of the approaches on the NIFTY50 datasets and tabulated their results for a cross-comparison. Table 2 shows an overall analysis of our proposed system with existing related works that have been tested on the NIFTY50 datasets, in terms of average error percentage.

We see that the proposed hybrid system performs very well with an average error percentage of just 3.57% for the NIFTY50 datasets which is a significant improvement from other works. The inter and cross comparative analysis helps provide a better representation of our proposed system's capabilities.



**Fig. 4** Metrics comparison of the models with different datasets

**Table 2** Comparison of error percentages with related works

	Model	Error percentage (%)
Mehtab et al. [11]	Univariate LSTM	8.7
Jain et al. [12]	ANN with custom configuration	8.41
Devi et al. [13]	ARIMA	16.26
Proposed System	Univariate hybrid ETS-LSTM	3.57

## 5 Conclusion

In this work, we have proposed and subsequently validated a new architecture for stock price forecasting. A statistical forecasting approach was coupled with an AI-backed one to predict and forecast stock price values by remapping the governing trends. The model follows a hierarchical system consisting of a global portion trained across multiple time series parts (LSTM weights) and a time series specific part (ETS coefficients and seasonal components). Deseasonalization and normalization are performed using ETS-inspired algorithms that isolate the major components of individual time series. Preprocessed time series are forecasted using LSTM cells, which capture long-term seasonal correlations and provide more efficiency owing to their incorporation of dilated recurrent skip connections and a spatial shortcut path from lower layers. The proposed architecture was used to estimate the stock prices of companies that work in the finance sector featured in the NIFTY50 index. Resultant metrics illustrate the model's significantly reduced error rate and improved

accuracy when compared to its individual classical models, as well as traditional ML models. With a maximum RMSE of 4.27, MAE of 1.95, and MAPE of 0.12, the proposed algorithm proves efficient in improving stock price. Future direction of the research can be extended toward improving the transparency of the algorithm to build human trust by providing interpretable rationalization through explainable artificial intelligence (XAI).

## References

1. Lee, H.S.: Exploring the initial impact of COVID-19 sentiment on US stock market using big data. *Sustainability* **12**(16), 6648–6648 (2020)
2. Umer, M., Awais, M., Muzammul, M.: Stock market prediction using machine learning (ML) algorithms. *ADCAIJ: Adv. Distributed Computing Artif. Intell. J.* **8**(4), 97–116 (2019)
3. Singh, R., Srivastava, S.: Stock prediction using deep learning. *Multimed. Tools Appl.* **76**, 18569–18584 (2017)
4. Pang, X., Zhou, Y., Wang, P., Lin, W., Chang, V.: An innovative neural network approach for stock market prediction. *J. Supercomput.* **76**(3), 2098–2118 (2020)
5. Li, A.W., Bastos, S.: Stockmarket forecasting using deeplearning and technical analysis: a systematic review. *IEEE Access* **8**, 185232–185242 (2020)
6. Dudek, G., Pełka, P., Smyl, S.: A hybrid residual dilated LSTM end exponential smoothing model for mid-term electric load forecasting. *arXiv preprint arXiv:2004.00508* (2020)
7. Liu, H., Long, Z.: An improved deep learning model for predicting stock market price time series. *Digital Signal Processing* **102**, 102741–102741 (2020)
8. Yadav, A., Jha, C.K., Sharan, A.: Optimizing LSTM for Time Series Prediction in Indian Stock (2020)
9. Wang, Y., Guo, Y.: Forecasting method of stock market volatility in time series data based on mixed model of ARIMA and XGBoost. *China Commun.* **17**(3), 205–221 (2020)
10. Mehtab, S., Sen, J.: A Robust Predictive Model for Stock Price Prediction Using Deep Learning and Natural Language Processing. Available at SSRN 3502624 (2019)
11. Mehtab, S., Sen, J., Dutta, A.: Stock price prediction using machine learning and LSTM-based deep learning models. In: *Symposium on Machine Learning and Metaheuristics Algorithms, and Applications*. Springer, Singapore (2020)
12. Jain, V.R., Gupta, M., Singh, R.M.: Analysis and prediction of individual stock prices of financial sector companies in nifty50. *Int. J. Inf. Eng. Electron. Bus.* **10** (2), 33 (2018)
13. Devi, B.U., Sundar, D., Alli, P.: An effective time series analysis for stock trend prediction using ARIMA model for nifty midcap-50. *Int. J. Data Mining Knowl. Manage. Process* **3** (1), 65 (2013)
14. Roondiwala, M., Patel, H., Varma, S.: Predicting stock prices using LSTM. *Int. J. Sci. Res. (IJSR)* **6**(4), 1754–1756 (2017)
15. Smyl, S.: A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *Int J Forecast.* **36**(1), 75–85 (2020)
16. Siami-Namini, S., Tavakoli, N., Namin, A.S.: A comparison of ARIMA and LSTM in forecasting time series. In: *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 16 (2018)
17. Moghar, A., Hamiche, M.: Stock market prediction using LSTM recurrent neural network. *Procedia Computer Sci.* **170**, 1168–1173 (2020)
18. Vassamopoulos, S., Bertels, K., Almudever, C.G.: Comparing neural network based decoders for the surface code. *IEEE Trans. Comput.* **69**(2), 300–311 (2019)

# Smart Contract for Academic Certificate Verification Using Ethereum



Shivani Pathak, Vimal Gupta, Nitima Malsa, Ankush Ghosh, and R. N. Shaw

**Abstract** A certificate is a document that a person receives after completing a course. This certificate can be used for recruitment purpose and getting admission in a college or institution. There is no system in place for institutes/universities to share information place to detect counterfeit credentials in the admissions process. As a result, a blockchain-based approach for storing certificates is needed that will prevent admittance forgeries via fake certifications. The proposed system is based on blockchain technology, which boosts privacy, security, and openness efficiency while lowering costs. To combat this issue, the proposed system must be both immutable and transparent. This article gives an overview of step-by-step development methodology to build a blockchain-based academic certificate verification system. A smart contract is a critical component of this system. Using Sublime Text 3 as a text editor with Remix Ethereum as an integrated development environment, a smart contract called AcadCerti was created for holding certificates. Using this AcadCerti smart contract, certificates can be saved on the blockchain that can be utilised by the company's or admissions team's during recruitment or admissions process. Because blockchain technology has a transparency aspect, the same certificate can be viewed by the employer, admissions staff, and student. They all have access to this information at the same time.

**Keywords** TruffleIDE · Blockchain technology · Certificate verification · Smart contracts · Remix · Sublime text 3

---

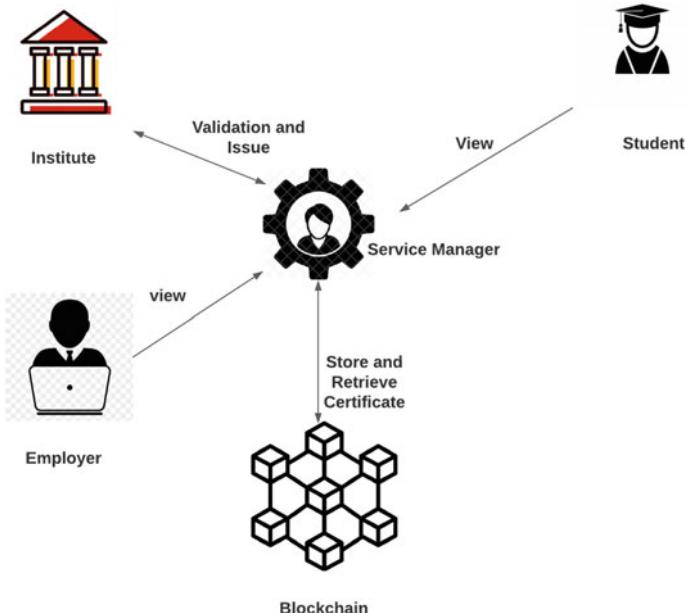
S. Pathak · V. Gupta · N. Malsa (✉)  
JSS Academy of Technical Education, Noida, India  
e-mail: [nitimamalsa@jssaten.ac.in](mailto:nitimamalsa@jssaten.ac.in)

V. Gupta  
e-mail: [vimalgupta@jssaten.ac.in](mailto:vimalgupta@jssaten.ac.in)

A. Ghosh · R. N. Shaw  
University Center for Research & Development (UCRD), Chandigarh University, Mohali, Punjab, India  
e-mail: [r.n.s@ieee.org](mailto:r.n.s@ieee.org)

## 1 Introduction

A good academic record is a tremendous since it is an asset demonstrates the knowledge of the holder, competence, and expertise, and it is used to apply for scholarships and employment. People try to make phoney certificates because they are so valuable. Fake academic certifications have long been a concern in higher education. Forgery of certifications has become a serious problem, and the existing method makes it impossible to tell whether a document is genuine or not. The initiative, which is built on blockchain, aims to address this issue. Immutability, consensus, authenticity, security [1], and transparency are all important aspects of blockchain-based systems [2]. A blockchain ledger is safe and secure, and it resists physical damage. The information is saved electronically and may be viewed at any time from any location. Many companies rely on their employers to authenticate papers. Blockchain was first used to construct cryptocurrencies [3], but it is currently envisioned as a versatile technology that will be employed in a variety of fields [4]. The technology's main use is in finance for predicting bitcoin prices using machine learning algorithms [5–7]. The paper [8] describes one such use. Figure 1 illustrates the suggested framework. A smart contract that is not optimised will cost more than the required gas, resulting in an overcharge to the user. As a result, gas cost optimisation has been achieved to a significant degree, and the number of transactions can be increased to further improve the situation [9]. The contract begins its execution when a student requests to an academic institution for certificate. The university recognises the learner and asks for the hash of his or her account, to which the student would want to give the certificate. The certificate is issued to the blockchain as a transaction by the educational institution after receiving the account. After that, the Miners validated the transaction. Any company who wishes to see a student's certificate must submit an application for a certificate. The service supervisor gives the employer all of the student's educational information that has previously been collected on the chain. The company may make proper judgments in the recruiting process after examining all of the information linked to the student's schooling. A distributed application (dApp) development method called "AcadCertificate" is detailed in this work. The suggested framework for the AcadCertificate is described in Sect. 1 of the document. The second section is a literature review, which discusses current techniques for verifying certificates and their qualities. The third section is methodology, which explains how the research was conducted procedures for putting together AcadCertificate dApp, as well as the essential dependencies. There are five phases to creating the dApp AcadCertificate in total. Sublime Text 3 and the Remix Ethereum IDE are used for the first three phases. Step 4 may be found in the IDE for Truffles. The fifth step is performed using Metamask as a wallet. The findings of smart contract testing are described in Sect. 4's result section with the Ethereum. IDE has been reworked. Conclusion and future prospects are discussed in Sect. 5.



**Fig. 1** A proposed framework for certificate verification system

## 2 Literature Survey

There are a variety of certificate verification systems available. The following are a few of them: Blockcerts is a blockchain-based certification system that tracks a list of a person's qualifications and verifies them on the blockchain. Because the procedure is entirely digital, the original issuer is not required to verify a person's credentials. Blockcerts offer their own set of benefits and drawbacks. The key security problem, certificate revocation problem, and identity problem of the Certificate Awarding Institution are only a few of the key drawbacks [10, 11].

Smartsert is a method comparable to another one developed for Jordan's Al-Zaytoonah University. The system was created to store the credentials of students. They have access to the system and may retrieve credentials as needed. This technology makes use of a database that is 100% accurate and protects data security and secrecy. Because this system is prone to assaults, additional emphasis on security measures is required [12, 13]. Recordkeeper is a blockchain-enabled service that securely saves and maintains data and information. The blockchain-based application has a number of advantages, including data security, immutability, and transparency. The verification of certificates is done using recordkeeper. Healthcare, property registry, supply chain management, and other associated services are some other blockchain applications. Assualts against Recordkeeper, like Smartsert, are common [14]. VECefblock is another form of prototype that has been created. It was created

to address the issue of false certificates in Vietnam. VECefblock was built on the Hyperledger Fabric blockchain technology. This prototype made advantage of Amazon EC2 is a public cloud platform provided by Amazon. This is a working prototype includes the entire system's conception as well as a data querying system that a data querying system that is efficient [15].

Another research was conducted established a digital application academic for the purpose of certificate verification utilising the Ethereum is a distributed ledger, which decreased costs and enhanced accuracy [16]. For certificate verification, a framework based on robotics has been developed. A smart contract has been created to store a student's certificates [9]. A solution was developed to solve the challenges of certificate dependability and security. By utilising current open-source platforms, it is straightforward to implement [17, 18]. There is now a hashing function employed to maintain the integrity of the files in all of the systems suggested or have been developed thus far employing. The blockchain is a type of technology. MD5, SHA-256, and CRC32 are a few examples the most often used hashing methods. Common consensus methods include Proof of Work (PoW), Proof of Stake (PoS), Delegated Proof of Stake (DPoS), Proof of Elapsed Time, and others. Other consensus methods, for instance, the dynamic average consensus method [19, 20], are also examined. There are several forms of smart contracts developed as blockchain systems such as Ethereum, Hyperledger, and others have continued to expand. Smart contracts are computer programmes that represent a non-trusting party's agreement. They use a collection of procedural policies for scenario-response to build business logic. They also have decentralised, shared programmes that are distributed on a block chain. Deployment should take a smart contract that is well-written that automates on behalf of the signing parties, and the agreement is executed [21, 22]. They give a service a variety of services, including as financial services and market forecasting. Security concerns and privacy disclosure, on the other hand, are obstacles for this type of technology foreshadows time ahead study. Contracts are a type of business logic that may be used in a variety of ways. Expressed using solidity is an example of a high-level language. They are functions for embedding that are called when an event happens. They have immutability characteristics, making them irreversible in nature. During various stages of the contract's lifetime, the construction of a contract encounters distinct problems. These problems, creation, deployment, execution, and conclusion, to name a few issues, are explored in this research [23]. When a smart contract is activated, it begins to execute, transforming from one state to the next. The blockchain makes it easier to keep track of the state of smart contracts execution of smart contracts when particular there are trigger conditions and occurrences [24]. Another study on blockchain technology describes step-by-step approach for storing a certificate on blockchain [25]. Paper on healthcare monitoring systems [26] shows a smart contract for healthcare monitoring, and further gas optimisation techniques have been applied on smart contract to reduce the gas cost.

### 3 Methodology

This paper describes a gradual, step-by-step approach for the development of proposed system CERTbchain's distributed application from start to finish (dApp) (through a smart contract that verifies educational certificates). A Web application is built in this case not part the protocol of the blockchain but is connected to it. This technology assists the company in authenticating the employee's identity authenticity of certifications stored on the blockchain. Because student certificates include sensitive information about students, they must be stored securely on the blockchain.

A smart contract (AcadCerti) is developed to store the certificates. Before inserting a certificate, the Institute must be registered. The Service Manager acts as a conduit between blockchain network and its users. (See 1st figure for an example.) An editor for text called the text editor Sublime Text 3 [27] is used to create a smart contract built on the foundation of solidity. Further the smart contract's functioning is tested using an Ethereum-based platform called Remix IDE [28]. Truffle IDE (version 4.0.4), Node.js for Web applications, and Metamask client (3.14.1) plug-in for Chrome dependencies are the tools you will need other softwares utilised to create the system. The processes for building the CERTbchain dApp are as follows, along with the requisite dependencies.

#### 3.1 *Formulating a Digital Contract*

Before deploying a smart contract, it is necessary to create one [29]. Figure 2 illustrates the AcadCerti smart contract's structure. The following five elements are listed in order of importance:

1. Name of the contract AcadCerti.
2. Owner, sendtoRecords, ipValue, facts, and working are application-varied specifics.
3. The contract's modifiers define the contract's rules and regulations.
4. Two distinct functions the contract has two functions: setAcadCerti() and setactive().
5. The contract's revert() method is used to handle any issues that occur.

#### 3.2 *Smart Contract Writing*

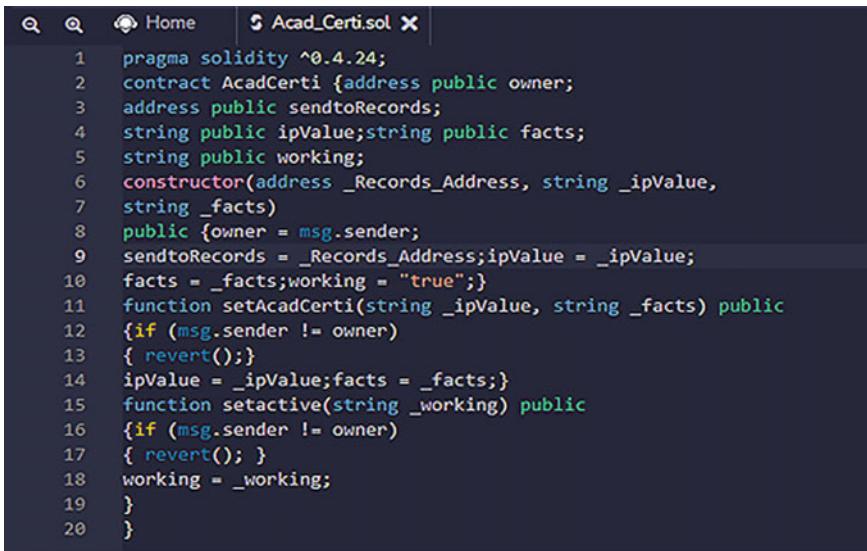
Figure 3 depicts a screenshot of the AcadCerti smart contract's written code. Pragma solidity code 0.4.24 displays the solidity version to the compiler for the AcadCerti smart contract in development. AcadCertismart contract contains a number of public variables that have been used to store contract-related data:

AcadCerti // Smart contract's full name
//variables for data and state
make a public statement owner
make a public statement sendtoRecords
a public string ipValue
a public string facts
a public string working
modifications // policies or rules
//functions
a function setAcadCerti(...)
a function setactive(...)
revert()
//handler for errors

**Fig. 2** Design for “AcadCerti” smart contract

- owner is a public variable containing the contract’s owner’s address.
- sendtoRecords is a public variable that contains the student’s account’s hash address.
- ipValue is a public variable that contains the certificate image’s hash.
- facts there is also a public optional argument that carries the certificate’s description.
- working is another public variable that stores the “True” or “False” value, which is the default value“True”.

AcadCerti is a smart contract created by AcadCerti and has the following features:



```

1 pragma solidity ^0.4.24;
2 contract AcadCerti {address public owner;
3 address public sendtoRecords;
4 string public ipValue;string public facts;
5 string public working;
6 constructor(address _Records_Address, string _ipValue,
7 string _facts)
8 public {owner = msg.sender;
9 sendtoRecords = _Records_Address;ipValue = _ipValue;
10 facts = _facts;working = "true";}
11 function setAcadCerti(string _ipValue, string _facts) public
12 {if (msg.sender != owner)
13 {revert();}
14 ipValue = _ipValue;facts = _facts;}
15 function setactive(string _working) public
16 {if (msg.sender != owner)
17 {revert();}
18 working = _working;
19 }
20 }

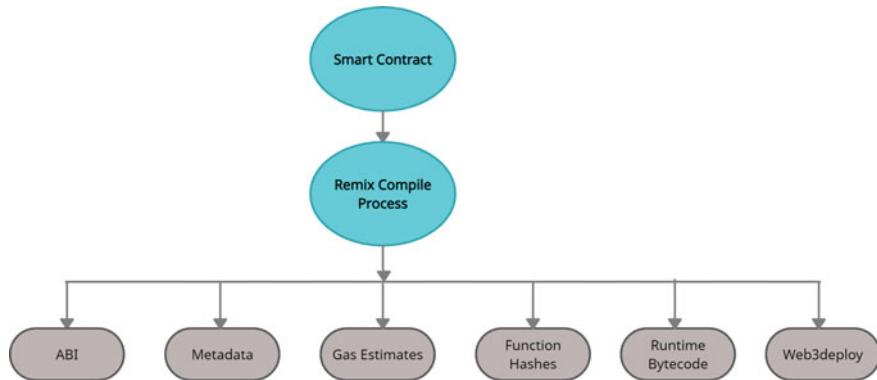
```

**Fig. 3** Smart contract code for AcadCerti

- constructor (sendtoRecords, ipValue, facts)—When constructing a new academic certificate, this function Object() [native code] is called and the value of the certificate is specified using the parameters value, where the contract’s admin is set to ‘sender account address’ and the active is set to “True”.
- setAcadCerti (sendtoRecords, ipValue, facts)—The arguments given with the setAcadCerti function are used to set the contract variables’ values.
- setactive (working)—This parameter’s value is “True” if it’s active, and “False” if it’s been removed but may be revived.
- As an error handler, the revert() method is utilised. If the communication does not originate from the contract’s original owner, the status of the contract is reverted to its original state.

### 3.3 Import the Smart Contract, Compile It, and Test It

1. Add the smart contract to your system: The contract is authored in the Sublime Text 3 text editor and then compiled into an Ethereum-based IDE.
2. Produce artifacts: After the compilation process, the resulting artifacts are shown in Fig. 4.  
A JSON file is produced once a smart contract is constructed, as seen in Fig. 5. The following are the major components:



**Fig. 4** Artifacts created as a result of the compilation contract [30]

- “Application Binary Interface (ABI) for Contracts”—Is a term that refers to a set of functions that have values for their parameters as well as their return values.
- “Metadata”—It summarises fundamental data information to make identifying and working with specific data instances easier.
- “Gas estimates”—A cost estimate for gas is calculated and saved in a JavaScript Object Notation document.
- “Hashes of functions”—refers to the function prototype’s first four bytes that allow a transaction to call the function.
- “Runtime bytecode”—This is the code that describes a smart contract and is kept on the blockchain.
- “Web3deploy”—Web3 refers to blockchain-based decentralised applications.

The following is the procedure for executing and deploying a AcadCerti smart contract:

- The contract is run by clicking the run button.
- The environment is configured using the JavaScript VM(London) (refer to Fig. 6).
- Remix creates ten default accounts. The administrator’s account is one of the accounts created (see Fig. 7 for an example). On the Remix screen, you can see the Web interface while deploying the contract, which contains the values of the variables (see Fig. 8 for an example).

accountAdd:0x5B38Da6a701c568545dCfcB03FcB875f56beddC4

ipHash:0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2 details: M.Tech.  
Degree Certificate.

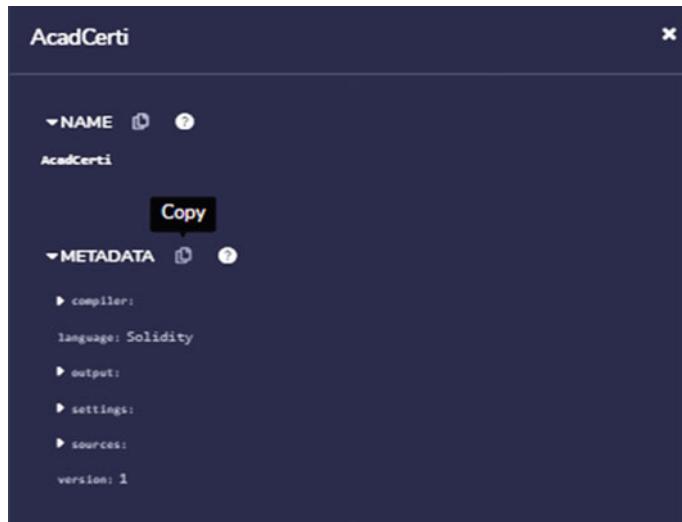


Fig. 5 AcadCerti created artifacts

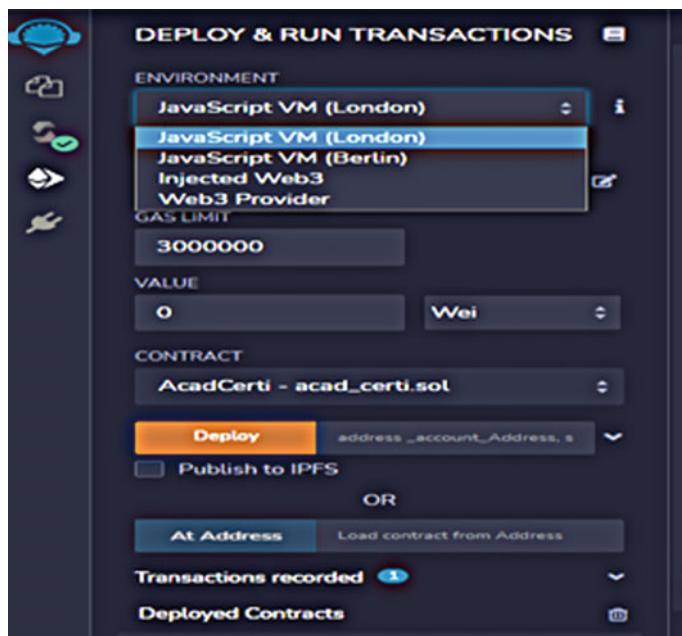


Fig. 6 Environment: Web interface using JavaScript virtual machine

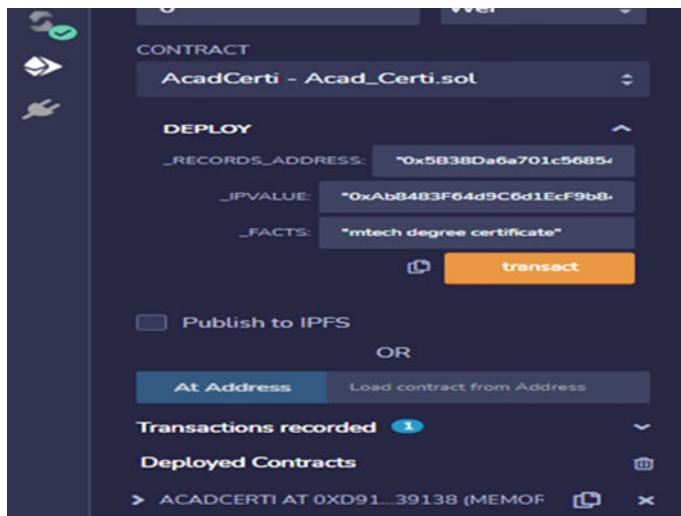


Fig. 7 Default accounts on the Web interface

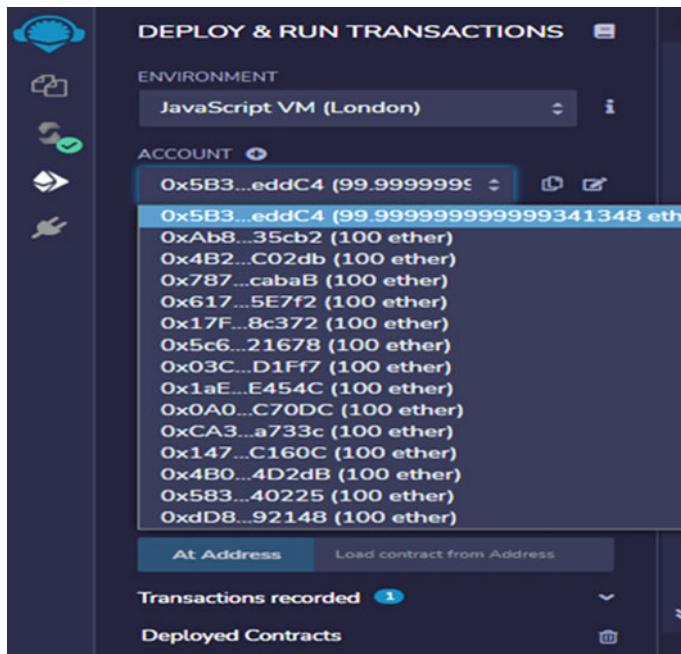


Fig. 8 The deployed Web interface AcadCerti contract

### 3.4 Component Integration in Dapps

Truffle IDE [31] is a development, deployment, and testing environment for end-to-end dApps.

The processes for putting all of the module of the AcadCertificate dApp are as follows:

1. Make a directory: The mkdir command is used to create a directory. Then, from this directory, run the “truffle init” command to create the AcadCERT—bchain dApp’s basic structure. After running the truffle init command, the following files and directories will appear.
  - Folder for contracts—Smart contract migration.sol is included in the source files for smart contracts.
  - Folder for migration—The smart contract is deployed by truffle using a migration method. This is for keeping track of changes.
  - Folder for test—It includes both JavaScript and solidity smart contract testing.
  - The file truffle.js and truffleconfig.js are two different types of truffle.js files—Truffle is a configuration file including blockchain network id, IP address, and RPC port information.
2. Develop a smart contract: The smart contract is compiled using the “truffle compile” command. This command generates the artifacts of a built contract, resulting in the creation of a build folder. This command is also in charge of displaying any smart contract syntax mistakes. The truffle has now been set up.
3. Produce the test: The command “truffle develops” generates a 10-account local test blockchain while integrating these accounts to Metamask, and it also shows seed words. Create a new file titled xyz contracts.js in the migrations a directory where the smart contract may be deployed. Fill up the blanks in the contracts.js and is a file created by xyz.
4. Execute the contract that has been compiled created: “truffle migrate—reset” is the command to use. On the screen, the statement “saving successful migration with contract name” shows. It preserves artifacts produced by the compiler for the smart contract “-reset” guarantees that the smart contract’s most recent version is deployed. Now, its time to deploy or transfer it to a test chain. Before implementing the smart contract, this test chain was deployed. Smart contract is deployed using the truffle.js file. It defines the Web app, and the network communication via the host IP address and port number is connected. Now, start a new terminal and type the following command “truffle develop”. It creates a test chain with ten accounts (account number 0 to account number 9). It will show ten distinct addresses as well as their private keys. Addresses are 160 bits long, and private keys are 256 bits long. After running this programme, mnemonics (seed words) become apparent. The built smart contract now has to be deployed on the chain of tests. Drag contracts.js into the migrations folder to do so using xyz. Run the command “truffle migrate—reset”. The name of the network, the migration

basis, the smart contract and the artifacts are all displayed. The developed smart contract was constructed and distributed via the blockchain.

5. Include a file for testing purposes: The command “truffle test” generates a environment for testing in which the smart contract that has been deployed may be tested. For smart contracts, there are both negative and positive tests used. JavaScript and solidity may both be used to create test cases. Insert the into the test script the folder for testing. Use the truffle test command to display the result of a fail/pass the test cases.

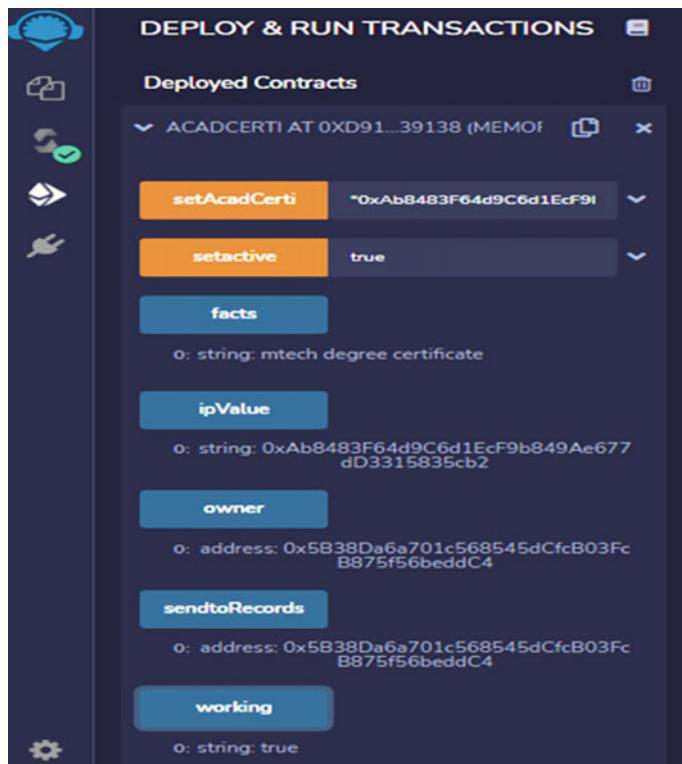
### ***3.5 Application of Metamask to the Front End of the Chain***

It is a browser plug-in that allows users to access Ethereum-based dApps. It connects to truffle’s blockchain to manage the accounts, serving as a connection between the app’s front end and the blockchain node that holds the accounts. The transaction may be signed using Metamask’s interface. It distributes the guess points required for transaction completion.

To get started with Metamask, go to matamask.io and install it using Chrome. Connect Metamask to the blockchain server; it acts as a connector between the blockchain server and the Web. In the Metamask wallet, there are several networks to choose from; the local test chain (port 9545) is chosen. This port can be used to expose accounts. Metamask may be used for the decentralised applications management. Write a directive npm run dev for the Web interface. It will launch a Web browser. The essential steps to execute the dApp may be completed using this online interface. The first and most important step in getting into the system is to register. Following registration, a request for a certificate is forwarded to the Educational Institute. The institute then acknowledges the student and requests his account hash. Once the student’s account hash is obtained, the institute begins the process of saving certificates. Students, employers, and other people may now see the same information.

## **4 Result**

The AcadCerti smart contract’s testing results are described in this section. The smart contract was tested using the Remix Ethereum IDE. Figure 9 depicts the results of the testing, which tell whether or not the smart contract AcadCerti is functioning properly. The contract includes the owner, sendtoRecords, ipValue, facts, and working fields, as well as the setactive and setAcadCerti methods. The contract’s functioning is tested by assigning values to these fields. The following are the values that have been assigned to the fields: networking: “True” The value “True” initiates the AcadCerti contract, whereas “False” deactivates it. The “True” value is supplied here. As a result, the contract is activated.



**Fig. 9** Results of smart contract AcadCerti

setAcadCerti:0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2, M.Tech. Degree Certificate.

The value of the Records Address parameter is set automatically when the function Object() [native code] is called. The values for the remaining there are two variables specified by means of filling inside the certification's ipValue information. Working: displays the string that was sent to set active.

Owner:0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 The contract was signed at the time of deployment; owner address was picked from a list of ten default addresses.

Facts: string: M.Tech. Degree Certificate facts As this is a string type variable, a string specified at the time of storing the certificate displays here. ipValue: Is a string: 0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2.

Because this ipValue variable is of the string type, it displays as a simply click the ipValue check. This is the address of the certificate that is specified in ipValue when the contract is deployed.

sendtoRecords:0x5B38Da6a701c568545dCfcB03FcB875f56beddC4.

Records the address for the student's records, which was provided at the time of deployment, is included in the add. As a result, the smart contract is operating according to its specifications.

## 5 Conclusion

A framework for developing a distributed application called AcadCertificate is proposed. Furthermore, the paper outlines a step-by-step procedure for creating a distributed blockchain-based academic certificate verification programme. There are five steps to take explained in the written document. The following are the steps: (1) Formulating a digital contract, (2) Smart contract script, (3) Import and create a smart contract, compile it, and put it through its paces, (4) Integration of Dapp(AcadCertificate) components, (5) Application of Metamask to the front end of the chain.

AcadCerti, a smart contract for keeping certificates, has been developed. In addition, contract functional testing results have been completed. Owner, sendtoRecords, ipValue, facts, and working are application-specific variables with values specified at deployment time. These settings are the same as throughout the contract's testing period (refer to Fig. 9). As a result, the smart contract AcadCerti has been proven to work properly. To improve the capacity of the AcadCertificate dApp, additional functionalities may be added to the smart contract or new smart contracts can be created. In addition, the same technology may be used to a variety of different applications. Furthermore, a gas cost review may be performed, and optimisation approaches can be used to lower the contract AcadCerti's gas cost.

## References

1. Rajawat, A.S., Rawat, R., Barhanpurkar, K., Shaw, R.N., Ghosh, A.: Blockchain-Based Model for Expanding IoT Device Data Security. In: Bansal, J.C., Fung, L.C.C., Simic, M., Ghosh, A. (eds.) *Advances in Applications of Data-Driven Computing. Advances in Intelligent Systems and Computing*, vol. 1319. Springer, Singapore
2. Casino, F., Dasaklis, T.K., Patsakis, C.: A systematic literature review of blockchain-based applications: current status, classification, and open issues. *Telemat. Inform.* **36**, 55–81 (2019)
3. Malsa, N., Vyas, V., Gautam, J.: RMSE calculation of LSTM models for predicting prices of different cryptocurrencies. *Int. J. Syst. Assur. Eng. Manag.* (2021)
4. Li, Y.: Emerging blockchain-based applications and techniques. *SOCA* **13**, 279–285 (2019). <https://doi.org/10.1007/s11761-019-00281-x>
5. Gautam, J., Atrey, M., Malsa, N., Balyan, A., Shaw, R.N., Ghosh, A.: Twitter data sentiment analysis using Naive Bayes classifier and generation of heat map for analyzing intensity geographically. In: Bansal, J.C., Fung, L.C.C., Simic, M., Ghosh, A. (eds.) *Advances in Applications of Data-Driven Computing. Advances in Intelligent Systems and Computing*, vol. 1319. Springer, Singapore (2021)
6. Malsa, N., Singh, P., Gautam, J., Srivastava, A., Singh, S.P.: Source of treatment selection for different States of India and performance analysis using machine learning algorithms for

- classification. In: Pant, M., Kumar Sharma, T., Arya, R., Sahana, B., Zolfaghariinia, H. (eds.) Soft Computing: Theories and Applications. Advances in Intelligent Systems and Computing, vol. 1154. Springer, Singapore (2020)
- 7. Bedi, P., Goyal, S.B., Rajawat, A.S., Shaw, R.N., Ghosh, A.: A framework for personalizing a typical web search sessions with concept-based user profiles using selective machine learning techniques. In: Bianchini, M., Piuri, V., Das S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore
  - 8. Malsa, N., Vyas, V., Gautam, J., Shaw, R.N., Ghosh, A.: Framework and smart contract for blockchain-enabled certificate verification system using robotics. In: Bianchini, M., Simic, M., Ghosh, A., Shaw R.N. (eds.) Machine Learning for Robotics Applications. Studies in Computational Intelligence, vol. 960. Springer, Singapore (2021). [https://doi.org/10.1007/978-981-16-0598-7\\_10](https://doi.org/10.1007/978-981-16-0598-7_10)
  - 9. Malsa, N., Vyas, V., Gautam, J., Shaw, R.N., Ghosh, A.: Reduction in gas cost for blockchain enabled smart contract. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6. IEEE (2021)
  - 10. Smolenski, N.: Top 10 reasons to use Blockcerts (2018) [Online]. Available: <https://medium.com/learning-machine-blog/top-10-reasons-to-use-blockcerts-ec7d29f2712>
  - 11. Jirgensons, M., Kepenieks, J.: Blockchain and the future of digital learning credential assessment and management. *J. Teach. Educ. Sustain.* **20**(1), 145–156 (2018)
  - 12. Szalachowski, P. (2020). SmartCert: redesigning digital certificates with smart contracts. arXiv preprint. [arXiv:2003.13259](https://arxiv.org/abs/2003.13259)
  - 13. Kanan, T., Obaidat, A.T., Al-Lahham, M.: SmartCert blockchain imperative for educational certificates. In: IEEE Jordan international Joint Conference 2019 on Electrical Engineering and Information Technology (JEEIT), pp. 629–633
  - 14. Lemieux, V.L., Hofman, D., Batista, D., Joo, A.: Blockchain technology and recordkeeping. ARMA International Educational Foundation (2019)
  - 15. Garba, A., Chen, Z., Guan, Z., Srivastava, G.: LightLedger: a novel blockchain-based domain certificate authentication and validation scheme. *IEEE Trans. Netw. Sci. Eng.* (2021)
  - 16. Al Harthy, K., Al Shuhaimi, F., Al Ismaily, K.K.J.: The upcoming Blockchain adoption in higher-education: requirements and process. In: 2019 4th MEC International Conference on Big Data and Smart City (ICBDSC), pp. 1–5. IEEE (2019)
  - 17. Baldi, M., Chiaraluce, F., Frontoni, E., Gottardi, G., Sciarroni, D., Spalazzi, L.: Certificate validation through public ledgers and blockchains. In: ITASEC, pp. 156–165 (2017)
  - 18. Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System (2008). Available: <https://bitcoin.org/bitcoin.pdf>
  - 19. Zhao, J., Lin, Z., Huang, X., Zhang, Y., Xiang, S.: TrustCA: achieving certificate transparency through smart contract in blockchain platforms. In: 2020 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS), pp. 1–6. IEEE (2020)
  - 20. Aldana-López, R., Gómez-Gutiérrez, D., Defoort, M., Sánchez-Torres, J.D., Muñoz-Vázquez, A.J.: A class of robust consensus algorithms with predefined-time convergence under switching topologies. *Int. J. Robust Nonlinear Control* **29**(17), 6179–6198 (2019)
  - 21. Wang, S., Yuan, Y., Wang, X., Li, J., Qin, R., Wang, F.: An overview of smart contract: architecture, applications, and future trends. In: IEEE Intelligent Vehicles Symposium (IV). Changshu 2018, pp. 108–113 (2018). <https://doi.org/10.1109/IVS.2018.8500488>
  - 22. Alharby, M., Aldweesh, A., Moorsel, A.V.: Blockchain-based smart contracts: a systematic map- ping study of academic research. In: 2018 International Conference on Cloud Computing, Big Data and Blockchain (ICCBB), Fuzhou, China, pp. 1–6 (2018). <https://doi.org/10.1109/ICCBB.2018.8756390>
  - 23. Aloqaily, M., Boukerche, A., Bouachir, O., Khalid, F., Jangsher, S.: An energy trade framework using smart contracts: overview and challenges. *IEEE Netw.* **34**(4), 119–125 (2020)
  - 24. Delmolino, K., Arnett, M., Kosba, A., Miller, A., Shi, E.: Step by step towards creating a safe smart contract: lessons and insights from a cryptocurrency lab. In: International Conference on Financial Cryptography and Data Security, vol. 9604, pp. 79–94. Springer, Berlin, Heidelberg (2016)

25. Malsa, N., Vyas, V., Gautam, J., Ghosh, A., Shaw, R.N.: CERTbchain: a step by step approach towards building a blockchain based distributed application for certificate verification system. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 800–806. IEEE (2021)
26. Malsa, N., Vyas, V., Singh, P.: Blockchain-enabled smart contract optimization for healthcare monitoring systems. In: Cloud Computing Enabled Big-Data Analytics in Wireless Ad-Hoc Networks, pp. 229–250. CRC Press
27. <https://www.sublimetext.com/docs/3/>. Accessed on 05 Sept 2020
28. <https://remix-ide.readthedocs.io/en/latest/>. Accessed on 05 Sept 2020
29. Dolgui, A., Ivanov, D., Potryasaev, S., Sokolov, B., Ivanova, M., Werner, F.: Blockchain-oriented dynamic modelling of smart contract design and execution in the supply chain. Int. J. Prod. Res. **58**(7), 2184–2199 (2020)
30. <https://www.coursera.org/lecture/smarter-contracts/processing-smart-contracts-compile-artifacts-demo-uR4xY>. Accessed on 15 Sept 2020
31. <https://www.trufflesuite.com/docs>. Accessed on 16 Sept 2020

# Comprehension of Climate Change with IoT-Enabled CNN



Priyanka Singh, Debaroti Sammanit, Rabindra Nath Shaw,  
and Ankush Ghosh

**Abstract** With the fast-pacing adoption of concepts of Industry 4.0 and upcoming Industry 5.0 as far as the adoption of low-cost deployment is concerned, wireless network-based device-free sensing (DFS) is of great interest and has successfully demonstrated the feasibility of IoT in climate change detection. In extreme climates when the received signals are severely attenuated, the accuracy of classification of this technology is known to decrease remarkably; while neural nets have enhanced performance, they only function properly when trained with large numbers of evaluated based. Therefore, in extreme climates, where adequate samples are hard to procure, reasonable sensor exactness is still unspecified. We look at the possible application of anomaly detection methods based on machine learning. This study examines the performance of a top-down, phenomenology model generated using a neural net and large global average monthly temperature data. The neural network algorithm accurately forecasts an increase and drop in temperature throughout the next ten years by generating explicit depictions using the month-to-month temperature records of 30 years. In this paper, we constructed a deep classification system for the convolutional neural network and showed how beneficial deep learning is to address the problems of climate patterns.

## 1 Introduction

Climate change will have a variety of complicated and unknown effects on integrated social-ecological systems (societies). It will have reversible and irreversible effects on ecological processes, cycles, and structures. Reduced sea ice, increased permafrost thawing, increased heat waves and heavy precipitation, and reduced water

---

P. Singh · D. Sammanit

Department of Civil Engineering, Amity School of Engineering and Technology, Amity University Uttar Pradesh, Noida, India

R. N. Shaw · A. Ghosh (✉)

University Center for Research & Development (UCRD), Chandigarh University, Mohali, Punjab, India

e-mail: [ankushghosh@gmail.com](mailto:ankushghosh@gmail.com)

supplies are all expected effects. These effects can extend beyond man-made political boundaries, disrupting resource supply networks, economics, and environmental services, and altering how we live and interact. Multiple governance options, technological choices, and implementation measures will be necessary to ensure that societies can raise their resilience to stress and disruption while adjusting to improve their sustainability. In the context of various urban-rural landscapes, ecosystems, and geographies, strategies will also need to be devised, integrated, and altered through time. In regulating, communicating, monitoring, assessing, choosing, implementing, reasoning, and discovering, societies are increasingly relying on smart Internet of Things devices, data science approaches, and intelligent learning algorithms. However, the overall capability and aptitude of societies to successfully adjust to climate change impacts, uncertainties, and hazards are mostly determined by the cumulative influence of individual appraisals, which are heavily reliant on data. Data aids evidence-based decision-making as well as scientific and technical innovation by allowing a communal understanding of the underlying causes of many effects, triggers, and trade-offs. To successfully decrease integrated risks and establish collective resilience that underlies a sustainable future, a greater knowledge of the effective use and potential advantages of data is essential.

## 2 Initiatives for the Adoption of New Technologies

As the Kyoto Convention terminates and the Paris Arrangement becomes effective in 2020, numerous nations commit to guarantee the decrease of carbon dioxide ( $\text{CO}_2$ ) discharges and an earth-wide temperature boost [1, 2]. Significant measures of carbon dioxide are created during energy creation (i.e., power, gas, and so forth) through petroleum products, for example, coal and oil [3, 4]. South Korea is the eighth biggest energy customer on the planet [5]. The South Korean government as of late formed the “Green Structure Development Help Act” to fill in as a lawful system for advancing eco-accommodating structures by 2020 fully intent on diminishing ozone-depleting substance discharges by 26.9% [6, 7]. The vital components of this demonstration incorporate the upgrade of energy effectiveness for existing and new structures and the advancement of low-energy structures. Power utilization has been consistently expanding at a yearly pace of 5.4% between 2000 and 2013 inferable from the development of offices, expanded creation of semiconductors/petrochemical items/electric force plants, moderate duties for private power utilization, and temperature impacts [2]. Energy-serious structures are structures that devour more than 2000 tons of oil a year like emergency clinics, lodgings, instructive offices, business structures, media communications offices, and apartment complexes [7]. For these substantial energy utilization structures, brilliant lattice frameworks joined with environmentally friendly power or ESS are required for the productive administration of complete energy utilization [8].

### 3 IoT for Reducing Changing Course of Climate

The Web of Things (IoT) is about more than streamlining our common luxuries with more astute wearable gadgets, carrying new efficiencies to industry and coordinations, and upgrading our homes and work areas with more responsive innovation [9–12]. IoT arrangements are now easing back the ruinous course of environmental change. Worldwide trailblazers are playing a functioning job, adjusting IoT gadgets to global norms for environment assurance. The best-case situations that the Assembled Countries Intergovernmental Board on Environmental Change (IPCC) depicts require forceful relief of worldwide fossil fuel by-products, calling for them to tumble to levels from 40 to 70% lower than their 2010 levels. Also, in any event, arriving at that objective will just lethargic a worldwide temperature alteration. At the new levels, the IPCC projects, normal temperatures could in any case move by 2 degrees Celsius by 2050. The Environment Gathering, a worldwide NGO, advocates an inescapable change to Drove lighting, particularly open spaces and for road lighting, to cut fossil fuel by-products by 1.4 million tons yearly. The US Division of Energy upholds those objectives through its Outside Lighting Gas pedal program, which offers specialized, monetary, and administrative assistance. Meanwhile, inventive producers are committed to making interest in Drove deliver considerably more prominent profits by creating city-wide computerized lighting networks that capacity as IoT-empowered advanced overhangs. When incorporated into these organizations, associated sensors can improve city productivity, helping reroute vehicle traffic, for instance, or making it simpler and quicker for drivers to leave, further decreasing discharges. Ericsson Exploration projects that what they call the shrewd network could alone cut ozone-depleting substance outflows by 3.9% by 2030. This consistently creating IoT-empowered energy supply organization, which can identify and adapt to nearby changes in energy use, gives various green advantages [13–16].

Urgent carbon investment funds can be acknowledged in the creating scene too. Minimal expense, low-power IoT gadgets could bring accurate horticulture (which limits the utilization of compost, pesticides, and water) to modernizing nations considerably more rapidly than has been conceivable up to now. That implies staying away from long stretches of inefficient over-utilization of water, fuel, and soil-added substances.

Early reception of IoT observing and rerouting for maritime freight delivery shows that it is feasible to diminish fuel utilization by up to 15%. Protection upkeep that evades five-week redesigns for on-the-spot fixes could likewise lessen the need to assemble repetitive armadas [17, 18].

### 4 Reducing Carbon Footprints

Modern IoT estimates the effect of modern cycles and human action through sensors that can screen an entire scope of components, including everything from air

and water quality, to surveying contamination levels around manufacturing plants, streams, and urban areas. The innovation can likewise distinguish the more back-handed effects of environmental change by estimating things, for example, flood and waterway levels, wind speed, land disintegration, the exercises of honey bees and bee colonies, and following creatures or vegetation in affected regions [19, 20]. “The web of things is the advanced skin of our planet,” says Alex Gluhak, head of innovation (IoT) at Computerized Sling. “By estimating the genuine condition of the world through sensors, we become mindful of existing issues and can follow them over the long run as we utilize explicit mediations to battle these issues.” Modern IoT can likewise fundamentally affect decreasing the carbon impression of cycles. It does as such by limiting the utilization of regular assets, including unrefined components, power, non-renewable energy sources, and water. Close by this, the innovation can decrease creation waste and assume a vital part in the following of material stream in the arising round economy. When joined with other advanced applications, like 5G and man-made consciousness (artificial intelligence), IoT could assist with cutting carbon by 15%, as indicated by the World Financial Discussion. “IoT can assist make with detecting of the crude information delivered each moment in large numbers of associated gadgets that make up business tasks, inventory network, and associated items,” says Andy Stanford-Clark, boss innovation official at IBM, UK, and Ireland. “IoT innovation, particularly when combined with artificial intelligence, can further develop asset effectiveness, diminish contamination, and invigorate new reasoning and advancement.”

## 5 Where is IoT Tech Handling Natural Issues?

In cultivating, accuracy horticulture is utilized to limit the utilization of water, manure, and pesticides. The innovation screens soil minerals, temperature, and dampness. This improves and increments yields and limits the utilization of the two assets and land [21, 22]. IoT sensors in the dirt and climate, close by the utilization of calculations, or “develop plans,” can work on the administration of homestead assets. Modern IoT could likewise assist with lessening the hurtful effect of ozone-depleting substances delivered by animals. “There’s potential for decreased methane from ruminants through domesticated animals wellbeing checking, for instance observing dietary wellbeing and temperature of creatures to recognize and treat creatures, which assists with lessening nursery discharges,” clarifies Dr. Nilufer Tuptuk of the Division of Software engineering, College School London. IoT is the advanced skin of our planet. In the assembling business, sensors are utilized to arrange items autonomically, accordingly enhancing creation. Modern IoT is additionally being utilized to screen the energy utilization of assembling gear, empowering administrators to recognize wasteful hardware. Further exact observing outside components can decrease blunders, bringing about not so much waste but rather more effective utilization of materials. “Coordinations organizations can abbreviate conveyance courses through wise course arranging, following can abbreviate conveyance times

through new experiences into the store network, items can be found, fewer items are lost, which eventually positively affects the environment by decreasing all-out immediate and roundabout energy utilization,” says Pascal Vögeli of the ZHAW Zurich College of Applied Sciences, Switzerland. In the energy and utility market, savvy road lighting is being utilized to diminish utilization, and spillage sensors in water and gas pipes are utilized to recognize and fix misfortunes brought about by spills. Expansion of the lifetime of merchandise can likewise be ascribed to the presentation of IoT arrangements [23, 24]. Prescient upkeep of merchandise, including vehicles, electrical products, and development hardware, can empower longer use cycles. This implies fewer breakdowns and substitutions and at last a decrease in waste.

Improve garbage sorting and collection by collecting and recycling. Waste management systems in many places throughout the world are antiquated and resource-intensive. Installing smart bins with sensors that detect and indicate when the bin is approaching full capacity can make waste management more effective, allowing pickup when needed rather than on a set timetable. Furthermore, based on traffic and weather data, drivers’ routes can be improved to conserve gasoline. Well-functioning recycling and waste collection system can also inspire individuals to put more effort into sorting their trash, which is critical for a circular economy that tries to repurpose, remanufacture, and reuse materials. The use of sensors to automate the separation of different forms of garbage, such as plastics, paper, glass, and metal, is one area with a lot of potentials [25–27]. This is a task that most people dislike and in which some individuals make blunders.

Last but not least, companies must understand that while IoT can be a solution, it can also be a source of problems. As billions of IoT gadgets (and the linked items that they are a part of) reach the end of their useful lives, they may become part of the growing e-waste problem [19].

## 6 Methodical Approach to Technologies

### 6.1 Cloud Computing Model

This was the main proposed design for IoT frameworks and depends on the reason that handling of information from the different parts in the IoT framework should happen in the cloud. Distributed computing considers the remote getting to of continuously shared assets (registering, stockpiling, and administrations) over the organization. It ought to have the option to progressively dispense these assets without human intercession, timetable, or pool as fundamental and have the option to be gotten to from a wide range of stages [15]. The cloud can give both types of equipment just as programming administrations for savvy city applications. It enjoys the benefit that it gives a focal administration stage from which to notice, control the IoT framework just as to disperse order activities dependent on the got information. Additionally, this centralization likewise takes into account cloud frameworks to have adequately huge

registering and capacity limits subsequently permitting them to perform complex undertakings of information mining, design extraction, and making derivations from sensor information in savvy urban areas to utilize it in the most ideal way. Nonetheless, there are a couple of burdens with utilizing the distributed computing model for the IoT. To start with, communicating all accumulated information to the cloud expands network traffic, despite the fact that this may not be valid for applications in which estimations are not exceptionally regular, yet in different cases, this could build network costs [28]. Also, information transmission overheads might increment because of the enormous measure of information that should be sent by the numerous sensors existing in the shrewd city situation. One more detriment that the distributed computing model experiences is information inertness since the detecting units exist at the detecting layer and the dynamic/information handling happens in the cloud, this leads to information idleness in the transmission of detecting data, particularly when numerous gadgets begin sending information simultaneously. Organization dependability can be an issue when utilizing this model, with the enormous volume of information traffic on the organization, it probably will not be imaginable to authorize vigorous information transmission plots as IoT frameworks get greater.

## 6.2 Fog Computing Model

Since the majority of the data delivered in IoT happens toward the detecting end of the IoT framework, additionally called the edge, Mist processing was proposed in [29] to beat a portion of the issues of the distributed computing model for the IoT. Haze figuring gives more different dissemination of obligations that are directed by the distributed computing engineering by moving a portion of the handling to gadgets on the neighborhood organization. Regularly, Mist registering alludes to information handling that is done by switches and other organization gadgets inside the organization layer in the IoT. Since network gadgets these days progressively offer better computational capacities, one can use them to perform the simple procedure on the information. Tasks like conglomeration and assortment of sensor information, straightforward handling activities, and dynamics can be performed to diminish the measure of data stream toward the higher cloud layer. Questions that should be responded to for the dynamic interaction incorporate yet are not restricted to for, e.g., does the choice require the utilization of averaging for one amount and momentary upsides of the other? Is it conceivable to extrapolate information got for one amount and utilize the presently estimated incentive for another? Because of the past information for a given period, would one be able to give higher layers choice choices as opposed to simply information, consequently giving better quality data to the cloud layer subsequently bringing about better usage of cloud assets? Mist layers can confine dynamics since they approach the neighborhood condition of a given district [12]. This would be useful in executing conveyed dynamic instruments which may be important in certain applications. Besides, they likewise consider nearby organizations to be set up utilizing non-web advances, for example, ZigBee,

Bluetooth, RFID, and so on, where sensors and opposite end gadgets send information to the haze hub (additionally alluded to as passageways in such frameworks) which is associated with the cloud.

Information sent vertical by the Mist layer in the IoT chain of command would be utilized to acquire bits of knowledge into framework conduct, and to direct new standards of framework activity, this will commonly be completed in the cloud. Gadgets in the haze layer might be given dynamic direction from the higher cloud layer to guarantee smooth framework activity. Nonetheless, equilibrium should be struck concerning the division of obligations between the cloud and the haze layer keeping in view the expenses in question.

### ***6.3 Edge Computing Model***

The motivation behind Mist registering was to push a portion of the dynamic toward the edge of the organization. Lately, with progressively fit gadgets being fostered that are connected to “edge” hubs, straightforward dynamic, and handling of information have been progressively continued to these gadgets to decrease organization and gadget costs considerably further at the haze level and settle on for much more profound conveyed dynamic plans. Edge figuring alludes to information handling that is done at the “thing” level, i.e., by sensors and different gadgets in the IoT framework. One more idea about edge figuring as examined in characterizes the edge processing layer as a delegate layer between the mist and the “things” (sensors) instead of edge hubs [30, 31]. The distinction between them for this situation is the edge registering hubs go about as accumulation and dynamic units on a more limited size contrasted with Mist gadgets which act to give consistent availability and information honesty all through the IoT organization. The point of the Mist and the edge figuring standards is to decentralize the IoT framework for reasons for lessening cost, expanding versatility, and expanding heartiness.

## **7 Sensing Technologies**

Smart city technologies rely heavily on sensing. Sensors give the knowledge and data needed to develop smart city solutions. Due to the diverse nature of smart city projects and their varied components, a variety of sensors are used as part of these initiatives. The researchers propose a framework for comparing IoT sensors and a list of sensors they discovered in use for the Internet of Things. We have based our research on their findings to survey smart city sensing technologies. Ambient, motion, electric, biosensors, identification, presence, hydraulic, and chemical sensors are among the sensors used in the Internet of Things. Sensors are a critical component of smart city IoT systems since they enable interaction between the smart city system and the city’s residents as well as the development of new services. One thing to keep

in mind is that many of the sensors have several applications, as discussed [32, 33]. Furthermore, each given application will necessitate the measurement of a variety of physical qualities and the usage of a variety of sensors. Ambient, motion, electric, identification, location, chemical, and hydraulic sensors, for example, have all been discovered to be used in smart homes. Working with several sensors, each of which may have distinct output data kinds is a task that must be accomplished, as stated in the difficulties.

## 8 Conclusion

Given the abundance of IoT innovation today, the world could rapidly move to an associated and responsive organization accessible to all. Analysts are proposing a normalized ecological sensor network driven by IoT innovation—one stage toward that goal. Access to a checked and steady wellspring of carbon and environment information could assist with propelling agreement and make strategy simpler to plan and uphold. Overseeing environmental change with innovation is only the initial step. Public/private associations should develop, and policymakers' obligation to put resources into arrangements needs to increase. The Web of Things, as far as it matters for itself, will accomplish more than keep track of who is winning. It will give noteworthy environmental information. It will cut waste by working on the progression of individuals, energy, merchandise, and data. Furthermore, it will keep on adjusting, as specialists and pioneers arrive at a new agreement on the move we wanted to make to secure ourselves against rising temperatures and perilous environmental conditions. IoT can assume a key part in aiding battle environmental change by assisting us with decreasing our ozone-depleting substance discharges and living more economically. Thought pioneers across many organizations and enterprises are accepting IoT answers for making imaginative items and administrations that are both serious according to a business viewpoint and simultaneously more earth cognizant. A key to progress will be to use overseen network arrangements and work with an environment of accomplices with demonstrated capacities in their space to make IoT arrangements that can be conveyed around the world and scale rapidly. As worldwide temperatures are rising, an opportunity to act is present in case we are to forestall irreversible harm to our planet.

## References

1. Worldometers. World Population Forecast—Worldometers. 2019. Accessed on 9 Mar (2021)
2. Ahvenniemi, H., Huovila, A., Pinto-Seppä, I., Airaksinen, M.: What are the differences between sustainable and smart cities? *Cities* **60**, 234–245 (2017)
3. United Nations. About the Sustainable Development Goals—United Nations Sustainable Development. Accessed on 9 Mar (2021)

4. Cardullo, P., Kitchin, R.: Being a ‘citizen’ in the smart city: Up and down the scaffold of smart citizen participation in Dublin Ireland. *GeoJournal* **84**, 1–13 (2019)
5. Desdemouster, J., Crutzen, N., Giffinger, R.: Municipalities’ understanding of the Smart City concept: an exploratory analysis in Belgium. *Technol. Forecast. Soc. Chang.* **142**, 129–141 (2019)
6. Khan, M.S., Woo, M., Nam, K., Chathoth, P.K.: Smart city and smart tourism: a case of Dubai. *Sustainability* **9**, 2279 (2017)
7. Wu, S.M., Chen, T.C., Wu, Y.J., Lytras, M.: Smart cities in Taiwan: a perspective on big data applications. *Sustainability* **10**, 106 (2018)
8. Ejaz, W., Anpalagan, A.: Internet of things for smart cities: overview and key challenges. *Internet Things Smart Cities* 1–15 (2019)
9. Janssen, M., Luthra, S., Mangla, S., Rana, N.P., Dwivedi, Y.K.: Challenges for adopting and implementing IoT in smart cities: an integrated MICMAC-ISM approach. *Internet Res.* **29**, 1589–1616 (2019)
10. Sánchez-Corcuera, R., Nuñez-Marcos, A., Sesma-Solance, J., Bilbao-Jayo, A., Mulero, R.; Zulaika, U., Azkune, G., Almeida, A.: Smart cities survey: technologies, application domains and challenges for the cities of the future. *Int. J. Distrib. Sens. Netw.* **15** (2019)
11. Silva, B.N., Khan, M., Han, K.: Towards sustainable smart cities: a review of trends, architectures, components, and open challenges in smart cities. *Sustain. Cities Soc.* **38**, 697–713 (2018)
12. Atat, R., Liu, L., Wu, J., Li, G., Ye, C., Yang, Y.: Big data meet cyber-physical systems: a panoramic survey. *IEEE Access* **6**, 73603–73636 (2018)
13. Mridha, K., et al.: Web based brain tumor detection using neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), 2021, pp. 137–143 (2021). <https://doi.org/10.1109/ICCCA52192.2021.9666248>
14. Mridha, K., et al.: Deep learning algorithms are used to automatically detection invasive ducal carcinoma in whole slide images. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 123–129 (2021). <https://doi.org/10.1109/ICCA52192.2021.9666302>
15. Khan, Z., Anjum, A., Soomro, K., Tahir, M.A.: Towards cloud based big data analytics for smart future cities. *J. Cloud Comput.* **4** (2015)
16. Koubaa, A., Aldawood, A., Saeed, B., Hadid, A., Ahmed, M., Saad, A., Alkhouja, H., Ammar, A., Alkanhal, M.: Smart palm: an IoT framework for red palm weevil early detection. *Agronomy* **10**, 987 (2020)
17. O’Grady, M., Langton, D., O’Hare, G.: Edge computing: A tractable model for smart agriculture? *Artif. Intell. Agric.* **3**, 42–51 (2019)
18. Rojek, I., Studzinski, J.: Detection and localization of water leaks in water nets supported by an ICT system with artificial intelligence methods as away forward for smart cities. *Sustainability* **11**, 518 (2019)
19. Pardini, K., Rodrigues, J.J., Kozlov, S.A., Kumar, N., Furtado, V.: IoT-based solid waste management solutions: a survey. *J. Sens. Actuator Netw.* **8**, 5 (2019)
20. Mukhopadhyay, M., et al.: Facial emotion recognition based on Textural pattern and Convolutional Neural Network. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573860>
21. Ghosh, M., et al.: Robust face recognition by fusing fuzzy type 2 induced multiple facial fused image. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573871>
22. Shirazi, E., Jadid, S.: Autonomous self-healing in smart distribution grids using multi agent systems. *IEEE Trans. Ind. Informatics* **3203**, 1–11 (2018)
23. Andreão, R.V., Athayde, M., Boudy, J., Aguilar, P., de Araujo, I., Andrade, R.: Raspcare: a telemedicine platform for the treatment and monitoring of patients with chronic diseases. In: Assistive Technologies in Smart Cities. IntechOpen, London, UK, (2018)

24. Keane, P.A., Topol, E.J.: With an eye to AI and autonomous diagnosis. *NPJ Digit. Med.* **1**, 10–12. [[PubMed](#)]
25. Trencher, G., Karvonen, A.: Stretching, “smart”: Advancing health and well-being through the smart city agenda. *Local Environ.* **24**, 610–627 (2019)
26. Bedi, P., et al.: Impact Analysis of Industry 4.0 on real time smart production planning and supply chain management. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573563>
27. Tao, F., Cheng, J., Qi, Q.: IIHub: An industrial internet-of-things hub toward smart manufacturing based on cyber-physical system. *IEEE Trans. Ind. Inform.* **14**, 2271–2280 (2018)
28. Trakadas, P., Simoens, P., Gkonis, P., Sarakis, L., Angelopoulos, A., Ramallo-González, A.P., Skarmeta, A., Trochoutsos, C., Calvo, D., Pariente, T., et al.: An artificial intelligence-based collaboration approach in industrial IoT manufacturing: key concepts, architectural extensions and potential applications. *Sensors* **20**, 5480 (2020)
29. Wan, J., Yang, J., Wang, Z., Hua, Q.: Artificial intelligence for cloud-assisted smart factory. *IEEE Access* **6**, 55419–55430 (2018)
30. Chakraborty, R., et al.: Study and prediction analysis of the employee turnover using machine learning approaches. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573759>
31. Farag, S.G.: Application of smart structural system for smart sustainable cities. In: Proceedings of the 2019 4th MEC International Conference on Big Data and Smart City (ICBDSC), Muscat, Oman, 15–16 Jan (2019), pp. 1–5
32. Wang, Y., Ram, S., Currim, F., Dantas, E., Sabóia, L.A.: A big data approach for smart transportation management on bus network. In: Proceedings of the IEEE 2nd International Smart Cities Conference: Improving the Citizens Quality of Life, ISC2 2016—Proceedings, Trento, Italy, 12–15 Sept (2016), pp. 1–6
33. Lele, A.: Internet of things (IoT). *Smart Innov. Syst. Technol.* **132**, 187–195 (2019)

# Employing Adaptive Learning and Intelligent Tutoring Robots for Virtual Classrooms and Smart Campuses: Reforming Education in the Age of Artificial Intelligence



Ashraf Alam

**Abstract** As a consequence of the advent of new technologies, teaching and learning methods have evolved dramatically. Artificial intelligence (AI) applications in educational settings are becoming increasingly apparent as a result of rapid development of AI technology in recent years. Adaptive learning, smart campus, teacher evaluation, intelligent tutoring robots, and virtual classrooms are only a few of the applications of educational-AI that is explored in this article. After evaluating the impact of AI technology on teaching and learning, it is conclusively inferred that AI has a beneficial effect on both the quality of instruction provided by teachers and on the learning outcomes of students. Toward the end, the article discusses the possible challenges that AI applications in education may face, instances of AI's potential in helping schools to improve, and thereby promoting educational reforms.

**Keywords** Artificial intelligence · Education · Pedagogy · Adaptive learning · Smart campus · Teacher evaluation · Intelligent tutoring robots · Virtual classrooms

## 1 Introduction

Across the globe, advancements in research and development in artificial intelligence (AI) technology are occurring at an excessively fast pace. AI is a fast developing area of technology that is already being used in a wide range of disciplines and applications [1]. Education has changed considerably with the advent of AI technology, which has been ingrained in today's modern societal framework. Increasing number of people, particularly those from education sector, recognize the importance of advancements in AI technology. Numerous studies have been conducted to demonstrate the advantages of incorporating AI into classroom. It has had a significant impact on the way courses are delivered and students are taught [2]. Students' enthusiasm, initiative, and originality increase as a result of artificial intelligence's ability

---

A. Alam (✉)

Rekhi Centre of Excellence for the Science of Happiness, Indian Institute of Technology Kharagpur, Kharagpur, India  
e-mail: [ashraf\\_alam@kgpian.iitkgp.ac.in](mailto:ashraf_alam@kgpian.iitkgp.ac.in)

to continuously enhance and optimize their learning environment [3]. By combining these high-end technologies into education, teachers' monitoring abilities and fair and efficient classroom management could be significantly enhanced [4]. Research on application of artificial intelligence in education is developing at a pace comparable to that in other areas. AI has been more widely utilized in education as a result of relevant research results demonstrating beneficial application impacts and assistance in the improvement of educational practices. With the advent of artificial intelligence (AI) in education, it is now possible to seamlessly integrate and change teaching and learning. The article discusses the use of artificial intelligence (AI) in the classroom and how it works.

## 2 Review of AI's Application in Education

The world's attention has turned to artificial intelligence research after AlphaGo's 4:1 victory against world Go champion Lee Sedol [5] in the Go game. An increasing number of academics in artificial intelligence and education aim to help professors by creating smart campus settings using intelligent teaching and administration techniques. In education, images, facial recognition, adaptive learning, and other AI technologies are being utilized to increase teacher efficiency while also improving student learning experiences [6]. Additionally, artificial intelligence and big data may be used to fully evaluate and analyze educational data with the goal of reforming teaching for pedagogical quality enhancement [7]. The following subsections discuss the impact of AI on adaptive learning, teacher evaluation, intelligent tutoring robots, smart campuses, virtual classrooms, among other things.

### 2.1 *Adaptive Learning*

Students perform better and retain more knowledge when they use learning analytics and real-time analysis powered by artificial intelligence (AI). Adaptive learning systems are intended to aid students in their learning by combining assessment, instruction, learning, and practice [8]. Because of adaptive learning systems, teachers may collect data on students' learning patterns, suggest the optimal learning route for them based on their abilities, and then supply them with learning material in the form of online instructional videos to complete the loop. Teachers may utilize the human–computer interaction technology associated with artificial intelligence to assist them in responding to students' online inquiries; this helps especially when problems cannot be handled in physical classrooms [9]. Teachers improve teaching effectiveness by using adaptive learning technology from companies such as DreamBox Learning, BYJU's, and IBM Watson Education [10]. Artificially intelligent assessment and learning systems based on knowledge spaces, such as ALEKS, are extensively used in the USA. Yilmaz conducted a study to see how ALEKS

impacted the math abilities of middle school students [11]. According to the study findings, students' end-of-year mathematics performance increased statistically after getting ALEKS instruction. Fang, Ren, Hu, and Graesser in 2019 performed a meta-analysis to determine the efficacy of ALEKS in the classroom [12]. According to study conducted at the University of California, Berkeley, when utilized for shorter rather than longer periods of time, ALEKS was similar to conventional classroom teaching, and not better.

Teachers that use the ALEKS learning system to assist students in learning may find this research beneficial since it provides additional information on ALEKS. As with ALEKS, BYJU'S is an Indian educational program that utilizes artificial intelligence to assist students in their studies. BYJU'S integrates animated scenarios into its online lectures and activities to aid students in their comprehension of difficult scientific topics. Gamification enhances the learning experience by including it into the operating interface. Finally, BYJU'S system may suggest courses to students based on their abilities and interests [13]. According to Sruthi and Mukherjee in 2020, BYJU'S app has revolutionized Indian educational environments by integrating effective teaching and learning methods [14]. The vast majority of those who responded considered the program to be educational, amusing, and helpful. BYJU'S may also aid students in developing strong conceptual knowledge. Academics in India have examined how BYJU'S aid students in developing strong conceptual understanding [15]. However, some students may be reluctant to use it owing to the cost. The critical nature of English education for academic success has long been recognized. There are simply too many students in English courses for teachers to give personalized spoken teaching to each one. English courses often include cutting-edge technologies such as voice recognition and semantic analysis [16]. This method is utilized to assist students with their poor English pronunciation. When used effectively, it may assist teachers and students in engaging in one-on-one oral practice. Duolingo is a multilingual learning platform that emphasizes personalized instruction via the use of a game-based structure.

According to Tsai, studying using Duolingo increased learner autonomy in terms of organizing learning time, self-regulating on a consistent basis, finding new resources for learning, evaluating one's learning, and modifying learning techniques to achieve better learning results [17]. Zou created a platform for teaching college-level English that used artificial intelligence to assess its students' requirements and comprehension [18]. On this platform, artificial intelligence algorithms may be utilized to change both the speed and content of instruction. Students who used the platform to study English performed better on standardized exams than those who do not, suggesting that the platform may be able to assist children develop their autonomy while simultaneously increasing their English abilities. Additionally, Bin and Mandal in 2019 created a method for teaching collegiate English using artificial intelligence [19]. This approach will be used in English classes to increase the efficiency and quality of teaching.

## 2.2 *Teaching Evaluation*

Grading is simplified in classroom making use of AI technologies such as image recognition, prediction algorithms, and computer vision. The educational process is largely reliant on pupils' test-taking performance. Teachers in traditional education scenarios spend a significant amount of time developing questions, grading, and assessing students' performance, as well as analyzing test papers. As a result of AI, educational evaluation methods are becoming more scientific. Additionally, AI enhances the accuracy of educational evaluation results. Artificial intelligence may be used to create exam questions and to correct assignments and tests automatically [20]. Teachers often grade their students' homework and test papers manually. After reviewing assignments and exams for a long period of time, teachers get tired and over time get disinterested with their work. As a consequence of the long review procedure for test papers, errors are inevitably made. Teachers may profit from image recognition technology if it relieves them of the responsibility of grading and correcting assignments [21].

Furthermore, the probability of committing mistakes becomes very low. Artificial intelligence technology may help teachers save time and resources by correcting exam papers and detecting blank tests as well as suspected duplicate tests. IFLYTEK and New Oriental collaborated on the development of RealSkill, a tool for improving oral and writing abilities in IELTS and TOEFL exams using intelligent correction. Students may use RealSkill online platform to better prepare for the IELTS and TOEFL examinations. This platform performs intelligent grading, sentence-by-sentence correction of essay-type problems, behavioral analysis exercises, and learning diaries [22]. E-rater [23], an automated essay evaluation software, was created and published by the American Educational Testing Service (ETS). Along with enhancing students' writing skills, E-rater is capable of accurately and swiftly evaluating students' writings (including structure, grammar, and calculation of total score). To aid students in doing well on spoken English examinations, ETS has integrated AI technology into the assessment process and developed a Speech Rater engine [4].

While volumetric evaluation of instruction continues to have a role, AI provides a novel perspective and technique for assessing students' physical education courses. Ge, Yin, and Feng in 2018 created a browser/server architecture, a Visual Studio integrated open environment, and a SQL Server database operating on Windows platform to construct an autonomous learning system based on AI for college sports [15]. This platform may assist teachers in saving time in the classroom while also encouraging college students to increase their physical activity levels. Sports intelligence may be measured using Yong's 2018 approach for assessing sports teaching [6]. This technique of assessing sports instruction is based on the process of expert decision-making. These findings demonstrate that an AI-based system for evaluating physical education can provide new developmental methods for implementing and developing modern sports technology, as well as theoretical support and recommend for future development of scientific educational technology in classrooms.

## 2.3 Virtual Classroom

Cutting-edge virtual reality (VR) and augmented reality (AR) technologies have the potential to revolutionize the educational environment. By fusing the physical and virtual worlds, ubiquitous computing technology enables the creation of virtual classrooms and laboratories for students [17]. Teachers may use VR technology into the classroom to mimic difficult-to-explain instructional situations. Students in a smart classroom may benefit from a rich learning environment by observing natural occurrences and changes in objects that are not readily visible or detectable in the real world. When students are exposed with information in many dimensions, their visual, aural, kinesthetic, and other senses are stimulated, creating a strong sense of realism. To assist learning, pupils are provided information in a variety of methods. Abstract ideas and theories become intuitive and self-evident as a result of this process. Additionally, it assists instructors in their effectiveness by piquing students' interest in the subject being taught. Students may select whether or not to attend lectures on campus or to go with a hybrid virtual classroom, which holds significant potential for course attendance flexibility [8].

The term “virtual simulation laboratory” refers to the practice of recreating authentic experimental settings digitally using 3D modeling, multimedia, simulation, and virtual reality technologies. This technique results in the creation of computer operating environments similar to those utilized in conventional research. By using various operating settings, it is possible to fully replace, partly replace, or even partially replace traditional test operational connections [19]. For the experimenter, this may include the creation of a fully reproduced virtual experimental environment and experimental operational items, as well as sophisticated multimodal interactive equipment capable of merging virtual and real-world interaction [3]. In virtual classrooms, with the advent of AR and VR technologies, there's now no need for any apprehension concerning harm instigated by errors in experimental operations or fear on missing out on classic experiments because of experimental conditions. The results of experiments thus obtained in virtual environments are comparable to or sometimes even authentic and reliable compared to those attained in real setting. Through platform simulation, virtual experiments are carried out, emphasizing the relationship between experimental operations and outcome of simulation results [11]. For example, educators and pupils can benefit from virtual laboratories.

Experiments in chemistry [2], physics [13], biology [4], and other academic disciplines [15] can be executed and followed through in virtual laboratories. Students' practical abilities in a virtual environment gain from the interactive elements of the virtual simulation laboratory. Apart from economizing on resources, virtual experimental equipment also mitigates experimental risks [6]. However, according to Makransky, Terkildsen, and Mayer in 2019, studying science in virtual reality ecosystem may cause cognitive overload and distraction in pupils, resulting in poor learning outcomes [13].

## 2.4 Smart Campus

Due to campus's critical role in talent development, AI has emerged as a new developmental trend in higher education [8]. Artificial intelligence is essential for campus management and service delivery. To create a smarter campus, face recognition, hearing, and sensing technologies are all being utilized [9]. To develop intelligent management systems, researchers must gather and analyze massive amounts of data [4]. Due to the collaboration of educational management and artificial intelligence, it was feasible to develop a collaborative decision-making model that enabled early detection of educational system issues, more efficient resource allocation, and effective campus safety [1]. Suspicious individuals are effectively stopped from entering areas such as dorms, labs, and libraries that should be off-limits to non-school personnel making use of face recognition technology [2]. Face recognition may assist in keeping the school premises secure during card swapping by unauthorized person who make use of other person's credentials [14].

According to Upala and Wong, there is evidence that face recognition technology can also be used to borrow and return library books, completing identification verification through facial information while enabling autonomous borrowing and returning of books via the book lending and returning machine [14]. This would enhance library efficiency while decreasing labor costs. An and Xi in 2020 and Liu, Zhou, Zou, Yeh, and Zheng in 2018 in their research relied on image recognition to automatically detect and compute the prices of meals and commodities at canteens and campus supermarkets, and subsequently on face recognition for checkout and self-service purchasing [15]. Ramdani, Virgono, and Setianingsih in 2020 and Liu et al. In 2018, have published new researches on these subjects [17]. It is a good idea to install infrared fences around schools to keep an eye out for intruders or children who have fled through the barrier. By immediately notifying the relevant personnel of triggered alarm information and camera images when an infrared sensor is touched, we may remind them to perform on-site inspections or call authorities [14].

## 2.5 Intelligent Tutoring Robots

Numerous scientific and academic disciplines are availing facilities of robots in teaching and research. Among these fields are education, computer science and automated control, as well as materials and physical sciences. Industrial robots were primarily focused in early robotics research and development and were important in the field's progress [8]. Technology's educational potential is increasing, and as a consequence, robots are gaining popularity. In the 1960s, Prof. Papert at MIT's artificial intelligence department created the world's first teaching robot [4]. Educational robots are those that have been purpose-built to assist students in developing their analytical, creative, and practical skills in educational environments. Its features are

interactivity, openness, and scalability [5]. It makes use of artificial intelligence technologies such as voice recognition and emotion recognition (which analyzes facial expressions and tone) as well as bionic technology, which can mimic human actions while also posing a threat to humans via human-like abilities such as listening to what is said [11].

Intelligent tutoring robots are capable of a wide range of tasks, including instructing on a number of topics. Aid with teaching, assistance with learning, robotic administration of learning, etc., may all be classified into five main categories [15]. After completing a literature review on educational robotics applications based on tertiary education learning theory, Spolaor and Benitti in 2017 published their findings [5]. They examined studies indicating robots that may assist in the learning of new skills and information. The bulk of the studies examined in their research were based on inclusion of robots into computer science education. Rather than addressing the conflicts between student interests and that of the tutor's, tutoring robots begin by resolving conflicts between students' needs and that of the tutor's [14]. The employment of robots in classroom may be beneficial in sparking a student's enthusiasm for learning.

Pepper robots, such as the SoftBank-developed model [15], may be used educationally, for example, to increase students' interest in AI via play, at all stages of education, from elementary school to middle school to university. To further the advancement of artificial intelligence, The University of Nottingham Ningbo in China has introduced the Soft-Bank Pepper humanoid robot as the artificial intelligence ambassador, offering multilingual conversation, courses, and information retrieval services in their library. Pepper is a reality today, credit to which goes to SoftBank's artificial intelligence technologies [16]. The Adaptive Systems Research Group at the University of Hertfordshire has created robots that are extensively utilized to assist children with autism spectrum disorders [7]. The results indicate that autistic children are very active and respond well to encounters with instructional robots. Recent advances in artificial intelligence technology are enabling the development of increasingly complex robots that act like humans. Intelligent humanized tutoring robots will thus immensely benefit education and teaching in future.

### **3 Reforming Education in an Age of Artificial Intelligence**

As technology advances, AI is being more extensively used in a variety of areas, including education. Cognitive agents can execute a variety of cognitive activities, including speaking (human–computer conversation), hearing (machine translation, voice recognition), seeing (computer vision, image recognition, text recognition), and learning (machine learning, intelligent adaptive learning) [18]. The application of artificial intelligence technologies such as computer vision, natural language processing, and intelligent adaptive learning has altered traditional education and teaching. This has inspired colleges and teachers to rethink their curricula. In education, artificial intelligence offers the advantage of enabling the development of more

personalized ways of teaching and learning. Teachers and students now learn in a variety of methods as a consequence of AI. Dissonance may develop a customized learning plan for pupils based on their needs and current learning environment. This results in a more engaging learning experience for pupils while also improving their capacity and efficiency for learning.

Using big data and machine learning, in-depth evaluations of students' daily academic work, test performance, as well as personalized teaching recommendations for students with difficult-to-learn concepts may be performed [19]. As a result of this approach, student learning time may be reduced while learning efficiency is improved [6]. Adaptive learning technology has the potential to be helpful in the delivery of personalized education between computers and students. Artificial intelligence-based adaptive learning technology emulates teachers by training students one-on-one and endowing the learning system with a set of individualized teaching skills that students may apply in the real world [21]. Adaptive education gained popularity in the USA throughout late 1980s and early 1990s. Institutions of higher learning may purchase teaching resources from world-renowned adaptive education companies such as Knewton and Cognitive Tutor [2].

China adopted a technology-based approach to adaptive education considerably later than the USA and is much less progressed. The sector has expanded at a rapid rate during the last 15 years. When it comes to intelligent adaptive education, firms like the New Oriental and Yixue Education-Squirrel AI [12] are concentrating their efforts on K12 tutoring and English tutoring. Instructors will have more time to devote to providing humanistic care rather than administrative duties as a result of artificial intelligence. Nowadays, teachers spend a great deal of time looking through students' homework and test results. Repeating the same task consumes teachers' precious teaching and research time, not to mention time spent with students. Intuitive tutoring and assessment systems [9], educational robots [14], and other AIs can assist teachers with mechanically repetitive daily tasks such as correcting homework and test papers, relieving teachers of various responsibilities, and alleviating educators of the burden associated with extensive knowledge transfer. Additionally, artificial intelligence technology has the ability to improve instructors' abilities, allowing them to provide pupils with individualized and accurate instructional advice that they could not previously provide [3]. Additionally, AI enables teachers to spend more time and energy interacting with students, allowing them to devote more time and energy to each student's aptitude and moral development, as well as in overall physical and intellectual development. Rather than just imparting knowledge, teachers today act as mentors for their pupils, assisting them in learning and developing as a street-smart holistic human being. Additionally, AI promotes the creation of "smart" campuses as a third educational function [18]. Since the introduction of AI technology, education and teaching have seen significant changes.

Apart from facial and text recognition, other forms of artificial intelligence like as AR and voice interaction (VOI) also contribute to the development of smart campuses [22]. According to [19], artificial intelligence has the ability to improve student and teacher learning while simultaneously lowering management costs by

enhancing campus administration and services such as campus security, impact monitoring in classes, and school attendance. Due to the large number of students that live on campus, administration is unable to trace down all incursions. When facial recognition and body identification technologies are utilized in dorm administration, efficiency increases [4]. Student housing may benefit from increased security as a result of the usage of facial recognition technology powered by artificial intelligence. Teachers may monitor and analyze how often their students use their phones or glance up from their work by using artificial intelligence technology such as face recognition, human body analysis, and image identification in the classroom. This information may be used to assist kids in doing better in school and to guide educational decisions [10]. As a result, instructors and parents will have a greater understanding of their children's classroom performance, and teachers who want to enhance their teaching techniques may do so in real time. Educators will be able to rapidly identify absent students and give additional attention to them as a result of face recognition technology by monitoring attendance and providing attendance data based on facial recognition [6]. As a fourth advantage, artificial intelligence advances educational equality [21].

Numerous countries and regions face educational inequality as a result of economic development or geographic limitations. Students living in rural areas with little educational resources, as well as students living in big cities who are physically distant from their instructors, may benefit from AI technology's ability to transcend regional divides and decrease educational resource inequality. Through the use of Internet and artificial intelligence, it may be possible to provide individualized teaching to distant learners [4]. Numerous topics may be addressed remotely online by a single educator in areas with a scarcity of teachers. This enables students to interact with and learn from outstanding academics while also encouraging resource sharing among those educators. Teachers from all around the globe may now use artificial intelligence to check homework given by pupils from a variety of geographical regions [12]. This technology enables personalized teaching, intelligent adaptive learning, and capacity development for pupils. It may be thus be highly beneficial to educators worldwide. Combining online and offline teaching methods has the potential to improve the quality of instruction in regions with limited teaching resources and to provide professional development for instructors in remote areas.

## 4 Challenges of AI in Education

While incorporating artificial intelligence into classroom has the potential to improve education, it may also introduce unforeseen complications. Artificial intelligence in education will be more effectively utilized in the future by people who are aware of the possible drawbacks associated with its usage. Typically, these problems manifest themselves in the areas listed below. To begin, it is essential to safeguard the rule of law when artificial intelligence is integrated into the classroom. Developing countries face the danger of new technology widening the divide between rich and poor

as artificial intelligence advances. Similar to the digital divide, the ever-widening algorithmic divide threatens to deny many pupils access to educational opportunities enabled by artificial intelligence. Because the bulk of artificial intelligence algorithms developed in rich nations cannot account for the conditions that exist in impoverished countries, they cannot be utilized directly in such countries [19].

To create the conditions for artificial intelligence to improve learning, the education sector must overcome significant obstacles, such as lack of basic technologies and infrastructure. Second, before collecting, using, or disseminating data, we must examine the ethical and safety consequences. Personal student counselling/advice is one of the ethical issues highlighted by AI. Additional ethical issues include data gathering, data privacy, ownership of responsibility, and feed algorithms. Increasing supervision of AI technology and its products necessitates public discussions about the technology's ethics, accountability, and safety consequences. Additionally, it assists teachers in preparing for teaching aided by AI. Artificial intelligence use in classroom necessitates instructors to improve their digital teaching skills. To develop an easy-to-use approach for deploying artificial intelligence teaching aids, educational product makers must first understand how instructors operate. The ability of students to study independently is being emphasized more as a consequence of advancements in instructional techniques, which is a good development.

Education will be student-centered in the era of artificial intelligence, with students taking the lead on instructional activities and projects [17]. Students may use the intelligent teaching system to build personalized learning plans, choose their own learning materials, monitor their progress, and engage in cooperative learning activities in groups [14]. According to [12], instructors should emphasize the development of students' autonomous learning capacities throughout the teaching process, since different learning methods place a greater emphasis on students' self-regulation and self-management skills [7]. The sixth recommendation suggests that students interact with one another more often.

Pupils' social communication skills will decrease as an increasing number of students use artificial intelligence platforms for learning and machines as their main mode of communication. Students should assist and support one another while seeking higher education. Artificial intelligence-based education may provide a solution to this problem, resulting in remote learning paradigm with heavy focus on socialization. Students may learn online while simultaneously engaging in a variety of camps and social activities. Finally, as economic and technological globalization have progressed, the importance of artificial intelligence technology in education has become clearer. In a number of countries, artificial intelligence research has been elevated to a national priority. The revolutionary AI-based education ecosystem enables accurate, personalized, and adaptable educational services and administration. When it comes to the classroom use of artificial intelligence, schools, teachers, and students all face a range of problems and concerns. These problems can be resolved only via collaboration between educators, students, and other stakeholders in educational ecosystem.

## 5 Concluding Remarks

The use of artificial intelligence in education is expected to grow as technology advances. The debate about artificial intelligence and education may be better understood by examining how AI is being utilized in classrooms and the challenges it confronts. Application of AI in education assists instructors and students in navigating and using AI technology in teaching and learning process to enhance quality of teachers' instruction and students' learning ways, and also to diversity and personalize students' learning styles.

## References

1. Afra, S., Alhajj, R.: Early warning system: from face recognition by surveillance cameras to social media analysis to detecting suspicious people. *Phys. A: Stat. Mech. Appl.* (2020)
2. An, R., Xi, T.: Research on the service design of smart campus based on sustainable strategy—taking smart canteen as an example. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12202 LNCS, pp. 20–30. Springer (2020)
3. Azad, S., Chen, B., Fowler, M., West, M., Zilles, C.: Strategies for deploying unreliable AI graders in high transparency high-stakes exams. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2020)
4. Bergamin, P., Hirt, F.S.: Who's in charge?—dealing with the self-regulation dilemma in digital learning environments. *Knowledge Management in Digital Change*, pp. 227–245. Springer (2018)
5. Bin, Y., Mandal, D.: English teaching practice based on artificial intelligence technology. *J. Intell. Fuzzy Syst.* (2019)
6. Casanova, A.: BYJU's: how a learning app is promoting deep conceptual understanding that is improving educational outcomes in India (2018)
7. Chang, J., Lu, X.: The study on students' participation in personalized learning under the background of artificial intelligence. In: 2019 10th International Conference on Information Technology in Medicine and Education (ITME), pp. 555–558. IEEE (2019)
8. Chassignol, M., Khoroshavin, A., Klimova, A., Bilyatdinova, A.: Artificial Intelligence trends in education: a narrative overview. *Procedia Comput. Sci.* (2018)
9. Chen, L., Zechner, K., Yoon, S.Y., Evanini, K., Wang, X., Loukina, A., Tao, J., et al.: Automated scoring of nonnative speech using the SpeechRater SM v. 5.0 Engine. *ETS Research Report Series* (2018)
10. Cheong, K.H., Koh, J.M.: Integrated virtual laboratory in engineering mathematics education: Fourier theory. *IEEE Access* (2018)
11. Cui, W., Xue, Z., Thai, K.P.: Performance comparison of an AI-based adaptive learning system in China. In: *Proceedings 2018 Chinese Automation Congress, CAC 2018* (2019)
12. Deloitte: Global Development of AI-Based Education (2019)
13. Dong, Z., Zhang, Y., Yip, C., Swift, S., Beswick, K.: Smart campus: definition, framework, technologies, and services. *IET Smart Cities* **2**(1), 43–54 (2020)
14. Fang, Y., Ren, Z., Hu, X., Graesser, A.C.: A meta-analysis of the effectiveness of ALEKS on learning. *Educ. Psychol.* **39**(10), 1278–1292 (2019)
15. Fu, Y.: Research on the development trend of online education industry considering the influence of big data and artificial intelligence. *Advances in Intelligent Systems and Computing*, vol. 928, pp. 852–859. Springer Verlag (2020)

16. Ge, X.L., Yin, Y.W., Feng, S.: Application research of computer artificial intelligence in college student sports autonomous learning. *Educ. Sci. Theor. Pract.* **18**(5), 2143–2154 (2018)
17. Grams, D.: A quantitative study of the use of “DreamBox Learning” and its effectiveness in improving Math achievement of elementary students with Math difficulties. ProQuest LLC (2018)
18. Sarkar, S., et al.: Spintronics based systems for AI nano robots. In: 2018 International Conference on Computing, Power and Communication Technologies (GUCON), pp. 873–877 (2018). <https://doi.org/10.1109/GUCON.2018.8674931>
19. Hao, K.: China has started a grand experiment in AI education. It could reshape how the world learns. *MIT Technol. Rev.* (2019)
20. Kim, Y., Soyata, T., Behnagh, R.F.: Towards emotionally aware AI smart classroom: current issues and directions for engineering and education. *IEEE Access* (2018)
21. Knox, J.: Artificial intelligence and education in China. *Learn. Media Technol.* **45**(3), 298–311 (2020)
22. Krumm, J. (ed.): *Ubiquitous Computing Fundamentals*. Chapman and Hall/CRC (2018)
23. Kulkarni, P.V., Rai, S., Kale, R.: *Recommender System in eLearning: A Survey*, pp. 119–126. Springer, Singapore (2020)

# A Comprehensive System for Detecting Profound Tiredness for Automobile Drivers Using a CNN



Jitendra Singh Kushwah, Dhyanendra Jain, Prashant Singh,  
Amit Kumar Pandey, Sanjoy Das, and Prashant Vats

**Abstract** The sleeping is a found naturally condition of body and mind that is characterized by changed awareness, somewhat suppressed perceptual activity, decreased muscular activity and suppression of almost all muscle fibers, and decreased relationship with the environment. Whenever it comes to dangerous sleepiness, the car driver and those around his safety is solely dependent on their judgments. The said review shows a drowsiness detection & identification and continuing to drive behavior concealer methodology based on a communicative associate agent capable of distinguishing and attempting to avoid motorist drowsiness behind the rear axle, using a webcam to obtain actual pictures of the driver's face, and a representative displayed on the LCD, observes the driver's picture in order to heat of drowsiness and avoid a potential mishap. A revised yolov5 (YOLOv4)-based neural net is developed in this study to identify motorist sleepiness.

---

J. S. Kushwah

Department of Computer Science & Engineering, Institute of Technology and Management, Gwalior, Madhya Pradesh, India

e-mail: [jitendra.singhkushwah@itmgoi.in](mailto:jitendra.singhkushwah@itmgoi.in)

D. Jain · P. Singh (✉) · A. K. Pandey

Department of Information Technology, Dr. Akhilesh Das Gupta Institute of Technology & Management, New Delhi, India

e-mail: [prashant.ert@gmail.com](mailto:prashant.ert@gmail.com)

D. Jain

e-mail: [dhyanendra.jain@gmail.com](mailto:dhyanendra.jain@gmail.com)

A. K. Pandey

e-mail: [amitpandey33@gmail.com](mailto:amitpandey33@gmail.com)

S. Das

Department of Computer Science Engineering, Indira Gandhi National Tribal University Manipur, Manipur, India

e-mail: [sdas.jnu@gmail.com](mailto:sdas.jnu@gmail.com)

P. Vats

Department of Computer Science & Engineering, Dr. Akhilesh Das Gupta Institute of Technology & Management, New Delhi, India

e-mail: [prashantvats12345@gmail.com](mailto:prashantvats12345@gmail.com)

**Keywords** Convolutional neural network (CNN) · Driver drowsiness detection · Human–machine interface · OpenCV · YOLO

## 1 Introduction

The capability of major highways to handle transport (both passengers and freight) must maintain up with the economic expansion. With around 62.16 lakh kilometers of highway, India does have these cond-fastest-growing transit system. Regional highways, interstate expressways, state roads, major district roads, many district roads, and rural roadways are all included. In 2018, 151,417 people have been killed and 469,418 have been injured in road accidents in India, as per government stats. Nevertheless, this is likely an underestimation of fatalities because not all accidents are brought to court. As per the Public Security Advisory board (NSC), driver drowsiness causes approximately 1,200,000 crash risk, 72,000 concussions, and 900 deaths each year [1]. As according AAA, having trouble sleeping drivers directly relates to an approximate 9.5% of all accidents. Motorists' heads start to wobble as they grow sleepy, as well as the car may slip away from of the center of the lane. All stated previously vehicular and eye sight counter measures become visible only when the driver begins to slumber, which really is frequently too early to avoid a crash [2]. This study is organized into 4 segments: Following this introductory, Sect. 2 is linked publications, third section is methodology, Sect. 4 is experimentation and achieved findings, and fifth section is the conclusions and a prediction of future horizons. To summarize, the influence of sleepy driving on traffic accidents is well-known and, as such, an issue that must be addressed. Nonetheless, given the present investment in autonomous vehicles, automobiles from every major automaker, thereby planning for the future.

The research of this subject may be deemed superfluous in the age of self-driving autos. However, the following automobile analysis the automation process must be considered [3]. First and foremost, only cars with SAE level 1, 2 are permitted. Accessible as of now, that is, automobiles with modern driver assistance systems lane keeping and speed assistance systems (ADAS) capability for adjusting. These automobiles are far from perfect. Capable of driving on their own.

Second, even with SAE Level 3 and 4 cars, where automated driving is conceivable under certain conditions, the driver must still be willing to take control, sometimes after an extended period of non-driving. In fact, driver handoff tactics are now a hot topic, with drowsiness playing a significant role [4], underscoring the urgent need to further investigate fatigue detection. Finally, in the best-case scenario, level five automation will not be available until at least 2025 [3], meaning the transition from manual to fully autonomous driving will be slow and a significant portion of the population will continue to drive non-fully automated vehicles for many years to come.

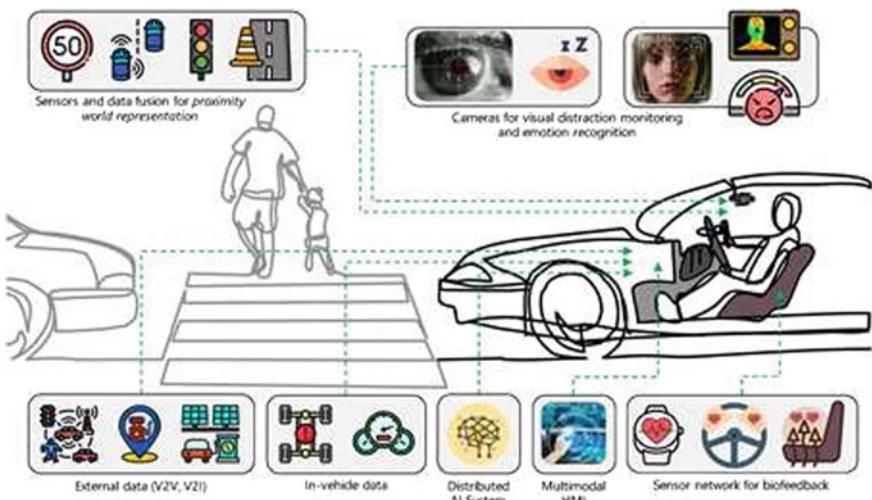
## 2 Related Work

Numerous studies have considered the very next physiological data to identify tiredness: electromyogram (EMG), electrocardiogram (ECG) electrooculogram (EoG), electroencephalogram (EEG). A few really scientists use the recorded EEG signal to detect driver fatigue through nystagmus. This same membrane potential between the ocular surface and the photoreceptors creates an electromagnetic current that takes into account the direction of the eye sockets; this electrostatic force is the evaluated EoG indication [3]. The possible methods are much more concerned with the identification of tiredness than with its evolvement or management. The main objective of an Inter-personal Interaction scheme is to maximize the efficiency of human-automated operative collaboration. To accomplish the above, the scheme must be easily interpreted by the motorist, boost his/her knowledge of the problem, and (greatest importantly) be believed [4]. The HMI discussed in this chapter is intended to capitalize on the Advisor Assistant's window of opportunity, i.e., to quickly portray its conceptual framework in order to: (1) Offer effective information and explanation whenever the scheme makes a decision. (2) Promote teamwork once sentient and done by robot's representatives start sharing actions and decisions. (3) Avoid providing irrelevant details when delegating actions and decisions to the car pilot in order to prevent perceptual and project environment over abundance [5]. The HMI is installed in a filled bidirectional instrument panel. It incorporates all essential driving instruction (e.g., present velocity, gearing, automated mode) and also evidence concerning the driver's state (e.g., if he/she is pre-occupied) and the actions necessary to achieve optimum driving performance (i.e., the suggested behavior). The HMI was created using concepts connected to the negotiated settlement interaction method. This implies that the HMI's primary objective is to "understand instead of alarm," collaborating with the driver to achieve a pleasurable, pleasant, and safe journey [6]. The HMI will notify the motorist on what the driver anticipates of him and offer additional especially true to the factors that led to the invitation for interplay, based on decisions made by the supervisor representative.

Diagrammatical interpretations are supplied through with a 3D interactive portrayal at the center of the HMI, in which the road climate and surrounding road actors are recreated (e.g., regarding map data plus vehicular detectors) and showcased through with an expressionistic portrayal. Communications delivered via acoustic input and message [7].

## 3 Proposed Work

The graphic below, for instance, depicts a scenario in which the driving and mechanization are exchanging control of the vehicle, and the automobile alerts the user that they are nearing an automobile which will be tracked as well as a computer process (Fig. 1).



**Fig. 1** Drawing of an HMI for regulate sharing between such a car driver

The diagram below illustrates a scenario in which the mechanization is involved and, as a result of a combination of sensorimotor limitations (i.e., dearth of clarity) and the application of a careful behaviors, it truly notifies the motorist that “vehicle going to follow” (CF) would then occur, unless motorist purposefully overrides the system that performs a conventional mechanical overtake. In just this case, this same 3D depiction is more concerned with identifying the rationale for the behaviors (i.e., the accessibility restriction) than with the real action demanded of the motorist, which is consigned to a tiny text in to hand corner of the window. Because it is directed at cultivating behaviors instead of expressly pressuring are sponse, this design need storely on implied conversations (Fig. 2) [8].

Eventually, the statistic below depicts the specific instance where the driver initiates an emergency warning manoeuvre; in this case, the explanation is given prior to the car’s real stop, allowing the driver to regain control prior to the car’s stop. The collaboration is supplied here to demonstrate the forthcoming decision/behavior of the automated processes, namely to stop in the exit lane (Fig. 3) [9].

#### 4 Experimental Setup and Results

The model has been implemented using Notepad++ (anaconda 3). As illustrated in Fig. 4, the model can recognize photos of driving vehicles, people, and other things such as pens, ties, cups, and so on.

The dataset for the pictures is ultralytics, as well as the data types of MS-COCO courses are in detectron dict order. 91 thing classes nICOCO-Stuff has (1–91), 91



**Fig. 2** Depicts an HMI notifying the motorist of the system-selected action (CF manoeuvre in this case)



**Fig. 3** Design of a HMI that notifies the operator of the rationale for an urgent maneuver's activation

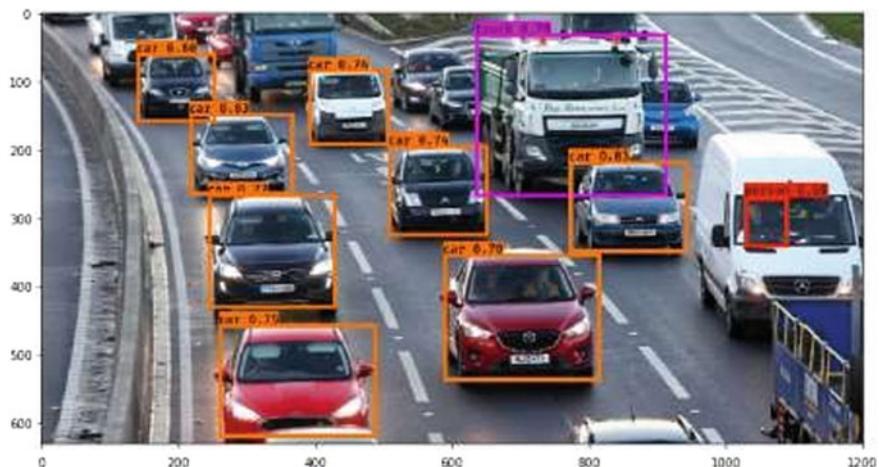
stuff classes (92–182) and 1 category “unmarked” to be consistent with COCO(0). It is worth noting that 11 of COCO’s item courses lack categorization comments (desk, blender, eyeglasses, door, hat, mirror, plate, hair brush, shoe, window, street sign) (Fig. 5).

The output from running results XYXY is as follows:

- Xmin
- Ymin
- Xmax
- Ymax



**Fig. 4** Model identifying the objects from the video

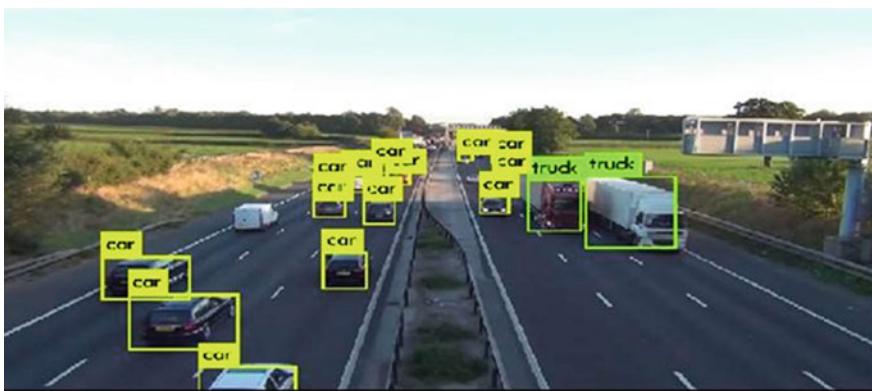


**Fig. 5** Model identifying the several objects in the frame

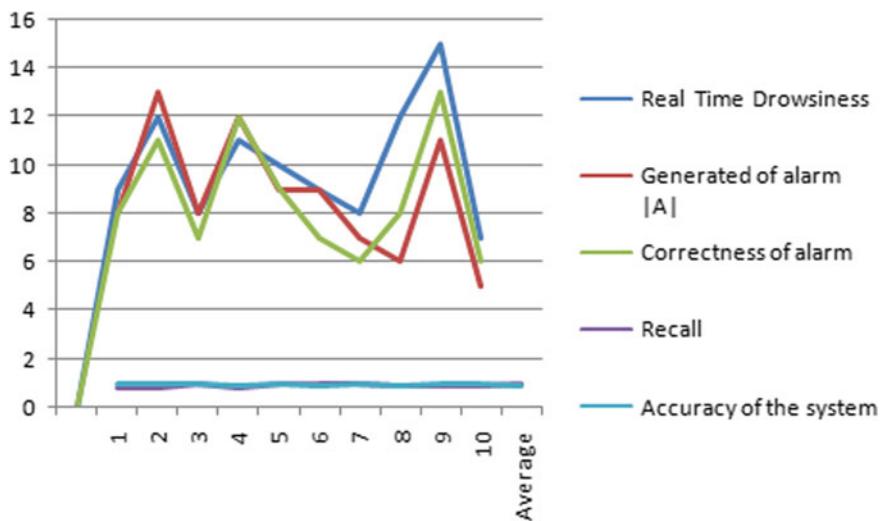
- Confidence
- Class.

To use an OpenCV circuit, we can connect the picture first from web camera and use the model approach to connect the web camera, capable of making detection systems in live time, as seen in Fig. 6 (Fig. 7).

Table 1 displays the tiredness detection results from the 10 test videos. The expert guidance in this phase is required to assess a driver's true sleepiness as indicated by the  $|RI|$  indices. Equation (16) calculates the recall rate based on the right alert index as well as those defined by the specialist.



**Fig. 6** Model identifying the objects from the open webcam



**Fig. 7** Graph representing improvement in system

$$\text{Recall} = (R_a)/(R)$$

The accuracy is determined by dividing the number of accurate alarms by the number of alarms generated by the system.

$$\text{Accuracy} = (R_a)/A$$

The suggested approach's findings for detecting the condition of the driver's tiredness have an excellent average Recall and Precision (near 0.95). The impact of blurring captured with an RGB camera, on the other hand, explains the faults that

**Table 1** Result of fatigue detection

Videos (Actual)	Real time drowsiness  R	Generated of alarm  A	Correctness of alarm	Recall	Accuracy of the system
			Ral		
1	9	8	8	0.8	0.99
2	12	13	11	0.82	0.95
3	8	8	7	0.95	0.95
4	11	12	12	0.81	0.88
5	10	9	9	0.98	0.99
6	9	9	7	0.95	0.85
7	8	7	6	0.99	0.99
8	12	6	8	0.88	0.87
9	15	11	13	0.87	0.92
10	7	5	6	0.9	0.99
<i>Average</i>				0.949	0.85

produced warnings. These mistakes have an impact on the detection of the iris and, as a result, the values of the descriptors.

Using this method, we can ensure an efficient system that can be used to prevent fatigue-related incidents on a significant scale. This technique is adaptable to all types of vehicles. At the point of drowsiness detection, we can also include additional features, like a message to be sent to another person preselected by the user, and the message can contain data like GPS position, driving speed, and so on. All of these factors can combine to reduce accident rates and save many people from serious injury and death.

## 5 Conclusion

This document compares and contrasts the two-schemes used here which is a juncture designed to check exhaustion eye gaze that was being discovered using an image sensor. This is accomplished to recognize the sleep problems to prevent a mishap. It is based on the idea of eye movements to achieve end outcome on studies about driver drowsiness warning and monitoring scheme to provide solutions to the problem of finding the state of day time sleepiness and a mathematics-based method is used the system at appoint in time to detect tiredness eye gaze which is identified using the image sensors. If the status of that has been determined and the alarm system has indeed been activated, based on 150 photographs of various persons that have been utilized and a complete description is been constructed using machine learning and embedded devices. It must have been put to the test and proved to be efficient in part. There is indeed a significant potential. The suggested technique distinguishes

between regular eye blink and sleepiness by detecting immobility as though the eyelid shave indeed been closed for a duration of 4 or even more frames. The created system is unobtrusive and may be corrected by incorporating different sensors. The structure is predicated on machine learning and employs the notion of graphics processing and alive camera with the assistance of OpenCV to recognize certain face traits for identification. It also lists some of the suggested system's drawbacks as well as solutions to those drawbacks.

## References

1. The Motor Vehicles Act 1988 (Section 41 of Chapter IV)
2. Lee, K.H., Kim, W., Choi, H.K., Jan, T.: A Study on Feature Extraction Methods Used to Estimate a Driver's Level of Drowsiness. IEEE (2019)
3. Hong, T., Qin, H.: Driver Drowsiness Detection in Embedded Systems. IEEE (2007)
4. Sarkar, D., Chowdhury, A.: Real time embedded system application for driver drowsiness and alcoholic intoxication detection. IJETT **10**(9) (2014)
5. Batchu, S., Praveen Kumar, S.: Driver drowsiness detection to reduce the major road accidents in automotive vehicles. IRJET **02**(01) (2015)
6. Mukhopadhyay, M., et al.: Facial emotion recognition based on textural pattern and convolutional neural network. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6 (2021). <https://doi.org/10.1109/GUCON50781.2021.9573860>
7. Fouzia, Roopa Lakshmi, R., Rathore, J.A., Shetty, A.S., Supriya, K.: Driver Drowsiness Detection System Based on Visual Features. IEEE (2018)
8. Dahiphale, V.E., Satyanarayana, R.: A real-time computer vision system for continuous face detection and tracking. IJCA **122**(18) (2015)
9. Pratama, B.G., Ardiyanto, I., Adji, T.B.: A Review on Driver Drowsiness Based on Image, Bio-signal, and Driver Behaviors. IEEE (2017)

# Satellite Data Investigation for Change Estimation During COVID Era by Fusing Pixel and Object-Based Technique



Amit Kumar Shakya , Ayushman Ramola, and Anurag Vidyarthi

**Abstract** In this investigation, an innovative combination of pixel-based change detection technique and object-based change detection technique is explored with the satellite images of Holy Masjid al-Haram, Saudi Arabia. The gray-level co-occurrence matrix (**GLCM**) method is used to quantify the texture of the remote sensing data through the texture classification approach on the satellite data in this work. **GLCM** produces results of the texture quantification in normalized form. Thus, applying a texture classification scheme on the satellite data is impressive to observe. Later maximum likelihood image classification approach is used for classification purposes. The classified information is categorized into four different classes. The kappa coefficient's value and the overall accuracy for the *pre-COVID* classified study area are **0.6532** and **76.38%**, respectively. During **COVID**, the classified study area presents the kappa coefficient and the overall accuracy of **0.7631** and **82.18%**, respectively.

**Keywords** *GLCM* · Texture · Satellite · *COVID* · Maximum likelihood classification

## 1 Introduction

Nowadays, several change detection methodologies are used in remote sensing and satellite application [1]. These techniques are, i.e., “pixel-based change detection (*PBCD*)” and “object-based change detection (*OBCD*)” [2]. *PBCD* technique is widely used in the earlier times in which information about a study area is obtained based on pixel arrangement in an image. This technique has several advantages as it provides information about the image pixels through spectral and spatial behavior

---

A. K. Shakya · A. Ramola

Sant Longowal Institute of Engineering and Technology (SLIET), Sangrur, Punjab, India  
e-mail: [xlamitshakya.gate2014@ieee.org](mailto:xlamitshakya.gate2014@ieee.org)

A. Vidyarthi  
Graphic Era (Deemed to be University), Dehradun, Uttarakhand, India

analysis. The other method for image classification is *OBCD* which provides information about the study area in terms of objects (classification classes). The classification area is converted into several independent classes, and these objects are compared with each other to obtain the amount of change developed in the study area. Image differencing, principal component analysis, image ratio, image regression, texture classification-based approaches are prominent *PBCD* methodologies [3]. “Classified object-based change detection (*COCD*),” “multi-temporal-based change detection (*MT OCD*),” “direct object-based change detection (*DOCD*)” methodologies are some prominent *OBCD* methodology [4, 5].

## 2 Background of Change Detection Techniques

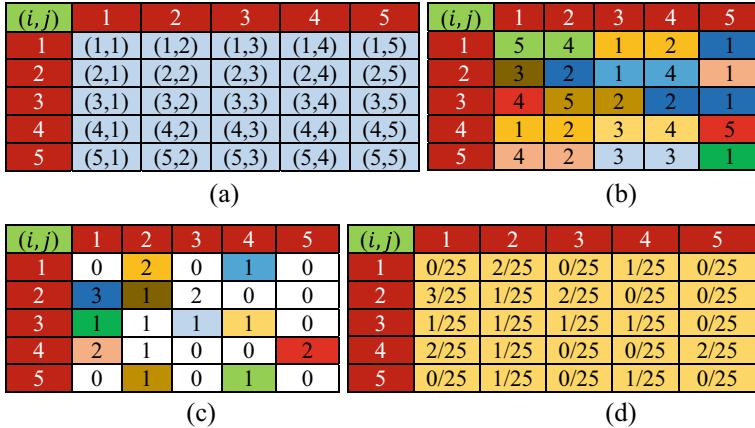
In remote sensing, information from satellite images can be obtained from several methods. This technique provides spectral, spatial, visual, numerical data about the images. These techniques provide information about several remote sensing data. *Harlick* was the first one to develop *GLCM* [6]. He presents features categorized into fourteen different categories to classify an image’s texture. Several texture analysis applications include *DICOM* image [7], *PALSAR* image, multispectral image analysis [8–11], etc.

Scientist *Gotlieb* divided these features into four distinct groups based on several parameters [12]. In remote sensing applications, visual identification of the features is important because the changes developed in any study area can be identified through observable human behavior. Texture visual features include “homogeneity,” “energy,” “correlation,” and “contrast” [13]. This research uses visual texture features to investigate *pre-COVID* and during *COVID* multispectral images of Holy Masjid al-Haram, Mecca, Saudi Arabia [14]. The *GLCM* formation of an investigating image is presented in Fig. 1. The pixel position of the input image location is illustrated in Fig. 1a. Input image of dimension  $5 \times 5$  is presented in Fig. 1b. The *GLCM* image of the test image is illustrated in Fig. 1c. The normalized image of the *GLCM* image is presented in Fig. 1d.

*GLCM* of the image depends on two parameters “distance” and “angle of orientation.” The distance represents space between the “pixel of interest” and the “neighboring pixel.” The orientation angle varies from  $0^\circ$  to  $315^\circ$ . Table 1 presents *GLCM* features affecting human vision.

### OBCD methodology

“Object-based change detection (*OBCD*)” classifies the input data into several classes. These classes can be classified as water, urban, soil, etc. In our case, we have classified objects as pilgrims, ground, rooftop, shadow, etc. When two images are compared, following *OBCD* methodologies, they are compared with the individual “user accuracy, producer accuracy, commission error, omission error, overall accuracy, and kappa coefficient” [15]. Based on these parameters, different classification techniques known as “minimum distance classification (*MDC*)” [16], maximum



**Fig. 1** **a** Pixel location **b** Test data dimension  $5 \times 5$  **c** *GLCM* of test image **d** Representation of the *GLCM* image in normalized form

**Table 1** *GLCM* visual features

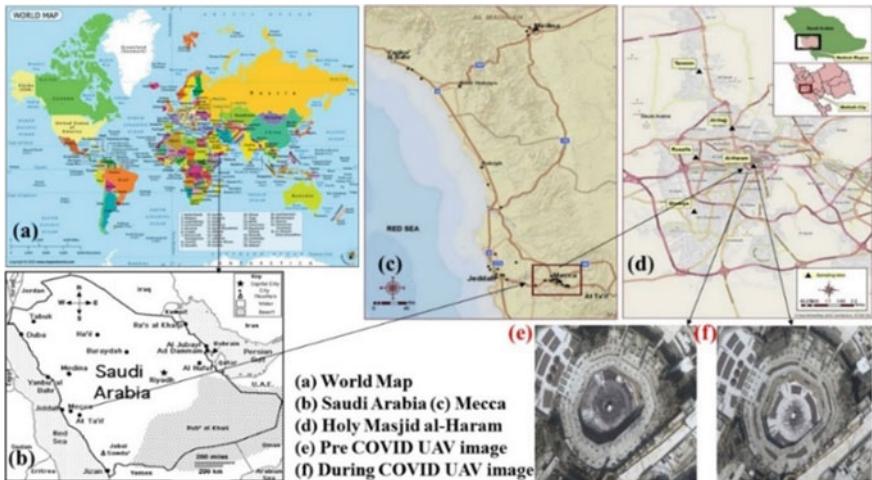
S. No.	Features	Expression	References
1	Homogeneity	$f(i, j) = \sum_i \sum_j \frac{1}{1+(i-j)^2} p(i, j)$	[5]
2	Energy	$f(i, j) = \sum_i \sum_j \{p(i, j)\}^2$	[5]
3	Correlation	$f(i, j) = \frac{(i, j)p(i, j)-\mu_x\mu_y}{\sigma_x\sigma_y}$	[5]
4	Contrast	$f(i, j) = \sum_0^{n_g-1} n^2 \times \left\{ \sum_{i=1}^{n_g} \sum_{j=1}^{n_g} p(i, j) \right\}$	[5]

likelihood classification (*MLC*) [17], spectral angle mapper (*SAM*) [18], support vector machine (*SVM*) [19], Mahalanobis distance classification (*MaDC*) [20], parallelepiped classification (*PC*) [21], and spectral information divergence (*SID*)” [22] are developed and used to classify remote sensing data.

### 3 Experimental Results

In this experiment, two satellite images of Holy Mecca, Saudi Arabia, are obtained from [14]. The *pre-COVID* and during *COVID* images are examined through *GLCM*-based texture classification approach.

Texture quantification is performed through this approach, and feature values are obtained. The changes developed in the images are represented through an object-based change detection minimum distance image classification approach. Through



**Fig. 2** **a** World view **b** Study map of Saudi Arabia **c** Geographical location of the Holy Mecca **d** Geographical location of the Holy Masjid al-Haram **e** Satellite image of Holy Masjid al-Haram before *COVID* **f** Satellite image of Holy Masjid al-Haram during *COVID*

these classification approaches, image changes in the geographical location are presented. The change developed can easily be monitored and is visually identified. Figure 2 presents the geographical location of the study area Holy Masjid al-Haram, Mecca, which lies in the elevation of the Hejaz region. It has a height of 277 m, which is 909 ft above sea level. It is located at 21°23' North and 39°51' East longitude. It is divided into 34 districts [23]. Figure 2a represents the world view map. Figure 2b presents the political map of Saudi Arabia. Figure 2c illustrates a satellite image of Mecca. Figure 2d represents the geographical location of the study area Holy Masjid al-Haram. Figure 2e represents the satellite image of the holy Masjid al-Haram in the normal situation before the coronavirus spread. Figure 2f presents the satellite image of holy Masjid al-Haram during the time of coronavirus. We can observe changes developed in the sacred Masjid al-Haram area due to fewer people and changing geographical conditions. These satellite images are initially quantified through *GLCM*-based approach for analyzing modifications produced in the study areas' texture features. Later through *OBCT* techniques, the changes developed are highlighted to present the created changes visually. The texture feature quantification of Holy Masjid al-Haram before the *pre-COVID* and during *COVID* era is presented in Tables 2 and 3. The *GLCM* is calculated for eight different orientations and four different distances between the pixel pairs.

**#Notation** for Contrast (**Con#**), Correlation (**Corr#**), Energy (**Eng#**), Homogeneity (**Hom#**)

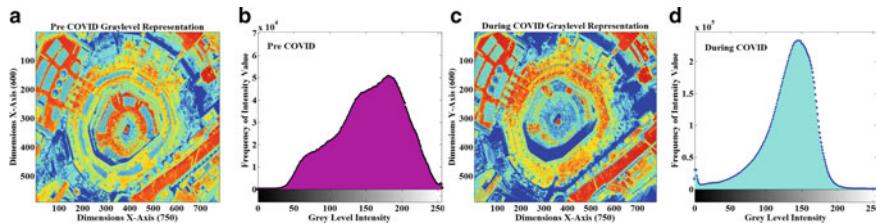
Through the gray-level image classification approach, the *pre-COVID* and during *COVID* images of the study areas are investigated. The visual representation of the *pre-COVID* and during *COVID* images of the test area is presented

**Table 2** GLCM features quantification for the *pre-COVID* image

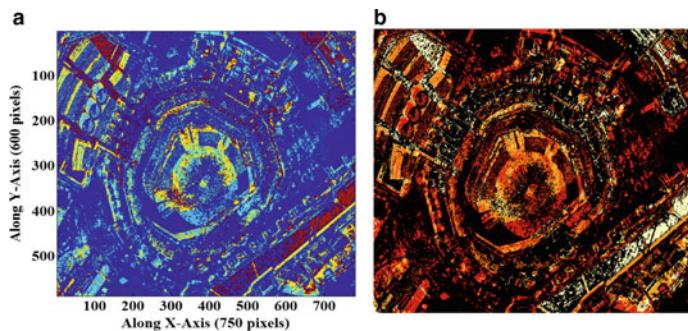
S. No	Texture features	Angle	Quantification of the visual texture features				Average
1	Con#	0°	0.0776	0.1345	0.1742	0.2013	0.1469
	Corr#		0.9051	0.8354	0.7867	0.7535	0.8202
	Eng#		0.3774	0.3421	0.3212	0.3083	0.3372
	Hom#		0.9628	0.9364	0.9193	0.9082	0.9317
2	Con#	45°	0.1150	0.1826	0.2178	0.2398	0.1888
	Corr#		0.8592	0.7764	0.7332	0.7061	0.7687
	Eng#		0.3526	0.3157	0.2996	0.2905	0.3146
	Hom#		0.9448	0.9149	0.9005	0.8918	0.9130
3	Con#	90°	0.0835	0.1445	0.1828	0.2065	0.1543
	Corr#		0.8977	0.8230	0.7761	0.7470	0.8109
	Eng#		0.3708	0.3336	0.3140	0.3032	0.3304
	Hom#		0.9586	0.9302	0.9138	0.9040	0.9266
4	Con#	135°	0.1099	0.1741	0.2087	0.2310	0.1809
	Corr#		0.8654	0.7868	0.7444	0.7169	0.7784
	Eng#		0.3559	0.3204	0.3043	0.2948	0.3189
	Hom#		0.9473	0.9188	0.9047	0.8959	0.9167
5	Con#	180°	0.0776	0.1345	0.1742	0.2013	0.1469
	Corr#		0.9051	0.8354	0.7867	0.7535	0.8202
	Eng#		0.3774	0.3421	0.3212	0.3083	0.3372
	Hom#		0.9628	0.9364	0.9193	0.9082	0.9317
6	Con#	225°	0.1150	0.1826	0.2178	0.2398	0.1888
	Corr#		0.8592	0.7764	0.7332	0.7061	0.7687
	Eng#		0.3526	0.3157	0.2996	0.2905	0.3146
	Hom#		0.9448	0.9149	0.9005	0.8918	0.9130
7	Con#	270°	0.0835	0.1445	0.1828	0.2065	0.1543
	Corr#		0.8977	0.8230	0.7761	0.7470	0.8109
	Eng#		0.3708	0.3336	0.3140	0.3032	0.3304
	Hom#		0.9586	0.9302	0.9138	0.9040	0.9266
8	Con#	315°	0.1099	0.1741	0.2087	0.2310	0.1809
	Corr#		0.8654	0.7868	0.7444	0.7169	0.7784
	Eng#		0.3559	0.3204	0.3043	0.2948	0.3189
	Hom#		0.9473	0.9188	0.9047	0.8959	0.9167

**Table 3** GLCM features quantification for the during COVID image

S. No	Texture features	Angle	Quantification of the visual texture features				Average
1	Con#	0°	0.0929	0.1753	0.2354	0.2748	0.1946
	Corr#		0.9142	0.8381	0.7826	0.7462	0.8203
	Eng#		0.3591	0.3257	0.3059	0.2931	0.3209
	Hom#		0.9587	0.9311	0.9132	0.9010	0.9260
2	Con#	45°	0.1300	0.2221	0.2751	0.3081	0.2338
	Corr#		0.8800	0.7949	0.7459	0.7153	0.7840
	Eng#		0.3380	0.3030	0.2861	0.2760	0.3008
	Hom#		0.9425	0.9117	0.8952	0.8847	0.9085
3	Con#	90°	0.0895	0.1634	0.2159	0.2520	0.1802
	Corr#		0.9174	0.8492	0.8008	0.7674	0.8337
	Eng#		0.3549	0.3198	0.3006	0.2890	0.3161
	Hom#		0.9568	0.9277	0.9101	0.8986	0.9233
4	Con#	135°	0.1329	0.2252	0.2749	0.3060	0.2348
	Corr#		0.8773	0.7921	0.7461	0.7173	0.7832
	Eng#		0.3382	0.3040	0.2877	0.2779	0.3020
	Hom#		0.9423	0.9120	0.8962	0.8861	0.9092
5	Con#	180°	0.0929	0.1753	0.2354	0.2748	0.1946
	Corr#		0.9142	0.8381	0.7826	0.7462	0.8203
	Eng#		0.3591	0.3257	0.3059	0.2931	0.3209
	Hom#		0.9587	0.9311	0.9132	0.9010	0.9260
6	Con#	225°	0.1300	0.2221	0.2751	0.3081	0.2338
	Corr#		0.8800	0.7949	0.7459	0.7153	0.7840
	Eng#		0.3380	0.3030	0.2861	0.2760	0.3008
	Hom#		0.9425	0.9117	0.8952	0.8847	0.9085
7	Con#	270°	0.0895	0.1634	0.2159	0.2520	0.1802
	Corr#		0.9174	0.8492	0.8008	0.7674	0.8337
	Eng#		0.3549	0.3198	0.3006	0.2890	0.3161
	Hom#		0.9568	0.9277	0.9101	0.8986	0.9233
8	Con#	315°	0.1329	0.2252	0.2749	0.3060	0.2348
	Corr#		0.8773	0.7921	0.7461	0.7173	0.7832
	Eng#		0.3382	0.3040	0.2877	0.2779	0.3020
	Hom#		0.9423	0.9120	0.8962	0.8861	0.9092



**Fig. 3** **a** Visual representation of change in the pre-*COVID* image **b** Histogram plot of the *pre-COVID* image **c** Visual representation of change in the *during COVID* image **d** Histogram plot of the *during COVID* image



**Fig. 4** **a** Gray-level image differencing representation of the *pre-COVID* and during *COVID* image of the study area **b** Thermal view of the image differencing map

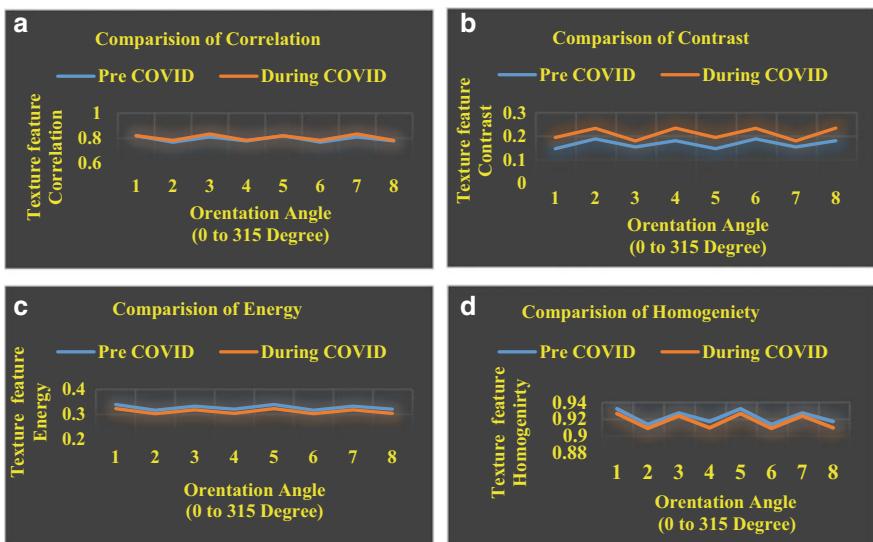
in Fig. 3. The histogram plot of the *pre-COVID* and during *COVID* images represents the changes developed in the study area.

The differencing image technique can visually represent the change that occurred in the pre- and during *COVID* images of the study area illustrated in Fig. 4a. The thermal view of the differencing image map is presented in Fig. 4b. Results of comparison of texture features for the *pre-COVID* and during *COVID* images are shown in Fig. 5a–d.

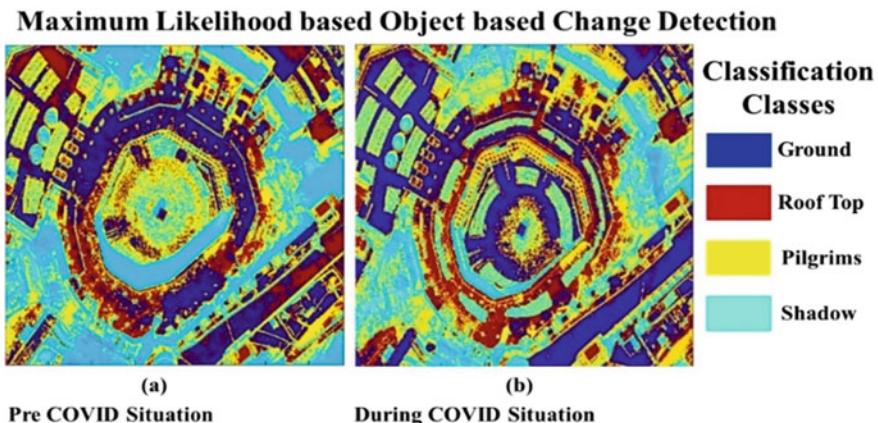
With Google Earth's assistance, the dataset training is performed by manually selecting appropriate classification regions for the four classes known as ground, rooftop, pilgrims, and shadow. The classification accuracy is computed using the ground truth through longitude and latitude coordinates of the respective area. In *OB**CD*, maximum likelihood-based change detection is an efficient image classification technique. Thus, the classification results of maximum likelihood classification are presented in Fig. 6.

The confusion matrix created for the *pre-COVID* and during *COVID* image using maximum likelihood classification is expressed in Tables 4 and 5, respectively.

The kappa coefficient's value and the overall accuracy for the *pre-COVID* classified study area are 0.6532 and 76.38%, respectively. Similarly, for the *COVID* classified study area, the kappa coefficient and the overall accuracy have values of



**Fig. 5** Comparison of *pre-COVID* and during *COVID* image features **a** Correlation **b** Contrast **c** Energy **d** Homogeneity



**Fig. 6** Maximum likelihood-based image classification

**Table 4** Developed confusion matrix for the *pre-COVID* image

Classes	Producer accuracy	User accuracy	Omission error	Commission error
Ground	85.52	88.64	14.00	11.34
Roof top	80.00	68.66	18.57	28.32
Pilgrims	76.00	85.42	21.23	12.05
Shadow	83.63	81.52	16.38	18.65

**Table 5** Developed confusion matrix for the during *COVID* image

Classes	Producer accuracy	User accuracy	Omission error	Commission error
Ground	100.00	65.00	1.00	35.00
Roof top	04.00	57.19	95.00	16.96
Pilgrims	98.00	82.05	02.00	16.96
Shadow	76.77	98.80	23.22	02.20

0.7631 and 82.18%, respectively. Thus, kappa value is approaching one, suggesting that the performed classification is reasonable. The study areas' overall classification accuracy is above 75%, offering that classification is close to reality.

## 4 Conclusion

This research work presents a combination of *PBCD* and *OBCD* methodologies. *PBCD*-based *GLCM* approach is ideal for the quantification of the image texture. Texture features can be quantified into four different categories. In this research work, texture features affecting the human visual are investigated. The *OBCD* approach is suitable to classify an image in terms of other classes. In this experiment, we have created four different classes and used the maximum likelihood image classification approach to classify the image. *Pre-COVID* and the during *COVID* satellite images of Holy Masjid al-Haram, Saudi Arabia, are investigated in this research work. The features derived from *GLCM* provide us information about the behavior of the visual texture features. Contrast and correlation show similar behavior. Similarly, texture feature energy and homogeneity show identical behavior when *pre-COVID* and *COVID* images are compared. The overall accuracy of the *pre-COVID* image is 76.38%, and the overall accuracy of during *COVID* replication is 82.18% using maximum likelihood classification. Thus, we have analyzed that *PBCD* and *OBCD* approaches can be used to classify a satellite image from the proposed research work. If someone is interested in obtaining the behavior of texture features for them, *GLCM* is a better option. But suppose the comparison between two image classes is required. It is better first to create the class using the *OBCD* technique and later make any comparison employing pixel-based *GLCM* approach or any other *OBCD* method as per the investigation requirement.

## References

1. Woodcock, C.E., Loveland, T.R., Herold, M., Bauer, M.E.: Transitioning from change detection to monitoring with remote sensing: a paradigm shift. *Remote Sens. Environ.* **238**, 111558 (2020). <https://doi.org/10.1016/j.rse.2019.111558>

2. Shakya, A.K., Ramola, A., Kandwal, A., Prakash, R.: Change over time in grey levels of multispectral Landsat 5TM/8OLI satellite images. Lecture Notes in Electrical Engineering, Mesra, Ranchi, pp. 309–356. Springer, Singapore (2019)
3. Shakya, A.K., Ramola, A., Vidyarthi, A.: Conversion of Landsat 8 multispectral data through modified private content-based image retrieval technique for secure transmission and privacy. Eng. Rep. **2**(12), e12273 (2020)
4. Hussain, M., Chen, D., Cheng, A., Wei, H., Stanley, D.: Change detection from remotely sensed images: from pixel-based to object-based approaches. ISPRS J. Photogr. Remote Sens. **80**, 91–106 (2013). <https://doi.org/10.1016/j.isprsjprs.2013.03.006>
5. Ramola, A., Shakya, A.K., Pham, D.V.: Study of statistical methods for texture analysis and their modern evolutions. Eng. Rep. **2**(4), e12149 (2020)
6. Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural features for image classification. IEEE Trans. Syst. Man Cybern. **SMC 3**(6), 610–621 (1973)
7. Shakya, A.K., Ramola, A., Pandey, D.C.: Polygonal region of interest-based compression of DICOM images. In: International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, India (2017)
8. Tripathi, S., Sharan, T.S., Sharma, S., Sharma, N.: Segmentation of brain tumour in MR images using modified deep learning network. In: 8th International Conference on Smart Computing and Communications (ICSCC), Kochi, India (2021)
9. Sharan, T.S., Tripathi, S., Sharma, S., Sharma, N.: Encoder modified U-Net and feature pyramid network for multi-class segmentation of cardiac magnetic resonance images. IETE Tech. Rev. **1**, 1–13 (2021). <https://doi.org/10.1080/02564602.2021.1955760>
10. Tripathi, S., Sharma, N.: Computer-based segmentation of cancerous tissues in biomedical images using enhanced deep learning model. IETE Tech. Rev. 1–15 (2021). <https://doi.org/10.1080/02564602.2021.1994044>
11. Shakya, A.K., Ramola, A., Vidyarthi, A.: Statistical quantification of texture visual features for pattern recognition by analyzing pre-and post-multispectral landsat satellite imagery. Nat. Hazard. Rev. **22**(4), 05021011 (2021)
12. Gotlieb, C.C., Kreyszig, H.E.: Texture descriptors based on co-occurrence matrices. Comput. Vis. Graph. Image Process. **51**(1), 70–86 (1990)
13. Shakya, A.K., Ramola, A., Kandwal, A., Mittal, P., Prakash, R.: Morphological change detection in terror camps of area 3 and 4 by pre-and post-strike through MOAB: b. In: Advances in Communication, Devices, and Networking, Sikkim, pp. 265–275. Springer, Singapore (2019)
14. REUTERS. Before and after the coronavirus, 17 March 2020 [Online]. Available <https://www.reuters.com/news/picture/before-and-after-the-coronavirus-idUSRTS3517B>. Accessed 10 Mar 2021
15. Shakya, A.K., Ramola, A., Kandwal, A., Prakash, R.: Comparison of supervised classification techniques with ALOS PALSAR sensor for Roorkee region of Uttarakhand, India. Int. Arch. Photogr. Remote Sens. Spatial Inf. Sci. **XLII-5**, 693–701 (2018). <https://doi.org/10.5194/isprs-archives-XLII-5-693-2018>
16. Samanta, A.K., Routray, A., Khare, S.R., Naha, A.: Minimum distance-based detection of incipient induction motor faults using Rayleigh quotient spectrum of conditioned vibration signal. IEEE Trans. Instr. Measur. **70**, 3508311 (2021). <https://doi.org/10.1109/TIM.2020.3047433>
17. Qiu, S., Sheng, W., Ma, X., Kirubarajan, T.: A maximum likelihood method for joint DOA and polarization estimation based on manifold separation. IEEE Trans. Aerosp. Electron. Syst. **1** (2021) <https://doi.org/10.1109/TAES.2021.3059094>
18. Zhuang, H., Deng, K., Fan, H., Yu, M.: Strategies combining spectral angle mapper and change vector analysis to unsupervised change detection in multispectral images. IEEE Geosci. Remote Sens. Lett. **13**(5), 681–685 (2016)
19. Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., Lopez, A.: A comprehensive survey on support vector machine classification: applications, challenges, and trends. Neurocomputing **408**, 189–215 (2020). <https://doi.org/10.1016/j.neucom.2019.10.118>

20. Chang, Z., Chen, W., Gu, Y., Xu, H.: Mahalanobis-Taguchi system for symbolic interval data based on kernel Mahalanobis distance. *IEEE Access* **8**, 20428–20438 (2020). <https://doi.org/10.1109/ACCESS.2020.2967411>
21. Bijeesh, T.V., Narasimhamurthy, K.N.: Surface water detection and delineation using remote sensing images: a review of methods and algorithms. *Sustain. Water Resour.* **68**(5), 1–23 (2020)
22. Chen, S., Xue, B., Yang, H., Li, X., Zhao, L., Chang, C.-I.: Optical remote sensing image registration using spatial-consistency and average regional information divergence minimization via quantum-behaved particle swarm optimization. *Remote Sens.* **12**(18), 3066 (2020)
23. Abdelrahman, K., Alamri, A.M., Al-Otaibi, N., Fnais, M.: Geotechnical assessment for the ground conditions in Makah Al-Mukarramah city, Saudi Arabia. *J. King Saud Univ. Sci.* **32**(3), 2112–2121 (2020)

# The Soiling Impact Effects on the Performance of the Solar Panels in Rabigh, Saudi Arabia



Abdullah Mohammed Zaki Khayyat, Abdulrahman Hamad Alharthi,  
Faisal Lafi Almohammadi, Ziyad Saad Almarwani,  
Nithiyanthan Kannan, and Youssef Mobarak

**Abstract** The primary goal of this study is to determine the impact of soiling on PV panels in Rabigh, Saudi Arabia. Dust collects naturally everywhere, and it is the most common nuisance in such as Saudi Arabia which has a vast desert area. According to this research work carried out, the collection of dust has harmed the efficiency and overall performance of the photovoltaic system. This soiling effect hinders sunlight from reaching PV panels, and it also shortens the life of solar panels in the long run. In comparison to the fall and winter seasons, the dust has a significant impact on solar panel performance in the summer and spring. To optimize the overall function of the system, the constant soiling effect must be cleaned at regular intervals.

**Keywords** Solar panel · Soiling effect · Temperature · I-V curve · PV curve · Natural dust

## 1 Introduction

Solar photovoltaic (PV) systems use solar cells to turn sunlight into electricity. Solar panels, a lithium-ion battery, a charge controller, and the electrical load make up the PV system. Solar PV panels are frequently installed on the rooftops of buildings [1–4]. An inverter connects the solar panels' direct current (DC) output to the building loads, converting the DC output into an alternating current (AC) to power the building. Monocrystalline silicon cell units, multi-crystalline silicon cell units, thick film silicon cells units, and amorphous silicon cells units are the most common types of solar PV cells. Solar power is being more widely used in recent years. In 2011, the US and Saudi Arabia erected a solar research facility near the hamlet of Al-Uyaynah. There was no electricity in the village, which was around 30 miles

---

A. M. Z. Khayyat · A. H. Alharthi · F. L. Almohammadi · Z. S. Almarwani · N. Kannan (✉) · Y. Mobarak

Electrical Engineering Department, Faculty of Engineering, Rabigh, King Abdulaziz University, Jeddah, Saudi Arabia  
e-mail: [nmajak@kau.edu.sa](mailto:nmajak@kau.edu.sa)

Y. Mobarak  
e-mail: [ysoliman@kau.edu.sa](mailto:ysoliman@kau.edu.sa)



**Fig. 1** Solar energy

northwest of Riyadh. The station is managed by the King Abdulaziz City for Science and Technology. An experimental assembly line for solar panels was built up by the agency. The solar cells were purchased from Taiwan, and the machinery for the production process came from Europe. Within a year, the line's capacity had tripled [5–8].

As depicted in Fig. 1, solar energy is solar radiation that can produce heat, trigger chemical processes, or generate electricity. The total amount of solar energy incidents on Earth considerably outnumbers the universe's present and future energy needs. If properly explored, this exceedingly spread power source can supply all future energy needs. Solar energy, in divergence from the finite conventional fuels such as coal blocks, oil, and gas, is expected to grow more appealing as a renewable energy source in the twenty-first century due to its endless supply and nonpolluting nature [9–12].

The Sun is a massive power source, and daylight is the most important power source that Earth receives. Its concentration, however, is rather low near the Earth's surface. This is mostly owing to the distant Sun's huge radial spread of radiation. Up to 54% of incoming sunlight is absorbed or scattered by the Earth's atmosphere and clouds, resulting in a very minor additional loss. Nearly half of visible light, half of the ultraviolet rays, and large quantities of infrared and other electromagnet-based radioactivity make up the sunlight that reaches the Earth.

## 2 Soiling Effects on PV Panels

In solar power systems, the soil is the deposition of particles on light-collecting surfaces. Because the deposited material either blocks or scatters incident light, electricity generation is diminished. Mineral partials, bird manures, yeasts, car exhaust

particles, and agricultural waste are all common soiling materials. Soiling affects traditional solar systems, focused photovoltaic systems, and intense solar (thermal) energy. Though, the belongings of soiling are more severe in intent schemes than in less concentrated systems. It is important to understand that “soiling” referred to both the procedure of accumulation and the substance that results, as depicted in Fig. 2.

There are several methods for reducing the impact of soiling. For solar power systems, the anti-soiling coating is a must-have feature. However, because of the lack of anti-soiling coatings, water cleaning has remained the most popular method. Soil erosion varies greatly between locations and even within areas. In rainy places, average soiling-induced power losses can be as low as 1%. In 2018, the worldwide average yearly energy loss due to soiling was projected to be between 5 and 10%. The loss of revenue from soiling is projected to be in the range of 3–5 billion euros.

Soiling is commonly caused by the deposition of airborne particles such as inorganic powder (Si and metal oxide salts), pollen, and smoke. Staining can be caused by a multitude of factors, including snow, ice, and frost, as well as industrial pollutants, sulfur acid particles, bird manures, dropping greenery, farming wastages, and the development of algae, yeasts, lichen, or bacteriological biofilms. The importance of each of these soiling devices is determined by their location.



**Fig. 2** Soiling on PV panels

Soiling either fully blocks or allows some light to pass through (hard shadow vs. soft shade) (soft shading). Soft shade scatters some of the transmitted light. Scattering distorts light, forcing rays to go in different directions. While diffuse light is sufficient for most photovoltaics, concentrated solar power and photovoltaics are more efficient. As a result, focused solar energy is additionally susceptible to contamination than traditional photovoltaic. Soiling-induced power loss in concentrated solar power is typically 8–14 times higher than in photovoltaics.

## 2.1 Working Principle of PV Cell

Solar panels are used to collect solar energy and convert it to electricity. They are made up of multiple layers and are meant to absorb solar light and generate electricity. As a result of the energy released by passing light striking the solar panel's surface, electrons are thrown out of their atomic orbits. The photovoltaic effect produces electric fields as a result of this process.

A solar energy generating system's fundamental structural unit is the solar cell, which captures electrical energy directly from light energy without any intermediate processes. A solar cell is sometimes known as a photovoltaic cell since it operates exclusively on its photovoltaic effect. A PV cell is a p–n junction semiconductor device. It is manufactured by fusing p-type (low electron concentration) and n-type semiconductor materials (high concentration of electrons). Extra electrons from the n-type attempt to diffuse to the p-side at the junction, and vice versa. Positive ion cores are disclosed in the n-side when electrons are transferred to the p-side, while –ve ions are revealed in the p-side when holes are transferred to the n-side. As a result, an electric field develops near the connection, causing the depletion zone to form. When photons with higher energy than the semiconductor's bandgap reach a solar cell, they are absorbed, resulting in the formation of an electron–hole (e–h) pair. Owing to the electrostatic force of the arena through the intersection, these e–h combines transfer to the n and p sides of the p–n junction, respectively. A potential difference between the two sides of the cell is established in this manner.

A solar cell usually features a –ve forward-facing interaction and a +ve backward contact. A p–n junction in a semiconductor material is comparable to a battery and sits between these two connections. If both sides are associated with an exterior circuit, the current will drift from the solar cell's positive to negative terminals.

## 2.2 Types of PV Panels

- **Monocrystalline solar panel**

Single-crystal panels are manufactured from a single silicon crystal and are also known as single-crystal panels. Their dark black appearance makes them easily

identifiable. They are the most durable solar panels due to their space-saving design. Single-crystalline solar panels are the most common name for single-crystalline solar cells. Their deep black color and sharp edges make them instantly recognizable.

Monocrystalline solar cells are the most space-efficient. In comparison to silicon solar cells, they take up the least amount of area. These solar cells are also extremely long-lasting, having a life expectancy of roughly half a century. They are also popular among homeowners who wish to extend the life of their warranties. Despite being universally regarded as the greatest system in its class, it comes at a very hefty cost. This is owing to a large amount of trash generated throughout the manufacturing process.

- **Polycrystalline solar panels**

These polycrystalline cells are made out of several silicon pieces. These cells have two silicon fragments instead of a single silicon core. This material is also less efficient than monocrystalline panels in terms of space and energy conversion. Furthermore, they are less heat resistant than monocrystalline panels.

- **Thin-film solar panel**

A light-absorbing layer 350 times thinner than that found in a normal silicon panel is used in thin-film solar cells. They are the lightest solar cells available due to their small form. Thin-film solar panels have a few advantages over regular solar panels, including lower installation costs and carbon offset.

For home solar installations, monocrystalline solar panels are the best option. They will pay a somewhat greater price, but they will get a system with a more discreet appearance that does not compromise performance or longevity. Furthermore, monocrystalline panels' high efficiency and power output ratings might result in greater cost savings over the system's lifetime. Monocrystalline solar panels are the most efficient type of solar panel, with efficiency ratings ranging from 15 to 20%. Monocrystalline panels are great for homes with limited roof space since they require fewer panels to generate the required electricity and have a higher efficiency rating. For the suggested model, a monocrystalline panel was chosen because of these benefits.

The performance of a solar cell (silicon) decreases as its temperature rises. As the temperature rises, the current decreases, lowering the cell voltage and lowering the productive capacity and efficiency. Solar cells have different effects on heat depending on the technology employed. Heat has less of an impact on thin films than it does on monocrystalline cells. Heat is a problem for semiconductors. The bandgap shrinks as the temperature rises, affecting the semiconductor's coefficients. The energy of electrons rises, but the energy required to break the bond between electrons and the nucleus falls. When the temperature rises, this causes an increase in current and a voltage drop.

### 3 Soiling Effect on PV Panels for the Real-Time I-V Curve

I-V characteristics of a solar cell curve are the intersection of the current and voltage curves of a PV cell diode in the nonappearance (dark) and existence (bright) of light. The standard “dark” currents in the diode rise when a cell is lit, changing the diode law to:

$$I = I_0 \left( e^{\left( \frac{qV}{nKT} \right)} - 1 \right) - I_L$$

where  $I_0$  = “dark saturation current.”

$q$  = the e charge.

$V$  = terminal voltage.

$n$  = ideal factor.

$k$  = Boltzmann constant.

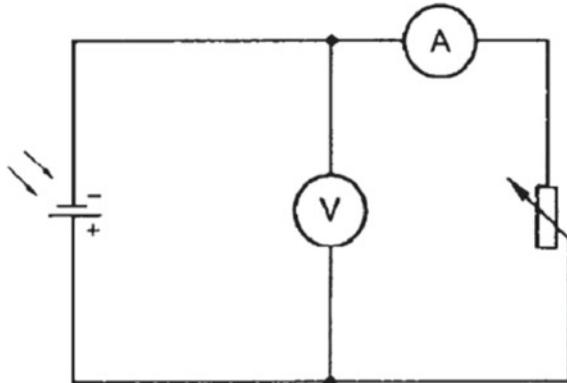
$T$  = temperature.

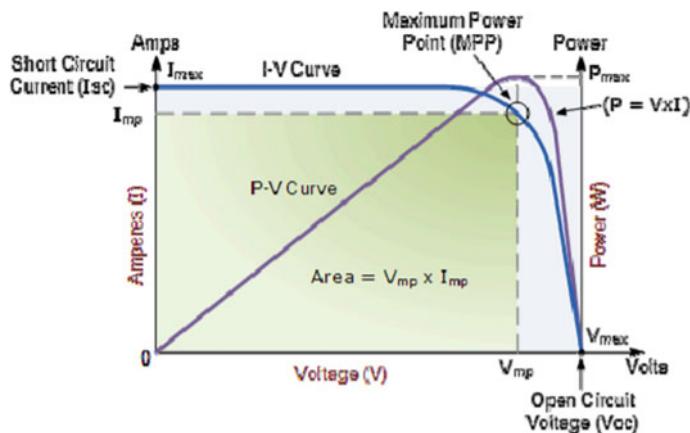
$I_L$  = light current.

A specific circuit for calculating I-V characteristics is displayed in Fig. 3. These parameters can be used to determine several solar cell metrics like short-circuit current (ISC), open-circuit voltage (VOC), fill factor (FF), and efficiency. The specific rating of a PV panel is influenced by these factors.

The SC current is the amps that flow via a PV cell when the voltage across it is no value and reached zero. Because of the creation and gathering of light-produced ions, this is the situation. For a perfect solar cell with a modest loss, the SC and current produced based on lights are the same. As a result, the SC current is the maximum current that can be pulled from the PV cell. The maximum voltage a solar cell can produce while no current is supplied is the open-circuit voltage. The forward bias on

**Fig. 3** Solar cell I-V characteristics circuit





**Fig. 4** Solar cell's typical I-V and power curves

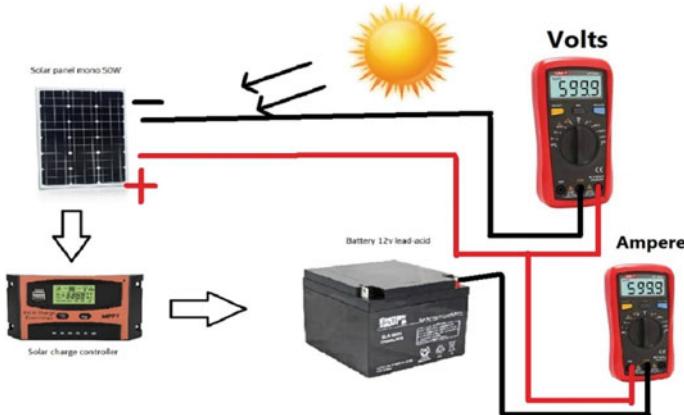
the PV cell caused by the bias of the PV cell intersection with the current produced by light determines the open-circuit voltage.

The “fill factor,” sometimes known as “FF,” is a parameter that, along with VOC and ISC, determines a solar cell’s maximum power output. The FF is calculated by dividing the maximum solar cell power by the product of Voc and Isc. The FF, as shown in Fig. 4, is a graphical depiction of the solar cell’s “squareness” as well as the size of the greatest rectangle that could fit in the I-V curve.

The most popular criterion for comparing the performance of solar cells is based mainly on efficiency. The efficiency of a PV cell is the ratio of energy generated to the energy received from the Sun. The band and strength of top angle positioned sunlight, the temperature of the PV cell, and the solar cell’s orientation all affect efficiency (Fig. 5).

Temperature and irradiance are two factors that affect the performance of solar PV modules. The OC voltage of solar modules differs with individual cell temperature. As the temperature increases because of outside changes or heat produced by power leaks during power production, the OC voltage (Voc) decreases rapidly. As a result, the output power is decreased. The solar panel temperature coefficient is computed by comparing the average PV module temperature in the module’s working environment to the STC information which is used to calculate the cell output in the design of a solar PV system. Irradiance has an impact on module performance as well, with a decrease in sunlight causing a drop in current and, as a result, a loss in power production. The performance of solar PV modules is influenced by a variety of parameters, including temperature and irradiance.

PV module open-circuit voltage changes with cell temperature. The open-circuit voltage (Voc) falls as the heat rises to owe to external modifications or temperature produced by local energy distribution during power generation production. The output power is reduced as a result. When building a solar PV system, the solar cell



**Fig. 5** Calculation of current and voltage in PV panel

temperature factor is calculated by comparing the desired optimum module temperature in the cell's working environment to the STC data used to compute the cell output. Irradiance has a similar effect on module performance, with a decrease in sunshine causing a decrease in current and, as a result, a loss in power output.

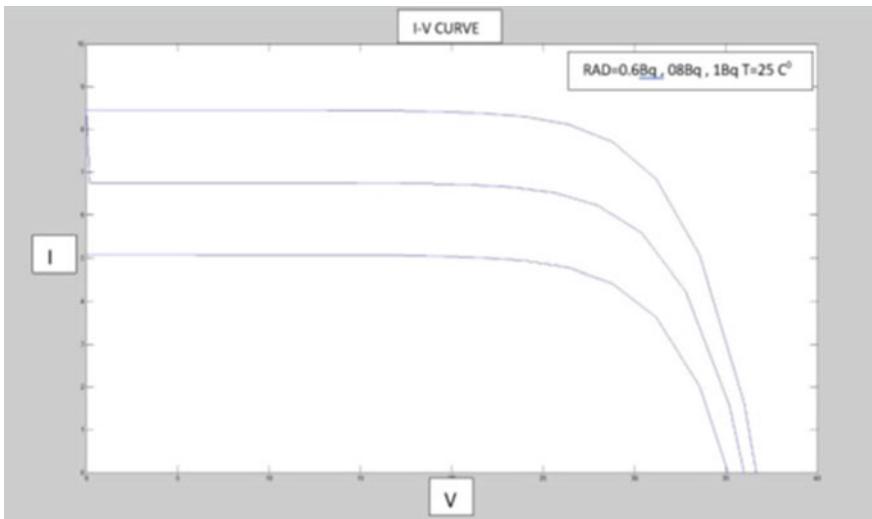
#### 4 Results and Discussions

The PV panel has been installed in Rabigh, Saudi Arabia, and performance analysis of the clean and dust panel has been analyzed. The current and voltage calculations of the PV panel have been measured, and the I-V curve and PV curve have been plotted using MATLAB as shown in Figs. 6, 7, 8, and 9. Since it is known that the temperature affects the performance of the solar panel output, the I-V and PV curve have been plotted for two different temperature such as 25 and 45 °C, respectively, as shown in Figs. 6, 7, 8, and 9.

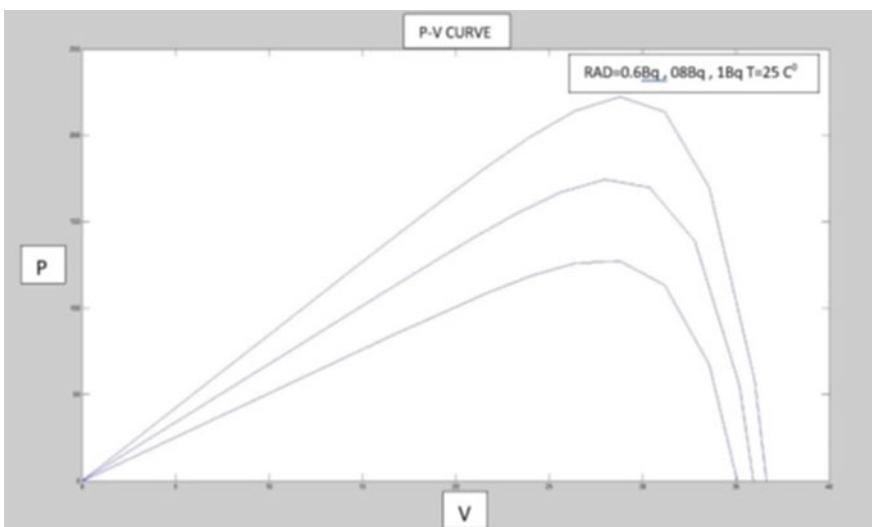
It is evident that as the temperature increases more than 25°, the output of the PV has declined compared to a significant value as shown in Fig. 10. This reading has been taken in spring and fall sessions, and it is expected that in summer, Saudi Arabia's average temperature is of very high value. So in summer, the power output of the panel will decrease significantly.

The output power of the panel at a different angle has been calculated, and it is proved that angle 45° is the best optimum angle for the installation of PV panel, which proves to have comparatively better power output compared to other angles as shown in Fig. 11.

Figure 11 shows the power output difference between clean and natural dust panels. Natural dust affects the performance of the panel. As the thickness of the accumulation of natural dust increases, there is a sharp decline in the performance

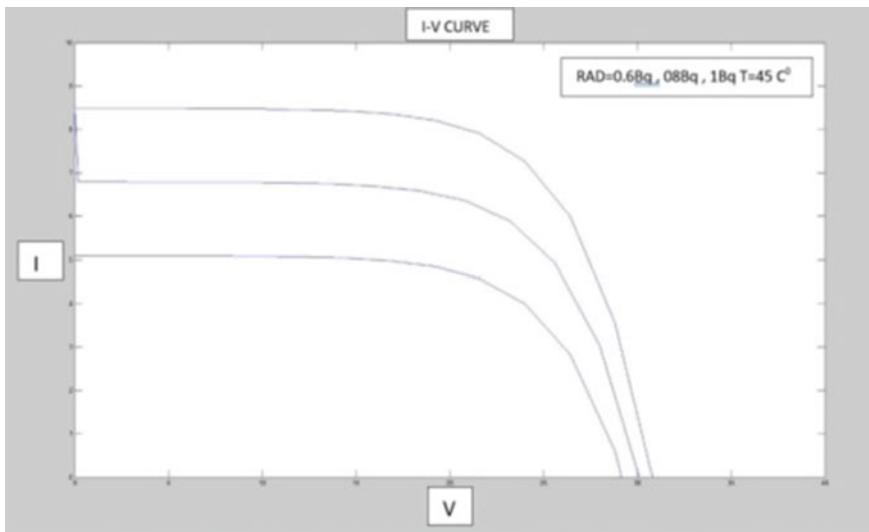


**Fig. 6** I-V curve  $T = 25\text{ }^{\circ}\text{C}$

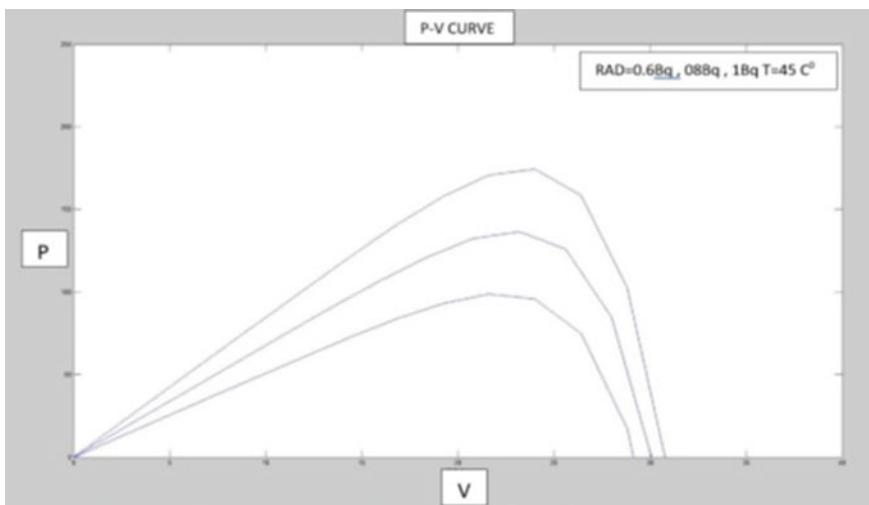


**Fig. 7** PV curve  $T = 25\text{ }^{\circ}\text{C}$

of the PV panel. This performance analysis indicates the PV panel needs to have a good cleaning technique at regular intervals and also it is also proved that angle, temperature, and irradiation affect the performance of the PV panel output.



**Fig. 8** I-V curve  $T = 45^\circ\text{C}$



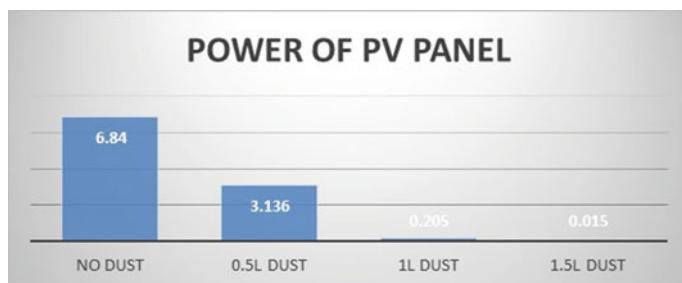
**Fig. 9** PV curve  $T = 45^\circ\text{C}$

## 5 Conclusion

The impact of the natural dust formed in the PV panel and the thickness of the dust has been measured, and the impression of soil on the performance of the PV panel has been analyzed. The dust drastically reduced irradiation that hits the panel, and it



**Fig. 10** Power output of the PV panels with different angles



**Fig. 11** Comparison of the performance of the clean and dust PV panel

is proved that the output of the PV panel reduced in a considerable factors such as temperature, panel angle is also considered, and the performance of the solar panels has been analyzed. It is clear that in Rabigh, the PV panel's dust must be removed at regular intervals by implementing a suitable cleaning mechanism. Each PV panel needs to be cleaned through dry, wet, or hybrid method by installing a simple cleaning device to maintain high power output.

## References

1. Mohamed, A., Hasan, A.: Effect of dust accumulation on performance of photovoltaic solar modules in Sahara environment. *J. Basic Appl. Sci. Res.* **2**, 11030–11036 (2012)
2. Mani, M., Pillai, R.: Impact of dust on solar Photovoltaic (PV) performance: research status, challenges, and recommendations. *Renew. Sustain. Energy Rev.* **14**, 3124–3313 (2010)

3. Etier, I., Tarabsheh, A.A., Kannan, N.: Shunt resistance spatial variations in amorphous silicon solar cells. *Microelectron. J.* **108**, art. no. 104960 (2021)
4. Sureshkumaar, G., Manoharan, S., Kannan, N., et al.: Double integral controller parameters for simulation analysis of the POESLLC insuperlift LUO converter. *Mater. Today: Proc.* (2021). <https://doi.org/10.1016/j.matpr.2021.01.467>
5. Etier, I., Al-Labadi, M.N., Jawarneh, A., Hammad, B., Kannan, N.: Real-time analysis model of single-action tracker, dual-axis solar tracker and fixed mounting of the photovoltaic system. *J. Green Eng.* **10**(11), 12439–12455 (2020)
6. VimalRaj, S., Suresh Kumar, G., Thomas, S., Kannan, N.: MATLAB/SIMULINK-based simulations on the state of charge on battery for electrical vehicles. *J. Green Eng.* **9**(2), 255–269 (2019)
7. Sureshkumaar, G., Kannan, N., Thomas, S.: MATLAB/SIMULINK based simulations of KY converter for PV panel's powered led lighting system. *Int. J. Power Electron. Drive Syst.* **10**(4), 1885–1893 (2019)
8. Etier, I., Anci Manon Mary, A., Kannan, N.: MATLAB based design and performance analysis of electronically commutated BLDC motor. *Indonesian J. Electr. Eng. Comput. Sci.* **24**(1), 22–28 (2021)
9. Sureshkumaar, G., Manoharan, S., Kannan, N., Etier, I., Mobarak, Y.A.: Double integral controller parameters for simulation analysis of the POESLLC in superlift LUO converter. *Mater. Today Proc.* (2021). <https://doi.org/10.1016/j.matpr.2021.01.467>
10. Nithiyanganthan, K., Loomba, A.K.: MATLAB/SIMULINK based speed control model for converter controlled DC drives. *Int. J. Eng. Model. Croatia, EUROPE* **24**(1–4), 49–55 (2011)
11. Jacob, D., Nithiyanganthan, K.: Smart and micro grid model for renewable energy based power system. *Int. J. Eng. Model. Croatia, EUROPE* **22**(1–4), 89–94 (2009)
12. Banerjee, A., et al.: Construction of effective wireless sensor network for smart communication using modified Ant Colony optimization technique. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems*, vol. 218. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-2164-2\\_22](https://doi.org/10.1007/978-981-16-2164-2_22)

# Machine Learning Models for the Electrical Power Generation by Savonius Vertical Axis Wind Turbine



Youssef Kassem , Hüseyin Çamur ,  
Ahmed Hamid Mohamed Abdalla Zakwan, and Nkanga Nkanga Amanam

**Abstract** This study evaluates the accuracy of feed-forward neural network (FFNN), cascade feed-forward neural network (CFNN), generalized regression neural network (GRNN), layer recurrent (LR), and radial basis function neural network (RBFNN) for estimating the electrical power of Savonius vertical axis wind turbine. To this aim, the electrical power (EP) of the proposed rotors was measured experimentally with various blade geometries and blade numbers. Various statistical indices were utilized to identify the best model for predicting the EP. The results demonstrated that all the proposed models are suitable for predicting the EP of developed rotors. Moreover, the findings indicated that the GRNN and LR performed better than other models.

**Keywords** Savonius wind turbine · Electrical power · FFNN · CFNN · GRNN · LR

## 1 Introduction

Due to the growth of populations and increasing the consumption of energy demand, renewable energy is an alternative solution for reducing the energy consumption demand and electricity cost [1–3]. According to Shahsavari and Akbari [4], renewable energy including solar and wind energy would help to reduce environmental impacts.

Wind energy is considered a clean source of energy and power generation source for solving the electricity crises [5, 6]. It can be converted to electricity using a

---

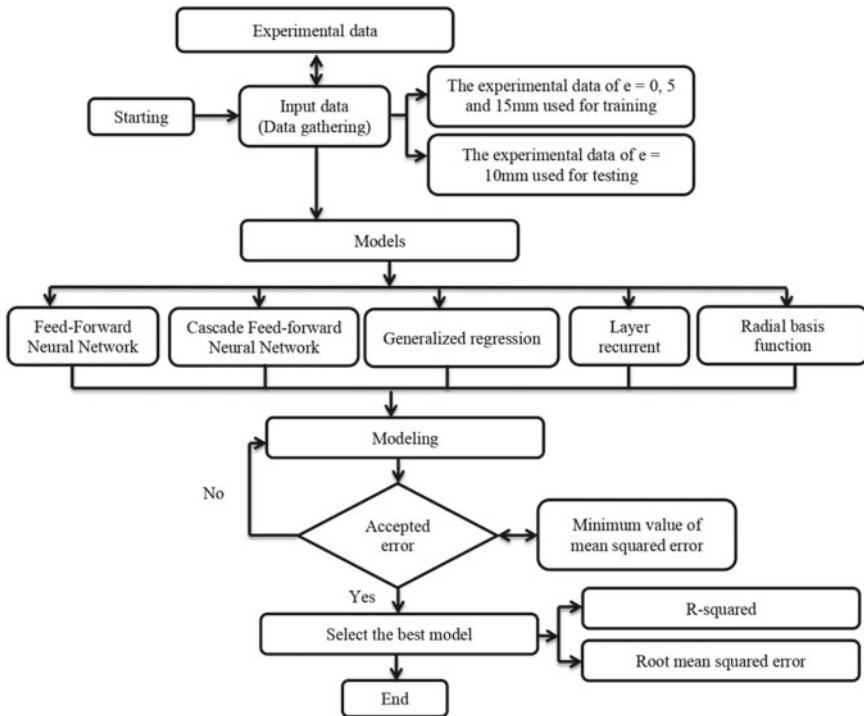
Y. Kassem · H. Çamur · A. H. M. A. Zakwan · N. N. Amanam  
Mechanical Engineering Department, Faculty of Engineering, Near East University, 99138  
Nicosia, North Cyprus  
e-mail: [youssef.kassem@neu.edu.tr](mailto:youssef.kassem@neu.edu.tr)

H. Çamur  
e-mail: [huseyin.camur@neu.edu.tr](mailto:huseyin.camur@neu.edu.tr)

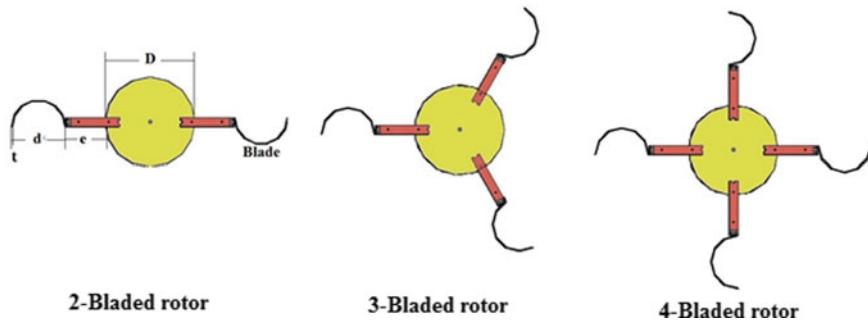
Y. Kassem  
Faculty of Civil and Environmental Engineering, Near East University, 99138 Nicosia, North Cyprus

wind turbine (WT). Wind turbines are categorized as horizontal axis wind turbines (HAWT) and vertical axis wind turbines (VAWT). Several studies have been investigated the performance of power generation of various types of WT particularly VAWT for use in urban areas [7, 8]. Recently, VAWTs particularly Savonius turbines (STs) have gained attention due to their advantages like operating at low wind speed in any wind direction [9].

Several scientific researchers [10–12] investigated the performance of these turbines. For instance, Kianfar et al. [10] experimentally and numerically investigated the effect of blade shape on the power coefficient of the rotor. Driss et al. [11] numerically examined the impact of arc angle on the performance of incurved Savonius wind rotor. Based on the previous studies, ST's performance depends on the rotor's geometries. Therefore, it needs to develop a model for estimating the ST's performance expressed by EP based on the geometry of the rotor. In this regard, the electrical power generation of the proposed rotors was estimated using FFNN, CFNN, RBFNN, GRNN, and LR models. Figure 1 illustrates the flowchart of the current study.



**Fig. 1** Flowchart of the procedure in the current study



**Fig. 2** 2D of the proposed rotors

## 2 Material and Methods

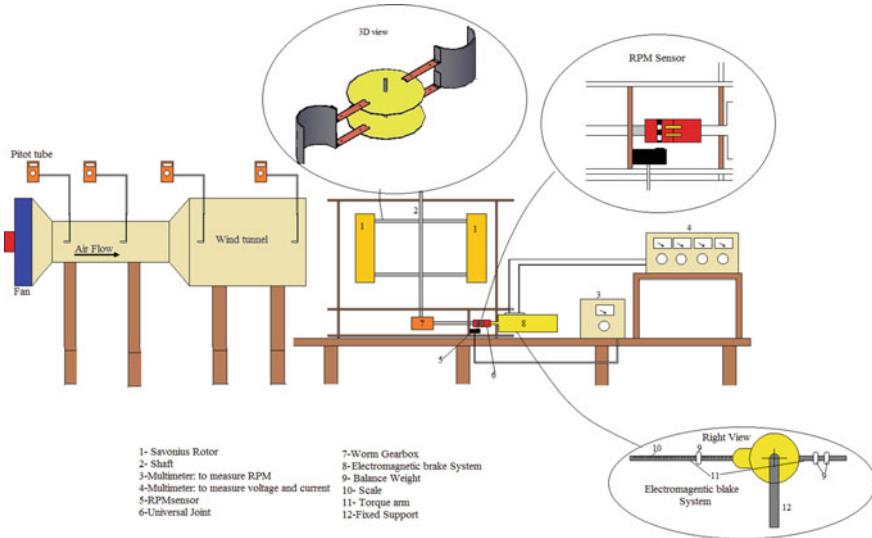
### 2.1 Experimental Model of Savonius Turbine and Apparatus

Figure 2 illustrates the two dimensions (2D) of the proposed rotor with various blade numbers (NB). In this study, the diameter of the rotor ( $d$ ) and external overlap ( $e$ ) are varied from 80 to 160 mm and 0 to 150 mm, respectively. Additionally, the rotor blade was made of PVC with a thickness of 3 mm. The rotor shaft was made of stainless steel with a diameter of 20 mm and a length of 700 mm. Moreover, two desks with a diameter ( $D$ ) of 300 mm and a thickness of 5 mm were made from fiberglass. These desks were placed on the top and bottom of the model. The performance of the Savonius turbine expressed by electrical power generation was measured using an electromechanical dynamometer as shown in Fig. 3. The description of the experimental study was given in Ref. [8].

Table 1 lists the statistical summary of the selected variables [ $e$ , NB,  $d$ ,  $H$ , wind speed (WS), revolution per minutes (RPM), mechanical power (MP), and electrical power (EP)].

### 2.2 ML Models

Machine learning (ML) is used as an alternative model to describe the complex relationship between the causal factors and responses [13]. Various ML models have been developed and utilized for different applications [13–15]. In this study, FFNN, CFNN, RBFNN, GRNN, and LR models are proposed to estimate the EP. The description of the developed models was given in Refs. [16–18]. In this work,  $e$ , NB,  $d$ ,  $H$ , WS, RPM, and MP are used as explanatory input variables. In this work, the training was done using data for  $e = 0, 5$ , and  $15$  mm, and the proposed model was utilized to estimate the EP for  $e = 10$  mm and then compared with the actual data.



**Fig. 3** Wind tunnel to measure the mechanical and electrical power for the proposed model

**Table 1** Statistical summary of used variables

Variable	Mean	Standard deviation	Minimum	Maximum
$e$	7.50	5.59	0.00	15.00
NB	3.00	0.82	2.00	4.00
$d$	11.33	3.40	8.00	16.00
$H$	40.00	8.17	30.00	50.00
WS	7.17	3.19	3.00	12.00
RPM	63.73	38.02	11.80	173.60
MP	2.32	2.81	0.01	12.24
EP	1.98	2.39	0.01	10.41

### 3 Results and Discussion

#### 3.1 Models

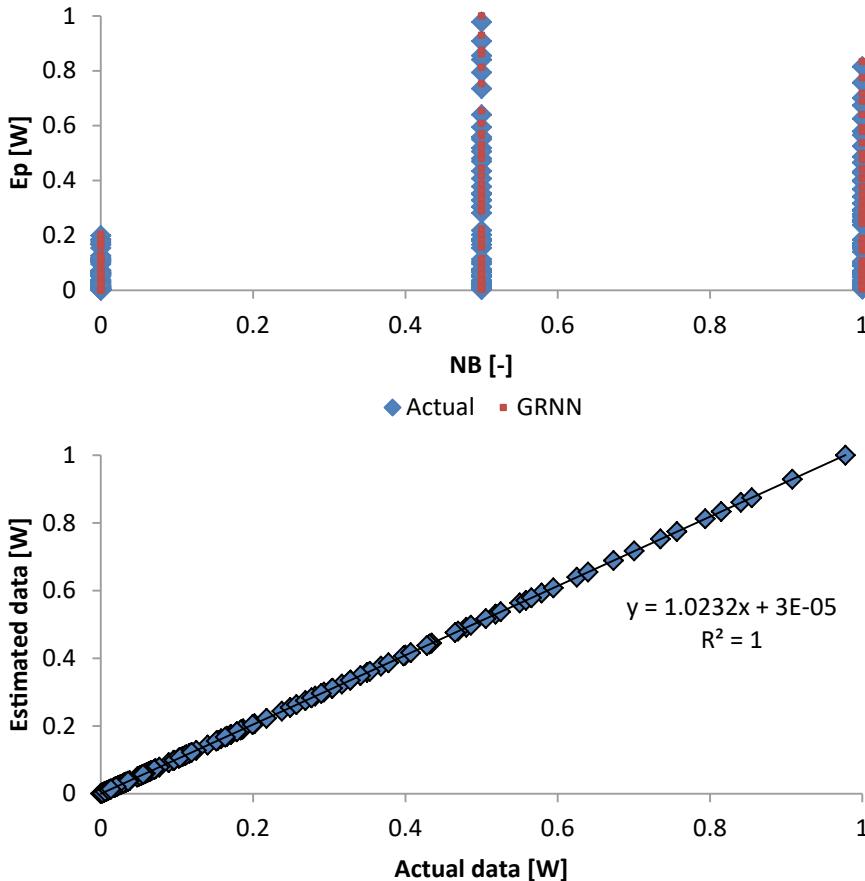
Aforementioned, five models were utilized to estimate the EP value of the proposed models using the experimental data. This study determines the best network configuration by trial and error method and selected based on the minimum value of mean squared error (MSE). In this study, TRAINLM was utilized as a training function.

Backpropagation is used to decrease in the gradient to minimize the MSE.

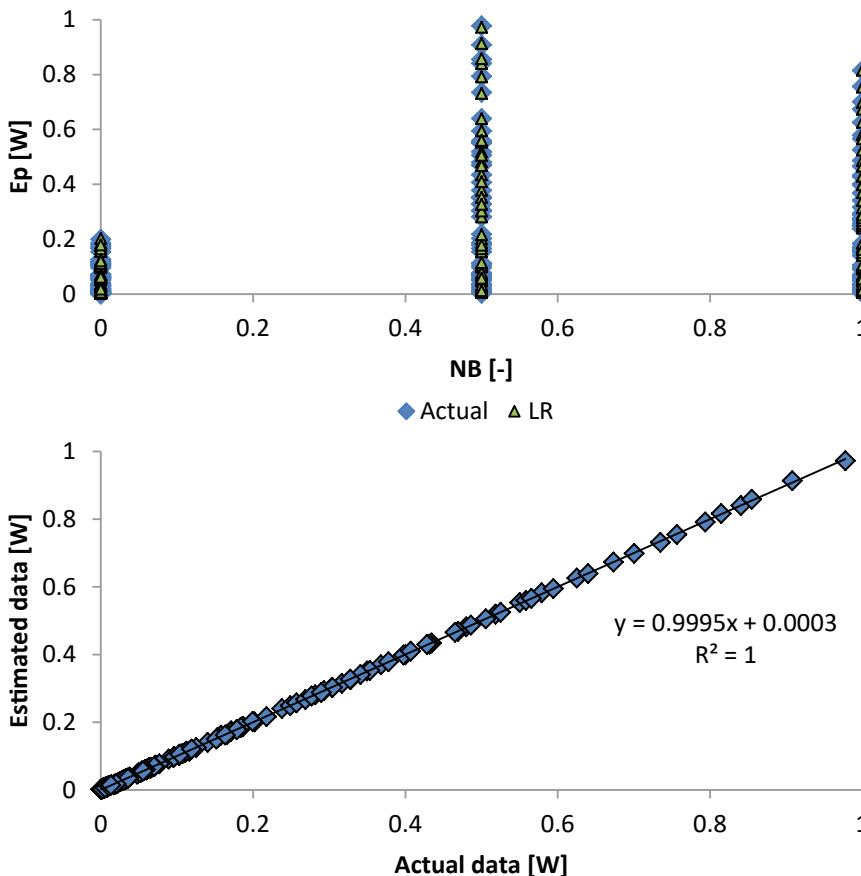
Based on the minimum value of MSE, it is found the following

- For FFNN, the value of MSE of  $5.327 \times 10^{-6}$  for the 7:1:1 with five neurons
- For CFNN, the value of MSE of  $3.826 \times 10^{-6}$  for the 7:1:1 with five neurons
- For RBFNN, the best spread and the maximum number of neurons were found to be 0.001 and 200, respectively.
- For GRNN, the best spread was found to be 0.002.

Figures 4, 5, 6, 7 and 8 compare the predicted data with the actual data of EP using the proposed machine learning approaches.



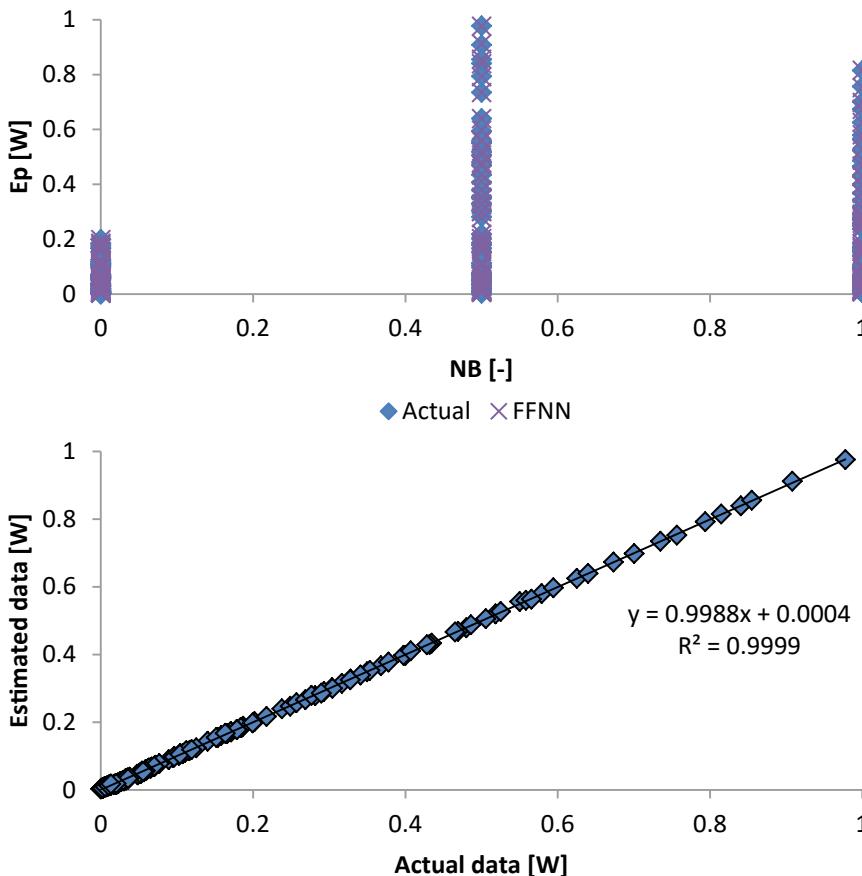
**Fig. 4** Comparison between measured and estimated data of EP using GRNN



**Fig. 5** Comparison between measured and estimated data of EP using LR

### 3.2 Performance Evaluation of Machine Learning Models, MLR, and QM

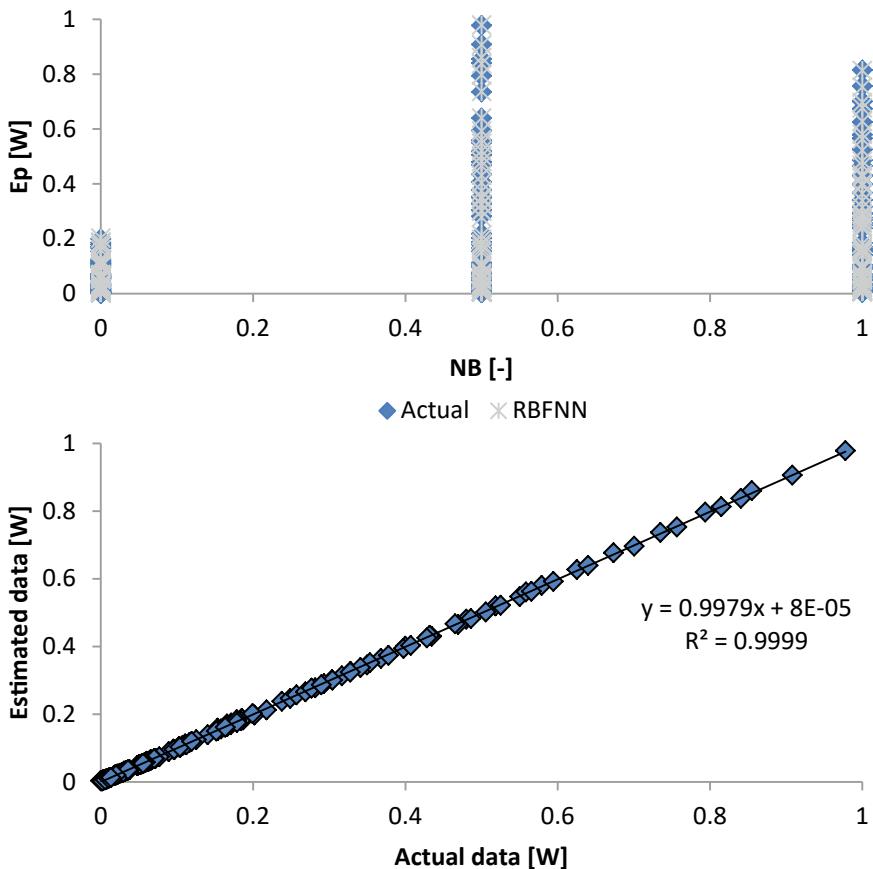
In this section, various statistical indices (R-squared and root mean squared error) were used to measure the performance of the proposed models. Table 2 lists the value of the used statistical indices. It is noticed that the maximum and minimum values of R-squared and RMSE were obtained from the GRNN model and LRR model, respectively.



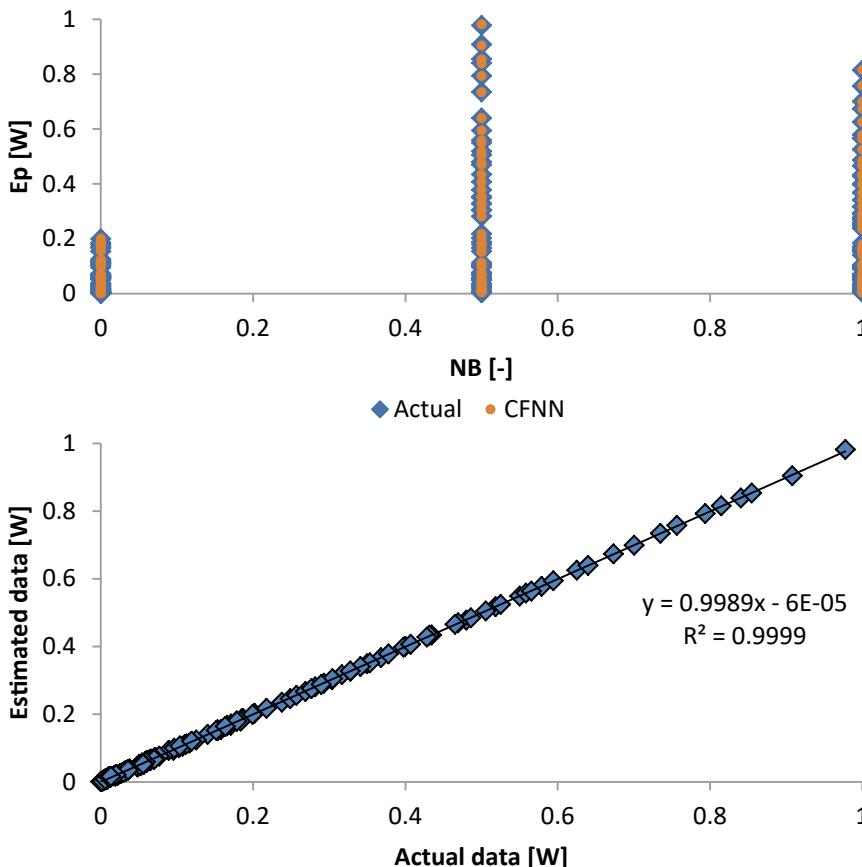
**Fig. 6** Comparison between measured and estimated data of EP using FFNN

#### 4 Conclusions

In this study, five ML models were proposed to estimate the EP of the Savonius rotor. In this work, external overlap, rotor diameter, blade height, wind speed, RPM, and mechanical power were utilized as input variables for the developed models. The results demonstrated that all developed models are applicable to estimate the EP of the rotor. Moreover, it is found that the GRNN and LR models have the highest accuracy among all models.



**Fig. 7** Comparison between measured and estimated data of EP using RBFNN



**Fig. 8** Comparison between measured and estimated data of EP using CFNN

**Table 2** Performance evaluation of the models

Statistical indicator	GRNN	LR	FFNN	RBFNN	CFNN
R-squared	1.00000	0.99995	0.99990	0.99991	0.99994
RMSE	0.00690	0.00157	0.00227	0.00222	0.00179

## References

1. Riahi, K., Van Vuuren, D.P., Kriegler, E., Edmonds, J., O’neill, B.C., Fujimori, S., et al.: The shared socioeconomic pathways and their energy, land use, and greenhouse gas emissions implications: an overview. *Glob. Environ. Change* **42**, 153–168 (2017)
2. VijayaVenkataRaman, S., Iniyar, S., Goic, R.: A review of climate change, mitigation, and adaptation. *Renew. Sustain. Energy Rev.* **16**(1), 878–897 (2012)
3. Arreynip, N.A., Joseph, E.: Small 500 kW onshore wind farm project in Kribi, Cameroon: sizing and checkers layout optimization model. *Energy Rep.* **4**, 528–535 (2018)

4. Shahsavari, A., Akbari, M.: Potential of solar energy in developing countries for reducing energy-related emissions. *Renew. Sustain. Energy Rev.* **90**, 275–291 (2018)
5. Alayat, M.M., Kassem, Y., Çamur, H.: Assessment of wind energy potential as a power generation source: a case study of eight selected locations in Northern Cyprus. *Energies* **11**(10), 2697 (2018)
6. Kassem, Y., Al Zoubi, R., Gökçekuş, H.: The possibility of generating electricity using small-scale wind turbines and solar photovoltaic systems for households in Northern Cyprus: a comparative study. *Environments* **6**(4), 47 (2019)
7. Loganathan, B., Mustary, I., Chowdhury, H., Alam, F.: Effect of sizing of a Savonius type vertical axis micro wind turbine. *Energy Procedia* **110**, 555–560 (2017)
8. Kassem, Y., Sefik, A., Çamur, H., Bahroun, A.A.: Experimental and numerical investigation of the influence of blade geometries and blade number on the performance of a newly developed Savonius-style wind rotor. *J. Eng. Appl. Sci.* **14**(24), 9788–9805 (2019)
9. Tahani, M., Rabbani, A., Kasaian, A., Mehrpooya, M., Mirhosseini, M.: Design and numerical investigation of Savonius wind turbine with discharge flow directing capability. *Energy* **130**, 327–338 (2017)
10. Kianifar, A., Anbarsooz, M., Javadi, M.: Blade curve influences on performance of Savonius rotors: experimental and numerical. In: Fluids Engineering Division Summer Meeting, vol. 49484, pp. 905–911 (2010)
11. Driss, Z., Mlayeh, O., Driss, S., Maaloul, M., Abid, M.S.: Study of the incidence angle effect on the aerodynamic structure characteristics of an incurved Savonius wind rotor placed in a wind tunnel. *Energy* **113**, 894–908 (2016)
12. Belkhier, Y., et al.: Adaptive linear feedback energy-based backstepping and PID control strategy for PMSG driven by a grid-connected wind turbine. In: Mekhilef, S., Favorskaya, M., Pandey, R.K., Shaw, R.N. (eds.) *Innovations in Electrical and Electronic Engineering. Lecture Notes in Electrical Engineering*, vol. 756. Springer, Singapore (2021). [http://doi.org/10.1007/978-981-16-0749-3\\_13](http://doi.org/10.1007/978-981-16-0749-3_13)
13. Kassem, Y., Gökçekuş, H., Alassi, E.: Identifying most influencing input parameters for predicting cereal production using an artificial neural network model. *Model. Earth Syst. Environ.* 1–14 (2021)
14. Kassem, Y., Çamur, H., Esenel, E.: Adaptive neuro-fuzzy inference system (ANFIS) and response surface methodology (RSM) prediction of biodiesel dynamic viscosity at 313 K. *Procedia Comput. Sci.* **120**, 521–528 (2017)
15. Kassem, Y., Çamur, H.: Prediction of biodiesel density for extended ranges of temperature and pressure using adaptive neuro-fuzzy inference system (ANFIS) and radial basis function (RBF). *Procedia Comput. Sci.* **120**, 311–316 (2017)
16. Kassem, Y., Gokcekus, H.: Do quadratic and Poisson regression models help to predict monthly rainfall? *Desalin. Water Treat.* **215**, 288–318 (2021)
17. Modaresi, F., Araghinejad, S., Ebrahimi, K.: A comparative assessment of artificial neural network, generalized regression neural network, least-square support vector regression, and K-nearest neighbor regression for monthly streamflow forecasting in linear and nonlinear conditions. *Water Resour. Manag.* **32**(1), 243–258 (2018)
18. Chu, Y., Fei, J., Hou, S.: Adaptive global sliding-mode control for dynamic systems using double hidden layer recurrent neural network structure. *IEEE Trans. Neural Netw. Learn. Syst.* **31**(4), 1297–1309 (2019)

# Performance Evaluation of SIMON and SPECK Block Ciphers to Secure IoT-Enabled Smart Cities



Monika Jangra and Buddha Singh

**Abstract** The Internet of things (IoT) has presented a smart world by interconnecting everyday objects to enable a superior everyday smart life. The IoT-enabled smart cities and smart mobility brought a completely revolutionary idea but with a compromise in the security. The IoT devices that offer smart applications like smart cities and smart mobility are generally resource-constraint and battery-operated. Raspberry Pi is a device widely used for such applications. SIMON and SPECK are the families of lightweight block ciphers that can offer encryption facilities to smart city applications. The paper presents the performance evaluation of the SIMON and SPECK family of lightweight block ciphers on the Raspberry Pi device. A trade-off between security and performance is observed between SIMON and SPECK implementations. The simulation results show that SPECK offers a good security level with better performance in Raspberry Pi hardware.

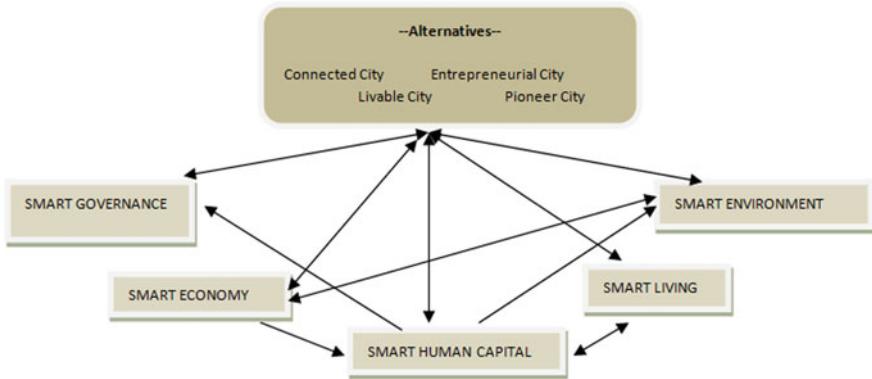
**Keywords** Internet of things · Smart cities · Security · Lightweight cryptography · Block ciphers

## 1 Introduction

The rapid development in the area of information and communication technologies (ICT) has taken the world to the smart world stage. The day-to-day things have transformed into smart things with the incorporation of various existing technologies from disciplines of sensing technology, communication technology, identification technology, etc. The collaborative network of smart things—famously known as the Internet of Things (IoT) [1]—satisfies end-users with smart needs. This intelligent growth is motivating personals from different areas to look out for the scope of IoT-enabled services in their areas too. One such area is IoT-enabled smart cities and smart mobility [1–5]. Figure 1 depicts an instance from the smart city with facilities for smart civil society [3]. Smart cities have changed the vision of living

---

M. Jangra (✉) · B. Singh  
Jawaharlal Nehru University, New Delhi, India  
e-mail: monikajangra0701@gmail.com



**Fig. 1** A smart civil society sub-network of a smart city

but this is offered with some serious security threats like a data breach and data exfiltration [3–6]. Traditional cryptography was strong enough to provide security for many applications in the desktop computing era. The methods from the traditional cryptography area are highly complex and resource-consuming that was easy to employ on high-end systems. The need for high resource usage makes them infeasible to employ for IoT-enabled smart cities.

The demand for security for lightweight applications gave birth to the lightweight cryptography area. The efforts of cryptographers have nurtured this area with many mechanisms that are well-suited for low-resource environments. PRESENT, CLEFIA, HIGHT, PICCOLO, ZORRO are some examples but they support a limited range of platforms [7–18]. SIMON and SPECK [18] were developed to support a wide range of platforms with a good level of performance in both hardware and software implementations. SIMON and SPECK are a family of lightweight block ciphers, developed by the US National Security Agency to provide security for resource-constrained environments and IoT applications like smart cities [18]. These were accepted by NIST Lightweight Cryptography Workshop in July 2015. The pervasive computing for smart cities is growing and rapidly changing, which inspired authors to develop efficient and flexible block ciphers for resource-limited applications. SIMON and SPECK are composed of simple operations that are flexible with any kind of hardware and software platform; i.e., they provide encryption facilities in an efficient and generalized way. Raspberry Pi devices are the resource constraint devices that are set up to operate on battery powers to deploy smart environments and applications. The SIMON and SPECK ciphers are lightweight ciphers used for encryption.

The authors state that SIMON and SPECK have good diffusion strength without using S-box for encryption. In this paper, the fact is empirically studied using the Avalanche effect metric. Later, in the paper, the ciphers are simulated on the Raspberry Pi device to evaluate their actual resource usage for IoT applications like smart cities and smart mobility. Paper organization is as follows: Sect. 2 includes the related

study to describe the basic structure of lightweight block cipher families SIMON and SPECK, Sect. 3 is the empirical study of the security strength and performance analysis of the cipher and includes the simulation results, Sect. 4 concludes the paper with Sect. 5 presents some of the future scopes.

## 2 Related Study: SIMON and SPECK

SIMON and SPECK are Feistal network-based block ciphers. They provide a wide range of parameters for encryption, i.e., from tiny block size to large block size and from low-level-security to high-level-security. Table 1 provides the parameter details of the SIMON and SPECK ciphers.

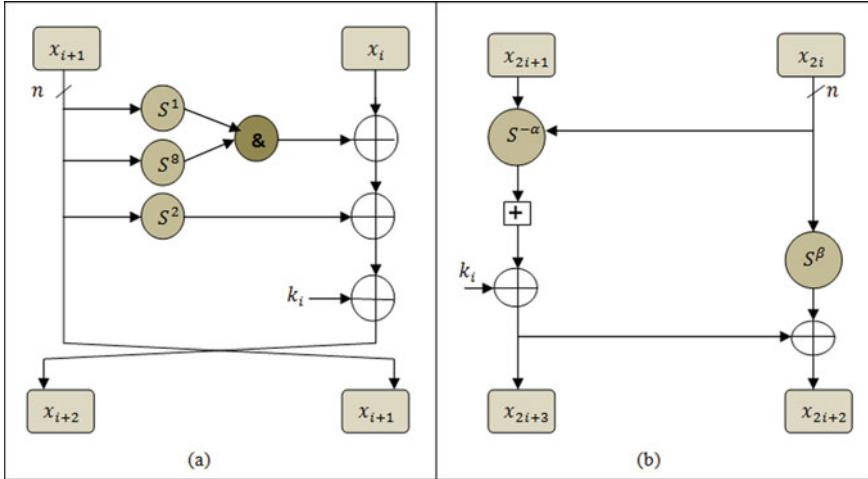
In this paper, we have given summarized implementation details of both block ciphers. The summarization includes an explanation of simple operations, round functions, and key schedules. For a detailed explanation of SIMON and SPECK lightweight ciphers, one can refer to work by Ray Beaulieu and co-authors [1].

### 2.1 SIMON and SPECK Operations

SIMON and SPECK operations can be broken into a collection of simple operations that are easily applicable to every platform and favors both hardware and software implementations. The simple operations used within SIMON and SPECK are Modular Addition and Subtraction, Bitwise XOR, Bitwise AND, Left Circular Shift, and Right Circular Shift. SIMON and SPECK get their nonlinearity from bitwise AND and modular addition, respectively, that have an efficient bias toward both software and hardware implementations. The rotation operations are done in the power of 8, i.e., byte shift or byte swap only so that rotation operations correspond to register to relabel only.

**Table 1** SIMON and SPECK parameters

Block size (in bits)	Key size (in bits)
32	64
48	72, 96
64	96, 128
96	96, 144
128	128, 192, 256



**Fig. 2** Feistel structure of **a** SIMON and **b** SPECK block ciphers

## 2.2 SIMON and SPECK Round Functions

SIMON and SPECK are two-stage Feistel structures. The notation for a SIMON and SPECK family member is like SIMON $2n$ , where  $2n$  represents block size and has a key size of  $n$  bits, a similar notation exists for the SPECK family also. Equation 1 represents the round function for SIMON.

$$R_k(x, y) = (y \oplus f(x) \oplus x) \quad (1)$$

where  $f(x) = (S_x \text{ and } S_x^8) \oplus S_x^2$ ;  $x$  and  $y$  are two intermediate ciphertexts of size  $n$ .  $k$  is the round key of size  $n$ .

Equation (2) depicts the round function for SPECK

$$R_k(x, y) = ((S^{-\alpha}x + y) \oplus k, S^{-\beta}y \oplus (S^{-\alpha}x + y) \oplus k) \quad (2)$$

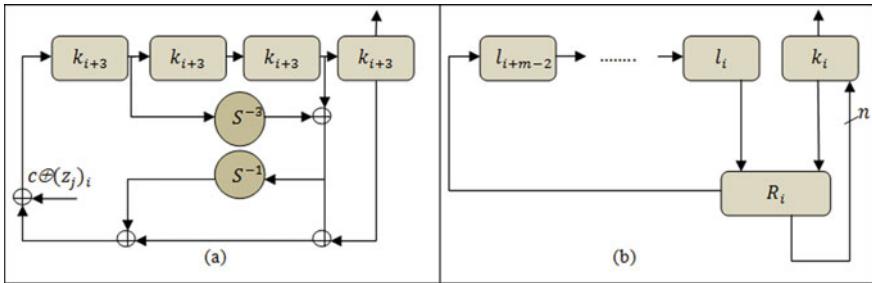
where  $\beta = 2$  and  $\alpha = 7$  for  $n = 16$ ;  $\beta = 3$  and  $\alpha = 8$  otherwise (Fig. 2).

## 2.3 Key Schedules

Simpler key schedules can be developed, but complex methods are intentionally used to avoid key-related issues. The SIMON key schedules are given in Eqs. (3) to (5).

$k_0, k_1, \dots, k_{m-1}$  are the keywords, where  $m \in \{2, 3, 4\}$

$$k_{i+2} = k_i \oplus (I \oplus S^{-1})S^{-3}k_{i+1} \oplus C_i \quad (3)$$



**Fig. 3** Key expansion schedules of **a** SIMON and **b** SPECK family of block ciphers

$$k_{i+3} = k_i \oplus (I \oplus S^{-1})S^{-3}k_{i+2} \oplus D_i \quad (4)$$

$$k_{i+4} = k_i \oplus (I \oplus S^{-1})(S^{-3}k_{i+3} \oplus k_{i+1}) \oplus E_i \quad (5)$$

where  $I$  is the identity matrix of size  $n \times n$ . The value of  $C_i$ ,  $D_i$ , and  $E_i$  depends on the value of  $m$ .

SPECK key schedule depends on its round function. The key schedule methods for SPECK are given in Eqs. (6) and (7).

$$l_{i+m-1} = (k_i + S^{-\alpha}l_i) \oplus i \quad (6)$$

$$k_{i+1} = (S^\beta k_i \oplus l_{i+m-1}) \quad (7)$$

where  $(l_{m-2}, \dots, l_0, k_0)$  are keywords. Figure 3 depicts the key expansion schemes used by SIMON and SPECK.

The research work [14, 18–21] includes the cryptanalysis of SIMON and SPECK. The analysis includes linear cryptanalysis, differential cryptanalysis, reduced round differential cryptanalysis, robustness testing and evaluations as per ISO certifications.

### 3 Experimental Results and Discussion

A good block cipher is one that provides strong ciphers, and at the same time, they have an efficient performance in both software and hardware platforms. But in reality, cryptographers have to deal with the trade-off of performances and strength in the real world. The security mechanisms should be designed as per the desired level of security and performance. If the application does not demand a high level of security then, simple methods are favored for crypt facilities; on the other hand, if the application is security-sensitive, then cryptographers compromise the performance in systems and resources they have. For example, a security method may have to round

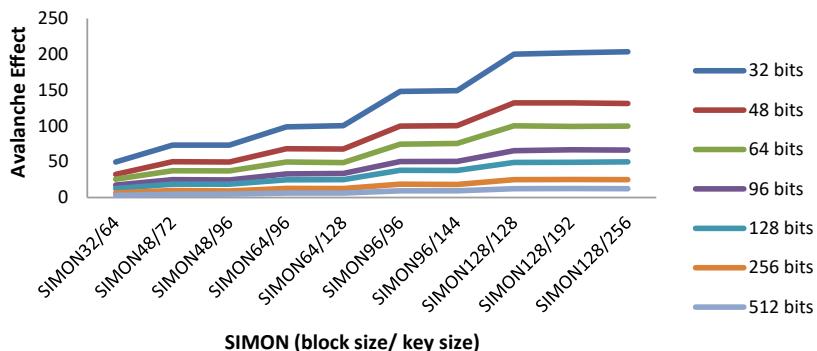
itself an absurd number of times for a secure ciphertext; which in turn can consume a lot of time, chip size, and other computing resources. This section involves the experimental results and discussions to evaluate the performance and security of the SIMON and SPECK family of block ciphers.

### 3.1 Security Analysis

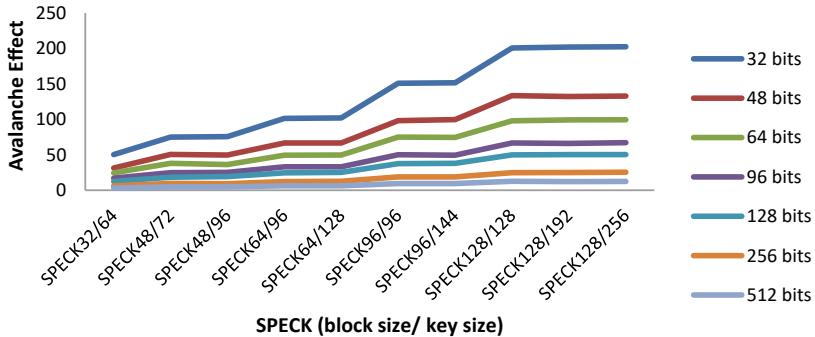
Most of the lightweight block ciphers employ the S-box for the differential property. S-box is the most optimal feature to increase the strength of the ciphers. The dependence of algorithms on S-box precludes the efficiency of serialization and algorithms that are not serializable have fewer opportunities for optimization [14]. SIMON and SPECK have left out the S-box feature. The independence of SIMON and SPECK ciphers from S-box makes them simpler and saves a lot of chip area. The experimental study presented in the paper empirically proves the differential strength of the block ciphers using the Avalanche Effect metric. The metric is the percentage change in the ciphertext bits if the plain text is changed by one bit only.

Figures 4 and 5 are the security performance of the SIMON and SPECK family of block ciphers for different sizes of plaintexts, respectively. The simulation patterns of SIMON and SPECK are similar, and some of the common observations are as follows.

1. For a constant *block\_size* and *key\_size*, as the size of plaintext data grows the security strength declines. The growth of the *plaintext\_size* increases number of blocks and the single bit change corresponds to the change in the single ciphertext block only, therefore resulting in the lower Avalanche effect.
2. For a constant *plaintext\_size* and *key\_size*, as the *block\_size* increases the security increases because the change is distributed over a larger number of bits. It is evident that security is maximum for block sizes 128 bit.



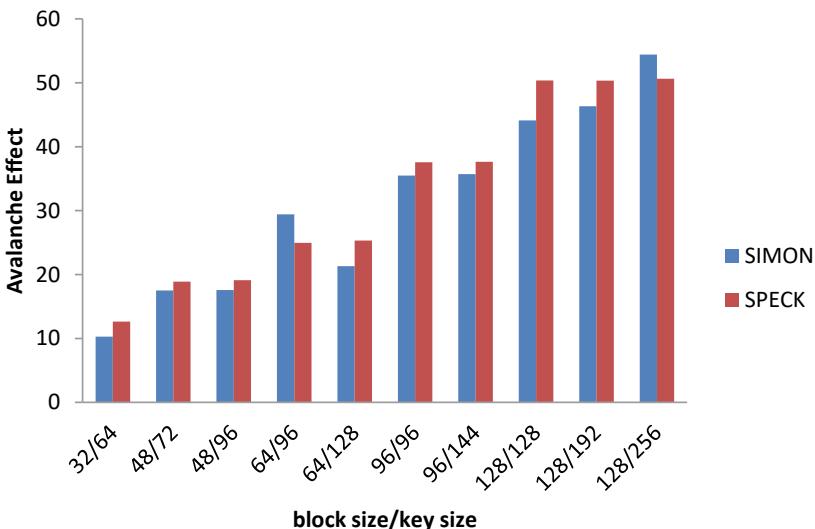
**Fig. 4** Avalanche effect of SIMON family of block ciphers with varying plaintext size



**Fig. 5** Avalanche effect of SPECK family of block ciphers with varying plaintext size

3. For a constant *block\_size* and *plaintext\_size*, larger the *key\_size* greater the security level offered. Since *key\_size* determines the number of linear equations required to mount linear attacks.
4. For, *plaintext\_size* < *block\_size*, the curves in Figs. 4 and 5 show more than 100% avalanche effect value because of additional data padding as the results have been evaluated according to *block\_size*, not *plaintext\_size*.

The optimal criteria of Avalanche effect for an optimal diffusion strength require the metric to reach at least 50%. From Fig. 6, SIMON and SPECK reach the 50% criteria with the larger *key\_size* and *block\_size* to attain the optimal value. The variants SIMON 128/256 and SPECK 128/128 for SIMON and SPECK, respectively,



**Fig. 6** Security comparison of SIMON and SPECK block ciphers family

**Table 2** System configuration

Raspberry Pi	
Model	Pi3 B+
Processor	Cortex-A53(ARMv8) 64-bit SoC@1.4 GHz
RAM	1 GB LPDDR2 SDRAM
Operating system	Raspbian OS (32-bit)

reach the optimal value ( $\geq 50\%$ ). In most of the variants, SPECK is outperforming in terms of security strength.

### 3.2 Performance Analysis

The authors of SIMON and SPECK have avoided the use of general bit permutation techniques claiming them to be less effective than simple circular shift permutations on different platforms. Moreover, bit permutation includes expensive data transpose operations and has larger code sizes. The use of simple circular shift operations is efficient but at the same, these are the reason for lower diffusion rates. Authors have tried to compensate for these effects by factoring a large number of rounds with good software performance. To evaluate performance for resource-limited operations, the ciphers are simulated on the Raspberry Pi device. The device configuration details are given in Table 2.

Table 3 is the comparative analysis results of the SIMON and SPECK family of block ciphers. From the table, it is observed that SIMON and SPECK both offer a satisfactory throughput on Raspberry Pi device but SPECK has almost double

**Table 3** Performance metrics for ECB mode

Size	Encryption time (ms)		Throughput (kbps)	
	SIMON	SPECK	SIMON	SPECK
32/64	52.60	33.33	0.47	0.75
48/72	38.50	22.22	0.64	1.12
48/96	38.50	22.93	0.64	1.09
64/96	33.81	19.61	0.73	1.27
64/128	35.27	19.78	0.70	1.26
96/96	28.29	14.71	0.88	1.69
96/144	28.87	14.35	0.86	1.74
128/128	27.78	11.95	0.89	2.09
128/192	27.81	12.29	0.89	2.03
128/256	29.19	12.84	0.85	1.94

**Table 4** Memory requirements of SIMON and SPECK

SIMON	SPECK	ISO/IEC requirement
14 kb	11 kb	<32 kb

throughput as compared to SIMON ciphers. SIMON uses the bitwise AND operation while SPECK uses the addition modulo that is an efficient choice for software implementations which makes SPECK perform well than SIMON.

As far as memory consumption is observed, the memory footprints of software implementations of SIMON and SPECK fall under the category of lightweight block ciphers. The details are given in Table 4.

## 4 Conclusion

The paper evaluates the performance of SIMON and SPECK lightweight block ciphers used for the encryption of data in resource constraint environments. The block ciphers are simulated on the Raspberry Pi model3 B device. The simulation results show that SPECK offers a good level of security with almost double throughput as compared to SIMON. Higher throughput by SPECK indicates the lower energy to be consumed on the IoT devices. SIMON and SPECK maintain the Security VS performance tradeoff quite well. Smart city stakeholders can use any of the methods for the encryption of data in communication in different modules as per security and performance requirements.

## 5 Future Scope

Raspberry Pi is a widely used IoT device for smart cities applications. The block ciphers can be simulated on the other frequently used IoT edge devices to generalize the software and hardware efficiency. These block ciphers can be tested for their performance with other modes of encryption also [22]. The bitwise operations like bit permutations can be utilized for security improvement without compromising with resource utilization levels [23].

## References

1. Atzori, L., Iera, A., Morabito, G.: The internet of things: a survey. *Comput. Netw.* **54**(15), 2787–2805 (2010)
2. Su, K., Li, J., Fu, H.: Smart city and the applications. In: 2011 International Conference on Electronics, Communications and Control (ICECC), pp. 1028–1031. IEEE (2011)

3. Lombardi, P., Giordano, S., Farouh, H., Yousef, W.: Modelling the smart city performance. *Innov. Eur. J. Soc. Sci. Res.* **25**(2), 137–149 (2012)
4. Bakici, T., Almirall, E., Wareham, J.: A smart city initiative: the case of Barcelona. *J. Knowl. Econ.* **4**(2), 135–148 (2013)
5. Hall, R.E., Bowerman, B., Braverman, J., Taylor, J., Todosow, H., Von Wimmersperg, U.: The vision of a smart city (No. BNL-67902; 04042). Brookhaven National Lab, Upton, NY (US) (2000)
6. Kalinin, M., Krundyshev, V., Zegzhda, P.: Cybersecurity risk assessment in smart city infrastructures. *Machines* **9**(4), 78 (2021)
7. McKay, K., Bassham, L., Sönmez Turan, M., Mouha, N.: Report on lightweight cryptography (No. NIST Internal or Interagency Report (NISTIR) 8114 (Draft)). National Institute of Standards and Technology (2016)
8. Miller, F., Vandome, A., McBrewster, J.: Advanced Encryption Standard (2009)
9. Duan, L., Liu, D., Zhang, Y., Chen, S., Liu, R.P., Cheng, B., Chen, J.: Secure data-centric access control for smart grid services based on publish/subscribe systems. *ACM Transaction on Internet Technology* (2016)
10. Saeed, A., Ahmadina, A., Javed, A., Larijani, H.: Random neural network based intelligent intrusion detection and prevention mechanism for IoT applications. *ACM Transaction on Internet Technology* (2016)
11. Sparrow, R.D., Adekunle, A.A., Berry, R.J.: LEOPARD: lightweight encryption operation permutation addition rotation and diffusion. In: 2016 10th International Conference on Signal Processing and Communication Systems (ICSPCS), pp. 1–5. IEEE (2016)
12. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J., et al.: PRESENT: an ultra-lightweight block cipher. In: International Workshop on Cryptographic Hardware and Embedded Systems, pp. 450–466. Springer, Berlin (2007)
13. Akishita, T., Hiwatari, H.: Compact hardware implementations of the 128-bit blockcipher CLEFIA. In: Proceedings of Symposium on Cryptography and Information Security—SCIS, Japan (2011)
14. Beaulieu, R., Treatman-Clark, S., Shors, D., Weeks, B., Smith, J., Wingers, L.: The SIMON and SPECK lightweight block ciphers. In: 2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 1–6. IEEE (2015)
15. Bansod, G., Raval, N., Pisharoty, N.: Implementation of a new lightweight encryption design for embedded security. *IEEE Trans. Inf. Forensics Secur.* **10**(1), 142–151 (2014)
16. Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.: The LED block cipher. In: Cryptographic Hardware and Embedded Systems (Lecture Notes in Computer Science), vol. 6917, pp. 326–341. Springer, Berlin, Germany (2011)
17. ISO: <https://www.iso.org/standard/78477.html>. Last accessed 2017/11/21
18. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., et al.: SIMON and SPECK: block ciphers for the internet of things. *IACR Cryptology ePrint Archive*, p. 585 (2015)
19. Ashur, T., Luykx, A.: An account of the ISO/IEC standardization of the Simon and Speck block cipher families. In: Security of Ubiquitous Computing Systems, pp. 63–78. Springer, Cham (2021)
20. Susanti, B.H., Permana, O.J.: Robustness test of SIMON-32, SPECK-32, and SIMECK-32 algorithms using fixed-point attacks. *J. Phys. Conf. Ser.* **1836**(1), 012006 (2021)
21. Rajawat, A.S., et al.: Blockchain-based model for expanding IoT device data security. In: Bansal, J.C., Fung, L.C.C., Simic, M., Ghosh, A. (eds.) *Advances in Applications of Data-Driven Computing. Advances in Intelligent Systems and Computing*, vol. 1319. Springer, Singapore (2021). [http://doi.org/10.1007/978-981-33-6919-1\\_5](http://doi.org/10.1007/978-981-33-6919-1_5)

22. Stallings, W.: Cryptography and Network Security, 4/E. Pearson Education India (2006)
23. Shi, Z., Lee, R.B.: Bit permutation instructions for accelerating software cryptography. In: Proceedings IEEE International Conference on Application-Specific Systems, Architectures, and Processors, pp. 138–148. IEEE (2000)

# Human Activity Recognition for Disease Detection Using Machine Learning Techniques—A Comparative Study



Arpan Adhikary, Koushik Majumder, Santanu Chatterjee,  
Rabindra Nath Shaw, and Ankush Ghosh

**Abstract** Human activity test is now rapidly adopted for disease detection. Many diseases can be detected using different human activities like walking, running, standing or performing any tasks. Parkinson's disease (PD), schizophrenia, mood disorder, chronic obstructive pulmonary disorder (COPD), peptic ulcer and asthma are some of those. These diseases can be sometimes incurable if treatment is not done at early stages. But, clinical tests and diagnosis are time-consuming. Machine learning gives the opportunity to predict and detect the diseases in less time. So, it is now becoming an important area of research. Some works have already been done by the researchers in the area of early detection and prediction of the above-mentioned diseases using machine learning (ML) and deep learning (DL) techniques. In this paper, we have carried out a detailed survey of different works in this field. We identified the weaknesses of the existing works. Based on these, we tried to find the future scope of the works to enhance their performance in the field of disease detection using human activity.

**Keywords** Human activity recognition (HAR) · Parkinson's disease (PD) · Chronic obstructive pulmonary disorder (COPD) · Machine learning (ML) · Deep learning (DL)

## 1 Introduction

Parkinson's disease (PD) is a common nerve and brain disorder. It generally affects the people above 60 years. But, people of different ages may suffer from PD depending upon some external scenarios. It may be genetic, or if someone is exposed to medicines or pesticides i.e. reserpine, phenothiazine, etc., for a long period, then

---

A. Adhikary · K. Majumder · S. Chatterjee

Department of Computer Science and Engineering, Maulana Abul Kalam Azad University of Technology, Kolkata, West Bengal, India

R. N. Shaw · A. Ghosh (✉)

University Center for Research & Development (UCRD), Chandigarh University, Mohali, Punjab, India

e-mail: [ankushghosh@gmail.com](mailto:ankushghosh@gmail.com)

also PD may arise. People affected by PD have some symptoms like slowness of movement, tremoring, rigidity, speech disorder, etc. Based on these symptoms, we can classify PD patients from healthy patients. Before PD symptoms appear, some patients may have pre-motor stages which can last from 5 to 20 years. In that time, patients may have sleep behaviour disorder, olfactory loss, rapid eye movement, decreased sense of smell, gastrointestinal problems, etc. It is not fully curable and no proper medicine or surgery is developed that can prevent this disease. But using some medicines, PD's impact can be reduced. COPD is a life-threatening disease that affect the lung and has a bad impact on the respiratory system. It is one of the major causes of morbidity and mortality across the globe. This is now the fourth major cause of death worldwide. COPD mainly occurs if the subject is exposure to tobacco or chemical smoke, air pollution or industrial dust for a long period. It may be genetic in some cases. The major symptoms for COPD are chronic coughs, chest tightness, shortness of breath, wheezing, etc. This is not fully curable like PD, but proper diagnosis and therapy can minimize its impact. So, to prevent more harm due to these disease, ML and DL classifiers are now being used to predict and detect them in early stage. In this method, first a model is trained using relevant dataset and different classifier, so that the model can give maximum prediction result. After this, new patient's data is provided to the model to check how model performs. Different ML classifiers like support vector machine (SVM), k-nearest neighbour (kNN), random forest (RF), decision tree (DT) and DL classifiers like convolutional neural network (CNN), AdaBoost and XGBoost are used by the researchers to train the prediction model.

## 2 Related Work

Recently, many researchers took different ML and DL classifiers into consideration to classify PD and COPD. For PD, some popular features i.e. freezing of gait (FoG), stiffness and hand movements were taken to classify PD, whereas for COPD, researchers took cardiac problem, TB history, breathlessness, wheezing, asthma, hypertension as input parameters and classified COPD from healthy people and also its severity. Some of the related works are discussed in this section.

Juutinen et al. [1] collected the data using smartphone sensors and distinguished PD from healthy people. For this study, nine ML classifiers were used for which kNN gave the highest accuracy of 84.5%, sensitivity of 88.5% and specificity of 81.3%.

Mazilu et al. [2] used a combination of smartphone and wearable accelerometer to detect FoG. They build an online FoG detected system, which gave an auditory feedback when FoG was detected. For classification, random trees (RT), RF, DT, pruned decision tree, NB, Bayes nets, kNN with one and two neighbours, MLP, bagging and boosting were used. AdaBoost gave them the best AUC, sensitivity, specificity and F1 score of 98–99%.

Tripoliti et al. [3] used six accelerometers and two gyroscope sensors to detect FoG for PD patients. For this, they took NB, RF, DT and random trees (RT) classifiers.

RF gave them the best accuracy of 96.11%, sensitivity of 81.94% and specificity of 98.74%.

Zhan et al. [4] implemented an Android application to keep track on the PD motor symptoms. For this study, the researchers took RF classifier with 500 trees. They collected the data before and after patient's medication. The patients were passively monitored for over 46,000 h. RF classifier was used for this study and achieved 71.4( $\pm 4$ )% of accuracy.

Anakal et al. [5] developed a clinical design support system (CDSS) model using multi-layer perceptron neural network (MLPNN) with three layers to detect PD severity. They used SVM, NN and DT classifiers to find the final prediction model.

Hussain et al. [6] built an AI system to diagnose PD severity. For classification, RF, SVM, gradient boosting machine (GBM), XGBoost and kNN classifiers. They built a soft voting ensemble (SVE) model to get the maximum accuracy. SVE gave the maximum accuracy of 91.0849% compared to others.

Wang et al. [7] developed an identification model to identify acute exacerbations in chronic obstructive pulmonary diseases (AECOPDs). The researchers used electronic medical records (EMRs) of COPD patients. For classification, RF, SVM, LR and kNN were used. LR and SVM gave almost same result.

Biichlin et al. [8] developed an ML tool that took the saliva samples of COPD patients and healthy people to classify the patients. For this, they took XGBoost, SVM, gradient boosting, LR and ANN. Gradient boosting gave the maximum sensitivity of 100% compared to others. A permittivity biosensor was used for saliva sample's characterization.

### 3 Discussion

Juutinen et al. [1] collected data from Satakunta Hospital District, Pori, Finland in 2018. A total of 97 subjects were taken for the whole study protocol. Before taking the measurement, the subjects were classified based on Finnish unified Parkinson's disease rating scale, section V (FIN-UPDRS-V). Using this, the patients were categorized with rating of 0–5, where 0 indicated no symptoms and 1–5 indicated the increasing severity of PD symptoms. The patients who had less severe symptoms were excluded. Walking test data was collected using smartphone sensors. Smartphone was attached to the patient's waist with a belt. Then the patients were asked to walk 20 steps, and the walking data was collected. As the data was collected using smartphone, the recordings having both accelerometer and gyroscope data were taken for pre-processing. Finally, 58 data were selected for the dataset. The researcher's main focus was to differentiate the walking features of PD patients and healthy people. In this step, individual steps were detected and the inactive periods were rejected. From the individual steps that were segmented from the walking test, 201 statistical features i.e. mean, median and standard deviation were calculated in each step. In the feature selection process, three methods i.e. minimum redundancy maximum relevance (mRMR), sequential forward feature selection (SFS) and

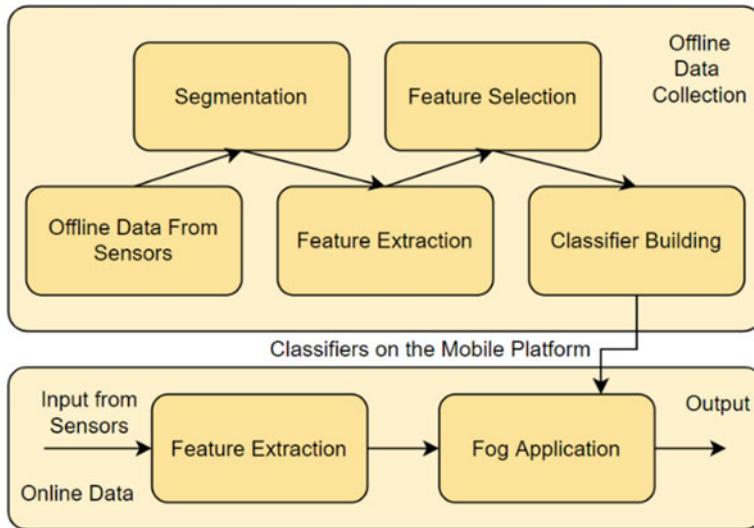


**Fig. 1** Methodology for PD classification using smartphone sensors [1]

sequential backward feature selection (SBS) were compared. mRMR gave the best result as it calculates the features depending on the number of input parameters and selects optimal features at each step. The optimal feature set was classified by 9 ML algorithms i.e. classification tree, linear discriminant analysis (LDA), Gaussian kernel, ensemble, logistic regression (LR), SVM, kNN, naïve Bayes (NB) and RF. They achieved the best accuracy of 84.5% and error rate of 15.5% using kNN [9–12] (Fig. 1).

Mazilu et al. [2] developed an online FoG detection system. They proposed a wearable assistant that was composed of wearable accelerometer sensor and smartphone. The main purpose of the system was to automatically detect FoG and give feedback to the patients. This system was composed of two systems. Firstly, the FoG detection classifier was trained with collected labelled data. For this, the most discriminative features that distinguish FoG from normal gait. The features were selected using correlation-based feature subset selection. The selected features were then used to train the classifier. Secondly, in the online phase, FoG was classified using Android application. To classify FoG, the app used the serialized classifier to that was serialized using Weka API in the previous offline training process. The application took the last recorded data by the sensor. Then it extracted the necessary features and passed to the classifier. The application provided some feedback to the patients if FoG was detected. The system was tested on DAPHNet dataset [8]. For data collection, the sensors were attached to the ankle, knee and on the lower back of each subject. Then the patients were asked for a random walk in a hall with 360° full turn and certain stop. While walking, the subjects walked forward, turned back and walked backward. Data collection was done in patient's own space to get the unbiased data of patient's daily activity. The collected data was then pre-processed in a fixed time interval. Then a window-based classification was done. For classification, the researchers used random trees (RT), RF, DT, NB, Bayes nets (BN), kNN, multi-layer perceptron (MLP) and boosting techniques. Using RF, they achieved the AUC of 99.82% for 1s window and AUC of 99.98% for 4s window. AdaBoost gave the best AUC of 99.85% for 1s window and AUC of 99.98% for 4s window (Fig. 2).

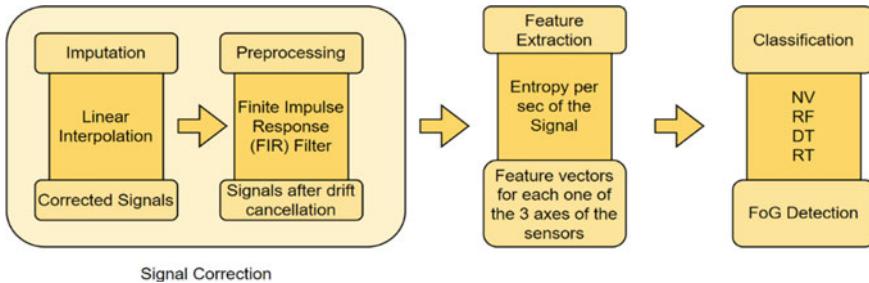
Tripoliti et al. [3] detected FoG events for PD patients using wearable sensor's signals. The experiment was done on 16 subjects where five were healthy people, five PD patients with gait symptoms and six without gait symptoms. For data collection, the patients were asked to perform some activities i.e. lying on the bed, standing up from the chair, walking a short distance, opening or closing door, etc., six accelerometer and two gyroscope sensors were used to collect the data. Accelerometer sensors were placed to right wrist, left and right legs, chest and waist of subject's body, and gyroscopes were placed on the patient's chest and waist. The data from the sensors



**Fig. 2** Methodology for online FoG detection system [2]

was collected using Bluetooth and portable storage device. After data collection, the recorded signals were pre-processed to replace the imputation and missing values in the dataset. Dataset contained imputation as sensors were disconnected sometimes and weak signals were captured due to large distance. Linear interpolation was taken to perform the imputation. After filtering and imputation calculation, entropy was calculated for every sensor's signals. Two factors were used for FoG identification. Entropy was used to detect FoG due to its non linearity feature, and the differences between healthy people and PD patients were observed using plotted histogram. For classification, feature vector was used to differentiate FoG from normal gait. Classification was performed using NB, RF, DT and RT algorithms. For waist data, the best result was obtained using DT, RF and NB. Whereas for chest data, the best result was obtained using DT, RF and RT. RF gave the best accuracy of 96.11%, sensitivity of 81.94% and specificity of 98.74% (Fig. 3).

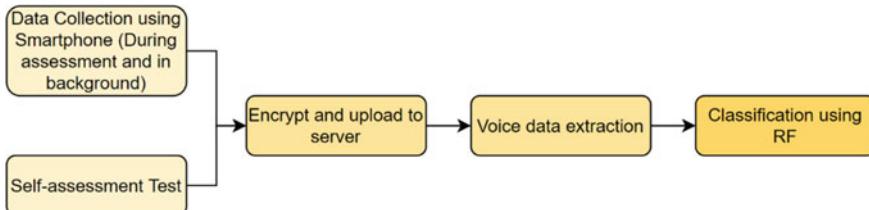
Zhan et al. [4] developed a smartphone-based high-frequency remote monitoring platform to monitor the symptoms of PD. They built a smartphone-based monitoring system named HopkinsPD to measure the PD symptoms. The data was collected actively at the time of clinical check-ups. Passively data was collected in the background the gain more features and increase the model's accuracy. Mainly phone usage logs, GPS location, Wi-Fi parameters, etc., were taken to measure the subjects' social behaviours and movements. From the collected data, five features i.e. voice, balance, reaction time, gait and dexterity were extracted. First, the application was installed in each subject's mobile to get high accessibility. Active tests were initiated and self-administered by the subjects in different times during the day. The application was running continuously in the background to captured passive data. Monitoring the subject passively gave an easy way of monitoring the patients without



**Fig. 3** Flowchart of automatic detection of FoG [3]

interrupting their daily activities. HopkinsPD is a fully automated system, so it captures and compresses the data without human intervention. Then the collected data was encrypted and uploaded to the server storage. Though data was collected both actively and passively, only active data was taken for feature extraction. In voice data extraction step, the 20 s samples were divided into 40 data frames in 5 s each. Dexterity features were extracted based on stay duration (how long subject touched the smartphone screen), move duration (interval between one finger release to another finger touching the screen). For classification, the researchers used RF classifier and achieved  $71.0(\pm 0.4)\%$  accuracy (Fig. 4).

Anakal et al. [5] designed a system to detect COPD and its severity. Firstly, a SVM model was designed for the initial evaluation of COPD. The model first classifies the COPD patients from healthy people. Then the model identified the severity of disease. For this, a multi-layer perceptron neural network (MLPNN) with three layers was used to design clinical design support system (CDSS) to find the severity. MLPNN was trained using back-propagation algorithm. After severity measurement, the researchers then detect the interaction of drugs and their side effects on patients to improve the system. Firstly, the basic details of the patients were collected. The details include medical history i.e. smoking history, allergy, alcohol consumption rate, cardiac problem, TB history, breathlessness, wheezing, asthma, hypertension and other symptoms of COPD. After collecting the details, a spirometer test was conducted to find the COPD severity. After severity diagnose, a proper treatment and management video were designed and given to the patients.

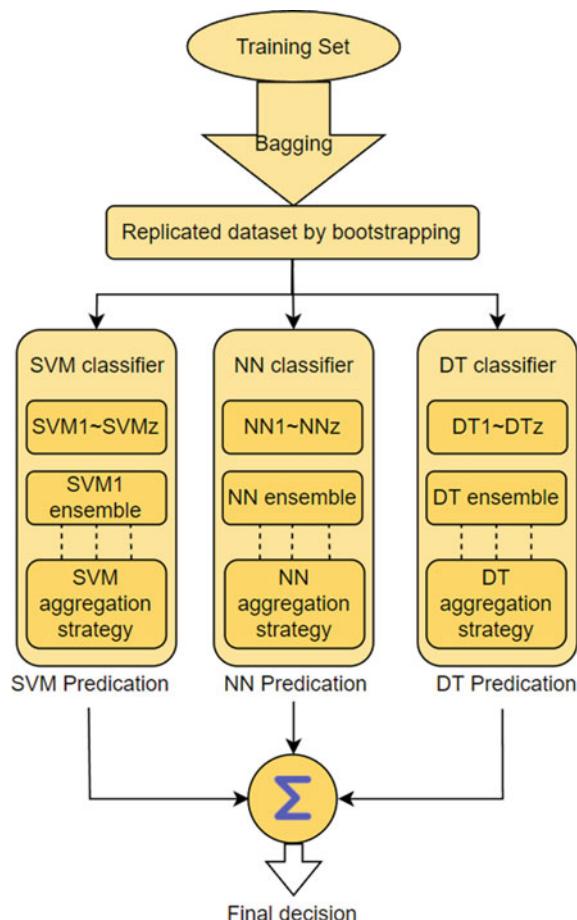


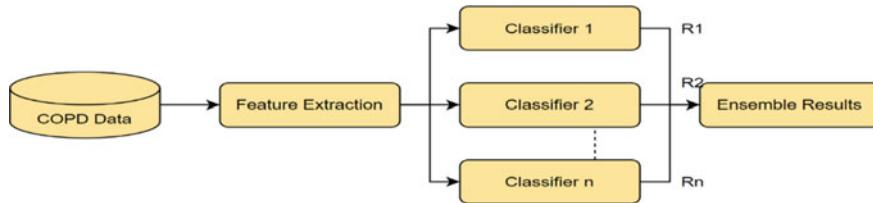
**Fig. 4** Methodology for PD identification using voice data extraction [4]

CDSS also included a drug-interaction checker that help doctors to check the drug-effect on patients. For classification, they used SVM, neural network (NN) and DT on the training set (Fig. 5).

Hussain et al. [6] built an AI system to detect COPD severity. For this study, the data was collected from Inje University Paik Hospital, Busan, Korea. The dataset contained 2900 COPD patients' details, who were classified as mild COPD and severe COPD. For feature extraction, recursive feature elimination (RFE) technique was used to select the optimal feature subset. A total of 54 features were collected for this study. Out of those features, 30 features were selected with RFE. From those 30 features, 24 features were taken after consulting to the experts. The dataset was splitted in 80:20 ratio, where 80% data was taken for training purpose and rest for testing. For classification, five state-of-art ML classifiers i.e. RF, SVM, gradient boosting machine (GBM), XGBoost and kNN were taken. But there were some certain limitations in the implementation of the classifiers, so a soft voting

**Fig. 5** Workings of ML techniques in the CDSS component [5]

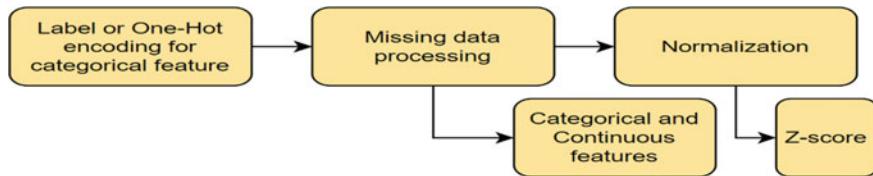
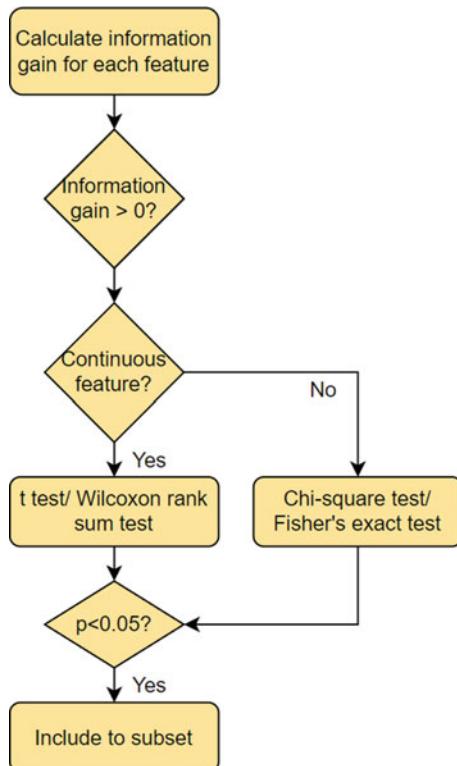
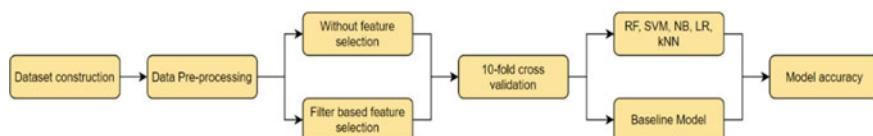




**Fig. 6** Architecture of SVE classifier for COPD classification [6]

ensemble (SVE) model was developed to get the maximum accuracy. For cross-validation, a five-fold cross-validation technique was done. The researchers achieved the average accuracy of 87.21% using RF, 90.225% using GBM, 88.0773% using XGB, 86.3587% using kNN and 88.1847% using SVM. SVE method provided the maximum accuracy of 91.0849% compared to others. SVE also gave the best AUC of 96.8656. The produced confusion matrix consisted of two labels where label 0 represented mild PD and label 1 represented severe PD. At label 0 matrix, 423 subjects out of 466 were correctly predicted. In label 1, 425 subjects from 465 were correctly predicted (Fig. 6).

Wang et al. [7] developed an identification model for acute exacerbations in chronic obstructive pulmonary diseases (AECOPDs) to disease the mortality rate caused by COPD. For this study, the electronic medical records (EMRs) of the patients were collected from China-Japan Friendship Hospital. A total of 563 data were collected which included the features like biomedical test, vital sign, diagnosis, demographics, etc. But, patients who have asthma, more than eight comorbidities were excluded from the dataset. The data was also excluded for the patients for whom AECOPDs diagnosis was missing or more than 30% features were missing. The patients aged between 36 and 80 years who were diagnosed as COPD and who were discharged from the hospital after treatment were included to the dataset. After data collection, 33 input features and one output features were extracted from the dataset. Here, the categorical features were converted using label encoding. Categorical missing data was replaced with the most frequent category associated with the corresponding feature. Continuous missing values were replaced with the average of the corresponding feature. Five ML algorithms i.e. RF, SVM, NB, LR and kNN were compared for the identification of AECOPDs. LR gave the best result when feature selection was not performed. After feature selection, result for all the five algorithms was improved. But SVM, along with LR gave better result than the others. Before feature selection, LR gave the best sensitivity of 0.79, specificity of 0.80 and AUC of 0.83 than the others. After feature selection LR gave the sensitivity of 0.83, specificity of 0.79 and AUC of 0.90. Whereas, SVM gave the sensitivity of 0.80, specificity of 0.83 and AUC of 0.90 (Figs. 7, 8 and 9).

**Fig. 7** Flowchart of data pre-processing [7]**Fig. 8** Flowchart of feature extraction [7]**Fig. 9** Flow diagram of proposed methodology [7]

## 4 Comparative Study

In the above-discussed papers, we found different classification techniques were used that were used to classify the diseases and their severity. The data was collected mainly using smartphone and body-fixed sensors. For COPD, the data was collected from the hospitals. In this section, we are going to compare the paper's outcome and their drawbacks.

In [1–3] and [4], the researchers used smartphone inbuilt sensors and body-fixed sensors to collect the data. In these papers, PD classification was done. For this, kNN, RF, NB, RF and SVM classifiers were used. In [1], kNN gave the best accuracy. kNN is mainly used for classification technique. Depending on the similarity of the preceding data points, it classifies new data points. Parameter tuning is done to find the best value for any point. It calculates the nearest neighbour of any point using Euclidian distance. But, it is computationally expensive. kNN also perform less efficiently for high-dimensional data. If dataset contains missing values, then also its performance becomes less efficient. SVM performs better than kNN as it is easier to interpret. SVM separates the features and classifies data points using hyperplane. SVM can be used for both classification and regression problem. It is suitable for a greater number of features in the dataset. But SVM has certain problems. It has more computational time. If target classes overlap, then SVM's performance gradually decreases. RF gave better classification result than SVM. RF is built using multiple DT. All the DTs have their own prediction and the tree with maximum votes becomes the model's prediction. For data pre-processing, DT needs less effort. It has no effects of missing values while tree structuring. Though ML algorithms gave a decent accuracy for PD classification, the accuracy can be increased using ensemble of different ML classifiers.

In [6] and [7], RF, SVM, kNN and XGBoost were used. For both the papers, SVM gave comparatively better result. But the process used in [7] is more attractive. The researchers ensemble the classifiers and differentiate COPD patients from healthy people. Also, they differentiate mild COPD from severe COPD. This work can enhance the monitoring system for COPD patients. As COPD is incurable and no proper surgery is found for this. So, constant monitoring can reduce the severity. XGBoost was used in [7], but it suffers from overfitting due to improper tuned parameters. Interpretation and visualization are hard for XGBoost. It is also a slower process than RF. For [8], LR gave the best accuracy, but for high-dimensional dataset it shows overfitting problem. Also, LR cannot solve non-linear problems due to its linear decision surface. Though, the papers have some decent accuracy in PD and COPD classification, we found certain drawbacks in the works. For PD classification, wearable body-fixed sensors and smart phone sensors were used, but proper security mechanism was not mentioned (Table 1).

**Table 1** Advantages and disadvantages of above discussed papers

Heading level		Example	Font size and style	
Juutinen et al. [1]	Classification tree, Gaussian kernel, linear discriminant analysis, ensemble, kNN, LR, NB, SVM and RF were used	Highest accuracy of 84.5% and an error rate of 15.5% was achieved using kNN classifier	This process is acceptable due to the availability of smartphone and use of default Android sensors	Subjects took extra steps while walking test. So, some data was biased
			Working with motor symptoms (walking test) made the work useful	The measured data was manually transferred to database
Mazilu et al. [2]	RT, RF, DT, pruned decision tree, NB, BN, kNN with one neighbour and two neighbours, MLP, bagging and boosting were used to classify gait from healthy people	AdaBoost gave the best sensitivity, specificity, <i>F</i> 1 score and accuracy of 98–99%	Online FoG detection system gave opportunity to classify PD from healthy people	Training and testing data was randomly selected. So, biasness was there
			Number of sensors were less for the study	A user dependent algorithm can be built as a future work
Tripoliti et al. [3]	NB, RF, DT and RT classifier on FoG	About 96.11% of accuracy was achieved using the RF classifier	Automated system was built based on patient's movement types (walking, turning, or standing) for FoG detection	Only 16 subjects were involved in this study. Some misclassification was there as different duration of the FoG events was taken
			Sensor data give brief understanding of the subjects' movement	A large dataset can be taken as a future work to enhance the accuracy

(continued)

**Table 1** (continued)

Heading level		Example	Font size and style	
Zhan et al. [4]	Implemented a smartphone application to track the motor symptoms of PD	Random forest achieved $71.4(\pm 4)\%$ accuracy	Easily available software was used for remote monitoring	Easily available software was used for remote monitoring.
			Clinical monitoring was done at low frequency	Clinical monitoring was done at low frequency. based on clinical data, patients were observed remotely and frequently with high frequently data
			Based on clinical data, patients were observed remotely and frequently with high frequently data	
Anakal et al. [5]	SVM, NN, DT and were used to the model	Clinical design support system (CDSS) was designed to keep track to the COPD patients	CDSS system helps to diagnose COPD more accurately and efficiently	A portable CDSS device can be developed without attaching computer devices
			This system classified different stages of COPD	
Hussain et al. [6]	For classification, RF, SVM, gradient boosting machine (GBM), XGBoost and kNN classifiers	SVE gave the maximum accuracy of 91.0849% compared to others	Enhance the monitoring system for COPD patients	Ensemble method was used for this study, which faces overfitting and underfitting issues at the time of training and validation
				Only two classes, mild and severe, were taken into consideration. More classified classes can give more accuracy and treatment probability

(continued)

**Table 1** (continued)

Heading level		Example	Font size and style	
	They also built an ensemble classifier		As this system classified mild COPD with severe COPD, so treatment becomes easier	The model was not tested in real-life scenario to test the accuracy
Wang et al. [7]	For classification, RF, SVM, LR and kNN were used	LR and SVM gave almost same result	This model can be used as a support to the technician for COPD diagnosis	Dataset taken for this work was small, a large dataset can be tested on the model to check the accuracy
		LR gave AUC of 0.90, whereas SVM gave the AUC of 0.90	Electronic medical records were taken, so inclusion of unbiased data is less	More features can be extracted from the dataset  The dataset was taken only from one region. Data from multi-region should be tested on the model to validate the accuracy

## 5 Conclusion and Future Scope

From the discussion of above papers, we found that PD and COPD are incurable and one of the major causes for mortality worldwide. So, detecting these diseases at very early stages can prevent more harm. For this, human activities can be taken into consideration. Patients suffering from PD and COPD face many abnormalities in their daily life. By screening these abnormalities, the diseases can be classified. For early PD detection, pre-motor symptoms should be taken along with motor symptoms to get better result. Collecting medical data using wearable sensors and smartphone inbuilt sensors is a critical task. It can be a huge threat if some data is compromised. So, to secure data collection process, IOT sensors can be used with proper security framework. Secured IOT architecture can ensure proper confidentiality and integrity to the critical medical data at data collection step. Data transferring to the server also has some risk if proper secure channel is not used. We can consider secure edge server for this. Edge server with proper security infrastructure gives the mechanism to collect the data, filter it and securely transfer it to cloud server. Cloud servers confirm the availability of the data, so that doctors can easily access the data anywhere anytime.

In some existing works, it is observed that data collection is mainly done at the time of clinical tests. But IOT devices can be used for data collection at patient's own space. Large dataset with a greater number of features can be captured using this. Availability of smartphone makes this process easy. Also, it is still not clear, what are the proper reasons for PD and COPD. So, if the reasons can be optimized, then better treatment and monitoring can be given for the diseases.

## References

1. Juutinen, M., Wang, C., Zhu, J., Haladjian, J., Ruokolainen, J., Puustinen, J., Vehkaoja, A.: Parkinson's disease detection from 20-step walking tests using inertial sensors of a smartphone: machine learning approach based on an observational case-control study. *PLoS ONE* **15**(7), e0236258 (2020)
2. Mazilu, S., Hardegger, M., Zhu, Z., Roggen, D., Tröster, G., Plotnik, M., Hausdorff, J.M.: Online detection of freezing of gait with smartphones and machine learning techniques. In: 2012 6th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops, pp. 123–130. IEEE (2012)
3. Tripoliti, E.E., et al.: Automatic detection of freezing of gait events in patients with Parkinson's disease. *Comput. Methods Programs Biomed.* **110**(1), 12–26 (2013)
4. Zhan, A., Little, M.A., Harris, D.A., Abiola, S.O., Dorsey, E., Saria, S., Terzis, A.: High frequency remote monitoring of Parkinson's disease via smartphone: platform overview and medication response detection (2016). arXiv preprint [arXiv:1601.00960](https://arxiv.org/abs/1601.00960)
5. Anakal, S., Sandhya, P.: Clinical decision support system for chronic obstructive pulmonary disease using machine learning techniques. In: 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), pp. 1–5. IEEE (2017)
6. Hussain, A., Choi, H.E., Kim, H.J., Aich, S., Saqlain, M., Kim, H.C.: Forecast the exacerbation in patients of chronic obstructive pulmonary disease with clinical indicators using machine learning techniques. *Diagnostics* **11**(5), 829 (2021)
7. Wang, C., Chen, X., Du, L., Zhan, Q., Yang, T., Fang, Z.: Comparison of machine learning algorithms for the identification of acute exacerbations in chronic obstructive pulmonary disease. *Comput. Methods Programs Biomed.* **188**, 105267 (2020)
8. Biichlin, M., Plotnik, M., Roggen, D., Maidan, I., Hausdorff, M., Giladi, N., Troster, G.: Wearable assistant for Parkinson's disease patients with the freezing of gait symptom. *IEEE Trans. Inf. Technol. Biomed.* **14**, 436–446 (2010)
9. Sinha, T., et al.: Analysis and prediction of COVID-19 confirmed cases using deep learning models: a comparative study. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [http://doi.org/10.1007/978-981-16-2164-2\\_18](http://doi.org/10.1007/978-981-16-2164-2_18)
10. Palimkar, P., et al.: Machine learning technique to prognosis diabetes disease: random forest classifier approach. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [http://doi.org/10.1007/978-981-16-2164-2\\_19](http://doi.org/10.1007/978-981-16-2164-2_19)
11. Rajawat, A.S., et al.: Sleep Apnea detection using contact-based and non-contact-based using deep learning methods. In: Bansal, J.C., Paprzycki, M., Bianchini, M., Das, S. (eds.) Computationally Intelligent Systems and their Applications. Studies in Computational Intelligence, vol. 950. Springer, Singapore (2021). [http://doi.org/10.1007/978-981-16-0407-2\\_7](http://doi.org/10.1007/978-981-16-0407-2_7)
12. Mridha, K., et al.: Web based brain tumor detection using neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 137–143 (2021). <http://doi.org/10.1109/ICCCA52192.2021.9666248>

# Use of Various Optimization Algorithms in the Energy Minimization Problem Domain of WSN: A Survey



Sudip Kumar De , Avishek Banerjee , Koushik Majumder , Rabindra Nath Shaw, and Ankush Ghosh

**Abstract** A wireless sensor network (WSN) is a collection of sensor nodes that are geographically distributed and wirelessly connected. A wireless sensor node is a small device that can sense various physical measurements from its environment and after converting them into electrical signals it can send and receive a limited quantity of data to other nodes or a sink node in the network. The sensed data is used to facilitate the users in different ways. A wireless sensor node typically contains a microcontroller unit (responsible for processing), a transceiver unit (responsible for communication), a data storage unit (responsible for storing information), one or more sensors, and a battery stored power unit (responsible for supplying energy to other units). The power storage capability of the small battery is quite limited. If the use of energy is not prioritized for the supplementation of energy to various components, a WSN might soon dry out. One of the prime focuses of researchers in the field of WSN is the minimization of energy. If the energy can be minimized then the sustainability of the network will be maximized. In this paper, the energy minimization problem has been addressed along with the usage of optimization algorithms.

**Keywords** Wireless sensor network (WSN) · Energy management strategies · Deployment · Clustering · Duty cycle · Data aggregation · Optimization algorithms

---

S. K. De

Department of Information Technology, Asansol Engineering College, Asansol, West Bengal, India

A. Banerjee

School of Computer Science Engineering and Technology, Bennett University, Greater Noida, Uttar Pradesh, India

K. Majumder

Department of Computer Science and Engineering, MAKAUT, Kolkata, West Bengal, India

R. N. Shaw · A. Ghosh (

University Center for Research & Development (UCRD), Chandigarh University, Mohali, Punjab, India

e-mail: [ankushghosh@gmail.com](mailto:ankushghosh@gmail.com)

## 1 Introduction

A wireless sensor network (WSN) is a self-contained network with several nodes and a few controllers known as sink nodes. The processing unit, a communication unit, memory unit, sensor unit, and battery unit are the five major but small units that make up the WSN nodes. The sensor unit detects various types of external analog signals, the processing unit processes the data and converts it to processed data, the memory unit stores the processed data, and the communication unit establishes connections between neighboring nodes. The battery unit is responsible for providing power to the entire node.

Wireless sensor networks (WSNs) have grown in popularity in recent years as a result of industry 4.0-related research for a variety of reasons. The following are some of the benefits of employing WSNs in Industry 4.0.

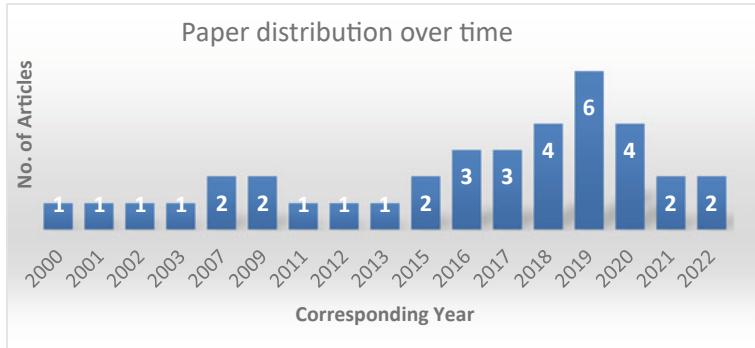
1. **Deployability:** Due to the ease with which WSN nodes and sink nodes can be deployed, WSN is being used in a variety of technical applications, such as pressure and temperature in a blast furnace, where human intervention is difficult.
2. **Accessibility:** This attribute allows the network to be accessed easily through a strong connection model using WSN nodes and sink nodes, though hardware reliance is a major aspect that influences the network's accessibility.
3. **Commercialization:** In the recent era, the commercialization of WSN nodes has also been an advantage in terms of easy availability of the essential equipment to build a powerful networking system.

Though the WSN has some drawbacks, such as limited computing capability due to its small designs, limited communication range between two neighboring nodes, and limited sensing capacity of coupled sensors, it has a lot of potentials. Despite its limitations, the WSN performs admirably in a variety of challenging situations. Many optimizations and soft computing techniques have been developed by the researchers to overcome challenges in a variety of areas, including coverage area optimization, power consumption optimization, duty cycle optimization, routing protocol optimization, topology control optimization, cost optimization, and so on.

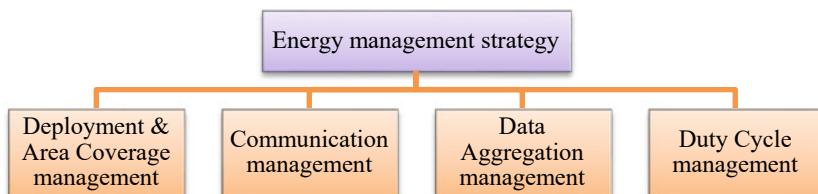
**Motivation:** Various optimization techniques for energy management strategies in WSNs have been investigated in this survey, which will be useful to researchers. The authors considered 37 research articles from the year 2000 to the present for the descriptive analysis on energy management strategies, as shown in Fig. 1.

## 2 Existing Energy Management Strategies in WSN

The energy management strategies in WSN can be classified as:



**Fig. 1** Paper distribution over time



## 2.1 Deployment and Area Coverage Management

Wireless sensor node deployment entails dispersing active WSN nodes throughout a target region. The deployment approach might be planned or haphazard. The haphazard or random deployment of WSN nodes increases the number of redundant nodes, whereas the planned node deployment strategy can reduce redundant nodes. Even with a significant number of redundant nodes deployed, coverage excellence cannot be assured. The deployment approach has a direct impact on the coverage quality of the network, network communication, and energy requirements of the network. The area coverage parameter of the WSN focuses on how adequately a target region is tracked. The coverage quality has a considerable impact on the functioning of a wireless sensor network (WSN) system. The network's energy requirements may be minimized with an optimal deployment strategy, allowing it to last longer. The energy minimized sensor deployment strategy is an NP-complete problem. For that reason, researchers are trying to use metaheuristic algorithms to get optimal or near-optimal solutions.

In 2021, Banerjee et al. [1] proposed different deployment strategies (random deployment, S-pattern and Spiral-pattern) to minimize the consumed energy by WSN nodes. The DE-QPSO, a hybrid algorithm with a single objective function, was utilized to build an energy-efficient network.

In 2019, Wang et al. [2] proposed two flower pollination algorithms (FPA): The first is an improved flower pollination algorithm (IFPA) that improves the FPA algorithm's convergence speed and precision; the second is a non-dominated sorting multi-objective flower pollination algorithm (NSMOFPA) based on FPA that effectively maintains population diversity. The NSMOFPA optimization method results in a more optimal WSN deployment in terms of energy consumption, coverage area, etc.

In 2017, Benatia et al. [3] proposed a multi-objective genetic algorithm (GA) to develop a multi-objective optimization deployment strategy (MODS). In this paper, the authors had proved that energy consumption directly depends on the distance between the sink node and WSN node. And the distance between WSN nodes and the sink node can be optimized using an effective deployment strategy which will help in the optimization of the WSN lifetime.

In 2018, Céspedes-Mota et al. [4] proposed a multi-objective differential evolution algorithm (MODEA) to optimize the sensor deployment process over the target area for the maximization of area coverage and minimization of the network's energy consumptions. It shows a good result; however, the research does not take into account the energy consumption caused by data communication: transmission and reception.

In 2019, Xiang et al. [5] suggested a hybrid node deployment approach employing the cuckoo search (CS) metaheuristic algorithm, and it was discovered that their optimization technique can both minimize energy consumption and improve coverage area. According to the authors they claim that their algorithm can balance the regional coverage and energy consumption. The authors had considered moveable WSN nodes. Their focus was on minimizing the number of movements of the WSN nodes. If the number of movements is reduced then the energy loss in process of movement will be also reduced. And this approach helped them to minimize the energy requirement of the network. But the relationship between the energy loss and node movement process is not depicted in the paper.

In 2020, ZainEldin et al. [6] have proposed a deployment strategy using a multi-objective metaheuristic algorithm, IDDTGA. The genetic algorithm (GA) has been used to find the optimal locations of the WSN nodes considering the mobility feature of the WSN nodes to reduce the coverage hole. The main focus of the paper is to develop an improved deployment strategy to minimize the network cost by minimizing WSN nodes with the greatest possible coverage area.

In 2022, Banerjee et al. [7] have proposed the algorithm for the construction of an energy-efficient WSN. The authors have used the Least Distance Connect First (LDCF) deployment strategy with the K-Mean algorithm and a hybrid algorithm, GA-SAMP-MWPSO, to minimize the overall energy requirement of the network. The algorithm delivers a better outcome than the existing algorithms (ACO, GA, PSO).

## 2.2 Communication Management

The data, collected from various nodes, should be transmitted to the sink node through the network. This data transfer requires a significant amount of battery power. Researchers have shown their interest to develop energy-efficient WSN communication management strategies. *Clustering and routing algorithms, as well as MAC protocol design*, are used in communication management methods to construct an energy-efficient WSN. To build an energy-efficient WSN communication strategy, one should concentrate on the factors like conserved battery power, energy dissipation rate, amount of energy harvesting from various activities such as vibrations, light, temperature, etc., collision avoidance rate, latency degradation rate, packet loss rate, avoidance of data redundancy, efficient routing with proper bandwidth utilization and so on.

The MAC protocol design can be classified as contention-based [8], contention-free [9], non-collision-based [10]. The contention-based allows several nodes to share the same radio channel without prior cooperation. The contention-free protocol typically uses the time division multiplexing access (TDMA) technique and relies on the fact that all the nodes are synchronized with time. It works well for the small network but for the larger network, the synchronization of the time is not always feasible. Although the CSMA/CD, CSMA/CA protocol eliminates collisions once a node has obtained a channel, collisions can still occur if many nodes begin sending data at the same time. A non-collision-based protocol resolves collision in the contention period to eliminate the chance of collision.

Although optimization techniques were less utilized in the design of the energy-efficient MAC protocol, they were widely used in clustering and routing algorithms to construct energy-efficient networks.

Ye et al. [8] designed the S-MAC protocol, which is specifically developed for wireless communication networks. The authors identified a few causes of energy wastage: the collision of packets, packet overhearing, controlling the less useful packet sending overhead, idle packet listening. If the above factors are considered then an energy-efficient network can be designed. It has been stated that a collision causes data to be resubmitted, and data resubmission increases communication delay as well as energy needs. When a WSN node picks up packets that are supposed to be for other WSN nodes then this thing is known as overhearing, and this overhearing of data can lead to an inefficient design of the network because this is another reason for energy loss. Having good control over less useful or most useful packets screening leads to an energy-efficient network. The result shows that the S-MAC protocol saves more energy compared to the energy used by the 802.11-like MAC protocol.

Rajendran et al. [10] developed Traffic-Adaptive Medium Access (TRAMA), a noble protocol that delivers collision-free unicast, multicast, broadcast transmission by managing the idle packet listening factor and employing transmission schedules to avoid collisions. It is stated that when the transmission load in a contention-based system, such as S-MAC, grows, the risk of packet collision increases, lowering

channel utilization and wasting battery power. The TRAMA protocol can be useful to design an energy-efficient WSN.

Alvi et al. [9] suggested a non-collision-based protocol that combines the Knapsack optimization method with a contention-free or TDMA-based protocol for better link utilization. By minimizing the control overhead by assigning 1 byte short ID to every node, energy consumption is also reduced.

Clustering is the process of splitting a region of interest into several subregions, or clusters. One of the most important strategies for extending the life of a network is clustering. Routing is the process of finding routes or paths for the transferring of the data from the source node to the destination. Both clustering algorithms and routing algorithms have a significant influence on the energy minimization process of WSN.

Heinzelman et al. [11] had proposed a cluster-based communication protocol, low-energy adaptive clustering hierarchy (LEACH). This algorithm distributes energy load at random and integrates data fusion into the routing protocol to reduce overall network energy consumption. The LEACH algorithm provides Quality of Service (QoS) while also extending the network's lifespan. The LEACH routing protocol has been designed considering single-hop connectivity between CH and base station (BS). When the coverage area is large, then the distance between the CH and base station might not be appropriate for LEACH routing using single-Hop communication. To address this issue the multi-hop LEACH protocol has been proposed by Biradar et al. [12]. The multi-hop LEACH shows energy efficiency in WSN.

Manjeshwar et al. [13] had proposed a hierarchical cluster-based communication protocol, threshold-sensitive energy-efficient sensor network protocol (TEEN), for reactive (quick response time) WSNs which are suitable for real-time or time-constrained applications and which shows efficiency in energy consumption.

In 2007, Latiff et al. [14] applied the particle swarm optimization (PSO) meta-heuristic algorithm for optimization of energy consumption of the network. Using this optimization algorithm, the network's lifetime has been increased. It also shows that the PSO outperforms both LEACH and LEACH-C in terms of network coverage and total throughput.

In 2016, Vimalarani et al. [15] had proposed an enhanced PSO-based clustering energy optimization (EPSO-CEO) algorithm. The authors employed the enhanced PSO optimization algorithm to create clusters and to choose optimal cluster heads from the clusters. To evaluate the algorithm, authors had considered these factors: packet delivery ratio, packet dropping ratio, end-to-end latency, throughput, total residual energy, average residual energy, total energy consumption, and the simulation results prove that the authors' approach is better than the competitive clustering (CC) algorithm.

Saha and Gupta [16] try to address the problem of cluster head (CH) selection to solve the energy hole problem using harmony search-based clustering protocol. To minimize energy consumption, the thought of the mobile sink nodes has been introduced. When compared to the LEACH algorithm, it shows to be a superior method in terms of energy usage and the number of alive nodes.

Arjunan and Sujatha [17] introduced a hybrid optimization approach that combines a fuzzy logic-based cluster scheme with the ant colony optimization (ACO) technique. It has been compared with the existing protocols LEACH, TEEN, DEEC, etc. and it has been found that their hybridized algorithm works better than the existing protocols LEACH, TEEN, DEEC in terms of energy consumption, number of alive nodes. It eliminates the energy hole problem of the network.

In 2018 Javadpour et al. [18] also proposed an energy-efficient clustering algorithm using fuzzy logic and PSO algorithm. The fuzzy logic has been used to build the network and PSO has been used to choose the cluster head. This strategy extends the network's life by reducing the amount of energy required by the sensor network.

Krishnan et al. [19] proposed an enhanced energy-efficient clustering algorithm using the ACO optimization algorithm considering multiple mobile sink nodes for maximizing the energy efficiency of the network. This ACO-based algorithm outperforms LEACH, GA, and PSO and shows the maximization of the lifetime with reduced data loss.

Wang et al. [20] have proposed an enhanced PSO algorithm to a search of energy centers for cluster head (CH) selection. This approach tackles the problem of energy holes in the network, and it outperforms the other algorithm (VD-PSO) in terms of energy usage and network lifetime.

Vijayalakshmi and Anandan [21] proposed a multi-objective optimization algorithm, Tabu particle swarm optimization, by hybridizing simple PSO and generalized Tabu search algorithms. The distance between nodes and BS, as well as the energy level of nodes, were taken into account when designing the multi-objective problem. The PSO has been applied to find the local best position of the base station (BS). To find the global best position the hybridized Tabu-PSO algorithm has been applied. In terms of the number of alive nodes, packet loss rate, and average latency, the findings reveal that Tab-PSO beats the multi-hop-LEACH and PSO algorithms.

Anand and Pandey [22] proposed a GA-PSO algorithm to maximize the network's lifetime. For each cluster, a cluster head (CH) is determined using the GA method, which took into account the node's distance and energy factors. The selection of relay nodes is based on a genetic approach in which the first CHs are picked and a subset of cluster heads is designated as relay nodes. When a CH has data to send to a base station (BS) or sink node (SN), it will use the best routing paths using the PSO algorithm to send it to the BS through the relay nodes or directly to the BS. The network lifetime rises as a result of improved CH selection and relay nodes deployment strategy in the network. The GA-PSO algorithm outperforms other standard methods such as LEACH, HCR, GADA-LEACH, and EA-CRP.

Deepa and Suguna [23] proposed OQoS-CMRP, a cluster-based multipath routing protocol, to minimize network energy consumption by minimizing communication overhead and transmission latency. The authors employed a modified PSO meta-heuristic algorithm to create clusters, choose cluster heads, and solve energy hole problems.

Banerjee et al. [7] employed a modified K-Mean method in 2022 to determine the position of the sink node (SN) during the clusterization process. Later, to extend the network's lifespan, the hybridized algorithm GA-SAMP-MWPSO was applied.

In 2021, Banerjee et al. [24] suggested a modified ACO technique to create an energy-efficient smart WSN system in a smart city by minimizing the coverage path. They have chosen an area of Durgapur, India, as an experimental zone ( $3.62 \times 3.62 \text{ km}^2$ ) where the information related to longitude, latitude has been fetched from <http://overpass-api.de/api>. Their routing strategy brings lifetime enhancement to the network.

### 2.3 Data Aggregation Strategy

Data gathered by many WSN nodes are sent to the cluster head (CH), and subsequently from the cluster head to the base station (BS) using a multi-hop or single-hop architecture. There is always a possibility of redundant data transmission. Data aggregation is a process that extends the lifetime of a WSN node by eliminating redundant unnecessary data and reducing the number of transmissions. Minimizing the length of the data packet network's lifetime can also be maximized. Dhasian et al. [25] divided data aggregation into five categories: network-based, cluster-based, tree-based, and centralized aggregation.

Al-Karaki et al. [26] have proposed a **network-based** data aggregation strategy, grid-based routing and aggregator selection scheme (GRASS), to minimize the energy consumptions of the network. To prolong the lifetime of the network the authors have used the genetic algorithm (GA) for finding data aggregation points, routes from WSN nodes to the data aggregation points, and the route from data aggregation point to the base station. Their twin heuristic approach in routing as well as data aggregation improves network longevity.

In **cluster-based** aggregation, sensor-sensed data is sent to cluster head (CH) and after aggregating the received data, CH sends the data to the base station (BS). In cluster-based data aggregation, intra-cluster data redundancy mainly happens and it consumes less energy, but it suffers from low accuracy and high latency problems when a large number of packets arrive in a short period. Shao-liang et al. [27], LEACH, HEED are popular cluster-based data aggregation algorithms.

Lu et al. [28] have proposed a probabilistic **tree-based** aggregation protocol using the hybridization of two optimization algorithms: ant colony optimization (ACO) and genetic algorithm (GA). The applied multi-objective function takes into account four goals: reducing energy usage, lowering aggregation delay, minimizing transmission interference, and maximizing network longevity. Using ACO-GA, the routing scheme has been developed. Packets received at nodes become deterministic when an adaptive dynamic timer policy is used in the routing process. The dynamic timer strategy reduces transmission delay and increases the likelihood of data aggregation. Yu et al. [29] have proposed a distributed tree-base data aggregation algorithm to reduce aggregation latency using a collision-free scheduling greedy approach.

Xiao et al. [30] suggested a **centralized** data aggregation technique. They have given their focus on energy allocation schemes centrally to minimize the energy

requirement of the network. To get the optimum value of accuracy for the data aggregation, the heuristical genetic approach has been adopted.

## 2.4 Duty Cycle Management

Duty cycle management is another important factor for the durability of the WSN. The power supply to the WSN nodes is limited. As a result, nodes cannot constantly remain in an active state since they would dry out very quickly. The ratio of a sensor node's active phase to its sleep phase is referred to as the duty cycle. By optimizing the duty cycle of WSN nodes, network's lifetime can be extended.

De Paz Alberola and Pesch [31] developed a duty cycle management algorithm, DCLA, that estimates the ideal duty cycle using the reinforcement learning (RL) technique. The RL collects the network's analytical data (delay constraints and the likelihood of successful data delivery) to forecast incoming traffic demand. It has been found that DCLA can minimize energy consumption while also having low processing power and memory demand capabilities.

Sinde et al. [32] proposed an algorithm, E<sup>2</sup>S-DRL, which has three phases: clustering phase, duty cycle management phase, and routing phase. The algorithm's goal is to reduce the network's energy consumption while also increasing its longevity. During the clustering phase, they concentrated on the data aggregation method while performing zone-based clustering with a hybrid PSO and affinity propagation (AP) algorithms. In the second phase, deep reinforcement learning (DRL) has been applied for duty cycle management. In the third phase, ACO and firefly algorithm (FFA) have been used to find the route of communication while keeping transmission latency in mind. The DRL method is used to adaptively plan the scheduling modes (sleep, listen, and transmit) of each sensor node, and it ensures that each WSN node will have a unique time slot to avoid data collision during communication.

Xu et al. [33] proposed a hybrid optimization algorithm, PHGWO, to address the problem of duty cycle management for the minimization of energy consumption. The parallel operator has been used with the gray wolf algorithm, and it is noticed that the hybrid algorithm, PHGWO, performs better than GA, PSO algorithms in terms of energy consumptions of the network.

Liu et al. [34] presented QCGWO, a hybrid algorithm for extending network lifetime by optimizing sensor duty cycle while taking into account real-world factory circumstances. The integration of quantum computing with gray wolf optimization assists the QCGWO in avoiding a local optimum. When it comes to enhancing network lifetime, the QCGWO algorithm surpasses the GA and simulated annealing (SA) techniques.

Subramanian and Paramasivam [35] proposed a MAC-based QoS algorithm, by applying the concept of assigning priorities among the nodes (PRIN) while considering the factors like delay, energy requirement, the lifespan of the network. They attempted to minimize the sensor nodes' duty cycle to extend the network's lifespan.

Iala et al. [36] proposed an adaptive duty cycle protocol (AD-MAC) to reduce energy consumption where the transceiver is waiting for incoming packets by estimating the time frame of incoming data packets. As the AD-MAC does not need central synchronization so the scalability factor of this approach is high. In terms of minimizing energy usage and successful data delivery report, the AD-MAC protocol beats the IEEE 802.15.4 standard.

Khan et al. [37] developed a dynamic adaptive duty cycle algorithm that works in complex and dynamic environments without the need for external assistance. Using ANN, they predict the timeframe pattern of incoming data packets at which the receiver should be in an active state. Comparing the results of the dynamic adaptive duty cycle algorithm with the fixed duty cycle algorithm it has been observed that ANN-based dynamic adaptive duty cycle algorithm minimizes the overall energy requirement [38].

### 3 Conclusions

The construction of an energy-efficient WSN is always a great challenging task. Lots of factors can be considered to build an energy-efficient WSN. After examining the research articles on deployment methods, it was discovered that there are more types of deployment techniques that may be developed to reduce the WSN energy consumption. Many researchers had worked on the maximization of coverage area. By maximizing the coverage area, one can save the number of WSN nodes to be deployed. To reduce the total energy consumption in WSN transmission, researchers can use ANN to build efficient communication routes in the WSN. The researcher can concentrate on developing lossless data aggregation algorithms based on compressive sensing. The compressed data can be retrieved at the receiver's end if the data aggregation approach is lossless. If the pattern of sleep and wake states of WSN can be predicted in advance using ANN, SVM, or other sorts of decision-making algorithms, then the energy-minimized WSN can be constructed with minimum latency.

### References

1. Banerjee, A., Das, V., Biswas, A., Chattopadhyay, S., Biswas, U.: Development of energy-efficient and optimized coverage area network configuration to achieve reliable WSN network using meta-heuristic approaches. *Int. J. Appl. Metaheuristic Comput. (IJAMC)* **12**(3), Article 1 (2021)
2. Wang, Z., Xie, H., He, D., Chan, S.: Wireless sensor network deployment optimization based on two flower pollination algorithms. *IEEE Access* **7**, 180590–180608 (2019)
3. Benatia, M.A., Baudry, D., Louis, A., El-Hami, A., Mazari, B.: Multi-objective WSN deployment using genetic algorithms under cost, coverage, and connectivity constraints. *Wireless Pers. Commun.* **94**(4), 2739–2768 (2017)
4. Céspedes-Mota, A., Castañón, G., Martínez-Herrera, A.F., Cárdenas-Barrón, L.E., Sarmiento, A.M.: Differential evolution algorithm applied to wireless sensor distribution on different

- geometric shapes with area and energy optimization. *J. Netw. Comput. Appl.* **119**, 14–23 (2018)
- 5. Xiang, T., Wang, H., Shi, Y.: Hybrid WSN node deployment optimization strategy based on CS algorithm. In: 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), pp. 621–625. IEEE (2019)
  - 6. ZainEldin, H., Badawy, M., Elhosseini, M., Arafat, H., Abraham, A.: An improved dynamic deployment technique based-on genetic algorithm (IDDT-GA) for maximizing coverage in wireless sensor networks. *J. Ambient. Intell. Humaniz. Comput.* **11**(10), 4177–4194 (2020)
  - 7. Banerjee, A., De, S.K., Majumder, K., et al.: Construction of energy minimized WSN using GA-SAMP-MWPSO and K-mean clustering algorithm with LDCF deployment strategy. *J. Supercomput.* (2022). <https://doi.org/10.1007/s11227-021-04265-7>
  - 8. Ye, W., Heidemann, J., Estrin, D.: An energy-efficient MAC protocol for wireless sensor networks. In: Proceedings. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 3, pp. 1567–1576. IEEE (2002)
  - 9. Alvi, A.N., Bouk, S.H., Ahmed, S.H., Yaqub, M.A., Sarkar, M., Song, H.: BEST-MAC: bitmap-assisted efficient and scalable TDMA-based WSN MAC protocol for smart cities. *IEEE Access* **4**, 312–322 (2016)
  - 10. Rajendran, V., Obraczka, K., Garcia-Luna-Aceves, J.J.: Energy-efficient collision-free medium access control for wireless sensor networks. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, pp. 181–192 (2003)
  - 11. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, 10-p. IEEE (2000)
  - 12. Biradar, R.V., Sawant, S.R., Mudholkar, R.R., Patil, V.C.: Multihop routing in self-organizing wireless sensor networks. *Int. J. Comput. Sci. Issues (IJCSI)* **8**(1), 155 (2011)
  - 13. Manjeshwar, A., Agrawal, D.P.: TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. In: IPDPS, vol. 1, no. 2001, p. 189 (2001)
  - 14. Latif, N.A., Tsimenidis, C.C., Sharif, B.S.: Energy-aware clustering for wireless sensor networks using particle swarm optimization. In: 2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 1–5. IEEE (2007)
  - 15. Vimalarani, C., Subramanian, R., Sivanandam, S.N.: An enhanced PSO-based clustering energy optimization algorithm for wireless sensor network. *Sci. World J.* **2016** (2016)
  - 16. Saha, B., Gupta, G.P.: Improved harmony search based clustering protocol for wireless sensor networks with mobile sink. In: 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), pp. 1909–1913. IEEE (2017)
  - 17. Arjunan, S., Sujatha, P.: Lifetime maximization of wireless sensor network using fuzzy based unequal clustering and ACO based routing hybrid protocol. *Appl. Intell.* **48**(8), 2229–2246 (2018)
  - 18. Javadpour, A., Adelpour, N., Wang, G., Peng, T.: Combing fuzzy clustering and PSO algorithms to optimize energy consumption in WSN networks. In: 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People, and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), pp. 1371–1377. IEEE (2018)
  - 19. Krishnan, M., Yun, S., Jung, Y.M.: Enhanced clustering and ACO-based multiple mobile sinks for efficiency improvement of wireless sensor networks. *Comput. Netw.* **160**, 33–40 (2019)
  - 20. Wang, J., Gao, Y., Liu, W., Sangaiah, A.K., Kim, H.J.: An improved routing schema with special clustering using PSO algorithm for heterogeneous wireless sensor network. *Sensors* **19**(3), 671 (2019)
  - 21. Vijayalakshmi, K., Anandan, P.: A multi-objective Tabu particle swarm optimization for effective cluster head selection in WSN. *Clust. Comput.* **22**(5), 12275–12282 (2019)
  - 22. Anand, V., Pandey, S.: New approach of GA-PSO-based clustering and routing in wireless sensor networks. *Int. J. Commun. Syst.* **33**(16), e4571 (2020)

23. Deepa, O., Suguna, J.: An optimized QoS-based clustering with multipath routing protocol for wireless sensor networks. *J. King Saud Univ. Comput. Inf. Sci.* **32**(7), 763–774 (2020)
24. Banerjee, A., et al.: Construction of effective wireless sensor network for smart communication using modified ant colony optimization technique. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies*. Lecture Notes in Networks and Systems, vol. 218. Springer, Singapore (2022). [http://doi.org/10.1007/978-981-16-2164-2\\_22](http://doi.org/10.1007/978-981-16-2164-2_22)
25. Dhasian, H.R., Balasubramanian, P.: Survey of data aggregation techniques using soft computing in wireless sensor networks. *IET Inf. Secur.* **7**(4), 336–342 (2013)
26. Al-Karaki, J.N., Ul-Mustafa, R., Kamal, A.E.: Data aggregation and routing in wireless sensor networks: optimal and heuristic algorithms. *Comput. Netw.* **53**(7), 945–960 (2009)
27. Shao-liang, P., Shan-shan, L., Yu-xing, P., Pei-dong, Z., Nong, X.: A delay sensitive feedback control data aggregation approach in wireless sensor network. In: *Lecture Notes in Computer Science*, pp. 393–400 (2007). [http://doi.org/10.1007/978-3-540-72590-9\\_54](http://doi.org/10.1007/978-3-540-72590-9_54)
28. Lu, Y., Comsa, I.S., Kuonen, P., Hirsbrunner, B.: Probabilistic data aggregation protocol based on ACO-GA hybrid approach in wireless sensor networks. In: 2015 8th IFIP Wireless and Mobile Networking Conference (WMNc), pp. 235–238. IEEE (2015)
29. Yu, B., Li, J., Li, Y.: Distributed data aggregation scheduling in wireless sensor networks. In: *IEEE INFOCOM 2009*, pp. 2159–2167. IEEE (2009)
30. Xiao, S., Li, B., Yuan, X.: Maximizing precision for energy-efficient data aggregation in wireless sensor networks with lossy links. *Ad Hoc Netw.* **26**, 103–113 (2015). <https://doi.org/10.1016/j.adhoc.2014.11.014>
31. De Paz Alberola, R., Pesch, D.: Duty cycle learning algorithm (DCLA) for IEEE 802.15.4 beacon-enabled wireless sensor networks. *Ad Hoc Netw.* **10**(4), 664–679 (2012). <http://doi.org/10.1016/j.adhoc.2011.06.006>
32. Sinde, R., Begum, F., Njau, K., Kaijage, S.: Refining network lifetime of wireless sensor network using energy-efficient clustering and DRL-based sleep scheduling. *Sensors* **20**(5), 1540 (2020)
33. Xu, M., Zhou, J., Lu, Y.: PHGWO: a duty cycle design method for high-density wireless sensor networks. In: 2019 IEEE International Conference of Intelligent Applied Systems on Engineering (ICIASE), pp. 28–31. IEEE (2019)
34. Liu, Y., Xiao, J., Li, C., Qin, H., Zhou, J.: Sensor duty cycle for prolonging network lifetime using quantum clone grey wolf optimization algorithm in industrial wireless sensor networks. *J. Sens.* **2021** (2021)
35. Subramanian, A.K., Paramasivam, I.: PRIN: a priority-based energy efficient MAC protocol for wireless sensor networks varying the sample inter-arrival time. *Wirel. Pers. Commun.* **92**(3), 863–881 (2016)
36. Iala, I., Dbibih, I., Zytoune, O.: Adaptive duty-cycle scheme based on a new prediction mechanism for energy optimization over IEEE 802.15. 4 wireless network. *Int. J. Intell. Eng. Syst.* **11**(5), 105 (2018)
37. Khan, A.A., Jamal, M.S., Siddiqui, S.: Dynamic duty-cycle control for wireless sensor networks using artificial neural network (ANN). In: 2017 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), pp. 420–424. IEEE (2017)
38. Paul, A., et. al.: A neuro-fuzzy based IDS for internet-integrated WSN. In: Bansal, J.C., Paprzycki, M., Bianchini, M., Das, S. (eds.) *Computationally Intelligent Systems and their Applications. Studies in Computational Intelligence*, vol. 950. Springer, Singapore (2021). [http://doi.org/10.1007/978-981-16-0407-2\\_6](http://doi.org/10.1007/978-981-16-0407-2_6)

# Prediction of Glaucoma Using Machine Learning-Based Approaches—A Comparative Study



Tiyasha Dhara, Arpan Adhikary, Koushik Majumder, Santanu Chatterjee, Rabindra Nath Shaw, and Ankush Ghosh

**Abstract** Glaucoma, also known as silent theft of sight, is one of the incurable eye diseases. It causes eye nerve damage and sometimes blindness also. Once patients lose their vision, it cannot be brought back. In most of the cases, glaucoma has no symptoms or pain. So early detection is very difficult. But to prevent permanent damage, early detection and diagnosis are highly critical. Some works have already been done on early detection and prediction of glaucoma using machine learning (ML) and deep learning (DL)-based approaches. In these works, researchers considered glaucoma-affected eye and healthy eye images for glaucoma detection and prediction. In this paper, a detailed survey of the existing works in this area has been carried out. We pointed out the drawbacks and based on that, we tried to find the future scope of the works that can enhance the accuracy in the timely detection of glaucoma which can save a lot of patients from future blindness.

**Keywords** Glaucoma · Machine learning (ML) · Deep learning (DL)

## 1 Introduction

Glaucoma is a progressive nerve damage disease for human eye. When intraocular pressure of the eye becomes high ( $>21$  mmHg), the ophthalmic nerve becomes damaged. This condition of eye is called glaucoma. Ophthalmic nerve is responsible for sending the images to human brain. But damage of this nerve may cause blindness sometimes. Glaucoma can happen for different causes. A fluid named aqueous humour usually flows through a channel inside eye. If the channel gets blocked or excessive fluid gets produced, then it causes glaucoma. Some other symptoms are—chemical injury to eyes, eye infections, blocking of blood vessels in the eye, etc.

---

T. Dhara · A. Adhikary · K. Majumder · S. Chatterjee

Department of Computer Science and Engineering, Maulana Abul Kalam Azad University of Technology, Kolkata, West Bengal, India

R. N. Shaw · A. Ghosh (✉)

University Center for Research & Development (UCRD), Chandigarh University, Mohali, Punjab, India

e-mail: [ankushghosh@gmail.com](mailto:ankushghosh@gmail.com)

Glaucoma mainly occurs for the people over 40, but sometimes young adults and children can be affected by this. Diabetes patients can also have glaucoma sometimes. As glaucoma has no early symptoms, it is very difficult to classify it at its early stage. But because of the chances of blindness, it is important to detect glaucoma accurately. To detect patients with high risk, machine learning approaches are very helpful. Manual glaucoma detection is time-consuming and biased. So, we need some smart automated system to capture patients' data on time. By using ML and DL techniques, glaucoma can be predicted more accurately and quickly. Patients' symptoms can be taken as input parameters to train the model. Thus, the model will be familiar with the glaucoma symptoms. Then it will be able to differentiate between glaucoma patients and healthy eye patients. In the recent studies, for predicting glaucoma, researchers used different techniques such as random forest, support vector machine, logistic regression, k-nearest neighbour, decision trees. Some deep learning techniques, i.e. XGBoost, CNN was also used to train the model in some of the works to predict glaucomatous eyes.

## 2 Literature Survey

### 2.1 Related Works

In the recent past, ML and DL classifiers have been taken into consideration by many researchers. Using these classifiers, models are capable of classifying glaucoma more accurately. In this section, some related works in this area and their results are discussed.

Bowd and Goldbaum [1] measured the model's performance by using ML classifiers, i.e. multi-layer perceptron (MLP) and support vector machine (SVM). They achieved the best sensitivity of 65% and specificity of 72% using MLP. Glaucomatous eye was classified from healthy people in this work.

Shuldiner et al. [2] predicted the rapid progression of glaucoma. They trained the model using initial visual field (VF) data. For classification, SVM, artificial neural network (ANN), random forest (RF), logistic regression (LR) and naïve Bayes (NB) were used. SVM gave them the best result where area under curve (AUC) was 0.72.

Devecioglu et al. [3] proposed a model (self-organized operational neural network) for early glaucoma prediction. They compared self-ONN with deep CNN network. The result showed that the proposed model predicted better than CNN. Three benchmark models were used for this study.

Mary et al. [4] built a deep learning (DL) system, i.e. convolution neural network (CNN) for automatic glaucoma diagnosis. They achieved the AUC of 0.831 from ORIGA dataset and the AUC of 0.887 from SCES dataset.

Kim et al. [5] built a ML model that gave a very accurate prediction for glaucoma diagnosis. The researchers trained their model using RF, C5.0, SVM and KNN. Using RF, they achieved the best accuracy of 0.98.

Oh et al. [6] developed a ML predictive model to diagnose glaucoma. SVM, C5.0, XGBoost and RF algorithms were used for the prediction. XGBoost gave the best sensitivity of 0.941, accuracy of 0.947, specificity of 0.950 and AUC of 0.945.

Tekouabou et al. [7] built a predictive model based on different SVM kernels. The model was used to optimize automated glaucoma diagnosis. Patient's visual field data was taken for diagnosis. They achieved the best accuracy of 91.71% using linear SVM kernel.

Civit-Masot et al. [8] developed a diagnostic aid tool for glaucoma detection using eye fundus images. Two subsystems were trained and then tested independently. In the first subsystem, ML techniques were applied for optical disc and cup detection. Then the results were combined, and the features were extracted. In the second subsystem, CNN model was used for glaucoma detection. The results of both systems were combined to classify glaucoma positive cases. They achieved the best accuracy of 0.90 using ResNet50 architecture.

Grassmann et al. [9] proposed an automated classification algorithm for glaucoma prediction based on colour fungus images. Different CNN architectures were used for training the model. They achieved the overall accuracy of 63.3%.

An et al. [10] developed an algorithm to diagnose open-angle glaucoma. They took three-dimensional optical coherence tomography (OCT) data and colour fungus images as input data. CNN model was used for classification. They achieved an average accuracy of 95%.

Lee et al. [11] built a progression model to predict normal-tension glaucoma. For this, visual field examination was done for each subjects twice within 3 months of intervals. Functional changes on VF tests were classified for each eye. RF and extra-trees classifier were used for classification. RF gave the AUC of 0.811, and extra-trees classifier gave them the AUC of 0.811.

Son et al. [12] developed a DL model to find abnormality in the eye from retinal fungus images. One benchmark model and two external models were used for training and testing the model. The researchers developed a deep learning model for classification and highlighting of lesions. For benchmark dataset, they achieved the AUC of 96.2–99.9%. For external dataset, they achieved the AUC of 94.7–96.5%.

Thakur and Juneja [13] took some hybrid features from fungus images to classify glaucoma. For classification, SVM, NN, kNN, NB and RF were used. The best accuracy of 97.2% was achieved by SVM.

Asaoka et al. [14] classified visual fields between glaucoma patients with healthy people. Deep feed-forward neural network (FNN) was used as predictor. ML algorithms such as RF, SVM, gradient boosting and NN were used for classification. Deep FNN gave the best AUC of 92.6%.

Wang et al. [15] detected glaucoma using the full retinal nerve fibre layer (RNFL) thickness maps. SVM, kNN, two CNN models (RestNet-18 and GlaucomaNet) were used for this. RestNet-18 gave them the best accuracy of 0.905.

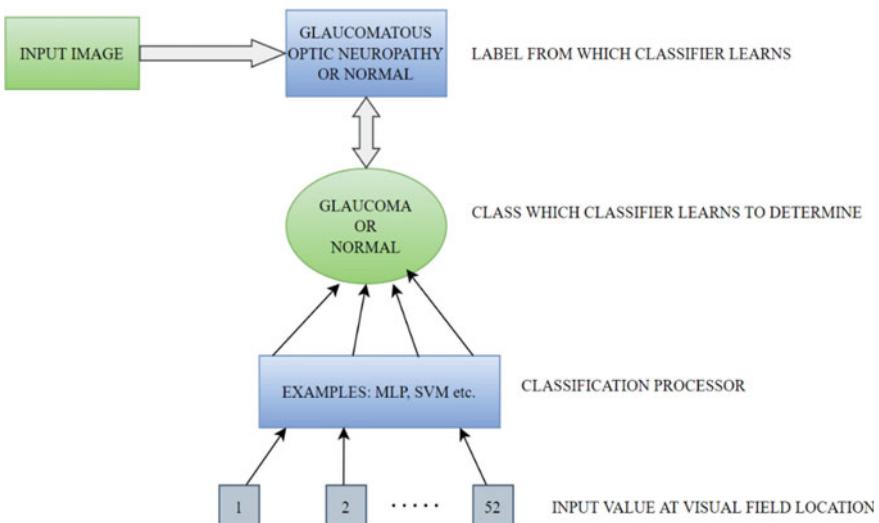
Ting et al. [16] developed a deep learning system (DLS) to detect diabetic retinopathy and related eye diseases using retinal images. The proposed system gave the maximum AUC of 98.0% for African American eye disease study. It was a

community-based study. For clinical-based study, the maximum obtained AUC was 0.983.

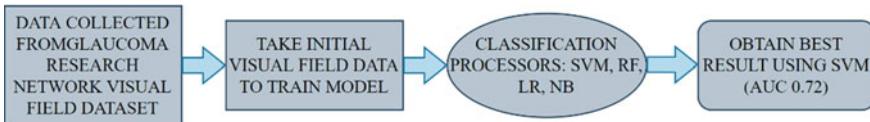
Mehtha et al. [17] built an automated system to detect glaucoma using clinical data and multi-modal retinal images. The data was collected from colour fundus photos (CFPs) and macular optical coherence tomography (OCT) scans. They built multiple models using clinical data and achieved the maximum AUC of 0.97.

## 2.2 Methodology Used in the Surveyed Papers

Bowd and Goldbaum [1] measured the performance of the SVM and MLP model. They achieved the best sensitivity and specificity using MLP. Receiver operating characteristics (ROC) curve and area under the ROC curve (AUC) were used to show the graph between true positive rate (TPR) and false positive rate (FPR). Here, TPR is denoted as sensitivity and FPR is denoted as specificity. A back-propagation neural network was trained based on visual field value. Multi-layer perceptron (MLP) was used to identify the glaucomatous standard automated perimetry (SAP) and visual field (VF) using the Humphrey 30-2 test of 76 points. Using dimensionality reduction, the researchers reduced the irrelevant features. To increase the accuracy and reduce the error rate, two resampling techniques, i.e. “n-fold cross-validation” and “Bootstrapping”, were used. In the case of “n-fold cross-validation”, the total dataset was divided into nsamples. MLP gave them best sensitivity of 65% and specificity of 72% (Fig. 1).



**Fig. 1** Methodology for glaucoma classification [1]

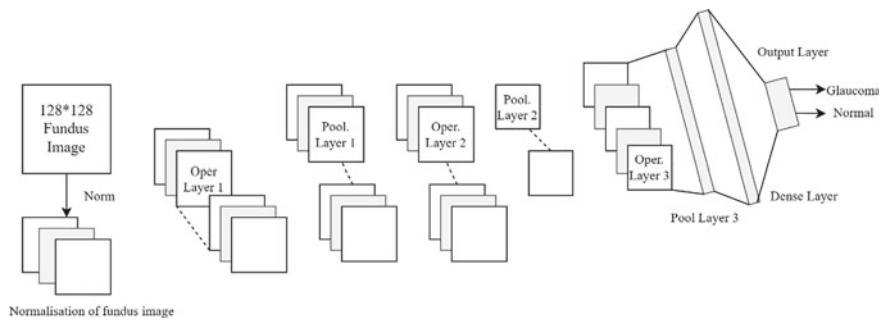


**Fig. 2** Proposed system diagram for glaucoma diagnosis [2]

Shuldiner et al. [2] predicted the risk of rapid progression of glaucoma. Initial VF and age were used for this as input parameters. The performance of test set was evaluated using AUC. Comparison of model performance was based on initial VF and first two VFs. 80% of the dataset was used as training data, and 20% was used as test data. For classification, SVM, RF, LR and NB were used. SVM model gave the most accurate result. For first two VF, best accuracy was obtained by RF. When SVM model was trained by one VF and RF models were trained by two VF values, the models' performance was almost same. They collected the data from “Glaucoma Research Network Visual Field Dataset”. For SVM, they used radial basis kernel for the rapid progressors and for the non-rapid progressors. They used three different sets of training and test data. Model performance was similar with the “fivefold cross-validation”. SVM and ANN gave them the best AUC of 0.72 (Fig. 2).

Devecioglu et al. [3] proposed self-organized operational neural networks (Self-ONNs) to detect glaucoma at the early stage. Fundus images were used for this work. The researchers took three datasets, i.e. ESOGU, ACRIMA and RIM-ONE, to train and test the proposed model. ESOGU dataset has 4405 healthy fundus images and 320 glaucoma fundus images. They proposed an ONNs model, which is a heterogeneous network model. This model encapsulates distinct nonlinear neurons. ONNs model is derived from generalized operational perceptrons (GOPs). This network gives better classification, when the data is scarce. It is also computationally efficient. The proposed model is based on Taylor-series function approximation. The model analysed the normalized RGB digital fundus image. It had three hidden layers. There were 32, 16 and 8 neurons in the hidden layers. The model extracted the optimal features from the raw fundus images by nonlinear transformation. Input layer of the model was 3 for three input colour channels. Output layer size was 2 for the number of classes. They compared the accuracy of their model with CNN. Using CNN, they achieved 68.0% of accuracy. The model gave them the accuracy between 71.5 and 75.3% for RIM-ONE dataset. For ESOGU dataset, they achieved 94.5% of accuracy with CNN and 95.0–100.0% of accuracy with self-ONN. For ACRIMA dataset, the model gave an accuracy of 81.4–91.8% where CNN gave them an accuracy of 80.0%. A tenfold cross-validation technique was applied to the benchmark models separately. It gave them better performance estimation and prevented over-fitting. For each fold, 5 BP runs were performed (Fig. 3).

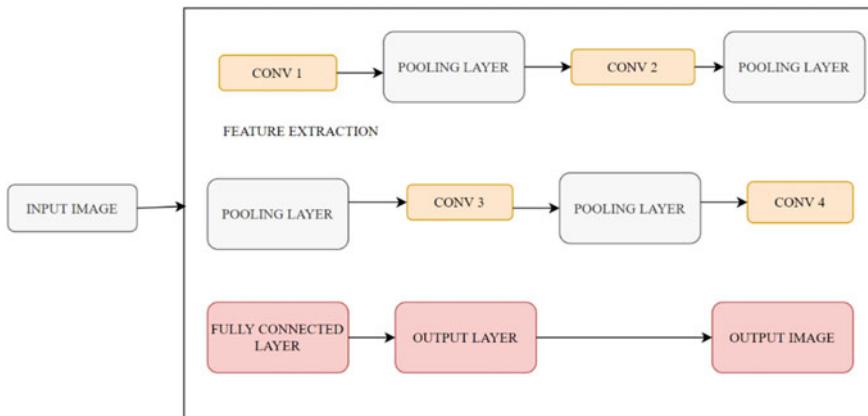
Mary et al. [4] used a convolution neural network (CNN) to automatically diagnose glaucoma from healthy eye. They used ORIGA and SCES dataset for this. ORIGA dataset had 166 glaucoma images and 4820 regular fundus images, whereas SCES dataset had 46 glaucoma images and 1672 regular fungus images. Glaucoma



**Fig. 3** Proposed glaucoma diagnosis framework [3]

identification was done using digital fundus image. Glaucoma was diagnosed using patient records, field vision monitoring, intraocular pressure and optical disc (OD). To implement the prediction model, the researchers used six CNN layers. The last layer predicted the glaucoma using “softmax classifier”. Softmax classifier is used to gather raw data and classify the output to 0 or 1. In each class, decimal probabilistic values are allocated by softmax. Softmax is implemented before the output layer. With the proper use of kernel, convolution step is performed over the input data and generates a feature map. To construct the feature map better, “Max-polling” was used. The dataset was extended with the help of “mark protection transformations”. The image overfitting problem was solved using configuration combination. The comparison was done between CNN and “state-of-the-art reconstruction-based approach”. 648 images were provided in the dataset, and 548 were used for testing. They achieved the AUC of 0.842 and 0.867 for ORIGA and SCES dataset, respectively. Each fundus image was diagnosed in the clinic. AUC curves were used to check the efficiency. To overcome the overfitting problem, overlap-pooling layers and response-normalization layers were followed frequently (Fig. 4).

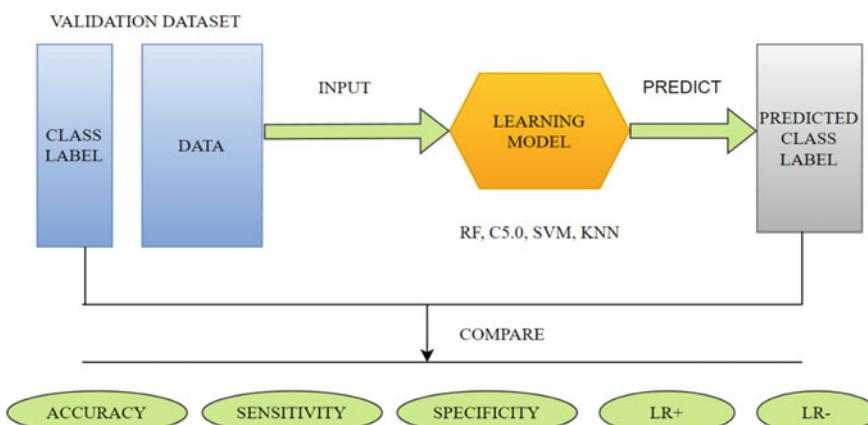
Kim et al. [5] developed the glaucoma prediction model. They used four different kinds of ML algorithms, i.e. C5.0, random forest (RF), support vector machine (SVM) and k-nearest neighbour (KNN). The best performance was achieved using RF (accuracy = 0.98, sensitivity = 0.983, specificity = 0.975, AUC = 0.979). They collected three types of medical records, i.e. “retinal nerve fibre layer (RNFL) thickness”, “visual field (VF) test parameter” and “general ophthalmic examination” to develop the model. After data collection, essential features were extracted. Then t-tests were applied to evaluate each feature. The dataset was split into 80% training data and 20% test data. In the last step of feature selection process, they performed a classification test on each combination of features. C5 algorithm was used on the collected feature. Age, ocular pressure, cornea thickness, RNFL4.mean, GHT, MD and PSD were in the final feature list. After classification, the mentioned features were compatible in glaucoma prediction. RF, C5.0 and SVM had strongly predicted glaucoma patients. Pruning was done automatically while constructing decision tree (DT). For prediction rules, C5.0 model used “RNFL4.mean, ocular\_pressure, MD and PSD” features.  $\text{RNFL4.mean} \leq 89.34$  and  $\text{MD} \leq -4.02$  were grouped as glaucoma patients. “Visual



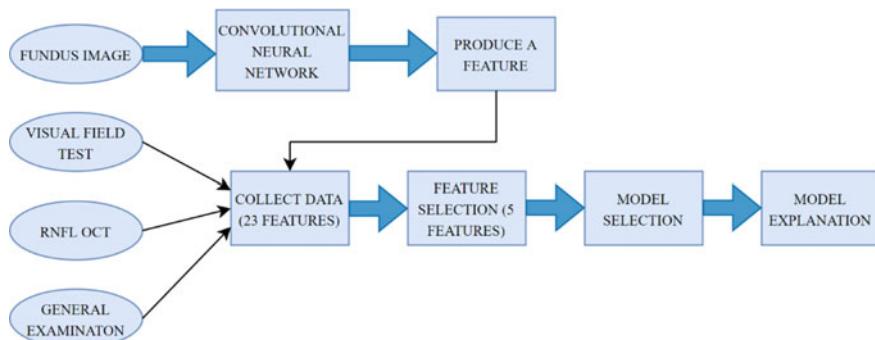
**Fig. 4** Methodology for glaucoma detection and prediction [4]

field index”  $\geq 97$  were classified into early glaucoma stage. The rate of early glaucoma diagnosis was 91.7%. Using RF, they achieved the best accuracy of 0.98, sensitivity of 0.983 and specificity of 0.975 (Fig. 5).

Oh et al. [6] built a ML-based prediction model for glaucoma. They achieved the best accuracy of 0.947 using XGBoost model. SVM, C5.0 and RF were also taken as benchmark models for classification process. Five features were selected for prediction—pattern standard deviation (PSD), retinal nerve fibre layer (RNFL) superior, RNFL inferior, RNFL temporal, intraocular pressure. The data collection was done from Gyeongsang National University Hospital and contained 1624 cases where 975 cases were of glaucoma. The dataset was split into 80% training samples and 20% test samples. Initially, there were 22 features in the dataset. Using chi-square feature selection measure the researchers selected 10 informative features



**Fig. 5** Methodology for glaucoma detection using visual field [5]

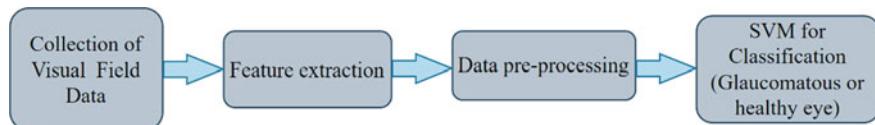


**Fig. 6** Methodology for glaucoma prediction using visual field and fundus images [6]

from the base dataset. SVM, RF, C5.0 and XGBoost classification algorithms were tested for the glaucoma prediction model. XGBoost gave the best accuracy of 0.947, sensitivity of 0.941, specificity of 0.950 and AUC of 0.945. They classified the output in different classes, i.e. healthy patient, glaucoma patient, increasing CSD and RNFL defects, healthy CDR and the absence of RNFL. Their model predicted glaucoma or non-glaucoma based on the RNFL thickness and VF test indices (Fig. 6).

Tekouabou et al. [7] built a computerized system to detect glaucoma. VF data of fundus eye was collected from Glaucoma Center of Semmelweis University. The performance of the model was determined based on the terms of accuracy, precision and AUC. ANN, RBF kernel, kNN and SVM were used to find the best accuracy. Linear, nonlinear (RBF) and polynomial kernel were used for classification. Due to unbalanced dataset, the researchers evaluated the model using AUC and precision performance measurement. For optimal classification, parameter optimization was done. For RBF kernel, the obtained precision was 100% for both test and validation. For linear kernel, it was 86.78% for test and 85.25% for validation. For polynomial kernel, precision obtained was 80.14% for test and 78.04% for validation. They achieved a precision of 100% using RBF kernel and accuracy of 91.7% using linear kernel. AUC of 88.30% was achieved using polynomial kernel (Fig. 7).

Civit-Masot et al. [8] trained a CNN network using brute force. Using eye fundus images, the researchers developed a diagnostic aid tool to detect glaucoma eye. Two subsystems were trained and tested independently. Firstly, segmentation and ML technique were applied to detect optic disc and cup. After detecting, they combined the features to extract the physical as well as positional feature. Secondly, they

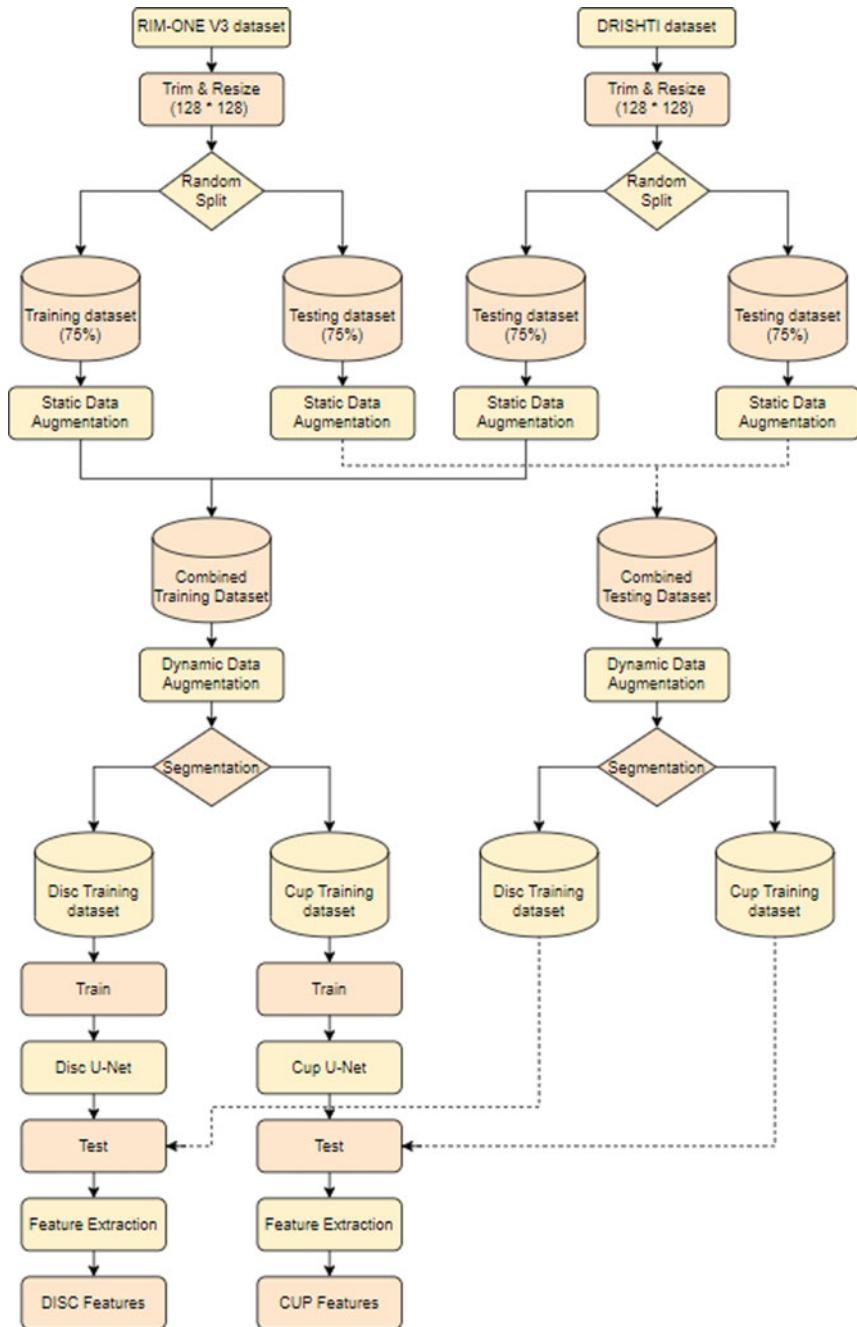


**Fig. 7** Methodology for glaucoma prediction using visual field [7]

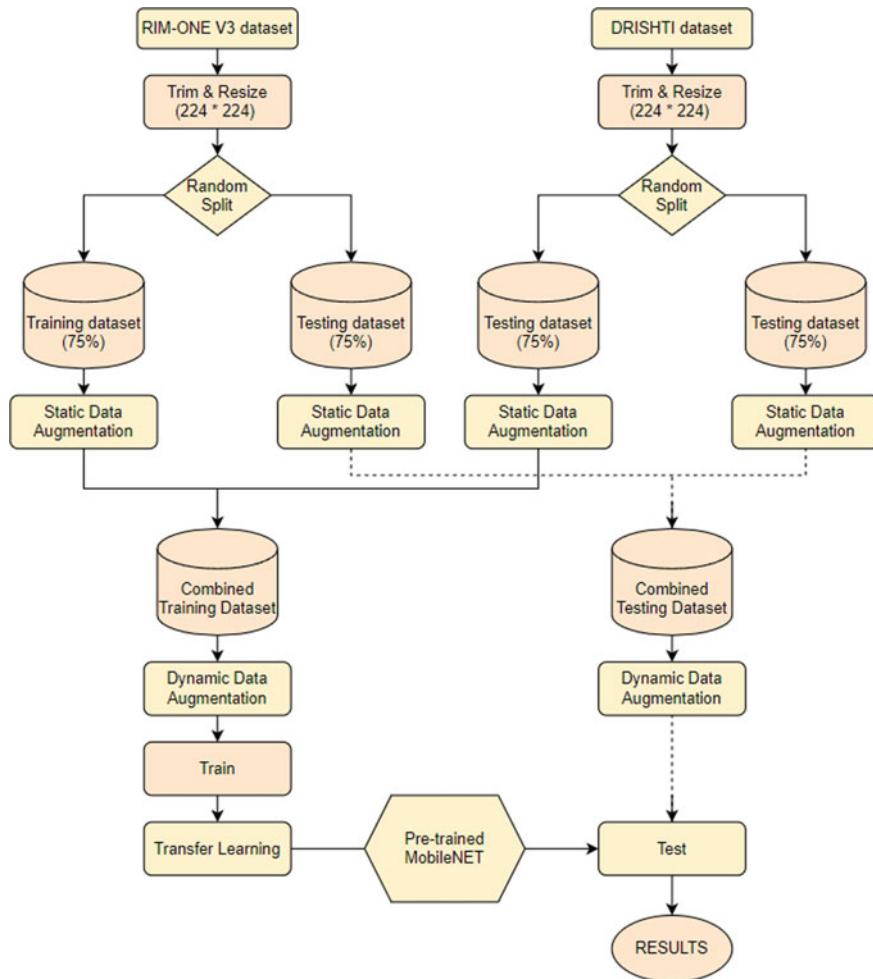
applied the transfer learning technique to the CNN model. The results were combined together to classify positive cases and improve the result. Feature reduction was done using ensemble method. RIM-One V3 dataset was collected from the University of La Laguna, and DRISTI-GS dataset was collected from the Aravind Eye Hospital. 75% of the collected data were used for training purposes and 25% image data for testing purposes in both the cases. In pre-processing stage, image trimming was done.  $128 \times 128$ -pixel images were used for segmentation. In static data augmentation stage, training image data was modified. In dynamic data augmentation phase, image zooming and rotating were done. In segmentation network, U-Net was used to segment disc and cut. Training process was done in 100 epochs. In the last stage, post-processing was done. For direct classification system,  $224 \times 224$ -pixel images were used in the pre-processing stage. In static dynamic augmentation stage, training was done on the combined dataset. In classification, MobileNet V2 network was used. The training stage was done in 50 epochs. Output of each system was given as input of the diagnosis aid tool. They achieved 90% specificity and 75% sensitivity through the segmentation system. Also, they achieved better ROC curve 0.93 using U-Net classifier than CDR-based classifier (Figs. 8, 9, 10 and 11).

Grassmann et al. [9] presented an automated classification algorithm for glaucoma detection and prediction. Participants were above 55 years of age. The subjects were separated into 13 different classes. Nine of which were AREDS stages, three were late AMD stages and one was for ungradable images. They took independent KORA dataset for the study. The proposed algorithm positively detected 84.2% of all fundus images which had some definite sign of both early or late AMD. The algorithm was able to classify 94.3% healthy images correctly. This method consisted of 4 stages. In the first stage, researchers pre-processed the colour fundus images. In second step, they used the output of the previous step to train multiple CNNs independently. Their aim was to train the CNN to compare the output with the true class iteratively. In third step, the RF algorithm was trained to build an ensemble-based model. In the final step, the final model was applied to predict the AREDS testing data. Six different CNN models were used to train the images. By combining different network architectures, they achieved an accuracy of 63.3% (Fig. 12).

An et al. [10] developed an algorithm for open-angle glaucoma diagnosis. The algorithm was based on the three-dimensional optical coherence tomography and the colour fundus. This system is capable of improving glaucoma diagnostic accuracy. The researchers took 208 glaucomatous samples and 149 healthy subjects' data. They created the thickness and deviation map with segmentation. Firstly, optic discs' fundus images were converted to grayscale format. Then, they mapped the fibre layer of disc retinal nerve. In the next step, the macular ganglion cell complex was converted to disc RNFL and macular GCC deviation. Data augmentation was performed to train the CNN model. To combine each CNN models' result, they trained the model using RF classifier. A tenfold cross-validation was done for the evaluation of the models. Using CNN, they achieved the AUC of 0.940 for colour fundus images, 0.942 for RNFL, 0.944 for GCC, 0.949 for RNFL deviation and 0.952 for macular GCC. The researchers took VGG19 CNN model which consisted of 19 layers and is commonly



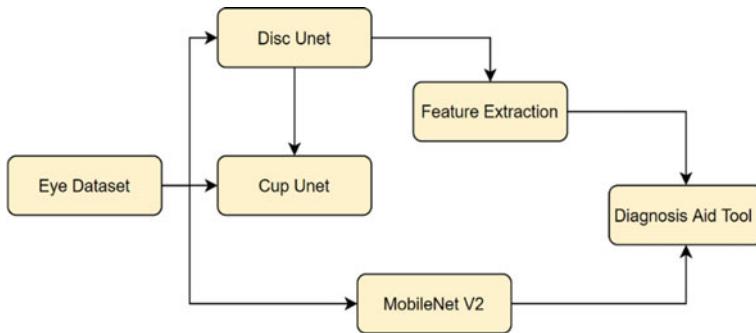
**Fig. 8** Disc and cup segmentation subsystem [8]



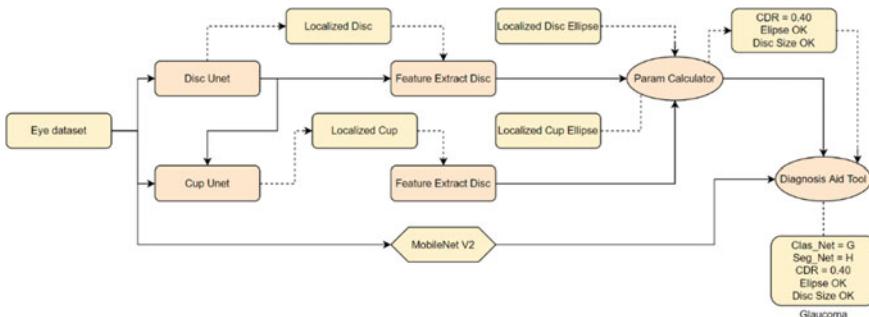
**Fig. 9** Eye fundus image classification [8]

used for image classification. For each feature, a CNN model was trained separately. After classification, performance evaluation and cross-validation were done (Fig. 13).

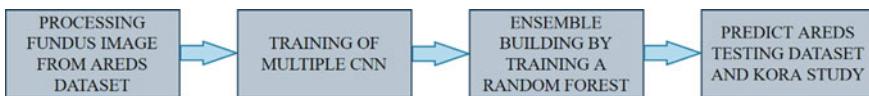
Lee et al. [11] predicted the progression of normal tension glaucoma (NTG). For this study, they took 155 NTG eyes' data of subjects aged between 20 and 40. From which, 110 samples were used for training and 45 for testing. A complete ophthalmic examination was done on all the subjects. This examination included visual acuity assessment, slit lamp biomicroscopy, gonioscopy, dilated stereoscopic examination, etc. Other test conducted on the subjects were central corneal thickness (CCT) measurement, axial length measurement, digital colour stereo disc photography, red-free RNFL photography, optic nerve head (ONH). VF examination was done twice on the patients for 36 months. For classification, the researchers took RF



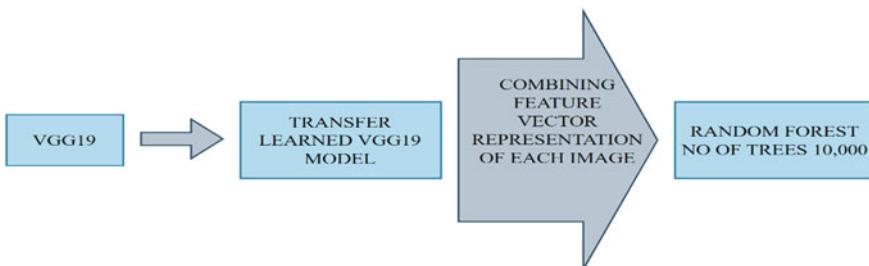
**Fig. 10** Diagnosis tool architecture [8]



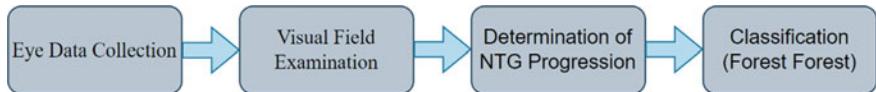
**Fig. 11** System diagram for glaucoma diagnosis [8]



**Fig. 12** Block diagram for glaucoma diagnosis [9]



**Fig. 13** Methodology for glaucoma diagnosis using CNN [10]

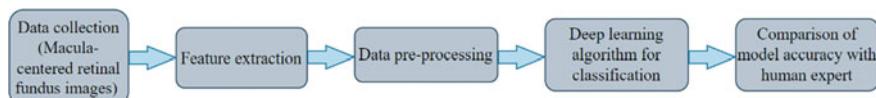


**Fig. 14** Methodology for normal-tension glaucoma diagnosis [11]

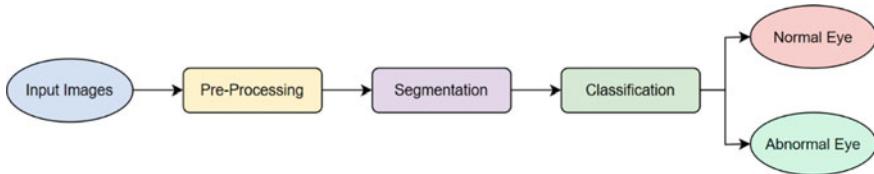
classifier and extra-trees classifier. As the number of subjects was less, so threefold cross-validation technique was performed on the training set. In this process, the training set was partitioned into three parts, of which only a single part was taken to validate the model. The remaining parts were used as training sets. This process was repeated 3 times. They achieved an AUC of 0.811 using RF and extra-trees classifier gave them AUC value of 0.881 (Fig. 14).

Son et al. [12] developed and evaluated a deep learning model to find abnormality in retinal fundus images. They used 103,262 images, from which 309,786 readings were used for training the model. For testing, the Indian Diabetic Retinopathy Image Dataset (IDRiD) and e-ophtha datasets were used. An external dataset (Messidor) was used for model comparison. Primary dataset was divided into 9:1 ratio where 10% was taken as test dataset. The remaining 90% was divided into 9:1 ratio for training and validation, respectively. The researchers used the training set for learnable weight optimization and used the validation set for tuning the hyperparameter. A deep learning architecture was built for classification, and a heatmap was used for lesions highlighting. A single-channel low-resolution image is called a lesion heatmap. Its value ranges between 0 and 1. This image was superimposed on the original image to detect glaucoma. Any image was classified as glaucoma if the lesion heatmap reached a certain threshold. Twelve neural networks were used in the screening system to detect glaucoma. For in-house dataset, the achieved AUC was high at 96.2–99.9%. For IDRiD dataset, the AUC ranged from 95.7 to 98.0%. For e-ophtha dataset, AUC was from 94.7 to 96.5%. This algorithm was consistent for haemorrhage, membrane, macular hole, RNFL, glaucomatous disc change, no glaucomatous disc changes, etc (Fig. 15).

Thakur and Juneja [13] derived some hybrid features from structural and non-structural data to classify fundus images. The structural features are disc damage likelihood scale (DDLS), cup-to-disc ratio (CDR), etc. Non-structural features are grey level run length matrix (GLRM), grey level co-occurrence matrix (GLCM), first-order statistical (FoS), higher-order spectra (HOS), higher-order cumulant (HOC), etc. For this study, DRISHTI-GS dataset was used, which had 101 retinal images. These images were labelled as glaucoma and non-glaucoma. In RIM-One dataset, 159 retinal fundus images were labelled as “glaucoma” and “non-glaucoma”. Total 260



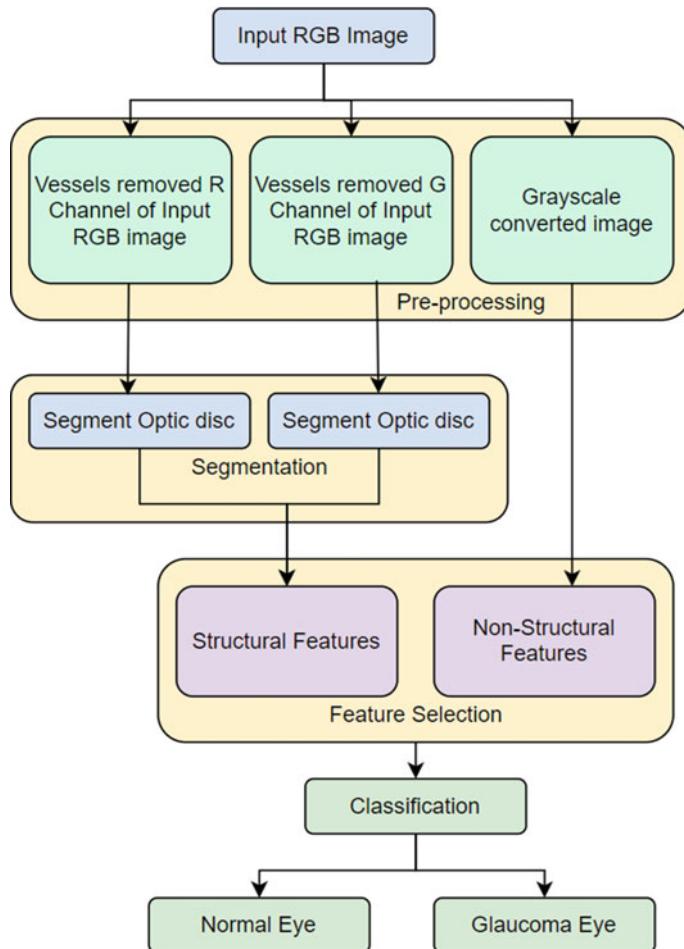
**Fig. 15** Methodology for glaucoma diagnosis using CNN [12]



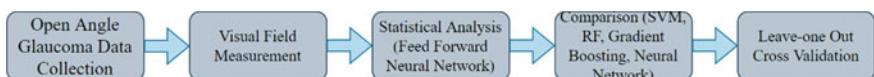
**Fig. 16** Generalized CAD system [13]

images were taken for the experiment. Initially, the input images were pre-processed. Outliers were removed, and segmentation was performed to extract the features from the input images. Classification was performed using SVM, neural network (NN), kNN, NB and RF. In the initial step, the input images were converted to red, blue and green channels for analysing the best-suited channel. Red and green channels were taken for segmentation. Level set-based adaptively regularized kernel-based intuitionistic fuzzy C-means (LARKIFCM) was used for disc and cup segmentation. Some significant features were taken for classification. Structural features were taken from segmented region of retinal images. Non-structural images were taken from grayscale converted retinal image directly. For feature selection, different techniques, i.e. information gain (IG), gain ratio (GR), correlation, relief feature ranking were used. The best accuracy was achieved using SVM. When 61 features were taken for evaluation, the accuracy was 91.6%. For 10 features, the achieved accuracy was 97.2%. All the other classifiers gave the accuracy between 76.3 and 88.8% for 61 features and 89.6 and 94.4% for 10 features. Threelfold and fivefold cross-validation techniques were used to validate the model performance (Figs. 16 and 17).

Asaoka et al. [14] classified visual fields (VFs) of perimetric open-angle glaucoma (OAG) from healthy eyes. For this work, 171 VF features were taken from 53 eye glaucoma samples and 108 samples from 87 healthy eyes. The main prospective for this work was to distinguish perimetric glaucoma VFs (PPGVFs) from healthy VFs. For this, the patients with open-angle glaucoma, who underwent VF measurement were identified first. The dataset was collected from University of Tokyo Hospital, Tokyo, Japan and Tajimi Municipal Hospital, Japan. In this work, the proposed model had 10 nodes in the first hidden layer and 3 nodes in the second hidden layer. At the output layer, a feed-forward neural network was used to discriminate between healthy VFs and PPGVFs. Two hidden layers were used for back-propagation technique. The researchers used sigmoid neural network. For cross-validation, leave-one-out cross-validation was done. In this method, the patient's data with all VFs from both eyes were used as validation set. For training set, all VFs from the remaining participants were used. This process was repeated for 200 iterations. To compare the performance, ML classifiers, i.e. RF, gradient boosting, SVM and neural network (NN), were used. AUC was used to evaluate the accuracy of discrimination for every method. AUC was obtained using each classifier. AUC of 92.6% was obtained using deep FNN. AUC obtained using other ML classifiers are as follows: 79.0% using RF, 77.6% using gradient boosting, 66.7% using neural network (Fig. 18).



**Fig. 17** Flow diagram of the methodology [13]

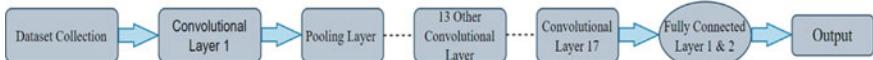


**Fig. 18** Methodology for open-angle glaucoma diagnosis [14]

Wang et al. [15] assessed the diagnostic accuracy of different ML models using full retinal nerve fibre layer (RNFL). The main purpose of this study was to classify glaucomatous eyes from healthy eyes using entire RNFL thickness maps. For this study, 93 glaucoma images from 69 patients and 128 healthy peoples' data were taken. The dataset was collected from Los Angeles Latino Eye Study (LALES). Firstly, local features were evaluated and extracted. Clinical history and examination, optic disc



**Fig. 19** Methodology for glaucoma diagnosis using GlaucomaNet [15]



**Fig. 20** Methodology for glaucoma diagnosis using ResNet-18 [15]

photography and VF results were independently determined. Four ML models, i.e. SVM, KNN and two conventional ML models, were used in this study. Two deep learning models, i.e. ResNet-18 and GlaucomaNet, were also used. GlaucomaNet network was designed by the researchers which had three convolutional layers. Each layer had an activation layer, a pooling layer and two fully connected layers. At the output layer, the model classified glaucoma from healthy eye. Data augmentation was done for both the DL models. All the models were tested using fivefold cross-validation technique. For each run, the dataset was divided into 8:2 ratio for training and validation, respectively. Total 5 iterations were done to achieve best accuracy. Based on the probability of each model's output, AUC was calculated. The best accuracy of 0.905, sensitivity of 0.860, specificity of 0.928 were achieved using ResNet-18 network. SVM gave them an accuracy of 0.869. KNN gave an accuracy of 0.860 and GlaucomaNet gave an accuracy of 0.878 (Figs. 19 and 20).

### 3 Comparative Study

From the above-discussed papers, we found different classifiers were used to detect glaucoma and its types. In some papers, fundus images were taken as input data. Whereas, in some other papers, retinal images were taken to find the abnormality in the eye. In most of the papers, ML models were used to classify glaucomatous eye from healthy eye. In other papers, the researchers implemented DL models to find the glaucomatous eye. In this section, a detailed comparison on the papers have been carried out.

In [2, 5–7], the researchers collected data from clinics and hospitals. In [2], SVM, ANN, RF, LR and NB were used to detect glaucoma. They achieved the best accuracy using SVM. SVM was also used in [13–15] for classification. SVM is used for both classification and regression, but it is mainly used for classification. To put the data points in correct class, the finest decision boundary is created. SVM implementation is done using kernel that converts data into its' required form. It uses kernel trick technique for nonlinear data. SVM has better generalization capability to prevent the over-fitting problem. It works better for regression problems. But SVM has certain drawbacks. For large dataset, it performs less efficiently. For target class's

overlapping, its performance is reduced. SVM also underperforms if the number of features for each data point is more than training data samples. Random forest is better than SVM for classification. It consists of multiple decision trees, and the tree with maximum votes becomes the model's decision. Random forest is suitable for large datasets. It is easier for interpretation. But decision trees work less efficiently for continuous variables. Any small change in data affects the tree structure. In [2] and [7], radial basis function (RBF) kernel was used. It is a commonly used kernel as it is simple. In these papers, SVM was used to predict the rapid progression of glaucoma with a single VF data. Researchers tried to combine all the classifiers mentioned to achieve the best performance. But they obtained less AUC using LR and ANN. For ANN, they got less accuracy due to heavy computational burden. For a smaller number of features, they got less accuracy using RF. In [6] SVM, C5, RF, XGBoost were used to measure the model's accuracy. It provides a parallel tree boosting and leading machine learning library for regression, classification and ranking problems. XGBoost handles large datasets well, and execution speed is also high. It is faster than gradient boosting. XGBoost has some inbuilt features for handling missing data.

In [3, 4, 8–10], CNN was used to classify the glaucomatous eye but different approaches were used. In [4], CNN network was constructed. CNN is used for image recognition, object detection, image classification, face recognition, etc. In CNN models, each input parameter is passed through a series of convolution layers for training and testing [18–20]. The layers are kernels or filters, pooling and fully connected layers. Data augmentation is done to extend and translate the image data to produce a horizontal reflection of data. To solve the over-fitting problem, configuration combinations are dropped out. In [8], a combined result of two dual CNN subsystems was used. Then ensemble approach was used to predict glaucoma. Ensemble is used to combine the multiple model's prediction results and find the best predictive performance. For segmentation, they used generalized U-Net. U-Net is a CNN architecture, that is used to segment the image data. It is a U shape symmetric architecture that consists of a contracting path and expansive path. CNN was also used in [10]. Transfer learning of CNN was used as input. The researchers performed dropout and data augmentation to train the CNN. After combining the results of CNN, they used RF trained for classification and removing the second fully connected layer of CNN.

In [11], RF and extra-trees classifiers were used on the collected dataset. This is an ensemble learning technique that takes the output from multiple decision trees and collects them in a forest to give the classification result. Both of these classifiers compose a large number of decision trees. Extra-trees use greedy algorithm to select the optimal split point, which is chosen randomly. While random forest uses bootstrap replicas, extra-trees use the whole sample data. As extra-trees form a greater number of trees than RF, the estimation is also better for extra-trees classifier. For data collection and storing, no proper security measures were discussed in the above papers. It is a major drawback for collecting sensitive data. Data should be collected and maintained with proper authentication and authorization (Table 1).

**Table 1** Comparison of the above-discussed papers

Paper name	Method used	Result obtained	Advantage	Disadvantage
Boyd and Goldbaum [1]	MLP and SVM were used to find glaucoma	96% specificity was obtained	Deep learning technique more accurately classifies glaucoma than machine learning techniques	Probability of risk progression for large dataset was not defined
Shuldiner et al. [2]	SVM, ANN, RF, LR, NB classifiers were used	Getting highest AUC (0.72) (Area Under Curve) with SVM and Artificial Neural Network	Use of initial VF value for prediction gave them the best result	Addition of MD slope value with VF value reduced the performance accuracy
Devecioglu et al. [3]	CNN on 3 benchmark datasets (Self-ONN)	They achieved better accuracy using Self-ONN than CNN. The achieved accuracy was 81.4–100%	Reduced complexity than deep CNN model  It can be easily integrated into the decision support system to detect glaucoma in real-time	Classification accuracy is not that high, it can be further improved by combining a segmentation network properly  Deep CNN with large no of neurons and more depths performed better than Self-ONN model
Mary et al. [4]	To extract or identify the features of glaucoma, CNN was used	Achieved AUC value of 0.842 and 0.867 from ORIGA and SCES dataset, respectively	The model defined unequal characteristics that help to identify secret symptoms of glaucoma	To enhance the accuracy, data raise approach can be used
Kim et al. [5]	They used four different ML algorithms, i.e. C5.0, RF, SVM and KNN for glaucoma prediction model	The best performance was achieved using RF (Accuracy = 0.98, sensitivity = 0.983, specificity = 0.975, AUC = 0.979)	In several cases, RF Classifier gave more accurate outcome than DT  Reduced the over-fitting problems of data  False negative rate of their model is also very low	It is complicated as well as the implementation is also very complex in nature. So, it gave slow prediction  Clinical test and model prediction results differed in some of the cases

(continued)

**Table 1** (continued)

Paper name	Method used	Result obtained	Advantage	Disadvantage
Oh et al. [6]	To build a predictive model SVM, C5.0, RF and XGBoost are used	XGBoost model gave the best accuracy = 0.947, sensitivity = 0.941, specificity = 0.950, AUC = 0.945	Using large number of feature decision making was easy	In some cases, clinical diagnosis detected pre-perimetric glaucoma, but their model predicted healthy eye
				Clinically diagnosed healthy data was predicted as glaucoma by the model
				The proposed model also produced 5.3% missed prediction cases
Tekouabou et al. [7]	Predictive model was trained using SVM kernels	In RBF kernel precision was 100% for both test and validation	Implementation of computerized model easily detect early-stage glaucoma	Dataset was very small in size and attributes were also very less. So, CNN was less suitable
		Linear kernel gave the best performance on accuracy = 91.7%		
Civit-Masot et al. [8]	Two CNN-based subsystems were used, and results of both systems were combined together. Ensemble approach to detect positive cases of glaucoma	Using ensemble, they got an accuracy of 0.88	Achieved more accurate result due to the use of ensemble approach than prior work	To improve the model performance, ensemble model needs to train with more data
Grassmann et al. [9]	CNN model was used to detect the glaucomatous patient	Achieved 63.3% accuracy	The algorithm took deep learning into consideration which is better in image classification than ML techniques	Accuracy is very less which can be further improved

(continued)

**Table 1** (continued)

Paper name	Method used	Result obtained	Advantage	Disadvantage
An et al. [10]	A combined result of CNN was used as input of RF to classify the glaucomatous eye	Got AUC of CNN 0.9740 for colour fundus image, 0.942 for RNFL thickness, 0.944 for macular GCC thickness, 0.949 for RNFL deviation, 0.952 for macular GCC deviation. RF combine all of those and got AUC of CNN = 0.963	Model has potential to predict early-stage glaucoma without taking VF data into consideration	Dataset was small. So, larger dataset is required to train the model well
Lee et al. [11]	RF and extra-trees classifier were used to predict the progression of NTG	RF gave the AUC of 0.811	The study was done, and the data was collected for 3 years	Number of subjects were less, so cross-validation was done on training set
		Extra-trees classifier gave the AUC of 881	The model achieved better AUC in predicting primary open-angle glaucoma	The dataset was limited to Korean people only, a larger dataset may be taken in a future study
Son et al. [12]	Deep NN model on IDRiD dataset and e-ophtha dataset	For IDRiD, AUC was 95.7–98.0%	It can visualize salient regions using a lesion heatmap	Heterogeneous dataset was not generalized for clinical application
				The AUC decreased when collective patterns were extracted from the training set
		For e-ophtha, AUC was 94.7–96.5%	Large dataset was taken for the work. So achieved accuracy is more	For training set and test set, the features were not exactly the same for all data
				The proposed algorithm performed rather less well when it was tested with some external dataset due to some heterogeneous images

(continued)

**Table 1** (continued)

Paper name	Method used	Result obtained	Advantage	Disadvantage
Thakur and Juneja [13]	SVM, NN, kNN, NB and RF on fundus images	Best accuracy was achieved by SVM. Accuracy was 91.6% for 61 features and 97.2% for 10 features	Hybrid features were extracted for better glaucoma diagnosis from retinal images	Accuracy can be further improved using deep learning approaches
			Feature ranking process reduced the training time for the classifiers	Dataset used in this experiment was small. A large dataset can be used for better accuracy
Asaoka et al. [14]	RF, SVM, gradient boosting, NN on visual field data of open-angle glaucoma eye	Deep FNN gave them the best AUC of 92.6%	Able to differentiate PPGVFs from VFs with high sensitivity and specificity as DL method was used	Due to the lack of external dataset, the accuracy is not tested properly
			This method is better for identifying PPG without optical coherence tomography (OCT) and stereophotography	Glaucoma disorders other than VFs were not included for the study
Wang et al. [15]	SVM, kNN, ResNet-18, GlaucomaNet on RNFL dataset	ResNet-18 gave them the best accuracy of 0.905 and sensitivity of 0.860	GlaucomaNet gave higher AUC than the other models	Intraocular pressure was not included for glaucoma diagnosis
			The model is less prone to overfitting and takes less time for training	Number of glaucomatous eye data was limited
				Accuracy was inconsistent while comparing the models as they had some minor differences

#### 4 Conclusion and Future Scope

From the discussed papers, we find that glaucoma mostly occurs in people over 40 years due to an increase of intraocular pressure in the eye. Initially, glaucoma may not cause any symptom in the affected people. Also, glaucoma is incurable. So early detection can prevent more harm. ML classifiers help us to detect glaucoma

from a healthy eye. In the exiting works, data collection for glaucoma was mainly done in the clinic. So, the collected data was limited. For better diagnosis, more data should be captured. We can extract a large number of features if we have enough data in hand. But data collection is difficult in the remote areas. It is always not possible for the remote patients to go for clinical check-up. Here, we can take smartphones into consideration. Smartphones are easily available and portable. So, we can collect data from the remote areas. Smartphone cameras are improving day by day, and some of those are capable to capture high-resolution macro-pictures. So, smartphones can be used to enhance the scope of data collection. After collecting the data, proper security mechanism should be maintained while storing it. But security aspect was not properly discussed in the surveyed papers. Medical data is sensitive, and it should not be compromised. Consequently, during data collection and storage, we need to be more careful regarding the security aspect of the data. For data collection, we can use IoT sensors with a proper security framework. IoT sensors are connected to Internet, so we can easily transfer the data to the cloud servers. These sensors along with the security framework can ensure confidentiality and integrity of collected data. For data storage, we can take cloud storage into consideration. Cloud provides us with better data security and availability. This can enable the doctors to easily access patient's reports remotely. It can be an important step towards telemedicine and remote check-up. There are many factors for which glaucoma can occur. But no proper reason is not found till now. So, if glaucoma reasons can be optimized, it will provide a better way for proper treatment for this disease.

## References

1. Bowd, C., Goldbaum, M.H.: Machine learning classifiers in glaucoma. *Optom. Vis. Sci.* **85**(6), 396–405 (2008)
2. Shuldiner, S.R., et al.: Predicting eyes at risk for rapid glaucoma progression based on an initial visual field test using machine learning. *PLoS ONE* **16**(4), e0249856 (2021)
3. Devecioglu, O.C., Malik, J., Ince, T., Kiranyaz, S., Atalay, E., Gabbouj, M.: Real-time glaucoma detection from digital fundus images using self-ONNs. *IEEE Access* **9**, 140031–140041 (2021)
4. Mary, J.J., Charanya, R., Shanthi, V., Sridevi, G.: Prediction of glaucoma disease using deep learning techniques. *Eur. J. Mol. Clin. Med.* **7**(9), 1447–1453 (2020)
5. Kim, S.J., Cho, K.J., Oh, S.: Development of machine learning models for diagnosis of glaucoma. *PLoS ONE* **12**(5), e0177726 (2017)
6. Oh, S., Park, Y., Cho, K.J., Kim, S.J.: Explainable machine learning model for glaucoma diagnosis and its interpretation. *Diagnostics* **11**(3), 510 (2021)
7. Tekouabou, S.C.K., Alaoui, E.A.A., Chabbar, I., Cherif, W., Silkan, H.: Machine learning approach for early detection of glaucoma from visual fields. In: Proceedings of the 3rd International Conference on Networking, Information Systems & Security, pp. 1–5 (2020)
8. Civit-Masot, J., Domínguez-Morales, M.J., Vicente-Díaz, S., Civit, A.: Dual machine-learning system to aid glaucoma diagnosis using disc and cup feature extraction. *IEEE Access* **8**, 127519–127529 (2020)
9. Grassmann et al.: A deep learning algorithm for prediction of age-related eye disease study severity scale for age-related macular degeneration from color fundus photography. *Ophthalmology* **125**(9), 1410–1420 (2018)

10. An, G., et al.: Glaucoma diagnosis with machine learning based on optical coherence tomography and color fundus images. *J. Healthc. Eng.* **2019**, 1–9 (2019). Article ID 4061313. <https://doi.org/10.1155/2019/4061313>
11. Lee, J., Kim, Y.K., Jeoung, J.W., Ha, A., Kim, Y.W., Park, K.H.: Machine learning classifiers-based prediction of normal-tension glaucoma progression in young myopic patients. *Jpn. J. Ophthalmol.* **64**(1), 68–76 (2020)
12. Son, J., Shin, J.Y., Kim, H.D., Jung, K.H., Park, K.H., Park, S.J.: Development and validation of deep learning models for screening multiple abnormal findings in retinal fundus images. *Ophthalmology* **127**(1), 85–94 (2020)
13. Thakur, N., Juneja, M.: Classification of glaucoma using hybrid features with machine learning approaches. *Biomed. Signal Process. Control* **62**, 102137 (2020)
14. Asaoka, R., Murata, H., Iwase, A., Araie, M.: Detecting preperimetric glaucoma with standard automated perimetry using a deep learning classifier. *Ophthalmology* **123**(9), 1974–1980 (2016)
15. Wang, P., et al.: Machine learning models for diagnosing glaucoma from retinal nerve fiber layer thickness maps. *Ophthalmol. Glaucoma* **2**(6), 422–428 (2019)
16. Ting, D.S.W., et al.: Development and validation of a deep learning system for diabetic retinopathy and related eye diseases using retinal images from multiethnic populations with diabetes. *JAMA* **318**(22), 2211–2223 (2017)
17. Mehta, P., et al.: Automated detection of glaucoma with interpretable machine learning using clinical data and multimodal retinal images. *Am. J. Ophthalmol.* **231**, 154–169 (2021)
18. Mridha, K., et al.: Web based brain tumor detection using neural network. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 137–143 (2021). <http://doi.org/10.1109/ICCCA52192.2021.9666248>
19. Palimkar, P., et al.: Machine learning technique to prognosis diabetes disease: random forest classifier approach. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems*, vol. 218. Springer, Singapore (2022). [http://doi.org/10.1007/978-981-16-2164-2\\_19](http://doi.org/10.1007/978-981-16-2164-2_19)
20. Mridha, K., et al.: Deep learning algorithms are used to automatically detection invasive ducal carcinoma in whole slide images. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 123–129 (2021). <http://doi.org/10.1109/ICCA52192.2021.9666302>

# Implementation of IoT-Based Health Care and Saline Monitoring System Using Arduino UNO



Rinkesh Pantawane, Amlan Jyoti, Sangeeta Sheoran, Yogeshwar Uikey, Shivali Sonarkar, and Rajesh Thakare

**Abstract** In this twenty-first century, automation is the new era of modern machines. The new medical industry has gone modernized but with some problems for which we delve into the topic of what kinds of issues people confront in hospitals and how smart technologies help or are meant to help doctors and nurses tackle these issues. One of the most typical issues that doctors face is that when a patient is sick, then the doctor must first administer saline to the patient. When the saline bottle is in place, they must have to check frequently the level of the bottle. It is impossible to keep track of everything all of the time because there are fewer doctors in underdeveloping nations like India, and it can be difficult for a single doctor to supervise multiple patients at the same time. The demand for health care has increased dramatically in recent decades, necessitating the development of well-equipped, efficient monitoring systems for health care facilities, which will benefit doctors and ensure that patients are protected from any type of health problem while undergoing treatment. Using IoT, we have introduced this solution for the benefit of the medical industry.

**Keywords** Arduino UNO · Health care · Internet of things · Medical assist device · Microcontroller · Saline · Sensor · Wireless networking technology

## 1 Introduction

The IoT technology, or Internet of things, is a relatively basic concept that entails connecting all physical places and things on the planet to the Internet. It is the interconnection of physical objects with electronics built into its architecture for them to communicate and experience interactions with one another and with the outside world. In the coming few years, Internet-of-things technologies would provide advanced degrees of services, effectively transforming how people lived. Health care, automation, agricultural, smart buildings, and wireless sensor networks are just a few of the fields where IoT has made a significant impact. Patients interaction with

---

R. Pantawane (✉) · A. Jyoti · S. Sheoran · Y. Uikey · S. Sonarkar · R. Thakare  
Department of Electronics Engineering, Yeshwantrao Chavan College of Engineering, Nagpur,  
Maharashtra, India  
e-mail: [17010574@ycce.in](mailto:17010574@ycce.in)

doctors was restricted to visits and text conversations until IoT was found. Healthcare organizations had no way of frequently assessing patient conditions and making appropriate recommendations.

Now though, when the Internet of things (IoT) was first introduced, it enabled gadgets to remotely monitor the health care of any type of patient, by keeping the patient safe and healthy by allowing specialists to give exceptional healthcare advice in the health system [1, 2]. As a manner patient interest and happiness have increased as interaction with doctors has become simpler and more convenient. By reimagining the area of devices and human touch in the provision of healthcare solutions, wireless technology (IoT) is certainly transforming the health industry. IoT technology in healthcare benefits patients, families, healthcare professionals, and hospitals [3].

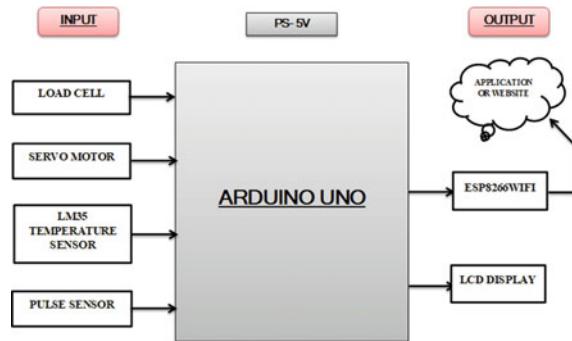
## 2 Related Works

A system was proposed at the IEEE 20th International Conference in 2018 [4]. In this paper, they suggest a smart saline level monitoring system that combines Internet of things (IoT) and sensing technologies [5, 6]. The load sensor and ESP32 Wi-Fi system-on-chip microcontroller were used to create this system. They calculated weight using a load sensor, then generated and published a message relevant to the voltage readout from the sensor using an ESP32 microcontroller. The MQTT-S distributed communication protocol, a TCP-based publish/subscribe protocol, was used. This automatic monitoring system will ensure that communications are delivered to their intended recipients on time [4].

IJET established a system in 2018. For monitoring, they used an LM35, a water level sensor, and tilt sensors, and the data is supplied and received. Internet access was enabled in the Node MCU module an open-source IoT platform from the nurse or doctor's end [7]. When the saline container's fluid level falls below a certain level, to the point that one of the LEDs glows, the remaining LEDs will glow as well, sign verify that the saline bottle is empty. If the patient is unconscious, this device is utilized to monitor his or her condition. They have used the ThingSpeak cloud platform. The data can be accessed on any web page or in the ThingSpeak application. All of the levels, or ranges, that are completed can be observed by the nurse [8].

One of the systems was introduced by IJEAST in 2020 [9]. They developed a saline surveillance device that measures saline levels continuously in the absence of healthcare personnel. They're utilizing an NRF module, an HX711 load cell, an LED, a buzzer, and a DC motor. The Arduino Nano is used to automatically track the patient's saline level. An Android OS smartphone is used to control the entire system. The main goal of the system is to keep track of the saline rate and alert the nurse when it reaches a certain level. When the saline is finished, the device uses a spring to automatically cease the flow of saline [9].

**Fig. 1** Proposed solution  
block diagram



### 3 Proposed Solution

We have introduced IoT in a beneficial method where with the help of IoT. It can communicate data to the nurse cabin via a wireless network. It can receive the data by putting a load cell on the saline bottle to see if the liquid saline is full or empty. When the procedure is complete, we must remove the needle from the patient's veins and replace it with a new saline solution.

The major goal of this system is to use an Arduino UNO to automatically track the patient saline level, and with the help of Wi-Fi send the data to the hospital computer system of the respective medical staff for verification.

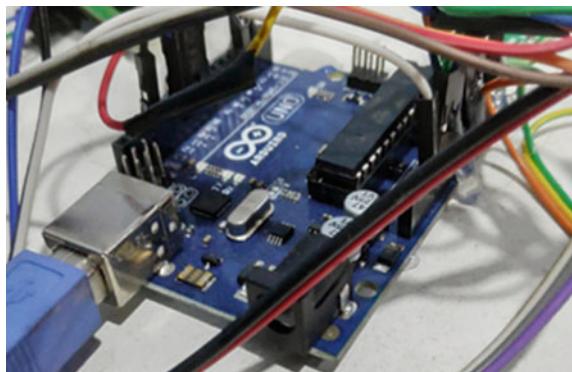
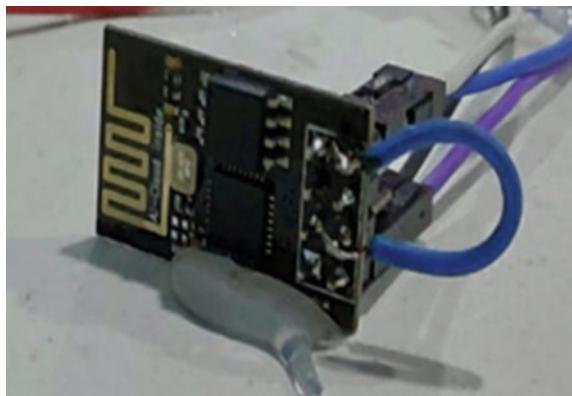
#### 3.1 Block Diagram

See Fig. 1.

### 4 Components Required

#### 4.1 Arduino UNO

The Arduino UNO is an open electronics prototyping board with a microprocessor that can be programmed using the Arduino IDE. Arduino is made up of two parts: a physical circuit board for programming and software (Fig. 2).

**Fig. 2** Arduino UNO**Fig. 3** ESP8266 Wi-Fi module

#### 4.2 *ESP8266 Wi-Fi Module*

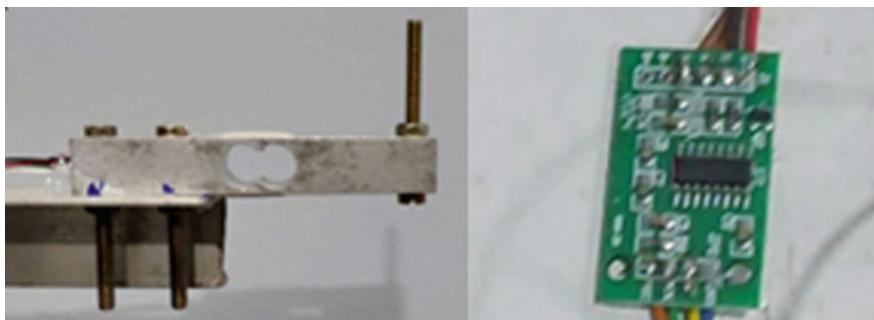
The ESP8266 Wi-Fi module is a surface-mountable Wi-Fi module with an embedded ESP8266 system on a chip that is both powerful and cost-effective (SoC). It is one of the most popular low-cost Wi-Fi modules for managing devices over the Internet (\$4.00 to \$10.00) (Fig. 3).

#### 4.3 *Load Cell with HX711 Chip*

A load cell is a device that measures the force applied to a load and turns it into a measured electrical output.

Weight changes with time, and the device can also detect strain and pressure on a surface.

The HX711 is an analog-to-digital converter (ADC).



**Fig. 4** Load cell with HX711 chip

It is a 24-bit A/D converter for load cell (HX711) (Fig. 4).

#### 4.4 Pulse Sensor

The pulse sensor uses a combination of a light-emitting diode and a photo-sensing APD. The LED is to be applied directly over a vein, thus when the heart beats there is blood flow within the vein. By measuring the blood flow, we can measure the heart rate. The pulse sensor has some circuitry for noise cancelation, reverse voltage protection, and amplification. It has three pins which act as a ground,  $V_{cc}$ , and signal. It needs a supply a min 3.3 V (Fig. 5).

**Fig. 5** Pulse sensor



**Fig. 6** LM35 temperature sensor



#### 4.5 LM35 Temperature Sensor

The output voltage of the LM35 series of precision automatic temperature devices is directly proportional to the instantaneous temperature in Celsius. It is a low-cost, high-accuracy sensor that rarely requires any external calibration (Fig. 6).

#### 4.6 LCD Display

A standard  $16 \times 2$  LCD display is used for displaying the output of the system. Proper interfacing is done that shows the amount of fluid that is remaining in the bottle. It takes commands from the microcontroller, and the output shows on the LCD screen (Fig. 7).

**Fig. 7** LCD display



**Fig. 8** MG995 servo motor

#### 4.7 *Servo Motor*

A standard DC servo motor consists of 3 pins of which 2 pins are for  $V_{cc}$  and ground, and the third is for transmitting and receiving signals. Appropriate position feedback sensors are used, now depending upon the rate of flow given to the servomotor will rotate (Fig. 8).

### 5 Software Used

#### 5.1 *Arduino IDE*

The Arduino IDE is an open-source platform that is very easy to use to write programs and dump them on various types of boards. Arduino IDE is a free and open-source electronic board. Arduino IDE has been used to build those as programming to microcontroller-based boards and other education sector development boards with third-party core compatibility. The Arduino IDE comes with a software library from the creation and implementation, which contains a variety of standard input and output routines.

## 5.2 Visual Studio

The Visual Studio is an included improvement surroundings (IDE) advanced Microsoft's graphical user interface (GUI), console, web packages, web applications, mobile applications, cloud, and Internet providers, among other things, are all being expanded. You might create both controlled and local code with the aid of this IDE.

Use tools in Visual Studio to easily upload associated solutions like Microsoft 365, Azure Project provider, and Application Insights as you design your application. C# with the Internet Frameworks, HTML and Java, or C++ may all be used to create applications.

# 6 Methods of Implementation

## 6.1 Model Implementation

Firstly, it needs to give it an external power supply for starting this system. In the breadboard, the horizontal pinpoints are ground and vertical points are  $V_{cc}$ . Arduino is connected with an external power supply. The pulse sensor is connected with analog pins of Arduino, the load cell is connected with HX711 Chip, and it is connected to digital pins, the LCD is connected with I2C, a servo motor is connected to digital pins, the Wi-Fi module is connected with digital pins, LM35 Temp. The sensor is connected to digital pins. After that, the sensor will generate data in analog format. Analog data will then be converted into digital format. Then it will send data on staff devices. A notification will pop up in the application. Then data will be shown in digital format, End. All components will work simultaneously (Fig. 9).

This is the proposed model of the system. This proposed system can calculate saline level, the temperature of the body, and pulse rate as well as it will stop the flow of the saline tube when the saline bottle is set to get finished and will send data on the companion application.

## 6.2 Flowchart

See Fig. 10.

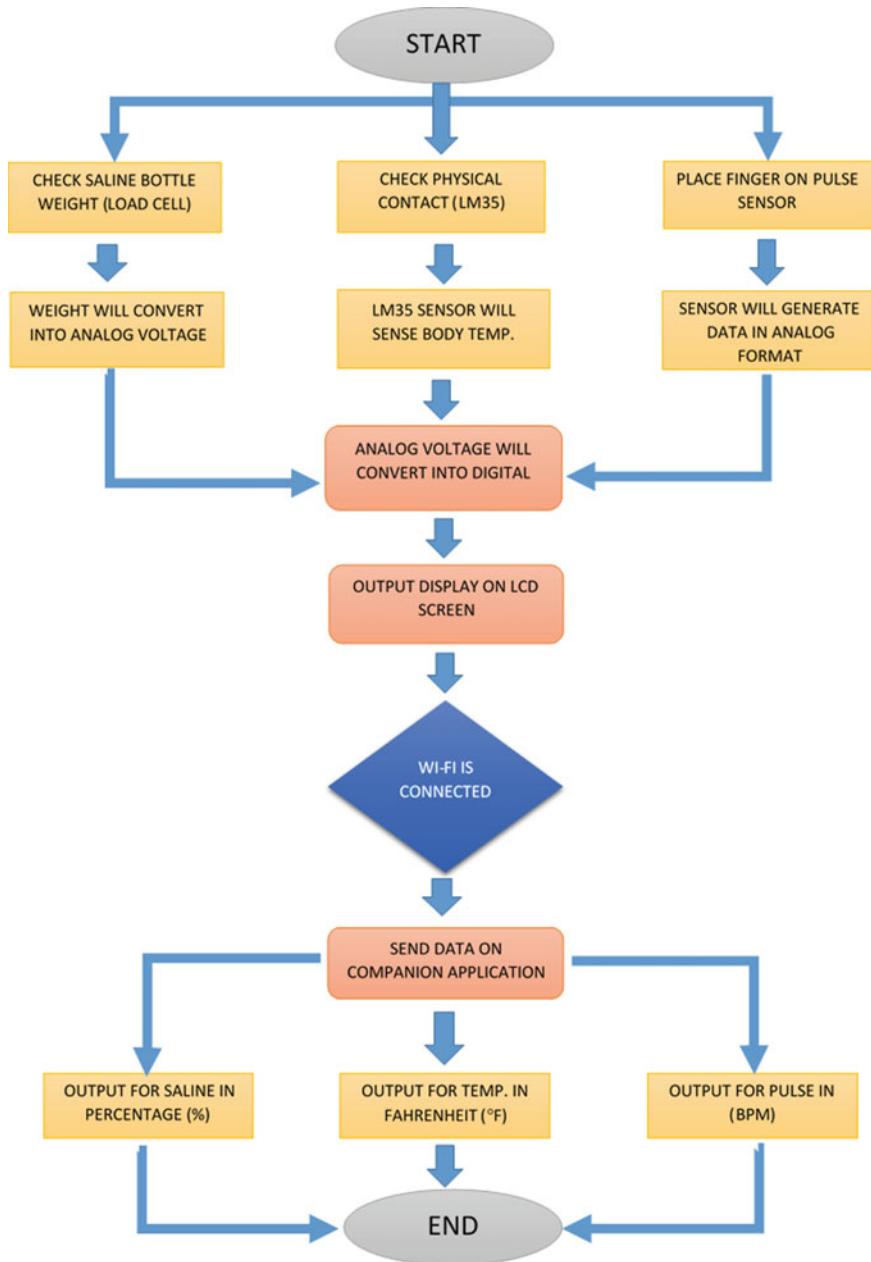
**Fig. 9** Proposed model

### 6.3 Size Table

Table 1 shows the different standard sizes of the bottle which can be used in this proposed system.

### 6.4 Application Window Implementation

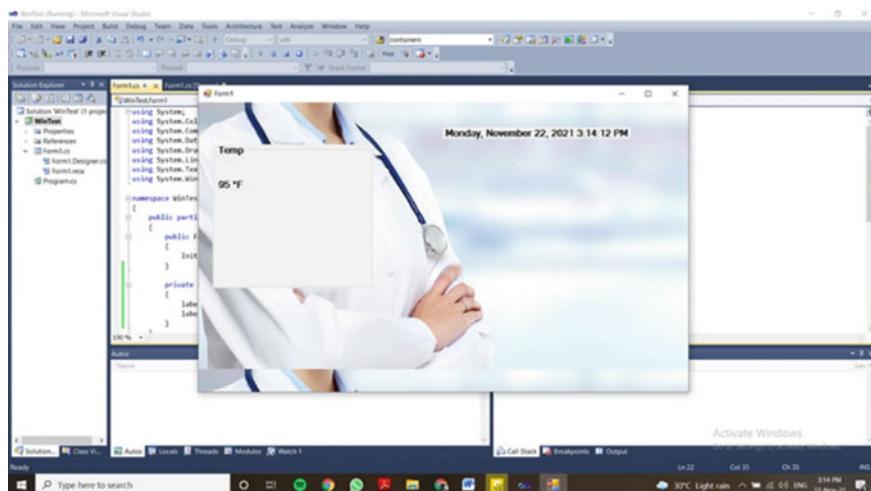
It needs Visual Studio to create an application for this system. This application will show the output of the components that are working in the system (Fig. 11).



**Fig. 10** Flowchart for the system

**Table 1** Standard size for saline bottles

S. No.	Saline bottle size (ml)	Weight (g)
1	100	100
2	250	250
3	500	500
4	1000	1000



**Fig. 11** Creating an application in visual studio

## 7 Result

1. The load sensor will monitor the amount of saline fluid in the bottle.
2. An alarm will be triggered if the amount of saline liquid is equal to or less than the essential element.
3. In conjunction with pulse rate, it will examine body temperature.
4. Notifications might be sent to the nurse or health professional as soon as the saline level falls below a certain level, indicating the need for a saline bottle replacement.
5. The need for a saline bottle is entirely determined by the many circumstances that the doctor has prescribed for the patient.
6. When the saline bottle is about to get empty, the servo motor will rotate 180°, and it will stop the flow of liquid by pinching the saline tube.
7. If a genuinely concerned patient isn't recognized by a nurse or doctor, the saline will be automatically stopped (Fig. 12).



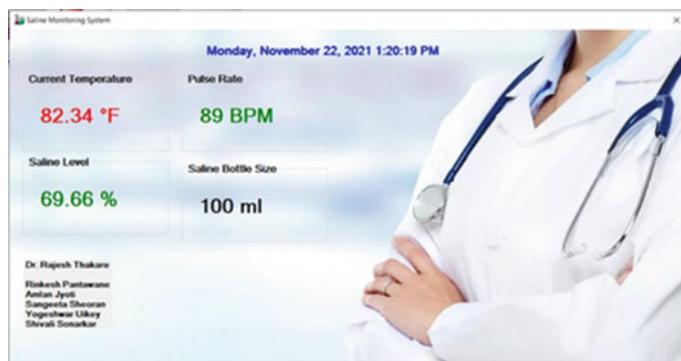
**Fig. 12** Output on the LCD screen for saline level



**Fig. 13** Output on the LCD screen for pulse and temperature

The saline level of the bottle is displayed on the LCD screen as shown in the image above. As much as the weight will change of saline bottles percentage levels will change accordingly (Fig. 13).

The above image is for the pulse rate, and body temperature of the person's body is getting displayed on LCD Screen. As a person will touch the sensor, it will change of pulse rate and temperature of the body accordingly (Fig. 14).



**Fig. 14** Application output window

This is an application window for the system which will continuously display the output of components that are used in this system. We have created four output blocks, first top left corner block is for temperature monitoring in farad. Second block is for pulse rate monitoring in BPM. Third block is for saline level monitoring, and fourth block is for the size of saline bottles according to that weight capacity will change.

## 8 Future Scope

1. It is possible to avoid the need for a monitor.
2. We can develop an android application for better monitoring.
3. We can use it all around the world thanks to the application of advanced IoT solutions.

## 9 Conclusion

Using the Internet of things, the proposed machine was created. Because the device is completely automated, no human interaction is required. In most hospitals, a nurse, medical practitioner, or caretaker checks the prevalence of saline on a manual basis. If the nurse is no longer accessible to monitor the saline, and it is still linked to the patient's body, the patient may be harmed. In this scenario, the proposed device is self-contained and saves the affected person's way of life. Because the nurse will not be necessary to monitor the patient's saline frequently, the gadget will be useful at night. Because the doctor, paramedic, or caretaker will receive an alarm notice as soon as the saline level exceeds the critical level, this system will automatically begin pinching the saline pipe to stop the saline flow at that time.

This project will be beneficial in both urban and rural areas, where the cost of medical treatment is expensive. Our method will lighten the burden of doctors and nurses and will provide more secure treatment to the patient. We are continuously working on it to make it a low-cost, high-performance automatic saline monitoring system. We want to involve more automation and include the latest computerized technology that can cut the stress for doctors, nurses, and medical staff. This equipment will assist nurses and staff in improving the speed and quality of their work by digitizing all statistics.

## References

1. Santosh, C., Kulat, K.D., Thakare, R.: A significance of VLSI techniques for low power real-time systems. *Int. J. Comput. Appl.* **1** (2010). <http://doi.org/10.5120/437-666>
2. Gupta, S., Kulkarni, M., Kulkarni, Y.: Smart saline monitoring system using load cell and RF sensor. *Int. Res. J. Eng. Technol. (IRJET)* **05**(06) (2018)
3. Amritha Ashok, K., et al.: Hospital assistance robots control strategy and machine learning technology. In: Bianchini, M., Simic, M., Ghosh, A., Shaw, R.N. (eds.) *Machine Learning for Robotics Applications. Studies in Computational Intelligence*, vol. 960. Springer, Singapore (2021). [http://doi.org/10.1007/978-981-16-0598-7\\_3](http://doi.org/10.1007/978-981-16-0598-7_3)
4. Ghosh, D., Agrawal, A., Prakash, N., Goyal, P.: Smart saline level monitoring system using ESP32 and MQTT-S. In: 2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom), pp. 1–5 (2018). <http://doi.org/10.1109/HealthCom.2018.8531172>
5. Gaikwad, V., Rathod, A., Ringe, P., Patil, V., Brigade, D.: Intelligent saline monitoring system. *Int. J. Innov. Sci. Res. Technol. (IJISRT)* **4**(5), 341–343 (2019). [www.ijisrt.com](http://www.ijisrt.com). ISSN 2456-2165
6. Pearline, S.P., Anushree, N., Aishwarya, L.: Saline infusion level detection and heart rate monitoring system. IJRASET5883, pp 637–641 (2016). <https://www.ijraset.com/fileserve.php?FID=5883>
7. Rajawat, A.S., et al.: Depression detection for elderly people using AI robotic systems leveraging the Nelder–Mead method. In: *Artificial Intelligence for Future Generation Robotics*, pp. 55–70 (2021). <http://doi.org/10.1016/B978-0-323-85498-6.00006-X>
8. Malleswari, B., Varma, P., Venkataram, N.: Smart saline level monitoring system using IoT. *Int. J. Eng. Technol.* **7**, 817 (2018). <http://doi.org/10.14419/IJET.v7i2.7.10986>
9. Kulkarni, T., Devare, A., Zende, S., Yadav, S., Biradar, A.: Design and develop a model for saline monitoring system. *Int. J. Eng. Appl. Sci. Technol.* **5**(1), 497–501 (2020). ISSN No. 2455-2143

# Blockchain-Based Academic Certificate Verification System—A Review



Shivani Pathak, Vimal Gupta, Nitima Malsa, Ankush Ghosh, and R. N. Shaw

**Abstract** Educational certificates, marriage registration records, students' records, healthcare records, and other types of information can all be recorded in a transparent and safe manner using blockchain technology. Professional, educational, and extracurricular certifications can be used to assess a student's achievements. Students can utilise these academic qualifications to exhibit their professional development and personal qualities. Such documents can be stored and maintained using blockchain technology. This clearly shows that this technology is essential to keep digital assets safe so that anybody can use them without fear of losing data and to maintain them up to date at a low cost. A review of several blockchain-based digital certificate verification systems is presented in this study. The blockchain system employs a variety of blockchain platforms, an owner authentication technique, and blockchain technology to store students' academic records as blocks. Various papers based on academic certification have been examined and compared in terms of their goals as well as potential process issues in the future.

**Keywords** Blockchain platforms · Academic certificate verification · Smart contracts · Distributed application · Remix IDE

---

S. Pathak · V. Gupta · N. Malsa (✉)  
JSS Academy of Technical Education, Noida, India  
e-mail: [nitimamalsa@jssaten.ac.in](mailto:nitimamalsa@jssaten.ac.in)

V. Gupta  
e-mail: [vimalgupta@jssaten.ac.in](mailto:vimalgupta@jssaten.ac.in)

A. Ghosh · R. N. Shaw  
University Centre for Research & Development (UCRD), Chandigarh University, Mohali, Punjab, India  
e-mail: [r.n.s@ieee.org](mailto:r.n.s@ieee.org)

## 1 Introduction

Academic qualification is a valuable asset since it demonstrates the holder's knowledge, competence, and expertise, and it is used to apply for scholarships and employment. People try to make phoney certificates because they are so valuable. Fake academic certifications have long been a concern in higher education. Forgery of certifications has become a serious problem, and the existing method makes it impossible to tell whether a document is genuine or not. Graduation diplomas and transcripts include personal information that should not be readily available to the public. As a result, there is a critical need for a technique that can ensure that the information in such a document is original, that is, that it came from a legitimate source and is not a forgery. Decentralised, distributed, secure, faster, transparent, and non-modifiable capabilities are all features of blockchain technology. These are more advantageous than current technology. It is a linked list-like data structure that publicly maintains data and transaction information over a peer-to-peer network. Various machine algorithms [1–3] and blockchain methods are available to prevent forgery in academic certificate.

Blockchain technology has risen to prominence as a novel technique to developing decentralised, dependable networks. It's a simple and effective approach to prevent sensitive data from being changed or deleted in the future. The blockchain technology has many applications in many information systems as a result of these helpful qualities. Blockchain technology aspires to create a decentralised environment in which transactions and data are not controlled by a single institution or a third party. While cryptocurrencies [4, 5] are an interesting and important application of blockchain technology, the technology has also gained traction in a variety of other areas, including tamper-proof documents, distributed ownership records, universal medical records, supply chain management, access control, and more.

A person's academic qualifications are great assets. When applying for a job, higher education and academic qualifications on a passport serve as proof of identification and eligibility. In India, where over five million people graduate each year, fake credentials are a major issue. In August 2014, the Crime Branch busted a group that was reportedly involved in fabricating bogus education credentials for the Goa board of secondary and higher secondary education, as well as the Goa university, and arrested three people. Fake credentials are relatively easy to come by in India, and they are very cheap. Lying about educational qualifications is a prevalent problem, and counterfeiting certificates is now relatively simple. Companies and institutes (in the case of higher education) must spend crores of rupees to verify applicants' academic certifications, experience certificates, and transcripts. Employers' verification processes will be simplified by storing credentials on the blockchain, which will aid in the battle against bogus certificates and misleading resumes.

The digital certificate [6, 7] will be issued in a secure environment [8] using a 256-bit encrypted private key and two-factor authentication to access the interface that will produce, sign, and insert the certificate into the blockchain, thereby sealing and tampering-proofing the certificate's information. Each graduate will also receive

a print certificate as well as a Digital Certificate Number (DCN). This DCN can be placed on resumes so that any employer can quickly check the certificate's validity. It's also a great location to save academic certifications, of course. The blockchain is the way certification will be done in the future. It is far more efficient, secure, and straightforward than anything now available in the business. It will also protect them from being copied or hacked. Verification of digital certificates generated on a blockchain may be done using an app, and authentication/verification can be immediate once the certificate holder permits access. Another benefit is that records cannot be tampered with since modifications to individual blocks require the agreement of all persons involved in the block's processing.

When it comes to humans, document verification is a regular occurrence. They are required to authenticate their documentation in all aspects of their lives, including new births, marriages, judicial processes, work purposes, and every other new step in their lives. Blockchain technology can be used to do this. With blockchain technology, document verification is no longer difficult and is also immune to erroneous representations. The papers to be validated are stored on a distributed ledger using blockchain technology. It is a cryptographic duplicate that is saved on the blockchain network, not a digital copy. Assume that a university has opted to use blockchain technology to keep its academic certificates. The conversion of document content to one-way hash code using cryptography will be the initial stage. The code is also kept on the blockchain. The one-way hash code is a string of codes that looks like a duplicate of the document. These lines of code serve as the document's key. When the user shows this document to someone else, the code stays the same as it was when it was put on the blockchain.

## 2 Literature Survey

In 2021, Naveen Kumar Dumpeti and Radhika Kavuri in their paper "A framework to manage smart educational certificates and thwart forgery on a permissioned block chain" [9]. A proposed approach shows how to leverage to construct a distributed application on the blockchain make it simple to create, problem, keep track of, and double-check certificates. The design is a digital certificate management system that is tamper-proof. When it comes to transaction speeds, authentication, and user experience, friendliness and more precise access permission levels, the suggested model is a system that works well.

In 2021, Kumutha K. and S. Jayalakshmi in their paper "The Impact of the Blockchain on Academic Certificate Verification System-Review" [10] proposed a this clearly demonstrates the need for a private blockchain Hyperledger fabric without coins to build a degree certificate authentication system in the education system, and it is simple to use for researchers and analysts on an open source ledger.

In 2021, Hrithik Gaikwad, Nevil D'Souza, Rajkumar Gupta, Amiya Kumar Tripathy in their paper "A Blockchain-Based Verification System for Academic Certificates" [11] offered a method for automating the document verification work.

Blockchain technology, Smart contracts, with their implementation, have the ability to change any industry a centralised system into a decentralised app. We discovered via our work that, in a typical way, verifying a certificate can take anything from a few depending on the situation, it might take anything from a few hours to many days type of communication between the two and their availability institutions (a business and an institution).

In 2021, Renato Q. Castro and Manuel Au-Yong-Oliveira in their paper “Blockchain and Higher Education Diplomas” [12] we propose that blockchain technology be used to ally higher education certifications and degrees, taking into account the global increase in international students, as well as the refugee problem-as well as the counterfeiting of diplomas and the sale of false certificates.

Shanmuga Priya R. and Swetha N. in their paper “Online Certificate Validation Using Block chain” [13] proposed a one of the most important aspects of blockchain technology is its data security. Blockchain is a massive, publicly accessible online ledger in which each node saves and verifies the same data. Certificate forgery is less likely with the envisaged blockchain-based system in place. In the system, the process of applying for a certificate and receiving an automated certificate is open and transparent. Finally, the system ensures the correctness and confidentiality of the information.

In 2020, Elva Leka, Besnik Selimi in his paper “Bcert-a decentralised academic certificate system distribution using blockchain technology” [14]. The suggested system, which uses blockchain to distribute academic certifications, adds value to the process of providing certificates in educational institutions by including all of the fundamental components of blockchain, such as traceability, provenance, certification, and authenticity.

In 2020, C. Roopa, Dr. R. C. Suganthe, Dr. N. Shanthi in his paper “Blockchain Based Certificate Verification Using Ethereum And Smart Contract” [15] proposed the certificate generating process is an open and transparent mechanism that allows any organisation to check information about any certificate.

In 2020, Jagtap Rajendra V., Satpute Kartiki V., Gaikwad Vishal B., Prof. Khatal S. S. in their paper “Smart Contract for Educational Digital Certificate using Block chain” [16] proposed the first dataflow clustering technique that clearly records the density of regions shared by micro clusters and uses this information for recovery is detailed in this overview.

In 2018, Rujia Li, Yifan Wu in his paper “Blockchain based academic certificate authentication system overview” [17] proposed the following is a list of the paper’s objectives: Our project is built on the Bitcoin blockchain, which is maintained by thousands of people in the cryptocurrency community. To be sure, assuming that Bitcoin will continue to perform successfully in the future is a risky assumption to make because there are a variety of factors that could cause it to fail. The blockchain ecosystem and business model are influenced by stakeholders. Table 1 shows the summary of the differences between the various applications, their implementation mechanisms, and the issues they face.

**Table 1** Summarise the differences between the various applications, their implementation mechanisms, and the issues they face

Title	Authors	Objectives	Future process challenges
A framework to manage smart educational certificates and thwart forgery on a permissioned blockchain	Dumpeti, N. K., and Kavuri, R.	Shows how to connect to peer, call chain code (proposal), respond to proposal, ledger update event, and ask for transactions to be sorted	The use of blockchain to create a distributed application and make it simple to generate, issue, maintain, and validate certificates. To maintain digital certificates, the model is a tamper-proof process
The impact of the blockchain on academic certificate verification system-review	Jayalakshmi, S.	Maintaining academic records for students with the implementation of a certificate verification mechanism based on Hyperledger	This clearly demonstrates the need for a private blockchain Hyperledger fabric without coins to build a degree certificate authentication mechanism in the education sector
Online certificate validation using blockchain	Shannuga Priya, R., and Swetha, N.	Smart contracts [18] make use of entire module payments, smart contract code, and interfaces to build up conditions based on the application	The system's certificate application and automatic certificate giving processes are both open and transparent. Companies or organisations can thus query the system for information on any certificate
Blockchain based certificate verification using ethereum and smart contract	Roopa, C., Suganthie, R. C., and Shanthi, N.	The application of document parameters such as document perspective, payment methods during transaction, and cost-effectiveness was completed	The certificate generating process is an open and transparent mechanism that allows any organisation to check information about any certificate. The suggested methodology has the benefit of storing all of the information needed to validate and authenticate the certificate on the blockchain. The potential employer does not need to contact the institution in order to authenticate the credential. It is a less time-consuming and low-cost procedure
A blockchain-based verification system for academic certificates	Gaikwad, H., DiSouza, N., Gupta, R., and Tripathy, A. K.	This paper implements a variety of framework application modules, including the oer module, blockchain module, webapp module, and email module	They discovered via their work that, in a typical way, verifying a certificate can take anything from a few hours to several days, depending on the type of communication and availability of the two institutions (company and institute). Our proposed technique proved successful in speeding up the document verification process and avoiding the need for both entities to participate in the verification
Blockchain based academic certificate authentication system overview	Li, R. and Wu, Y. (2018)	Verification applications (Extranet), issuing applications (Intranet) verification application with federated identity, issuance application with multi-signature and BTC-address-based revocation, blockchain, and MongоАDB local database	Assuming that bitcoin will continue to perform successfully in the future is a risky assumption to make because there are a variety of factors that could cause it to fail

(continued)

**Table 1** (continued)

Title	Authors	Objectives	Future process challenges
Blockchain and higher education diplomas	Castro, R. O., and Au-Yong-Oliveira, M.	Less diploma forgery and fraud, improved decentralisation and data quality, and increased data security minimises the danger of fraud (accurate, verified and validated data)	As a challenge, the process of verifying academic credentials would be more streamlined, clear, and dependable, perhaps assisting students in more areas than first anticipated and even preventing student fatigue (due to the uncertainty involved in the current process)
Smart contract for educational digital certificate using blockchain	Jagtap Rajendra, V., Satpute Kartiki, V., Gaikwad Vishal, B. and Khatal, S. S.	Data encryption, time-stamping, distributed consensus methods, and economic incentive systems are all used in these applications	Future challenges in the digital certificate verification online component of data flow mining techniques, we integrated the shared density graph with the method necessary to maintain graphs. When grouping elements in online data streams creates somewhat bigger MCs, this study indicates that shared density reclustering works well
Development and evaluation of blockchain based secure application for verification and validation of academic certificates	Leka, E., and Selimi, B.	Proposed the technology we offer will be a user-friendly application that can be used in a matter of seconds	Additional adjustments will be determined by the above-mentioned qualities. Of course, there will be more user interface upgrades, as well as updates that make the programme easier to use and fixes for any issues that surface during testing
A blockchain-based smart contract towards developing secured university examination system	Samanta, A. K., Sarkar, B. B. and Chaki, N.	The special aims of this work include application-specific smart contracts, model-based smart contracts, and tool design smart contracts	In the future, we'll strive to include more services into the smart contract to maximise the system's benefits. One of our feature development goals is to provide a universal cloud-based meta-tool. It is important to note, however, that the future growth of smart contracts will not be constrained by existing frameworks

### **3 Techniques, Methods, Analysis and Findings of Academic Certificate Verification**

#### ***3.1 Hash and Digital Certificate Techniques Used***

The certificate's hash and digital signature are created and placed on chain alongside the certificate. The proof is delivered to the user, as well as a customisable manner for the employer to view the proof. To give a finer level of access rights, IoT is connected with the Hyperledger blockchain. The proof can be submitted to an employer, who can then use blockchain to verify the document's integrity. If the proof presented does not match one already in the system marks it as a rogue certificate on the blockchain [9] (Naveen Kumar Dumpeti and Radhika Kavuri in 2021).

#### ***3.2 The Major Techniques Used in Bcert System***

- The issuer, which may be universities or training centres;
- The users, who may be students, employers, or academic institutions; and
- The accreditation body, which validates the certificate. Issuers are the only ones who can upload certificates to the blockchain; they can also add credentials, view them, and give credentials to users. Users can, on the other hand, see a list of their obtained credentials and choose whether or not to make their credentials public [14] (Elva Leka, Besnik Selimi in 2020).

#### ***3.3 Hyperledger Technique Mechanism Used in Certificate Mechanism***

The following steps were engaged in this process:

- Create a student identity and validate the certificate using a digital signature and the record's hash value. This permissioned blockchain's essential features include:
- Data privacy: data remains in the hands of the companies that own it.
- Automated, real-time verification from any location.
- Resistant to tampering and fraud.
- Permanence: the certificates will last beyond the organisations who issued them, removing the need for future verifications to rely on the issuing authority, national and international scalability [10] (Kumutha K. and S. Jayalakshmi in 2021).

### **3.4 Research Methods Used for Certificate Verification**

- System Design: Based on applicable technologies, a blockchain certificate system was established in this study. The system's application was built on the Ethereum platform and is managed by the Ethereum Virtual Machine (EVM) [19].
- Block Creation: A container data structure is referred to as a block. A block appears to be 1 MB in size on average (source). Every certificate number will be turned into a block here. For security, a hash code will be generated for each block.
- Verification: This module allows users to submit credentials such as a 10th grade mark sheet, a plus two grade sheet, college certificates, government certificates, and so on [13] (Shanmuga Priya R. and Swetha N.)

### **3.5 Algorithms Used for Certificate Verification**

- Consensus Algorithms: The blockchain consensus protocol has several specific goals, including reaching an agreement, collaboration, cooperation, equal rights for all nodes, and each node's necessary participation in the consensus process. As a result, a consensus algorithm seeks to find a common accord that benefits the entire network.
- Proof of Work (PoW): This consensus mechanism is put into practice in blockchain to choose a miner for the following block. Bitcoin accepts this PoW consensus algorithm. This algorithm's main purpose is to solve a tough mathematical enigma and deliver a solution rapidly. This mathematical task needs a lot of computing power; thus, the node that solves it is the most powerful mines the next block as rapidly as possible.
- Stakeholder Proof (PoS): Popular method for converting to PoW. Ethereum has switched from a PoW to a PoS consensus algorithm. Instead than investing in expensive gear to solve a complex puzzle, validators invest in the system's currency by locking up some of their coins as stake in this form of consensus process. If validators find a block that they believe can be added to the chain, they will approve it by placing a stake on it. All validators receive a reward proportionate to their bets based on the original blocks put to the blockchain, and their stake increases proportionately.
- Evidence of Capacity (PoC): Instead than investing in expensive equipment or burning coins in the Proof of Capacity consensus, validators are expected to donate their hard drive space. The more disc space validators have, the more probable it is that they will be picked to mine the next block and receive the block reward [15] (C. Roopa, Dr. R. C. Suganthe, Dr. N. Shanthi in 2020).

### ***3.6 Hashing Algorithm Used in Certificate Verification***

The hashing algorithm's core principle is the fact that transforms any length of data into a fixed length, which greatly simplifies the act of storing and looking for blocks while still ensuring security. The proposed system's algorithm is as follows:

- The certificate will be uploaded by the user.
- Information such as the user's it will be extracted the name, branch, and batch. The extracted data will be hashed.
- The hash will be looked for in the blockchain.
- Certificate is valid if such a hash exists [11] (Amiya Kumar Tripathy, Hrithik Gaikwad, Nevil D'Souza, Rajkumar Gupta in 2021).

### ***3.7 Functions Used for Certificate Authentication System***

- Verification applications The verification applications are in charge of ensuring the authenticity and integrity of previously issued certificates. It consists mostly of two parts: a web-based application and a client-based application. The verification applications use the blockchain API to retrieve the transaction and verification information, and the system then validates the verification information by comparing it to the receipt's checking information.
- Issuing Applications The certificates, applying, examining, turning over, and issuing are all handled by the issuing applications. It created a merkle tree using the certificates' hashes and sent the merkle root to the blockchain via APIs [17] (Ruijia Li, Yifan Wu in 2018).

### ***3.8 Main Concepts Contributions***

- Blockchain
- Blockcerts (also known as smart contracts)
- Verification and issuance of digital diplomas
- Initiatives
- Implementation Obstacles/Difficulties [12] (Renato Q. Castro and Manuel Au-Yong-Oliveira in 2021).

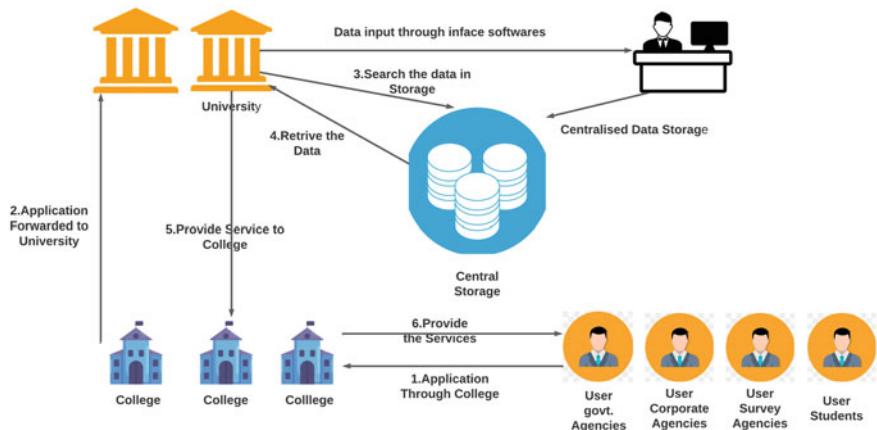
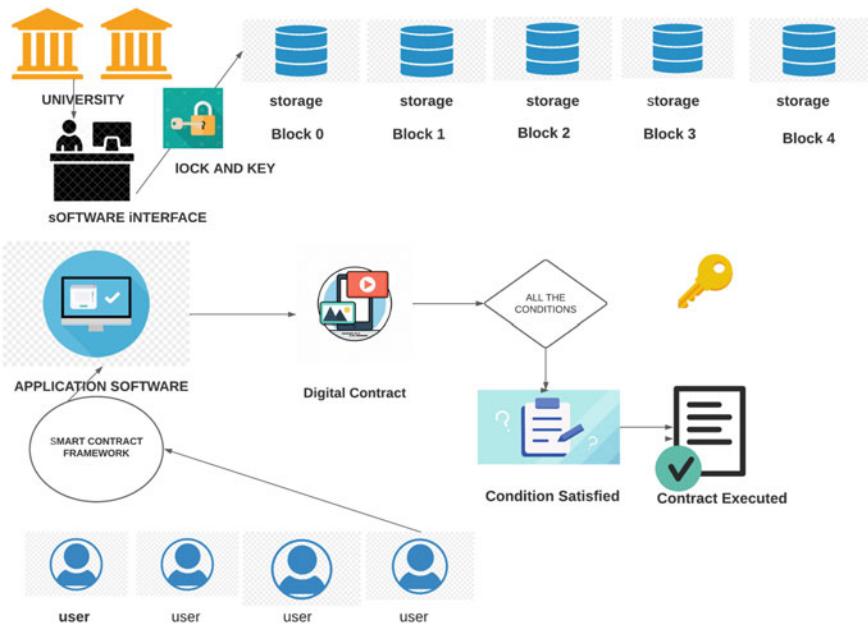
### ***3.9 Technology Used for Verification of Academic Certificate-Development Environment***

- Web3JS This allows for us to use, inspect, join, amend, use the blockchain to verify and validate contracts by allowing the client to interface with the blockchain network.
- IPFS (InterPlanetary File System) You gain access to additional increased efficiency, data, and private logs. It helps to make things go faster and safer operation of the internet. IPFS is a hypermedia protocol for peer-to-peer communication in a distributed file system to store and share data. It will be utilised in our project to store photographs, IDs, avatars, and other biometric IDs.
- PWA (Progressive Web Application) PWAs are online applications that are built and distributed. They're built with web technologies like HTML, CSS, and JavaScript in mind run across a variety of platforms and devices, including mobile phones and desktop computers. It will be utilised in this situation to create the verification procedure more accessible and available without sacrificing security, the need for extra gear or software.
- OpenShift Online This enables without having to bother about server architecture; we can deploy containers from pre-built images or other sources or incurring additional expenditures.
- Docker From the desktop to the cloud, Docker is the de facto standard for developing and sharing containerised applications. Docker enables us to create images that operate across numerous containers and can be easily scaled and redeployed.

### ***3.10 Methodology for an Application Case for Smart Contracts in the University Examination System***

The proposed smart contract for university examination methodology (Fig. 1). The system we suggest is broken into two sections. We suggested and constructed a smart contract in the first section to address our current difficulties. This section covers the contract and data flow between the university and various stakeholders in order to deliver a safe, trustworthy, and high-quality service in the shortest possible time (Fig. 2). The cost analysis and benefits of the proposed system are included in the second section, which aids in the system's implementation.

Algorithm for smart contracts proposed: The smart contract will be used to generate mark sheets, certificates, provisional certificates, and requests for record verification [20] (Ashis Kumar Samanta and Bidyut Biman Sarkar and Nabendu Chaki in 2021).

**Fig. 1** Data management system for university examinations**Fig. 2** University assessment of data management smart contract implementations

## 4 Conclusions

This work studied the principles and applications of 32 different studies by various authors, gathered material from well-known sources, and reviewed 32 different publications by various authors. Blockchain has a lot of potential in the education system. Blockchain is one of the most widely used and rapidly expanding technologies that is under utilised in a variety of industries, including health care, insurance, banking, e-voting, and supply chain management are just a few examples as well as certificate verification and digital identity, among other things. Blockchain is a technology that is thought to be safe in peer-to-peer and publicly dispersed networks due to the distributed nature of blockchain. It is possible to store large volumes of data to a more complicated network architecture. The majority of the papers show how blockchain technology is being used in Ethereum blockchain-based education system that verifies a degree's authenticity certificate.

## References

1. Gautam, J., Atrey, M., Malsa, N., Balyan, A., Shaw, R.N., Ghosh, A.: Twitter data sentiment analysis using Naive Bayes classifier and generation of heat map for analyzing intensity geographically. In: Bansal, J.C., Fung, L.C.C., Simic, M., Ghosh, A. (eds.) *Advances in Applications of Data-Driven Computing. Advances in Intelligent Systems and Computing*, vol. 1319. Springer, Singapore (2021)
2. Malsa, N., Singh, P., Gautam, J., Srivastava, A., Singh, S.P.: Source of treatment selection for different states of India and performance analysis using machine learning algorithms for classification. In: Pant, M., Kumar Sharma, T., Arya, R., Sahana, B., Zolfaghariinia, H. (eds.) *Soft Computing: Theories and Applications. Advances in Intelligent Systems and Computing*, vol. 1154. Springer, Singapore (2020)
3. Bedi, P., Goyal, S.B., Rajawat, A.S., Shaw, R.N., Ghosh, A.: A framework for personalizing atypical web search sessions with concept-based user profiles using selective machine learning techniques. In: Bianchini, M., Piuri, V., Das, S., Shaw, R.N. (eds.) *Advanced Computing and Intelligent Technologies. Lecture Notes in Networks and Systems*, vol. 218. Springer, Singapore (2022)
4. Malsa, N., Vyas, V., Gautam, J.: Blockchain platforms and interpreting the effects of bitcoin pricing on cryptocurrencies. In: Sharma, T.K., Ahn, C.W., Verma, O.P., Panigrahi, B.K. (eds.) *Soft Computing: Theories and Applications. Advances in Intelligent Systems and Computing*, vol. 1380. Springer, Singapore (2022). [https://doi.org/10.1007/978-981-16-1740-9\\_13](https://doi.org/10.1007/978-981-16-1740-9_13)
5. Malsa, N., Vyas, V., Gautam, J.: RMSE calculation of LSTM models for predicting prices of different cryptocurrencies. *Int. J. Syst. Assur. Eng. Manag.* 1–9 (2021)
6. Malsa, N., Vyas, V., Gautam, J., Shaw, R.N., Ghosh, A.: Framework and smart contract for blockchain-enabled certificate verification system using robotics. In: Bianchini, M., Simic, M., Ghosh, A., Shaw, R.N. (eds.) *Learning for Robotics Applications. Studies in Computational Intelligence*, vol. 960. Springer, Singapore (2021). [https://doi.org/10.1007/978-981-16-0598-7\\_10](https://doi.org/10.1007/978-981-16-0598-7_10)
7. Malsa, N., Vyas, V., Gautam, J., Ghosh, A., Shaw, R.N.: CERTbchain: a step by step approach towards building a blockchain based distributed application for certificate verification system. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), pp. 800–806. IEEE (2021)

8. Rajawat, A.S., Rawat, R., Barhanpurkar, K., Shaw, R.N., Ghosh, A.: Blockchain-based model for expanding IoT device data security. In: Bansal, J.C., Fung, L.C.C., Simic, M., Ghosh, A. (eds.) *Advances in Applications of Data-Driven Computing. Advances in Intelligent Systems and Computing*, vol. 1319. Springer, Singapore (2021)
9. Casino, F., Dasaklis, T.K., Patsakis, C.: A systematic literature review of blockchain-based applications: current status, classification, and open issues. *Telematics Inform.* **36**, 55–81 (2019)
10. Smolenski, N.: Top 10 reasons to use blockcerts [Online] (2018). Available: <https://medium.com/learning-machine-blog/top-10-reasons-to-use-blockcerts-ec7d29f2712>
11. Kanan, T., Obaidat, A.T., Al-Lahham, M.: SmartCert blockchain imperative for educational certificates. In: IEEE Jordan International Joint Conference 2019 on Electrical Engineering and Information Technology (JEEIT), pp. 629–633
12. Garba, A., Chen, Z., Guan, Z., Srivastava, G.: LightLedger: a novel blockchain-based domain certificate authentication and validation scheme. *IEEE Trans. Netw. Sci. Eng.* **8**(2), 1698–1710 (2021)
13. Jirgensons, M., Kapenieks, J.: Blockchain and the future of digital learning credential assessment and management. *J. Teach. Educ. Sustain.* **20**(1), 145–156 (2018)
14. Li, Y.: Emerging blockchain-based applications and techniques. *SOCA* **13**, 279–285 (2019). Available <https://doi.org/10.1007/s11761-019-00281-x>
15. Szalachowski, P.: SmartCert: redesigning digital certificates with smart contracts (2020). arXiv preprint [arXiv:2003.13259](https://arxiv.org/abs/2003.13259)
16. Al Harthy, K., Al Shuhaimi, F., Al Ismaily, K.K.J.: The upcoming blockchain adoption in higher-education: requirements and process. In: 2019 4th MEC International Conference on Big Data and Smart City (ICBDSC), pp. 1–5. IEEE (2019)
17. Lemieux, V.L., Hofman, D., Batista, D., Joo, A.: *Blockchain Technology and Recordkeeping*. ARMA International Educational Foundation (2019)
18. Malsa, N., Vyas, V., Singh, P.: Blockchain-enabled smart contract optimization for healthcare monitoring systems. In: *Cloud Computing Enabled Big-Data Analytics in Wireless Ad-hoc Networks*, pp. 229–250. CRC Press, Boca Raton
19. Masla, N., Vyas, V., Gautam, J., Shaw, R.N., Ghosh, A.: Reduction in gas cost for blockchain enabled smart contract. In: 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON), pp. 1–6. IEEE (2021)
20. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008) (whitepaper). Available: <https://bitcoin.org/bitcoin.pdf>

# Performance Analysis of VANETs Using Trusted Location and Trustworthiness of Nodes



Indrani Das , Sanjoy Das , Rishi Pal Singh , Ashwini Kumar , and Subrata Sahana

**Abstract** Vehicular ad hoc networks support numerous applications to ease the driving and safety of passengers, drivers, etc. To support the activities in VANETs, vehicular nodes can directly communicate through the wireless medium in vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) mode. The network's reliability improves if the trust level of messages disseminated is high. So, messages transmitted over the network should be authentic and ensure that these messages are routed and generated through trusted nodes. If the decision is taken based on the messages received from untrusted nodes, it will create many hazardous situations in the network. Our objective is to find correct information from the most trusted node, and the receiver node relies upon it. In this paper, our proposed method is based on trust value and the trustworthiness of vehicular nodes using the k-nearest neighbors (KNN) algorithm. The higher values of the trustworthiness of a node indicate higher chances for accurate information about the location. This is one of the primary objectives to compute trustworthy of a node. The performance analysis is done with parameter PDR with different network densities and source nodes. The result obtained through the proposed method is significant and trustworthy compared to other methodology.

**Keywords** Vehicle · Trust · Trustworthiness · Location · RSU

---

I. Das  
Assam University, Silchar, India

S. Das ()  
Indira Gandhi National Tribal University, Regional Campus Manipur, India  
e-mail: [sdas.jnu@gmail.com](mailto:sdas.jnu@gmail.com)

R. P. Singh  
Guru Jambheshwar University of Science and Technology, Hisar, India

A. Kumar  
Graphic Era Deemed to be University, Dehradun, India

S. Sahana  
Sharda University, Greater Noida, India

## 1 Introduction

The vehicular ad hoc network (VANETs) is an advancement of wireless networks [1]. This network is formed spontaneously among moving vehicles with or without the help of fixed infrastructure. The vehicular nodes can directly communicate through the wireless medium via V2V and V2I communication modes. The wireless medium may be homogeneous or heterogeneous technologies. Nowadays, traffic-related safety measures can be easily improved by improving transport efficiency. Many applications in VANET continuously broadcast information in the network about the availability of hospitals, tolls, accidents, road conditions, and many others. This information required authenticity and trustworthiness all the time. Otherwise, untrusted information cannot achieve VANET objectives for safe, efficient, and sustainable future intelligent transportation systems (ITS) [2]. The unique characteristics of VANET, like unlimited power supply, predicted mobility, highly computational capabilities [1], the faster movement of vehicular nodes, etc., differentiate this network from other wireless networks.

All the applications, vehicles are exchanging messages among themselves and thus help overcome the various issues and problems related to road accidents, traffic jams, etc. The trust management of communicated messages is very crucial because to maintain trustworthy communication. The high level of trustworthiness of the node is essential, so that transmitted messages should be trustful to a great extent to attain high reliability and secured travel [3]. Reputation and trust are also some crucial factors contributing to many decision-making activities. The opinion of one vehicle about others is based on trust value known as reputation [4].

The trustworthiness of data [5] and its authenticity is essential for the receiver node to make a reliable and efficient decision. The decision plays a crucial role in determining safety and taking critical traffic management in this network.

In this paper, the objective of the work is to improve the trustworthiness of the overall network. The messages which are transmitted over the network should be trustful and authentic. The proposed method is based on trust value and the trustworthiness of vehicular nodes using the k-nearest neighbors (KNN) algorithm. The trustworthiness of every node is evaluated in the network based on their responses to the source node request packet. Also, the packet delivery ratio has been computed to analyze performance. The trust value is the confidence in the message's authentication and replies upon communication among vehicles.

The paper is organized as in Sect. 1, we have discussed introduction, and in Sect. 2, work related to trustworthy is discussed. Section 3 discusses the algorithms of the proposed methodology of computation of the trustworthiness of the node. Section 4 includes various simulation scenario parameters and the result analysis. The paper is concluded in Sect. 5 with future direction.

## 2 Related Work

In VANETs, vehicle movement follows city road structure, urban area road structure, etc. The movements in urban, semi-urban, rural, city, and highway vehicles follow different road structures, and their velocity also varies accordingly. The traffic pattern also varies throughout the day and observed sparse, semi-dense, and dense traffic. During office and day-time traffic on the road on peak and late at night, early morning hours less traffic. In a densely populated traffic network, communication among vehicles is accessible, but scenario multi-hop communication pattern is followed in a sparsely populated network. Overall, communication depends on the communication link quality [6–10]. Mostly, applications supported by ITS need the exchange of reliable messages.

Further, vehicles frequently join and leave the network in this network, so identifying and authenticating a new vehicle is essential to maintain the trust value of messages disseminated by them. Also, validation of a new vehicle is important to ensure the security of the network. In [8, 11–13], trust evaluation-based security authentication methods are discussed. Two trust values are calculated: a direct and indirect evaluation based on securely authenticating the nodes. The direct trust method works based on recorded historical security data. When a node and the authority unit (AU) interact, security-related information is recorded to their local database. The AUs are easily accessible through the web. The data recorded is useful to evaluate and determine the credibility of a new vehicle. When nodes communicate in a group, it is their right to accept or not accept a new node. The acceptability of a new node joins in the network determined by their trust value and recommendation of other individual nodes. The selfish or malicious node does not accept the newly joined node. This condition is undesirable for the VANET scenario. Trust evaluation during the indirect method to identify the malicious nodes is very important and isolates them before calculating the indirect trust value. The correlation coefficient value is used during the recommendation trust value is useful in isolating the malicious nodes. Therefore, recommendation obtained from malicious nodes is discarded. In [14, 15], details on trust value computation based on the historical security events are discussed.

To maintain the trustworthy in the network, lots of scopes are still left for the computation of trust management in VANET, WSNs, etc. Only a few works addressed the issue of trust establishment in VANETs [16–18]. In [16], the authors proposed a trust computation method for the VANET environment. The trust management based on AHP presented in [17] shows significant improvements in the current security mechanisms. In [18], the authors discussed a trust and reputation infrastructure-aided approach called TRIP for VANETs. The trust value is determined by the reputation score awarded by other vehicles. In [19], an intrusion detection system, where watchdog mechanism is used to evaluate, nodes misbehave in the network. In [20], two security mechanisms are discussed that significantly improved the throughput of the network. In [6], an intrusion detection scheme has proven reliable in dealing with malicious nodes. In [21], a security system has been proposed against internal attacks

to ensure trustworthy vehicle-generated messages. This system uses a multi-segment technique over a gap Diffie-Hellman algorithm to ensure trustworthy messages are disseminated in the network and significantly reduce communication overhead. This scheme is easy to deploy and suitable for deterministic and non-deterministic network scenarios.

In [5], a decentralized novel trust management scheme has been proposed for VANET, and because of this feature, it is easy to implement. This scheme is robust with respect to various security threats, which are discussed and modeled numerically. This is useful in a real-time scenario because of the linear time complexity.

In [22], a new trustworthy scheme is proposed to handle message modification and fake message attacks. This scheme uses trust-level information of adjacent vehicles in two steps and helping in suppressing attack information. The trust value is computed at the first level. Here, the trust level means truth-telling probability or truth of the message of neighbor vehicles. A clustering technique is used to cope with the malicious node effect while computing the truth value of other nodes. This clustering mechanism suddenly removes the opinions of malicious nodes as outliers. In the second level, the trustworthiness of a message is determined by using a modified threshold random walk (TRW) [22]. In this scheme, the node disseminates trustworthy event information efficiently to neighboring vehicles so that node uses various network applications reliably. In [23], the authors discuss the concept of trust value of vehicles to assess the trustworthiness of data and how trustworthiness helps in improving the quality of the network performance. A self-organized “distributed trust computing framework (DTCF)” is used to compute the value of the trustworthiness of each vehicle.

The [24] trust model is proposed, where the trust value of nodes is computed to detect rogue nodes. Only the trustworthy nodes have participated in the routing process.

In [25], a novel trust model is proposed to deal with various attacks like black attack, slander attack, gray attack. The trust model is designed with the help of fuzzy theory, which computes the trust value of vehicles directly and indirectly. In [26], authors proposed a mechanism based on trust and reputation of nodes, with the help of these nodes transmitting bogus data that is easily detected. The model is tested with physical as well as the simulated environment.

Author [27] discusses vehicular traffic movements in a particular institution JNU, New Delhi. The optimal location for the deployment of RSU has been analyzed. A thorough analysis of traffic movements is observed that intersection points having dense traffic and deployment of RSU are needed in these locations.

The literature section discusses various existing works of VANETs dealing with multiple security issues. The malicious nodes, trust values of vehicular nodes' effect in networks are discussed.

### 3 Methodology

We have proposed a novel method to find the most trusted information about any things (hospital, restaurant, etc.) location based on vehicle reply [28]. In this method, the trust value of a node is computed based on the responses received from other nodes. The trust percentage of the message is calculated based on Eq. (1). The vehicle replied with a positive response for information asked by other nodes is awarded one (+1) point. The trustworthiness value of every node that exists in the network is calculated. If the trust value falls below 50%, access roadside unit (RSU) or TA. Based on the trustworthiness value, node accepts the response from the most trustworthy node (Fig. 1).

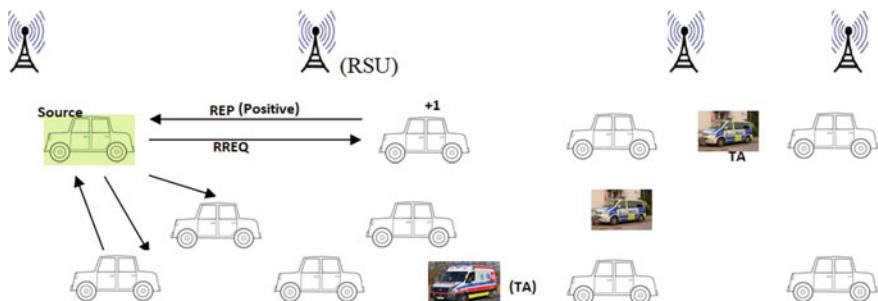
#### 3.1 Trust Percentage of Vehicles

The vehicle may want to find the location information for any school, college, shopping complex, toll plaza, etc., is available or not. The source vehicle broadcasts the request (REQ) packet to its neighbor vehicles and received (REP) reply messages. If the REP message is true, respective vehicles are rewarded with (+1) point. This reward point is a true response that keeps updating. The source vehicle computing the trust value on behalf of positive REP [28] message is received. The trust percentage is calculated through Eq. (1).

$$\text{Trust Percentage (\% age)} = \left( \frac{\text{Total number of positive responses}}{\text{Total Number of Reply}} \right) \times 100 \quad (1)$$

#### CASE 1: Trust percentage using vehicles [28]

The source vehicle receives REP messages from neighbor nodes and computes the ratio of trusted information about a location. Suppose, source node receives REP



**Fig. 1** Network scenario (RSU + TA)

messages from  $X_n$  number of vehicles,  $X_i$  number of vehicles reply is true, and  $X_j$  number of vehicles is negative or false. For each true response, a vehicle is rewarded with one point and helps in increasing its trustworthiness. The trust ratio is computed as  $[(x_i/X_n) * 100]$ . If the trust value is greater than 50%, then the source vehicle trusted the reply containing location information. Otherwise, the vehicle can compute location with the help of case 2.

#### **CASE 2: Trusted location with trustworthiness [28]**

The total number of true responses given by a vehicle means the number of times the true response given by the vehicle. The trustworthiness value is significant in the network to enhance network efficiency. This value is computed for each node in the network. Equation 2 does the computation of trustworthiness.

$$\text{Trustworthiness}(T) = \left( \frac{T_{\text{tr}}}{T_s} \right) \times 100 \quad (2)$$

$T_{\text{tr}}$  = Total number of true responses delivered by a node, and  $T_s$  = Total number of responses delivered by a node.

When the trust value falls below 50%, source node uses a trustworthy node to get location information. If there is no other node available other than the source node, then a trusted location is computed with the help of a roadside unit, according to case 3.

#### **CASE 3 (Trusted location by using RSU [28])**

In case the source node is alone, then it will ask RSU. It asked for location information from RSU and was fully trusted. If RSU is not available or not within its transmission range, then follow case 4.

#### **CASE 4 [Trusted location using trust authority (TA)] [28]**

If the roadside unit is not available, then location information can be received from TA like police vans, ambulance, etc. The notations and symbols used in the algorithm are listed in Table 1.

#### **Algorithm 1: Trusted\_Information\_Detection [28]**

##### **CASE 1:**

1. Source vehicle ( $S_i$ ) broadcast the REQ packet
2. if ( $d_{ij} \leq R$ ); All vehicles Receive REQ packet
3. Vehicles ( $N_j$ ) Reply with REP packet //  $N_j$  is neighbour of  $S_i$ , where  $j = 1, 2, 3, \dots, n$
4. For each vehicle
5. {
6.     for each (REP)
7.     {

**Table 1** Notations and symbols used in the proposed method

Notation and symbols	Meaning
$S_i$	Source
$d_{ij}$	Euclidean distance between source node $i$ and any node $j$
$R$	Transmission range
$N_i$	Number of neighbor node of $S_i$
$T_{tr}$	Total true responses received
$T_r$	Total responses received
$T$	Value of trustworthiness of node
$A, B, C$	Counter variable

```

8.         if (Location == Available)
9.         {
10.        Ttr += 1;      //Ttr is true response of each vehicle
11.        Tr += 1;      // Tr is total response of each vehicle
12.        A++          // counter to store only node saying location is present.
13.        Trustworthiness_of_each_node (T) = (Ttr/Tr)*100;
14.        }
15.        else
16.        {
17.          B++//B is counter denying the location is present.
18.          Tr += 1;
19.        }
20.      }
21.      Trust Percentage of Information A = [(A/C)*100] // C is counter store total
22.      number of reply
23.      Compute B = [(B/C)*100]
24.      }
25.      If (A > B)
26.      {
27.        Location is present;
28.      }
29.      else if (A <= B)
30.      Goto case-2

```

### CASE 2:

If ( $A \leq B$ )

{

1. Ask for the value of  $T$
2. Request to the most trustworthy node REQ.
3. Reply from the most trustworthy node will be answered.
4. If no vehicle is present goto case-3.

}

### CASE 3:

Ask to RSU for location

If (RSU != present)

else

Goto the Case-4.

### CASE 4:

Ask TA for location

Source node received trusted location.

Algorithm 1 discussed about trusted information detection works in the presence of ordinary vehicles, RSU, and TA. This algorithm is further enhanced with the help of K-nearest neighbor (KNN) technique discussed in Algorithm 2. The source node considered for Algorithm 1 remains the same. With the help of KNN, trustworthiness and trust percentage of vehicles are observed higher than Algorithm 1.

### Algorithm 2: Trusted\_Information\_Detection using KNN

Input Source Vehicle ( $S_i$ ) ( $i = 5, 10, 25$ , and  $35$ ) is fixed for different scenario.

The number of Nodes for each  $S_i$  is  $N_j$ .

```

 $X_{ij}$  = NearestNeighbors
    If ( $X_{ij} \leq 150$ )
        // Transmission Range set to 150m
    {
        Ttr +=1 ;      // Ttr is true response of each vehicle
        Y++ ;         // counter to store only node saying location is present.
    }
    else
    {
        Z+=1 ;      // Z is counter denying the location is present
    }
Total_response(Tr)= len(Xij)
Trust_worthiness_of_each_node( $T_{wn}$ )=  $(T_{tr} / Tr) * 100$ 
Trust percentage of information Y= [ (Y/C)*100] // C is counter store
total number of
//reply
Compute Z = [ (Z/C)*100]
If (Y>Z)
{
    Location is present;
}
elif
{
    Y<=Z;
}
Goto CASE-2 and So on as mentioned in Algo-1.

```

## 4 Computation of Trustworthiness of Node [28]

Each node maintains a buffer space to store the value of trustworthiness [7, 28]. The source node retrieved the value of trustworthy from each neighbor node and chose the best node. The reply received is reliable and trustworthy. The working of the algorithm follows the following steps.

### Step-1

Source vehicle sends a broadcast message, and neighbor nodes positive reply awarded one point and stored in the buffer. Two counter values are used to record the number of true responses received and store the number of responses received. The value of trustworthiness is computed as Eq. (2).

### Step-2

For every response, counter value of  $T_{tr}$  and  $T_t$  updating continuously and also the value of T.

### Step-3

In case, trust percentage <50%, then the vehicle will ask the trustworthiness of the nodes within the communication range and share the information stored in the buffer of each node.

### Algorithm 3: Trustworthiness of node

```

For each NODE // Number of node is N
For each REPLY // number of reply depends on number of neighbour
node present
    //at a particular moments
    If (REPLY==true)
    {
        True_Response = True_Response + 1;
        Total_Response = Total_Response + 1;
        Trustworthiness = (True_Response / Total_Response)*100;
    }
    else
    {
        Total_Response = Total_Response + 1;
        Trustworthiness = (True_Response / Total_Response)*100;
    }
}

```

## 5 Results and Discussion

The simulation work is carried through the NS-2 simulator. The parameters used for simulation are given in Table 2. We assume that vehicles are in a traffic point situation. The source node may be any node, but we have arbitrarily chosen nodes 5, 15, 25,

**Table 2** Simulation parameters

Parameter	Value
Number of nodes	40, 50, 60, 70, 80, 90, 100
Simulated area	1000 × 1000 m
Antenna	Omni-directional
Transmission range	150 m
Simulation time	1000 s

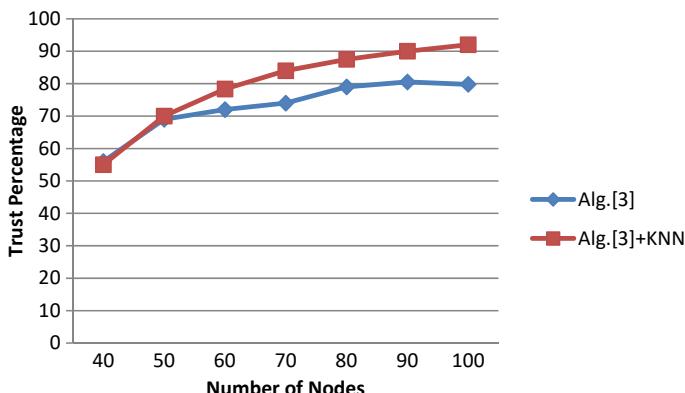
and 35 as source nodes for our simulation. The number of nodes considered is 40, 50, and up to 100. These source nodes asked all their neighbor node that any location is present or not. With different node densities, the trust value of various source nodes is computed and given in Table 4. In Table 5, the value of the trustworthiness of each node is listed. We have analyzed the proposed method based on different network scenarios and source nodes.

#### (A) CASE-A (*Source Node is 5*)

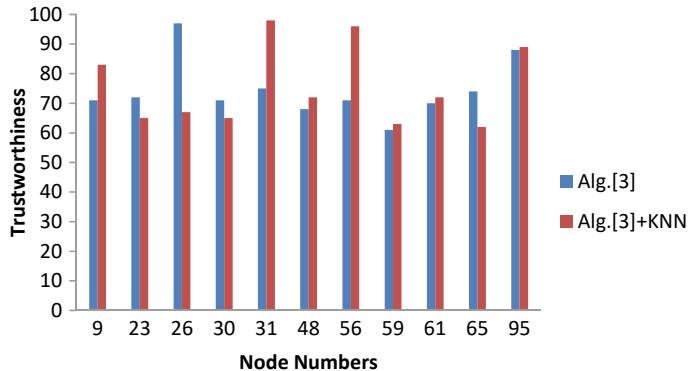
In CASE-A, the trust percentage and trustworthiness of nodes to source node 5 are computed. Figure 2 gives the value of trust percentage w.r.t. source node 5. The trust percentage obtained is always greater than 50% in different node densities. The maximum trust percentage achieved is 92% with KNN when node density is 100. Initially, in both the methods, trust percentage is increasing gradually. When node density reached to 70 nodes, KNN is giving a better trust percentage.

Figure 3 shows the trustworthiness of various nodes having a value 60% or more in both the algorithm. Node number 26 achieved the highest 96% trustworthiness value in algorithm [28], and node number 31 achieved 98% trustworthiness with KNN. In algorithm [28], total of 47 nodes trustworthiness value is 50% or above, whereas 49 nodes are in KNN.

#### (B) CASE-B (*Source Node is 15*)



**Fig. 2** Trust percentage of source node 5

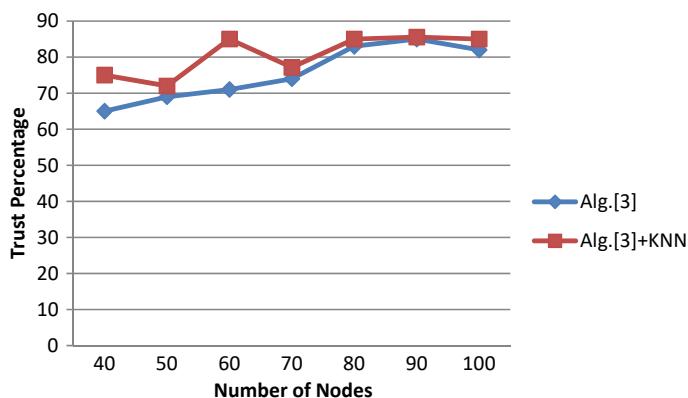


**Fig. 3** Trustworthiness of each node (source node is 5)

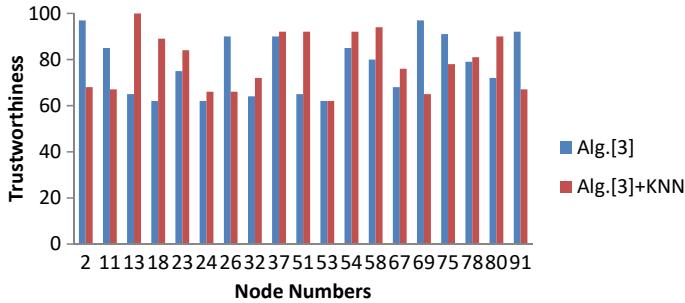
In CASE-B, the trust percentage and trustworthiness of nodes for source node 15 are computed. Figure 4 shows the value of trust percentage under different node densities. The trust percentage obtained is always greater than 50% for different node densities. The maximum trust percentage achieved is 85.56% with KNN and 85% with algorithm [28].

Figure 5 shows the trustworthiness of various nodes having a value 60% or more in both the algorithm. Node number 69 achieved the highest 97% trustworthiness value in algorithm [28], and node number 13 gained 100% trustworthiness with KNN. The number of nodes having a 60% trustworthiness value is increased as compared to source node 5. In algorithm [28], total of 54 nodes trustworthiness value is 50%, whereas 55 nodes are in KNN.

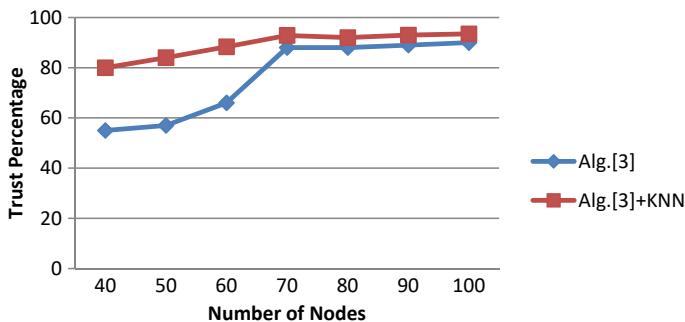
### (III) CASE-C Source Node is 25



**Fig. 4** Trust percentage of node 15 with varying node density



**Fig. 5** Trustworthiness of each node (source node is 15)



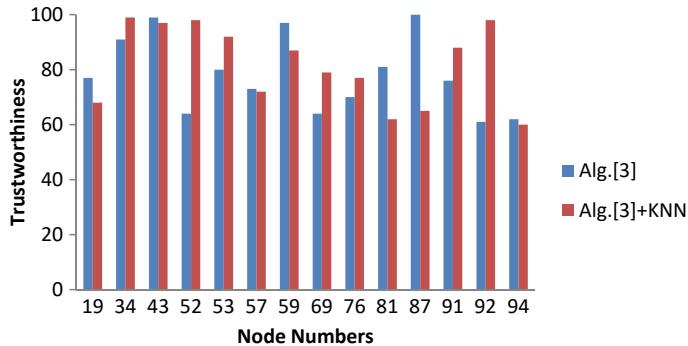
**Fig. 6** Trust percentage of node 25

In CASE-C, trust percentage and trustworthiness of nodes for source node 25 are computed. Figure 6 shows the trust percentage w.r.t. node 25 under different node densities. The trust percentage obtained is always greater than 50% in different node densities. The maximum trust percentage achieved is 93.5% with KNN, when node density is 100. In both the algorithms, trust percentage gradually increases.

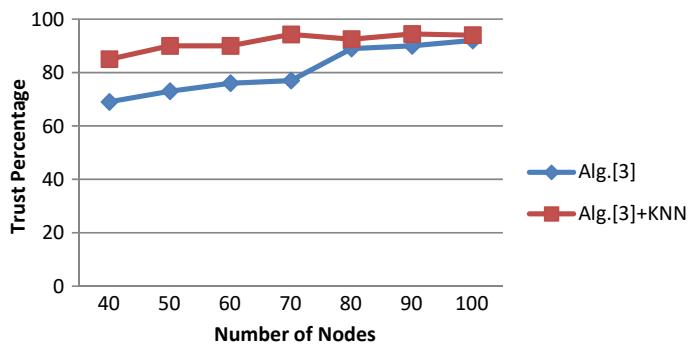
Figure 7 shows the trustworthiness of various nodes having a value of 60% or more in both the algorithm. Node number 87 achieved the highest 100% trustworthiness value in algorithm [28], and node number 34 gained 99% trustworthiness with KNN. The number of nodes having a 60% trustworthiness value is increased compared to source node 5 but decreases compared to source node 15. In algorithm [28], total of 46 nodes trustworthiness value is 50% or above, whereas 48 nodes are in KNN.

#### (IV) CASE-D: Source Node is 35

In CASE-D, trust percentage and trustworthiness of nodes for source node number 35 are computed. Figure 8 shows the trust percentage of source node 35 under different node densities. The trust percentage obtained is always greater than 50% in different node densities. The maximum trust percentage achieved is 93.5% with KNN when node density is 100 and 90% with algorithm [28]. The trust percentage gradually increases in both the algorithms.

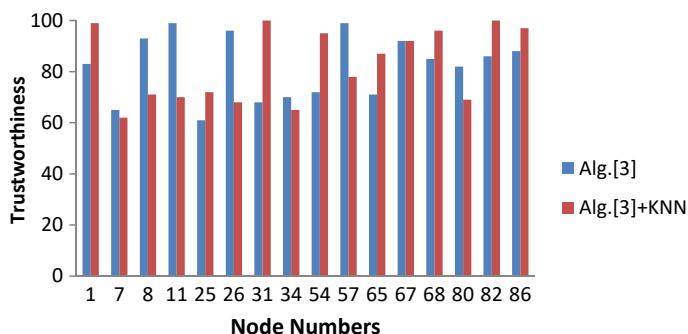


**Fig. 7** Trust percentage of node w.r.t. node 25

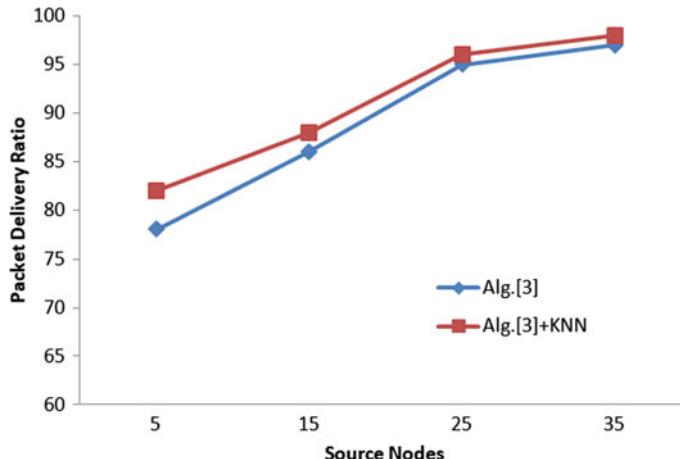


**Fig. 8** . Trust percentage of node w.r.t. node 35

Figure 9 shows the trustworthiness of various nodes having a value of 60% or



**Fig. 9** Trustworthiness of each node w.r.t node is 35



**Fig. 10** Packet delivery ratio of network with the different source node

more in both the algorithms. Node numbers 11, 57 achieved the highest 99% trustworthiness value in algorithm [28], and node numbers 31 and 82 reached 100% trustworthiness with KNN. The number of nodes having a 60% trustworthiness value is increased compared to source nodes 5 and 25 but decreases compared to source node 15. In algorithm [28], total of 46 nodes trustworthiness value is 50% or above, whereas 48 nodes are in KNN.

## 6 Packet Delivery Ratio

Figure 10 shows the packet delivery ratio of the overall network in different network scenarios. When the source node is 5, PDR value is lower as compared to other source nodes. Because during source 5, initially nodes are sparsely populated. The performance of algorithm [28]+KNN is better as compared to algorithm [28].

Both the algorithms are evaluated for four scenarios with different source node. The different scenario shows various values of trust percentage and trustworthiness. The value of trust percentage and trustworthiness is better with KNN and PDR values.

## 7 Conclusion

In this paper, we have analyzed the trust and trustworthiness of nodes. The trustworthiness of each node value helps the source node to find the most suitable trustworthy node. The trust percentage gradually increases with an increasing number of nodes in

the network based on their positive response. This factor plays a crucial role in establishing reliable communication, and without any fear, one can use this information as trusted information. This depends on the trustworthiness value; different source node has been arbitrarily selected and evaluated. This helps the source node to collect trusted information. Our proposed mechanism was enhanced with augmentation of the KNN algorithm, and the reliability of information significantly improved. The value of trustworthiness of nodes is varying in different scenarios. Our analysis of the trustworthiness of nodes with respect to the various source nodes is 5, 15, 25, and 35, respectively.

## 8 Future Scope

In this present work, we have studied the message authentication-based trustworthiness of individual nodes in the network. The proposed work used KNN algorithm, which can be further extended with other machine learning techniques. Also, road intersection points, traffic light scenarios may be used while computing message validity.

## References

1. Moustafa, H., Senouci, S.M., Jerbi, M.: Vehicular networks: techniques, standards, and applications. Chapter-1 Introduction to Vehicular Networks. Auerbach Publications Published April 9, 2009
2. Englund, C., Chen, L., Vinel, A., Lin, S.Y.: Future applications of VANETs. In: Campolo, C., Molinaro, A., Scopigno, R. (eds.) Vehicular Ad Hoc Networks. Springer (2015)
3. Sharma, K., Soni, S., Chaurasia, B.K.: Reputation and trust computation in VANETs. In: Proceedings of the IEMCON-14 Conference on Electronics Engineering and Computer Science, pp. 118–122 (2014)
4. Ding, Q., Jiang, M., Li, X., Zhou, X.H.: Reputation-based trust model in vehicular ad hoc networks. In: International Conference on Wireless Communications and Signal Processing (WCSP), pp. 1–6 (2010)
5. Shaikh, R.A., Alzahrani, A.S.: Trust management method for Vehicular Ad Hoc Networks. In: Singh, K., Awasthi, A.K. (eds.) Quality, reliability, security and robustness in heterogeneous networks. In: QShine 2013. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 115. Springer, Berlin, Heidelberg (2013)
6. Hasswa, A., Zulkernine, M., Hassanein, H.: Routeguard: An intrusion detection and response system for mobile ad hoc networks. In: Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, vol. 3, pp. 336–343 (2005)
7. Chaurasia, B.K., Verma, S.: Infrastructure based authentication in VANETs. Int. J. Multimedia Ubiquit. Eng. **6**(2), 41–54 (2011)
8. Tyagi, P., Dembla, D.: Investigating the security threats in Vehicular Ad Hoc Networks (VANETs): towards security engineering for safer on-road transportation. In: International

- Conference on Advances in Computing, Communications and Informatics, pp. 2084–2090 (2014)
- 9. Zhang, J.: A survey on Trust Management for VANET. In: International Conference on Advanced Information Networking and Applications, pp. 105–112. IEEE Computer Society (2011)
  - 10. Yang, X., Liu, L., Vaidya,N.: A vehicle-to-vehicle communication protocol for cooperative collision warning. In: 1st Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services, MOBIQUITOUS, pp. 114–123 (2004)
  - 11. Shukla, D., Arya, A.V., Das, S., Johri, P.: Security and attack analysis in VANET—a survey. In: IEEE International Conference on Computing Communication and Automation, 29–30 April 2016, pp. 625–630. <https://doi.org/10.1109/CCA.2016.7813797>
  - 12. Zhangm, S., Tao, J., Yuan, Y.: Anonymous Authentication-Oriented Vehicular Privacy Protection Technology Research in VANET, pp. 4365–4368. IEEE (2011)
  - 13. Zhang, C., et al.: An efficient message authentication scheme for vehicular communication. *IEEE Trans. Veh. Technol.* **57**(6), 3357–3368 (2008)
  - 14. Vaibhav, A., Shukla, D., Das, S., Sahana, S., Johri, P.: Security challenges, authentication, application and trust models for Vehicular Ad Hoc Network—a survey. *Int. J. Wirel. Microw. Technol. (IJWMT)* **7**(3), 36–48 (2017). <https://doi.org/10.5815/ijwmt.2017.03.04>
  - 15. Zhou, T., et al.: P2DAP—Sybil attacks detection in vehicular Ad Hoc networks. *IEEE J. Sel. Areas Commun.* **29**(3), 582–594 (2011)
  - 16. Chaurasia, B.K., Verma, S., Tomar, G.S.: Trust computation in VANETs. In: Proceedings of the International Conference on Communication Systems and Network Technologies (CSNT), pp. 468–471 (2013)
  - 17. Saraswat, D., Chaurasia, B.K.: AHP based trust model in VANETs. In: Proceedings of the IEEE 5th International Conference on Computational Intelligence and Communication Networks (CICN), pp. 391–393 (2013)
  - 18. Marmol, F.G., Perez, G.M.: TRIP, a trust and reputation infrastructure based proposal for vehicular ad hoc networks. *J. Netw. Comp. Appl.* **35**(9), 34–941 (2012)
  - 19. Obimbo, C., Arboleda, L.M., Chen, Y.: A watchdog enhancement to IDS in MANET. In: Proceedings of the IASTED Conference on Wireless Networks, pp. 510–515 (2006)
  - 20. Marti, S., Giuli, T.J., Lai, K., Baker, M.: Mitigating routing misbehavior in mobile ad hoc networks. In: Proceeding of the 6th ACM Annual International Conference on Mobile Computing and Networking, pp. 255–265 (2000)
  - 21. Viejo, A., Sebe, F., Domingo-Ferrer,J.: Aggregation of trustworthy announcement messages in Vehicular Ad Hoc Networks. In: VTC Spring 2009—IEEE 69th Vehicular Technology Conference, Barcelona, pp. 1–5 (2009)
  - 22. Shrestha, R., Nam, S.Y.: Trustworthy event-information dissemination in Vehicular Ad Hoc Networks. *Mobile Inf. Syst.* **2017**, Article ID 9050787, 16 (2017). <https://doi.org/10.1155/2017/9050787>
  - 23. Gazdar, T., Belghith, A., AlMogren, A.: DTCF: a distributed trust computing framework for Vehicular Ad Hoc Networks. *KSII Trans. Internet Inf. Syst.* **11**(3) (2017)
  - 24. Tripathi, K.N., Sharma, S.C.: A trust based model (TBM) to detect rogue nodes in Vehicular Ad-Hoc Networks (VANETS). *Int. J. Syst. Assur. Eng. Manag.* **11**, 426–440 (2020). <https://doi.org/10.1007/s13198-019-00871-0>
  - 25. Xia, H., Zhang, S., Li, B., Li, L., Cheng, X.: Towards a novel trust-based multicast routing for VANETs. *Secur. Commun. Netw.* **2018**, Article ID 7608198, 12 (2018). <https://doi.org/10.1155/2018/7608198>
  - 26. Chuprov, S., et al.: Reputation and trust approach for security and safety assurance in intersection management system. *Energies* **12**(23), 4527 (2019)
  - 27. Nidhi, Lobiyal, D.K.: Performance evaluation of RSUs deployment at dense intersections. *Int. J. Inf. Tecnol.* **13**, 1095–1099 (2021). <https://doi.org/10.1007/s41870-021-00642-w>
  - 28. Das, S., Das, I., Singh, R.P., Johri, P., Kumar, A.: Trust-based scheme for location finding in VANETs using trustworthiness of node. In: Jain, L., Balas, V.E., Johri, P. (eds.) Data and Communication Networks. Advances in Intelligent Systems and Computing, vol. 847, pp. 43–55. Springer, Singapore (2019)

# Author Index

## A

- Aakash Nakarmi, 199  
Abdullah Mohammed Zaki Khayyat, 429  
Abinash Sahoo, 295  
Abul Hasnat, 307  
Alharthi, Abdulrahman Hamad, 429  
Almarwani, Ziyad Saad, 429  
Almohammadi, Faisal Lafi, 429  
Amanam, Nkanga Nkanga, 441  
Amit Kumar Pandey, 407  
Amit Kumar Shakya, 417  
Amlan Jyoti, 513  
Anil Kumar Sagar, 147, 199  
Ankit Prakash, 157  
Ankush Ghosh, 317, 335, 369, 385, 463, 477, 489, 527  
Anurag Vidyarthi, 417  
Arjun, R. K., 357  
Arun Pandian, J., 89, 117  
Arpan Adhikary, 463, 489  
Aruna, R., 117  
Ashraf Alam, 395  
Ashwini Kumar, 541  
Avishek Banerjee, 477  
Ayushman Ramola, 417

## B

- Bandana Barman, 307  
Boronat, Fernando, 1  
Buddha Singh, 451

## C

- Çamur, Hüseyin, 441

## D

- Dayima Musharaf, 147  
Deba Prakash Satapathy, 295  
Debaroti Sammanit, 385  
Dhanush, B., 211  
Dhanya, D., 211  
Dhyanendra Jain, 407  
Dibyendu Barman, 307  
Divakar Kumar, 281

## F

- Fornés-Leal, Alejandro, 1

## G

- Ganzha, Maria, 1  
Gunpreet Singh Walia, 281

## H

- Hadya Jahangir, 199  
Harshad B. Nehate, 251

## I

- Indrani Das, 173, 541

## J

- Jaypal Singh Rajput, 41  
Jitendra Singh Kushwah, 407

## K

- Kameswara Rao, M., 81

Kanagalal Saideepak, 117  
 Kanchanadevi, K., 89, 117  
 Kassem, Youssef, 441  
 Kavita Tewari, 251  
 Koushik, C., 357  
 Koushik Majumder, 463, 477, 489  
 Krishna Mridha, 317, 335

**L**

Lacalle, Ignacio, 1  
 Lalitha Rangarajan, 67  
 Laxmi Rajput, 55

**M**

Mamta Mann, 183  
 Manish Sharma, 41  
 Milan Limbu, 317  
 Monika Jangra, 451  
 Munsifa Firdaus Khan, 173  
 Mupparaju Kavya Sree, 117

**N**

Nalin Kashyap, 281  
 Nihar R. Mohanta, 295  
 Nikita Sinha, 281  
 Nitima Malsa, 369, 527  
 Nithyanthan Kannan, 429

**P**

Palau, Carlos E., 1  
 Parth K. Kharkar, 251  
 Paprzycki, Marcin, 1  
 Prachi Waghmare, 33  
 Pranav, M. V., 357  
 Prashant Singh, 407  
 Prashant Vats, 407  
 Pratiksha S. Bhat, 251  
 Pratima Singh, 17  
 Priyanka Singh, 385

**R**

Rabindra Nath Shaw, 317, 335, 369, 385,  
 463, 477, 489, 527  
 Rajesh Thakare, 513  
 Rapti Chaudhuri, 225  
 Rashmi, J. R., 67  
 Rinkesh Pantawane, 513  
 Rishipal Singh, 183, 541  
 Rohit Kumar, 89

Rukmani, P., 211  
 Runa Saha, 317

**S**

Sandeep Samantaray, 295  
 Sangeeta Sheoran, 513  
 Sanjay Kumar, 55  
 Sanjoy Das, 281, 407, 541  
 Santanu Chatterjee, 463, 489  
 Sara Renjit, 237  
 Saravanan, A., 131  
 Satyam Kumar, 147  
 Saurav Kr. Gupta, 89  
 Seerat Musharaf, 199  
 Shaik. Mohissin Sultana, 81  
 Shakil Sarkar, 335  
 Shanvi Sharma, 17  
 Sharen, H., 211  
 Shekhar Karanwal, 99  
 Shivali Sonarkar, 513  
 Shivani Pathak, 369, 527  
 Shourjya Hazra, 89  
 Shridevi, S., 357  
 Shubham, 267  
 Shyamala Devi, M., 117  
 Sourav Simanto, 317  
 Stanly Felix, C., 131  
 Subrata Sahana, 281, 541  
 Sudip Kumar De, 477  
 Sumam Mary Idicula, 237  
 Suman Deb, 225, 267  
 Sunil Kumar Singh, 157  
 Swati Shinde, 33

**T**

Tella Malathi, 117  
 Tiyasha Dhara, 489

**U**

Umarani, M., 131

**V**

Varad S. Rane, 251  
 Vaño, Rafael, 1  
 Vimal Gupta, 369, 527

**Y**

Yogeshwar Uikey, 513  
 Youssef Mobarak, 429

**Z**

Zakwan, Ahmed Hamid Mohamed Abdalla,

[441](#)