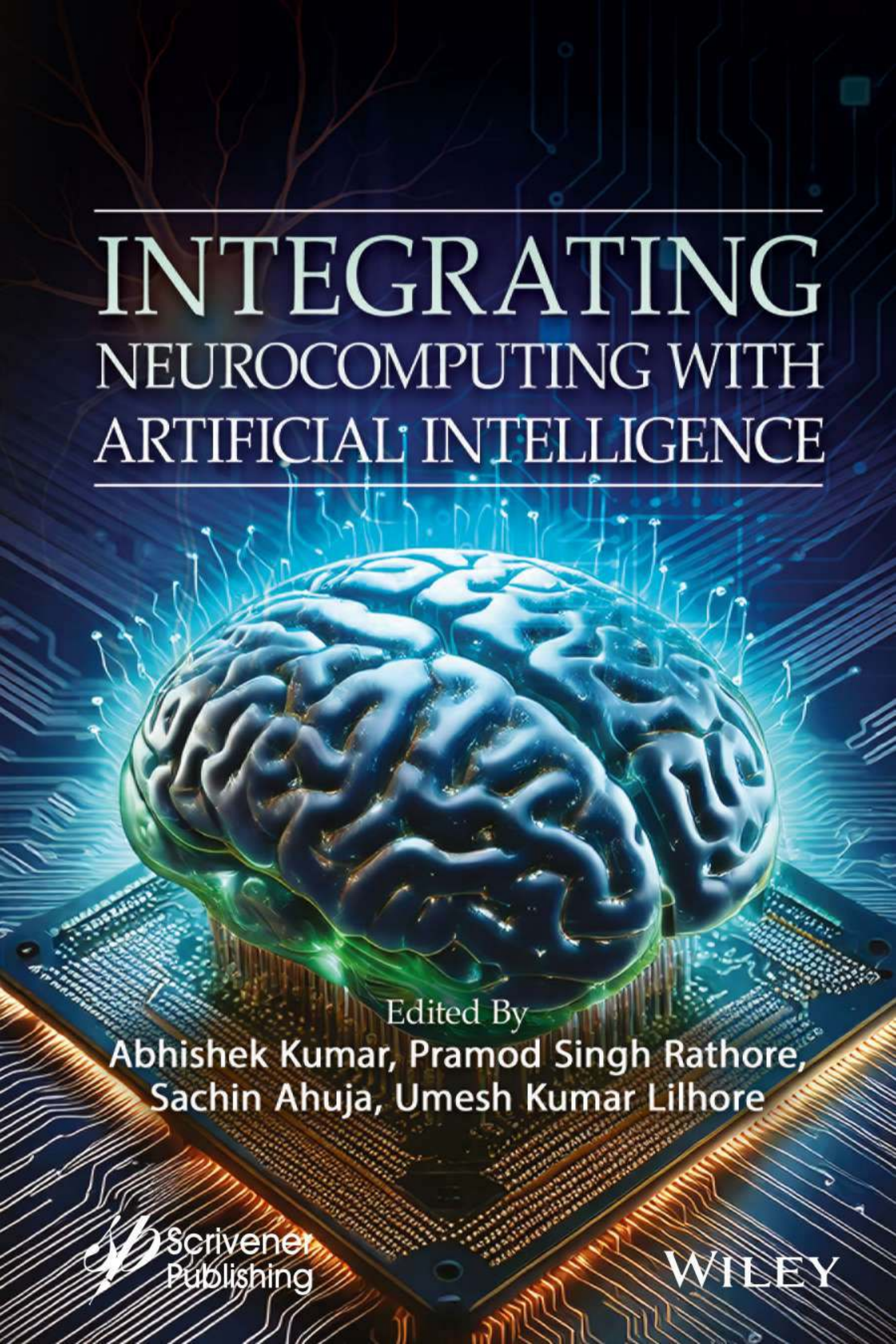


INTEGRATING NEUROCOMPUTING WITH ARTIFICIAL INTELLIGENCE

Edited By

Abhishek Kumar, Pramod Singh Rathore,
Sachin Ahuja, Umesh Kumar Lilhore

Scrivener
Publishing

WILEY

Integrating Neurocomputing with Artificial Intelligence

Scrivener Publishing
100 Cummings Center, Suite 541J
Beverly, MA 01915-6106

Publishers at Scrivener
Martin Scrivener (martin@scrivenerpublishing.com)
Phillip Carmical (pcarmical@scrivenerpublishing.com)

Integrating Neurocomputing with Artificial Intelligence

Edited by

Abhishek Kumar

Pramod Singh Rathore

Sachin Ahuja

and

Umesh Kumar Lilhore

Scrivener
Publishing

WILEY

This edition first published 2025 by John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, USA and Scrivener Publishing LLC, 100 Cummings Center, Suite 541J, Beverly, MA 01915, USA

For more information about Scrivener publications please visit www.scrivenerpublishing.com.

Wiley Global Headquarters
111 River Street, Hoboken, NJ 07030, USA

For details of our global editorial offices, customer services, and more information about Wiley products visit us at www.wiley.com.

Limit of Liability/Disclaimer of Warranty

While the publisher and authors have used their best efforts in preparing this work, they make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives, written sales materials, or promotional statements for this work. The fact that an organization, website, or product is referred to in this work as a citation and/or potential source of further information does not mean that the publisher and authors endorse the information or services the organization, website, or product may provide or recommendations it may make. This work is sold with the understanding that the publisher is not engaged in rendering professional services. The advice and strategies contained herein may not be suitable for your situation. You should consult with a specialist where appropriate. Neither the publisher nor authors shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages. Further, readers should be aware that websites listed in this work may have changed or disappeared between when this work was written and when it is read.

Library of Congress Cataloging-in-Publication Data

ISBN 9781394335688

Cover image: Generated with AI using Adobe Firefly
Cover design by Russell Richardson

Set in size of 11pt and Minion Pro by Manila Typesetting Company, Makati, Philippines

Printed in the USA

10 9 8 7 6 5 4 3 2 1

Contents

Foreword

The convergence of neurocomputing and artificial intelligence (AI) marks a transformative era in computational sciences. As AI continues to evolve, its intersection with neurocomputing has paved the way for brain-inspired models, cognitive computing, and adaptive intelligence, leading to groundbreaking applications across various industries. This book, *Integrating Neurocomputing with Artificial Intelligence*, provides a timely and comprehensive exploration of this emerging domain, offering insights into both foundational theories and cutting-edge advancements.

In the current technological landscape, AI has made significant strides in machine learning, deep learning, reinforcement learning, and natural language processing. However, despite these advancements, conventional AI systems often struggle with energy efficiency, real-time adaptability, and cognitive reasoning, areas where neurocomputing plays a crucial role. Neurocomputing, inspired by the structure and function of biological neural networks, provides novel computational paradigms that aim to mimic the brain's learning, perception, and decision-making abilities. This book delves into the integration of these fields, showcasing how neuromorphic computing, brain-inspired AI, and hybrid models can create more efficient, intelligent, and sustainable systems.

The chapters in this volume bring together leading researchers, engineers, and industry experts, presenting a multidisciplinary perspective on topics ranging from neuromorphic architectures, spiking neural networks (SNNs), bio-inspired computing, and hybrid AI models to their applications in healthcare, robotics, autonomous systems, cybersecurity, and smart environments. This compilation not only highlights state-of-the-art research but also underscores the challenges and opportunities that lie ahead in building more adaptive, interpretable, and scalable AI systems.

As industries increasingly adopt AI-driven solutions, the need for brain-like intelligence, real-time decision-making, and computational efficiency has never been more critical. This book serves as an essential resource for academicians, professionals, and students seeking to understand and contribute to the rapidly evolving field of AI-integrated neurocomputing.

I commend the editors and contributors for their remarkable effort in compiling this insightful volume. I am confident that *Integrating Neurocomputing with Artificial Intelligence* will inspire researchers, innovators, and practitioners to explore new frontiers in intelligent computing, ultimately shaping the future of AI-driven technologies.

Dr. Rashmi Agrawal
Professor and Associate Dean, School of Computer Applications,
Manav Rachna International Institute of Research and Studies (MRIIRS),
Faridabad, India

Preface

This book is organized into seventeen chapters. In Chapter 1, energy management in modern smart grids requires intelligent decision-making systems that optimize energy distribution and consumption. This chapter explores the synergy between fog computing and AI-driven energy models, presenting an architecture that enhances energy distribution efficiency. Using machine learning and neural networks, the authors demonstrate an advanced cloud-fog-based decision-making framework that ensures seamless energy optimization while addressing latency issues.

In Chapter 2, neural networks have revolutionized language processing and text analytics, yet challenges remain in integrating temporal dependencies efficiently. This chapter introduces a hybrid approach that combines Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks for superior performance in natural language processing (NLP) tasks. Through detailed simulations, the authors provide insights into architecture design, training techniques, and performance benchmarking.

In Chapter 3, this chapter explores the use of Industry 4.0 is driven by autonomous mobile robots, which require secure, real-time decision-making systems to navigate industrial environments safely. This chapter presents a cyber-physical security framework that leverages Software-Defined Networking (SDN) and Internet of Robotic Things (IoRT) to enhance robotic safety. The proposed system enables secure communication, real-time threat mitigation, and automated attack detection.

In Chapter 4, in this research chapter, Medical diagnostics have greatly benefited from AI-powered feature extraction and pattern recognition techniques. This chapter explores a hybrid neuro-fuzzy computing framework designed for disease classification and medical image analysis. The authors present a linguistic fuzzification model that enhances disease detection accuracy, offering significant contributions to biomedical AI applications.

In Chapter 5, the book chapter affords brief and general information regarding Advancements in neuromorphic vision systems are transforming robotic automation. This chapter presents an AI-powered neuromorphic vision-based control system for robotic drilling applications, improving precision, speed, and adaptive learning. The authors discuss sensor integration, event-based vision processing, and real-time control strategies, making this work highly relevant to industrial robotics and automation.

In Chapter 6, autonomous vehicles require real-time decision-making models that mimic human cognitive functions. This chapter discusses neuromorphic AI frameworks that enhance path planning, perception, and adaptive control in self-driving cars. The proposed neural engineering architecture integrates spiking neural networks and reinforcement learning to improve vehicle maneuverability and collision avoidance.

In Chapter 7, brain-Computer Interfaces (BCI) enable direct neural communication with machines, leading to significant advancements in humanoid robotics and assistive technologies. This chapter introduces an adaptive BCI system for humanoid robot control, focusing on steady-state visual evoked potentials (SSVEPs) and real-time signal processing techniques for enhanced human-robot interaction.

In Chapter 8, decision-making is a fundamental AI application across industries. This chapter explores deep learning-based decision-making models that improve operational performance in business, healthcare, and logistics. The authors present an Artificial Neural Network (ANN)-based framework, focusing on predictive analytics, optimization, and intelligent automation.

In Chapter 9, speech recognition plays a critical role in human-computer interaction and automated language translation. This chapter presents an AI-powered speech recognition framework leveraging Natural Language Processing (NLP) and deep learning. The authors discuss acoustic modeling, error correction, and real-time voice scoring, highlighting practical applications in education and AI-driven assistants.

In Chapter 10, in this chapter user give AI-driven medical imaging has enhanced early detection of ocular diseases. This chapter introduces deep learning-based neurocomputing models for classifying ophthalmological disorders using Optical Coherence Tomography (OCT) images. The proposed YOLOv3 and ResNet50 architectures improve diagnostic accuracy, offering valuable insights for automated medical analysis.

In Chapter 11, data security is critical in modern communication systems. This chapter presents an innovative multi-image steganography model using Deep Convolutional Neural Networks (CNNs). The approach introduces private keys for encrypted image transmission, ensuring high security, robustness against steganalysis, and effective information concealment.

In Chapter 12, biodiversity conservation benefits from AI-driven species classification models. This chapter presents a deep learning-based framework for automated honey bee subspecies identification, utilizing morphometric analysis and image processing. The proposed model significantly improves classification accuracy, demonstrating the potential of AI in entomology and ecological research.

In Chapter 13, neural networks have revolutionized speech recognition and acoustic modeling. This chapter explores spiking neural networks (SNNs) for automatic speech recognition (ASR), presenting models for large vocabulary speech processing and phoneme recognition. The authors analyze energy-efficient deep SNNs, making significant contributions to speech technology advancements.

In Chapter 14, this chapter engages in discusses Brain-Computer Interfaces (BCIs) enable direct neural interaction with humanoid robots, transforming rehabilitation and assistive technology. This chapter discusses a brainwave-controlled robotic system based on steady-state visual evoked potentials (SSVEPs), showcasing its applications in robotic control and cognitive computing.

In Chapter 15, this book chapter presents Medical data augmentation using Generative Adversarial Networks (GANs) improves diabetes prediction models. This chapter presents a GAN-based approach for simulating glucose monitoring data, enhancing machine learning models for hypoglycemia detection and personalized diabetes care.

In Chapter 16, in this study, Neuromorphic computing offers brain-inspired AI solutions for high-performance computing and edge intelligence. This chapter presents an in-depth analysis of spiking neural networks, reservoir computing, and quasi-backpropagation algorithms, highlighting their impact on neuromorphic hardware and AI applications.

In Chapter 17, this chapter explains the purpose of Quantum computing is reshaping AI by enabling parallel computation and probabilistic learning. This chapter explores the integration of quantum machine learning with neural networks, focusing on quantum-enhanced reinforcement learning, quantum annealing, and quantum convolutional networks (QCNNs).

Dr. Abhishek Kumar
Department of Computer Science and Engineering,
Chandigarh University, Punjab, India
Dr. Pramod Singh Rathore
Department of Computer and Communication Engineering,
Manipal University Jaipur, India
Dr. Sachin Ahuja
Department of Computer Science and Engineering, Chandigarh University,
Punjab, India
Dr. Umesh Kumar Lilhore
Department of Computer Science and Engineering, Galgotias University,
Greater Noida, UP, India

1

Integrating Fog Computing with AI Model on Decision Making for Distribution of Energy Management

Prajwal Hegde N.[1]*, Parvathi C.[2], Ajay Malpani[3], D. Suganthi[4] and Priya Batta[5]

[1]Department of Artificial Intelligence and Data Science, NMAM Institute of Technology, Nitte Deemed to be University, Karkala, Karnataka, India
[2]Department of Computer Science Engineering, BGSCET, Bangalore, India
[3]Department of Management, Prestige Institute of Management and Research (PIMR), Indore, India
[4]Department of Computational Intelligence, Saveetha College of Liberal Arts and Sciences, SIMATS, Thandalam, Chennai, India
[5]Dept. of CSE, Chandigarh University, Punjab, India

Abstract

New obstacles to effective energy organization for system process are emerging as the number of Internet of Things strategies and dispersed energy possessions in the next-generation spreading network continues to grow. One explanation is that the supervisory control and data achievement system has limited computing and storage capacity; thus, it cannot link all the large-scale resources. An innovative approach to energy management known as cloud-fog classified architecture is presented in this study as a means to meet the evolving demands of next-generation distribution networks. The utility and revenue model that developed based on this design included regular consumers, prosumers, and operators of the distribution system. Additionally, energy management might be automatically accomplished by integrating an AI module into the suggested design. This study employs neural networks at the fog computing layer to make regression predictions of power source output and energy use behavior. Moreover, at the network's cloud layer, a genetic algorithm was used to optimize prosumers' and customers' energy usage in accordance with the maximizing utility goal function. Results, including recorded

Corresponding author: prajwal.hegde@nitte.edu.in

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (1–14) © 2025 Scrivener Publishing LLC

customer use patterns and stakeholder income, show that the suggested strategies work with a sample of regular and prosumer consumers in a generic distribution network. Building next-generation distribution network real-time energy management systems may benefit significantly from this work as a reference.

Keywords**:** Internet of Things, energy management, regression, predictions, cloud layer

1.1 Introduction

Conventional power users who own these small-scale generating facilities are becoming prosumers due to the fast penetration of DERs into the distribution network (DN). This means that they use energy from the utility grid and produce it [1, 2]. More grid operating flexibility is being made possible by the rising number of active prosumers, who enable bidirectional energy flows. Both ecological concerns and the desire of home prosumers to reap the benefits of efficient energy transactions with the grid are propelling this shift [3, 4].

By 2020, experts predict that 26 billion gadgets will be linked to the Internet of Things (IoT). A new age has dawned with the advent of the Internet of Things (IoT), in which a wide variety of end devices and sensors are connected wirelessly or via wires using different forms of contemporary communication and the Internet [5–7]. An ever-increasing number of controllable units are a part of the DN's energy operation and management process, and the frequency of information and data exchange between various parties is on the rise due to the proliferation of IoT devices and the widespread use of energy cyberspace knowledge in the power grid [8, 9]. A new generation of distribution networks is possible because of the widespread adoption of smart devices and renewable energy sources (RES) in the distribution network [10, 11]. This next-generation delivery grid's real-time energy management is becoming increasingly important [12].

With the widespread use of distributed RES and the integration of massive Internet of Things (IoT) devices into next-generation distributed networks (DN), this study intends to tackle the problem of energy management and executive [13]. First, a hierarchical fog-cloud design is suggested for decision-making and energy management. AI technologies deployed independently in fog and cloud layers using the large-scale data created in DN may capture customers' consumption and RES production [14]. Users' energy consumption behavior may be captured by microeconomic theory as well [15]. The model encompasses several stakeholders, including regular customs, prosumers, and the distribution system operator (DSO).

Finally, the method's practicality is shown by optimizing retail power pricing and managing diverse DN stakeholders' income in real time [16, 17].

The paper is prepared like this. Section 1.2, lays down the groundwork for the cloud-fog hierarchical architecture that propose for DN energy management and explain how fog and cloud layer's function. Part 3 of the model describes the typical consumer, the prosumer, and the DSO. In Section 1.4, put the verification into action by incorporating AI technologies into the cloud and fog layers for energy organization and executive. At the fog layer, the main focus is on predicting power consumption and creating renewable energy sources (RES), while in the cloud, optimization of computations for particular objectives takes place. Optimal goal optimization with complete social welfare reproduction including ordinary users, prosumers, and DSO in a local DN based on utility and income models.

1.2 Methodology

1.2.1 Energy Management Using a Cloud-Fog Hierarchical Architecture

The suggested cloud-fog hierarchical architecture is mostly presented in this portion (see Figure 1.1). The fog computing deposits conduct gathering analysis and regression forecast by mining the fundamental data from the units of main consumers and prosumers in the DN. Using the cloud computation layer allows us to optimize the overall goal.

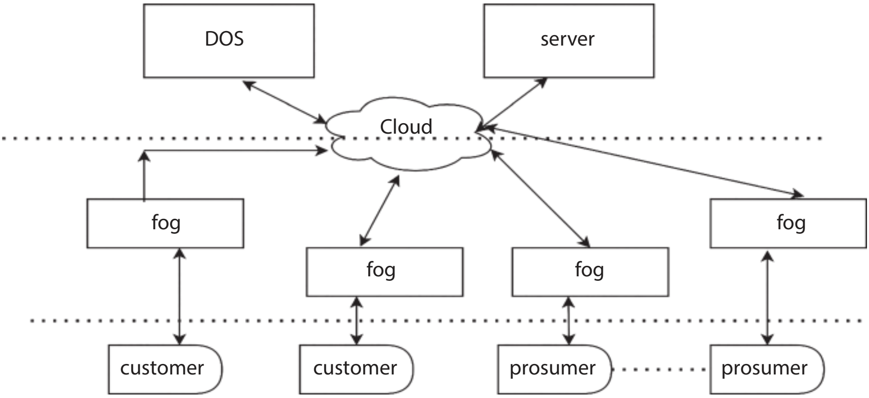


Figure 1.1 Energy management using cloud fog. Distribution operator.

1.2.2 Units Terminal

Customer or prosumer-installed DER (Internet of Things, photovoltaic, wind turbine, storage, etc.) and other IoT devices make up the terminal units of the next-generation DN. The communication and connectivity structure of the DN is shown in Figure 1.2. The database system stores data from clever meters and IoT devices. Fog layers, which are linked to the database systems, may conduct particular computational operations and prepare input for services supplied by higher layers. Wireless or wired protocols like Zigbee, 802.11, and 802.15 may facilitate communication between the device and the local area network gateway [18, 19]. Also, the gateway may gather data from utilities as well as Internet of Things devices; then, the terminal units and DN may communicate using the Open ADR protocol to operate the stated behaviors.

1.2.3 Operating Fog Layers

Fog computing involves placing databases and central processing units (CPUs) at the DN's designated nodes to handle requirements from users and DN operators [20, 21]. By storing and managing the terminal information, fog computing may alleviate the strain on cloud data processing and latency. Figure 1.2 shows that the user's smart meter and gateway may be communicated with by the fog computing nodes.

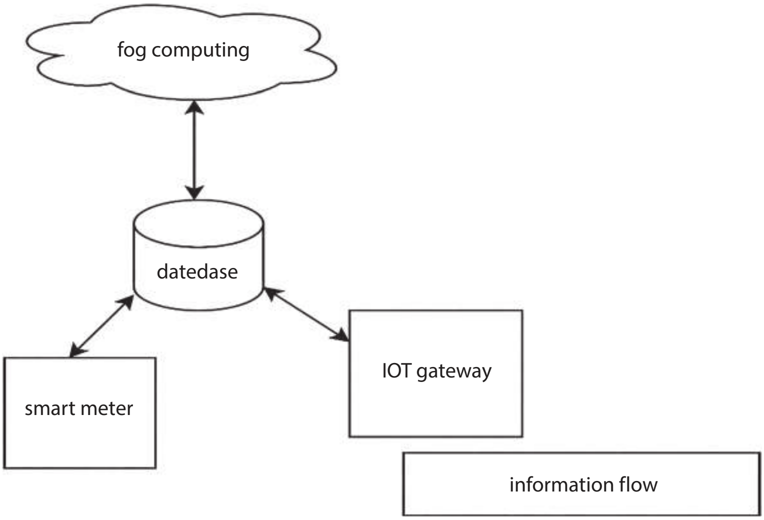


Figure 1.2 IoT device communication diagram.

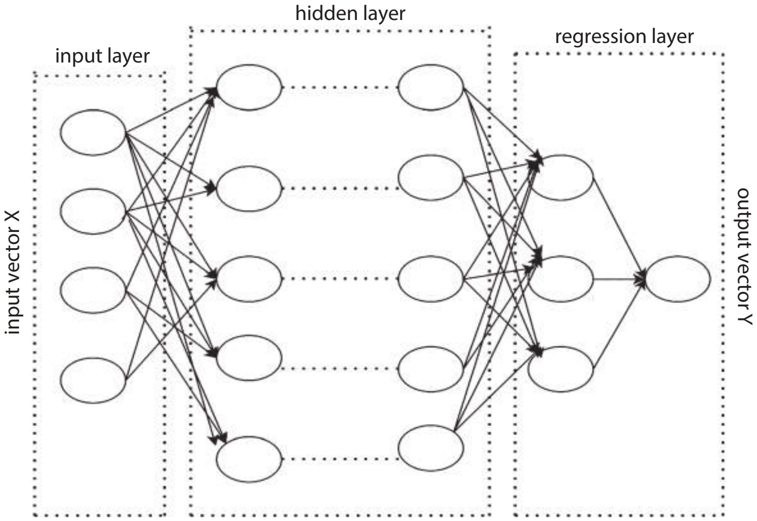


Figure 1.3 Neuronal network (NN) deployment diagram for fog layers.

Furthermore, under some scenarios, customers' power use behavior may be captured by AI modules put at the fog layers. As the amount of data continues to grow, artificial neural networks (ANN) have shown encouraging gains in machine learning and pattern recognition. Figure 1.3 shows how artificial NNs may be trained and utilized for regression analysis using sample data. A typical ANN has three types of input layers: hidden, regression, and general [22, 23]. For instance, in a regression study of a user's power consumption, the variables that may influence consumption behavior are the input data and the quantity of energy used is the output. Then, use NN's regression analysis to forecast how consumers and prosumers would consume. This lets distribution system operators get sufficient load management data from the fog computing layers. In addition, geographical data, weather reports, distributed power type, and other inputs may be used in reversion analysis using NN learning and exercise at fog layers using the outputs from RES for prosumers [24–26].

1.2.4 Operation of the Cloud Layer

Optimal scheduling, stability calculations, and market transaction participation are all responsibilities of the cloud layer, which makes decisions founded on data acquired from fog films and manages the energy consumption of the whole DN. A vast area network, like the Internet, may be used to communicate with fog and provide command information. The best choice will be assisted by the AI algorithm that is implemented on the cloud [27–29].

In this paper, GA a method that consistently solves large-scale discrete and nonlinear problems—to take on the cloud-established global optimization issue. Through the use of the goal function, GA encodes all potential issue solutions into a vector, with each gene being a component of the vector. In a manner similar to how mutation, trade, and natural selection work in biology, this method ensures that only the strongest will remain. As seen in Figure 1.4, new generations are produced by various GA processes such as selection, exchanging, and mutation based on the degree of fitness.

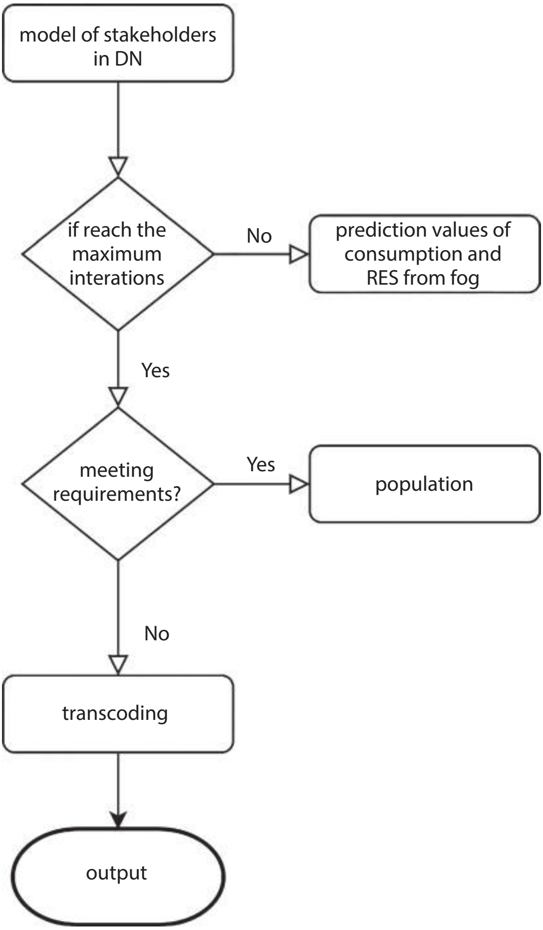

model of stakeholders in DN
if reach the maximum interations
No
prediction values of consumption and RES from fog
Yes
meeting requirements?
Yes
population
No
transcoding
output

Figure 1.4 Graph showing how the genetic algorithm may optimize decision making.

1.3 Modelling of Different Distribution Network Stakeholders

1.3.1 Consumers' Usefulness Model

Users' tastes and patterns of power use are often unrelated. Using the Internet of Things (IoT), smart beats, and big data knowledge makes it feasible to record each user's power consumption patterns. Collecting data like infection, electricity price, energy time, power, etc., allows one to study the electrical habits of clients. Using microeconomics' functional idea, this research constructs a utility model for residential consumers. Various efficacy functions $U(XIT,\ddot{\upsilon}IT)$ are chosen to represent the user's power level. The function represents customers' utility satisfaction. In this research, a quadratic equation function with diminishing bordering utility to define $U(XIT,\ddot{\upsilon}IT)$.

$$U\left(X_T^I,W_I^T\right)=\left\{W_I^T X_I^T-\frac{\propto}{2}\left(X_I^T\right)\right\}$$

Where it stands for the amount of energy the client uses at time slot T, ωIT describes how the user uses energy at time slot t, and α0, a predetermined value, denotes the unchanging circumstances.

Figure 1.5 displays how utility variations for different consumers as energy consumption increases. Additionally, it shows that, to varied

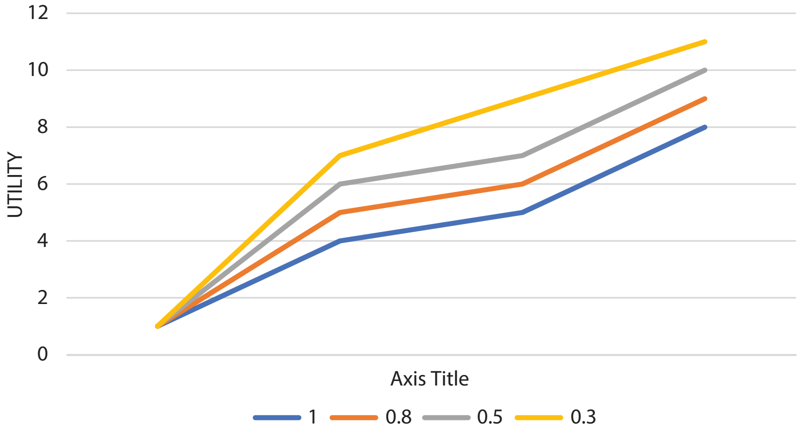


Figure 1.5 Energy consumption as a utility function for customers.

degrees, bordering utilities of all levels of pleasure are lessening. As an illustration of the declining impact, the marginal utility ϋ = 0.5 is used.

The customer's utility and the cost of buying power make up the customer's overall income. This leads to the following expression for customer's comprehensive utility in time interval t:

$$R_{CUMSTOMERS,I=\sum_t^T [u(x_{i,}^t wi]}$$

where xit is the amount of energy used by client i at time slot t and Pretailert is the real-time trade price for buying energy of consumers.

1.4 Results

This test utilizes a generic distributed network (DN) to spread 55 prosumers, each with a 20KW RES and 503 typical consumers. The suggested cloud-fog construction is used to optimize decision-making and execute energy management. Data generation rate and prosumer and customer locations dictate the placement and amount of fog nodes. Typically, the distribution grid's point of standard coupling (PCC) links the prosumers to a bus of the fundamental DN. Word trees and photovoltaics are the significant types of generators among prosumers. Tesla Powerwall's are used for storage, with a capital cost of ₹416250 INR and a lifespan of 15 years. Before calculating the real-time consumption characteristics ϋ, prosumers and consumers may use energy management techniques such as flow, regression, and forecasting at the fog layers for renewable energy sources (RES) and loads. The cloud layer may also maximize the DN's retail price and the total quantity of energy bought from the wholesale market.

1.4.1 Operating a Fog Computing System

F fog layers utilize multi-layer feed-forward NNs trained using the Levenberg-Marquardt approach to track and understand how much electricity each customer uses. After that, forecasts are generated using regression analysis. This test makes use of the load data from blond buildings in Germany. This is a list of the NN parameters: The data supplies the following details: 1) the input data is the expected value of the clients' incessant load; 2) the output data is time, temperature, humidity, and lighting; 3) this NN has a single hidden layer with ten neurons; 4) training uses 70% of the data, validation 15%, and testing 15%. 5) A network's generalizability being unaltered will cause NN training to end. This is where the regression analysis yielded its conclusions.

Figure 1.6 shows that after 34 generations of NN training, the root-mean-square error begins to meet the criterion, and after 40 iterations, it stops learning. A positive regression effect is often defined as a result greater than 0.9. With an R-value greater than 0.92, the reversion effect is considered adequate after training. Figure 1.7 shows the graph of normalized utility.

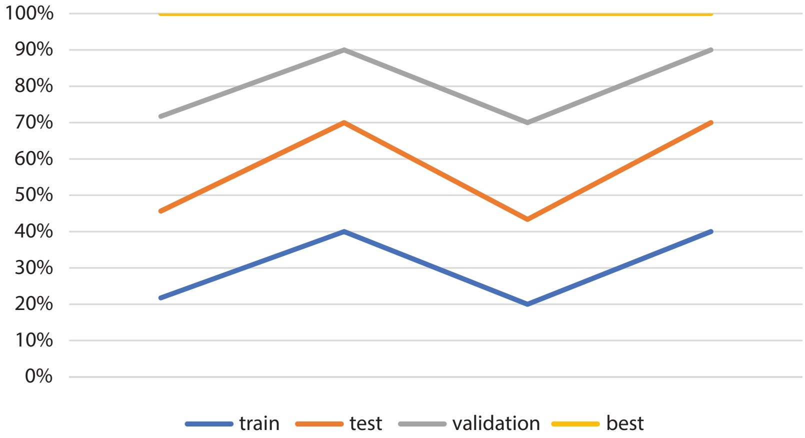


Figure 1.6 A trajectory for the generalization of NN learning.

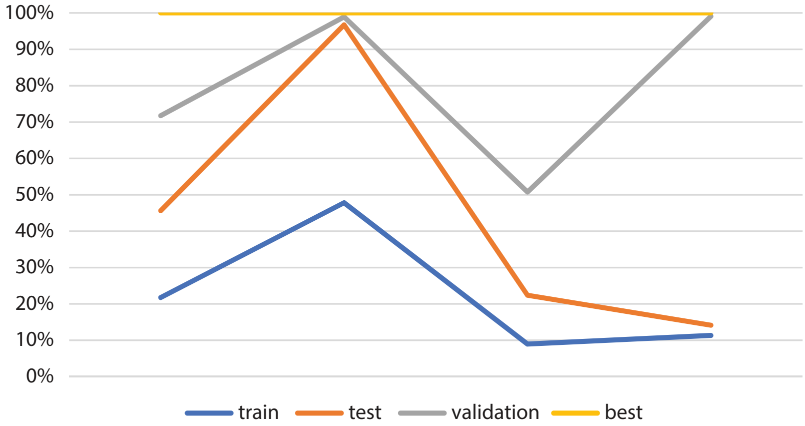


Figure 1.7 The utility function is available to customers in only 1 day.

Prosumers' RES output power may also be predicted using the NN incorporated in fog computing layers. Like load prediction by regression analysis, the procedure is straightforward. Even if they are not using NN to forecast RES output power here, since Another area of research is the use of NN for RES prediction; this test also includes a daily storage operation for arbitrage or prosumers. To make the test more efficient, the data of renewable energy sources that flow into the grid are taken straight from Open Energy Information (OpenEI).

1.4.2 Computing Operation Cloud

The goal of maximizing overall social welfare, as described in Section 4.1, is put into action at the cloud computation layer using data acquired from fog layers. Here are the specifics of the different parties involved:

Each time slot's ῡ is taken as v, the average value, when optimization is done in real-time at the cloud layer. On a global scale, in the cloud, the intelligent GA tool optimizes decision-making goals. The following parameters of GA are chosen:

- Cross-inheritance is set to 0.6.
- The mutation rate is 0.05.
- Maximum genetic algebra is 30.
- The population size is chosen as 40.

The input data of 13 hours to ensure the method is correct before computing the whole time (1–24 hours). The GA algorithm's convergence trajectory is seen in Figure 1.8.

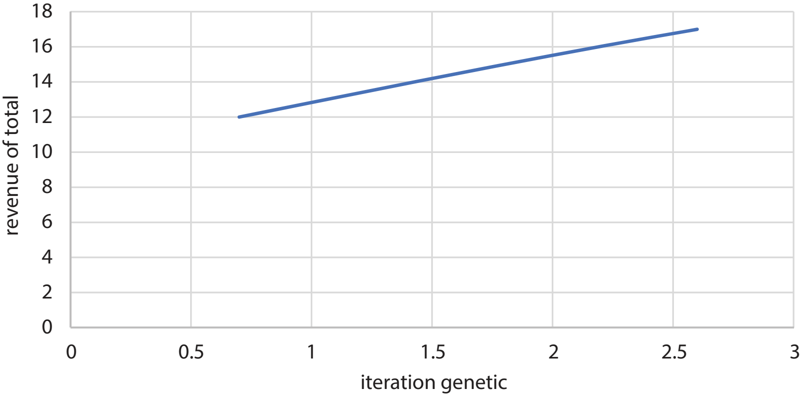


Figure 1.8 GA optimization convergence trace.

By the 23rd generation, a trend toward convergence in the outcomes becomes apparent. Figure 1.9 displays the outcome of the real-time purchase load.

Figure 1.10 shows the retail price of DN in real time, which is utilized to compute the real-time proceeds of stakeholders.

The DSO may develop a plan for wholesale purchase after the installation at the fog/cloud levels. The current wholesale market price is 0.353% per kilowatt-hour. Lastly, the daily earnings and expenses for each stakeholder. According to the above findings, the suggested energy management architecture can organize and handle various operational data in the DN

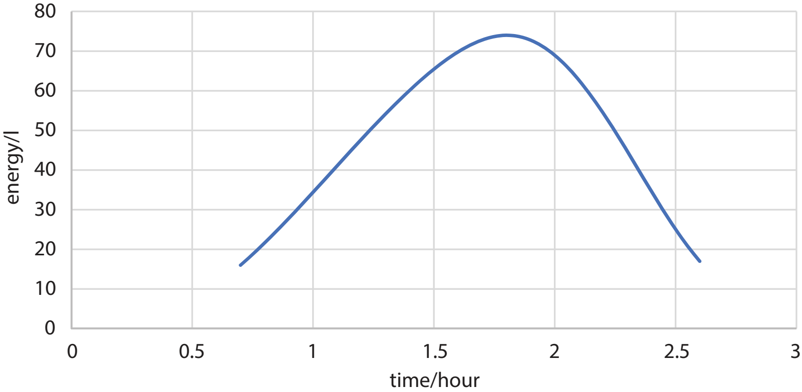


Figure 1.9 The total quantity of power bought from the wholesale market in a day.

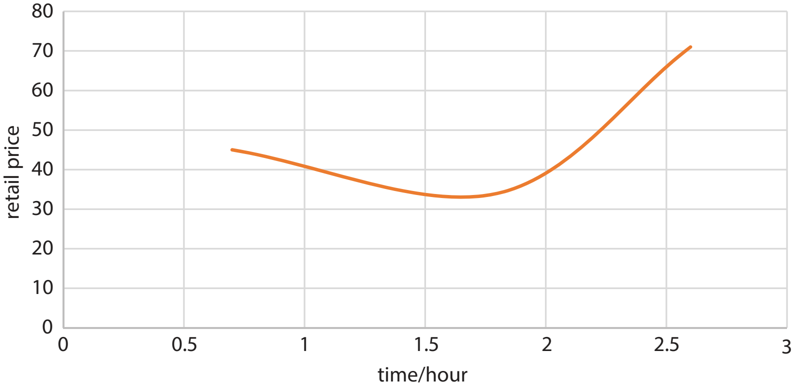


Figure 1.10 The retail price of electricity for 1 day.

in real time. The technology offers automatic energy management and an hourly executive procedure for the next-gen DN.

1.5 Conclusions

The pervasiveness of DERs and IoT devices in DN causes the system to create an increasing volume of data. In light of recent developments in the energy internet, big data, and control market reorganization, this paper suggests a hierarchical cloud-fog construction for DN energy management and executive. This article delves deep into models for regular people, those who use products for professional purposes, and DSOs. In addition, the fog layer's RES output and consumption behaviour can be captured using NN technology, and decision-making concerns in the DN can be solved by GA. A DN real-time energy management system may be built using this work as a guide. Additionally, the demand-side model will be enhanced by the increased involvement of new players in the next-generation DN, such as power-selling firms and operators of virtual power plants, as a result of the further reform of the energy market. Grid management will become more innovative and efficient due to advancements in AI technology. Consequently, the future research will focus on next-generation DN, intelligent energy management, and optimum decision-making with diverse stakeholders and sophisticated AI.

References

1. Coelho, V.N., *et al.*, Multi-agent systems applied for energy systems integration: State-of-the-art applications and trends in microgrids. *Appl. Energy*, 187, 820–8325, 2017.
2. Villegas-Ch, W., Arias-Navarrete, A., Palacios-Pacheco, X., Proposal of an Architecture for the Integration of a Chatbot with Artificial Intelligence in a Smart Campus for the Improvement of Learning. *Sustainability*, 12, 4, 1500, 2020.
3. Mohapatra, S.K., *et al.*, A sustainable data-driven energy consumption assessment model for building infrastructures in resource constraint environment. *Sustain. Energy Technol. Assess.*, 53, 1026975, 2022.
4. Chouikhi, S., Esseghir, M., Merghem-Boulahia, L., Energy consumption scheduling as a fog computing service in smart grid. *IEEE Trans. Serv. Comput.*, 16, 2, 1144–1157, 2022.
5. Petroșanu, D.-M., *et al.*, A review of the recent developments in integrating machine learning models with sensor devices in the smart buildings sector

with a view to attaining enhanced sensing, energy efficiency, and optimal building management. *Energies*, 12, 24, 4745, 2019.

6. Dolce, J.L., Automated power management and control. *NASA, Washington, Beyond the Baseline 1991: Proceedings of the Space Station Evolution Symposium. Volume 2: Space Station Freedom, Part 1*, 1991.
7. Boussaada, Z., *et al.*, Multi-agent systems for the dependability and safety of microgrids. *Int. J. Interact. Des. Manuf. (IJIDeM)*, 10, 1–135, 2016.
8. Ehsan, U., *et al.*, Expanding explainability: Towards social transparency in ai systems. *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021.
9. Shobole, A.A. and Wadi, M., Multiagent systems application for the smart grid protection. *Renewable Sustainable Energy Rev.*, 149, 111352, 2021.
10. Lai, V., *et al.*, Human-ai collaboration via conditional delegation: A case study of content moderation. *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, 2022.
11. Brandwajn, V. and Albuyeh, F., Integrated security analysis and remedial action package, in: *Power Systems and Power Plant Control*, pp. 215–219, Pergamon Press, Oxford, England, and New York, USA, 1987.
12. Fazlollahtabar, H., Internet of Things-based SCADA system for configuring/reconfiguring an autonomous assembly process. *Robotica*, 40, 3, 672–689, 2022.
13. Bi, J., *et al.*, Integrated deep learning method for workload and resource prediction in cloud systems. *Neurocomputing*, 424, 35–485, 2021.
14. Sefeedpari, P., *et al.*, Modeling output energy based on fossil fuels and electricity energy consumption on dairy farms of Iran: Application of adaptive neural-fuzzy inference system technique. *Comput. Electron. Agric.*, 109, 80–855, 2014.
15. Sah, S.K. and Joshi, S.R., Scalability of efficient and dynamic workload distribution in autonomic cloud computing. *2014 international conference on issues and challenges in intelligent computing techniques (ICICT)*, IEEE, 2014.
16. Amanna, A. and Reed, J.H., Survey of cognitive radio architectures. *Proceedings of the IEEE SoutheastCon 2010 (SoutheastCon)*, pp. 292–297, 2010.
17. Niu, Y., *et al.*, Smart construction objects. *J. Comput. Civ. Eng.*, 30, 4, 04015070, 2016.
18. Mahdi, M.N., *et al.*, From 5G to 6G technology: meets energy, internet-of-things and machine learning: a survey. *Appl. Sci.*, 11, 17, 8117, 2021.
19. Li, Z., Zhou, X., Liu, W., A geometric reasoning approach to hierarchical representation for B-rep model retrieval. *Comput.-Aided Des.*, 62, 190–202, 2015.
20. Khan, M., Seo, J., Kim, D., Towards energy efficient home automation: a deep learning approach. *Sensors*, 20, 24, 7187, 2020.

21. Shishegar, S., Duchesne, S., Pelletier, G., An integrated optimization and rule-based approach for predictive real time control of urban stormwater management systems. *J. Hydrol.*, 577, 124000, 2019.
22. Ashouri, M., Davidsson, P., Spalazzese, R., Quality attributes in edge computing for the Internet of Things: A systematic mapping study. *Internet Things*, 13, 100346, 2021.
23. Lai, P., *et al.*, Cost-effective user allocation in 5G NOMA-based mobile edge computing systems. *IEEE Trans. Mob. Comput.*, 21, 12, 4263–4278, 2021.
24. Zidan, A., *et al.*, Fault detection, isolation, and service restoration in distribution systems: State-of-the-art and future trends. *IEEE Trans. Smart Grid*, 8, 5, 2170–2185, 2016.
25. Kumar, A., Rathore, P.S., Dubey, A.K., *et al.*, Correction to: LTE-NBP with holistic UWB-WBAN approach for the energy efficient biomedical application. *Multimedia Tools Appl.*, 82, 39813, 2023, https://doi.org/10.1007/s11042-023-15604-6.
26. Kim, D., *et al.*, Design and Performance Analysis for Edge Intelligence-Based F-PMIPv6 Mobility Support for Smart Manufacturing. *Wireless Commun. Mobile Comput.*, 2021, 1–145, 2021.
27. Bhargava, N., Rathore, P.S., Bhowmick, A., A Methodology for the Energy Aware Clustering and Aggregate Node Rotation with Sink Relocation in MANET. *International Conference on Emerging Trends in Expert Applications & Security*, Springer Nature Singapore, Singapore, 2023.
28. Rathore, P.S. and Sarkar, M.K., Energy Efficiency using LEACH and ECSS based optimization. *2023 11th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON)*, IEEE, 2023.
29. Rathore, P.S., Kumar, A., García-Díaz, V., A Holistic Methodology for Improved RFID Network Lifetime by Advanced Cluster Head Selection using Dragonfly Algorithm. *Int. J. Interact. Multimedia Artif. Intell.*, 6, 2, 48–55, 2020.

2

Construction and Simulation of Hybrid Neural Network and LSTM to Language Process Model

Kiran Sree Pokkuluri[1]*, Ramakrishna Kolikipogu[2], K.S. Chakradhar[3], Rama Devi P.[4] and Mamta[5]

[1]Department of Computer Science and Engineering, Shri Vishnu Engineering College for Women, Bhimavaram, India
[2]Department of Information Technology, Chaitanya Bharathi Institute of Technology (A), Hyderabad, India
[3]Department of ECE, Mohan Babu University, Tirupathi, India
[4]Department of English, KLEF, Deemed to be University, Guntur, Andhra Pradesh, India
[5]Dept. of CSE, Chandigarh University, Punjab, India

Abstract

AI and machine learning, deep learning is where it is at right now. More and more academics are paying attention to it since it is a relatively young topic that has grown rapidly in the last time. In the current years, there has been a steady improvement in the presentation of CNN models on deep learning problems; these models are among the most significant classical structures in the field. Image classification, semantic separation, target identification, and natural language processing employ convolutional neural networks to autonomously learn sample data feature representations. After examining the typical CNN model's structure to improve performance through system depth and width, this paper examines a model that improves performance even more through an attention mechanism. This study finishes with a summary and analysis of the existing special model structure. A CNN model, hybrid CNN, and LSTM that incorporate text features with language knowledge may improve text language processing. Parameter optimization, text characteristics, and language competence increase TLP model

**Corresponding author*: drkiransree@gmail.com

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (15–32) © 2025 Scrivener Publishing LLC

accuracy. The suggested model outperforms the literature reference model with experimental findings on data sets showing an accuracy of 93.0%.

Keywords**:** CNN, LSTM, AI, machine learning, deep learning

2.1 Introduction

Many industries rely on text language processing, including those dealing with public opinion on networks, crisis PR, brand marketing, and many more. Netizens' sentiments, opinions, and inclinations regarding current social procedures, policy execution, and goods and services are reflected in the massive amounts of user comment data collected on online media [1, 2]. Both academics and businesses have invested a lot of time and energy into studying network review data analysis due to its high practical utility [3, 4]. In order to assist visitors in selecting their preferred vacation spot, researchers have examined the language processing issue using data from travel blogs [5, 6]. Comment data classification into negative, positive, and neutral categories is a key field of natural language processing research [7, 8]. Language processing faces significant obstacles in a network setting due to the nonstandard nature of text expression, which includes the use of acronyms, network neologisms, spelling and grammatical flaws, and other issues [9, 10]. Traditional machine learning techniques, deep learning algorithms, and dictionary-based approaches are the mainstays of language processing issue solving [1, 11]. Yadav *et al.,* offer a convolutional neural network language processing model that combines words, parts of expression, effective dictionary entries, and other external information [12]. This approach examines emotive and linguistic information to improve network text processing accuracy [2, 13]. Training the word vectors using the word vector learning model is first. Adding part of language and affective words creates feature data to eliminate word ambiguity and convey emotion [14, 15]. Natural language processing, semantic segmentation, image classification, and target identification all use convolutional neural networks to learn feature representations from sample data on their own [16, 17]. This research first looks at a model that uses network depth and width to increase performance, and then it looks at a model that uses an attention mechanism to boost performance even more. The model is a convolutional neural network [18, 19]. The current special model structure is summarized and analyzed to conclude this research. Text feature-language knowledge-convolutional neural network (LSTM) models, hybrid CNNs, and convolutional neural networks (CNNs) could enhance text language processing [20, 21]. Accuracy of TLP models is improved by optimization of parameters, text features, and language competency. train a traditional convolutional neural network (CNN) to classify images (LENETS). When it comes to processing text, the neural network method has also been quite effective. The phrase segmentation class was the pioneer in using convolutional neural networks to text categorization using deep learning. Following the generation of sentence vectors using LSTM cyclic neural networks, discourse vectors were generated in alignment with the sentence vectors, and sentiment categorization was performed at the discourse level [22, 23].

More in-depth study on neural network topology has been carried out by researchers as it greatly influences the model's impact. For instance, the Cola Emotional Classification Neural Network Model integrates the greatest aspects of many models including attention, Hybrid CNN and LSTM in an effort to overcome the limitations of individual neural network models. An NN model for character-level classification is suggested using a combination of Hybrid CNN and LSTM, taking into account the features of brief text categorization. Word vectors are a key component of neural network models; their capacity to represent text information is a key component of the model's efficacy. Google's 2013-word vector training tool, Word2vec, uses the CBOW and SKIP DRAM word embedding models to form the backbone of deep learning models used in NLP. Scientists refined the word vector training model to meet the demands of interpreting emotional information via language.

Model for embedding sentiment words to make the language processing framework work better, SSWE is utilized to train word vectors. Emotion dictionary and remote supervised data were used to train the word vectors that carry emotional information. One may use word and training vector, part-of-speech chain, and word disambiguation to enhance the capability of word vector text representation [24]. An important part of language processing is the ability to represent and make use of textual emotional qualities. Many different emotional traits and their combinations have been the subject of academic research. This research presents a model for a multiattention CNN that uses word, part-of-speech, and word-position attention matrices to analyze a target emotion. For the purpose of sentiment analysis on Chinese microblogs, a multilayer convolutional neural network model is suggested. This model incorporates many elements of emotion information, including words, part of expression, and word position. For object-level sentiment categorization, experts recommend using a convolutional neural network, which is a model that incorporates both object attentional mechanisms and part-of-speech information. Several types of convolutional neural networks have been suggested by researchers as a result of work on deep learning theory. Figure 2.1 shows the results of a literature search for model recognition rates on classification tasks, which were then sorted in order to facilitate model quality comparisons. In lieu of testing on ImageNet, the appreciation rate on CIFAR-100 or the MNIST dataset is provided, as some models fail to do so. One of these metrics is the TOP-1 recognition rate, which measures how likely it is that the CNN model's categorization prediction would be accurate.

A CNN model's top 5 recognition rate indicates the likelihood that it will correctly identify the top five categories out of all the categories it has been trained on. Convolutional neural networks have found use in a variety of applications, including target recognition, semantic segmentation, and white language processing, as a consequence of a string of ground-breaking research findings and ongoing improvements tailored to the needs of various tasks. The paper begins by providing a synopsis of the history of convolutional neural networks based on the information provided above. It then goes on to analyze a typical model of these networks using the following criteria: stack framework, residual structure, the mechanism of attention model method, ascent performance, and finally, the makeup of the special. Lastly, the essay delves into the common uses of CNN, including target recognition, division of semantics, and white language processing. It also covers the current issues and potential future directions for development of the deep convolutional neural network.

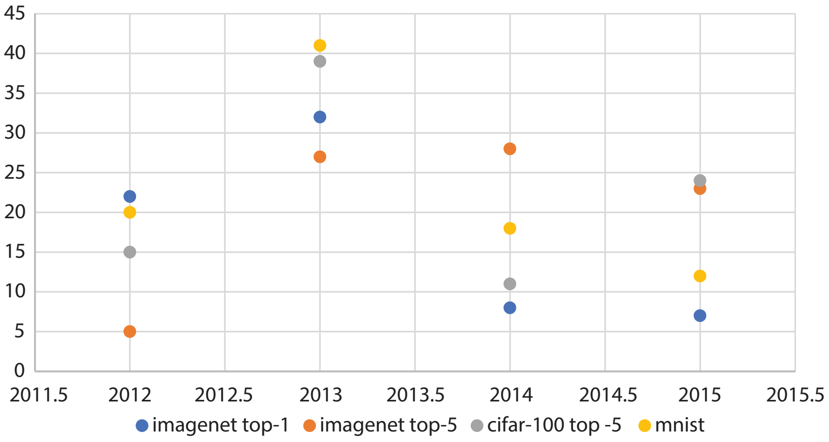


Figure 2.1 Classification task literature model recognition rates.

To perform sentiment analysis on brief texts, this research suggests a hybrid architecture of CNN and long short-term memories (LSTMs). The local information retrieved by the CNNs are fed into the RNNs. By substituting the pooling layer with a short and long-term memory model, may lessen the loss of resident info and get the input sequence's long-term reliance using a network model that consists of a convolution and cyclic layer. Although the results might be unpredictable at times, they are 6.4% better than the providing path machine approach and 1.3% better than the Hybrid LSTM and CNN with combining layer. In Section 2.2, the model and operational unit of a CNN are introduced, and the analysis approach in this study is applied to this model. Section 2.3 details the construction and simulation of a CNN based language processing model, and data analysis proves the method's superiority. Section 2.4 concludes the study and outlines the next steps.

2.2 Convolutional Neural Network

2.2.1 Operation Unit and Basic CNN

Pooling, nonlinear units, convolutional layers, and full connection layers make up the meat and potatoes of a convolutional neural network (CNN). It is common practice for CNN to include a pooling layer that rotates with the convolutional layer, followed by one or more fully associated layers for output. Occasionally, in order to make CNNs even more efficient, we use batch normalization and other procedures, and we use the global flat pooling layer instead of the whole connection layer. The CNNs construction is shown in Figure 2.2:

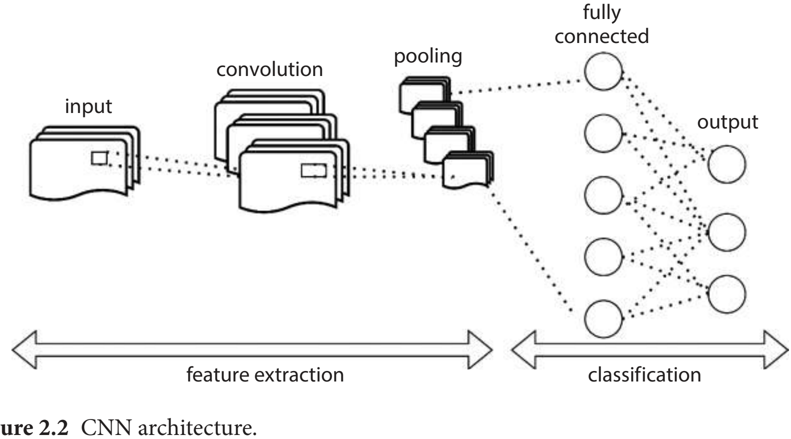


Figure 2.2 CNN architecture.

The NLP system's use of CNN has yielded excellent outcomes. A CNN has several complicated nonlinear activation functions, like RELU or Tank, and its convolutional layer resembles a sliding window on a matrix. Each neuron's input in a classical feedforward neural network is linked to every output in the layer below it, known as the full connection layer. Pooling is when CNN retrieve picture information based on the convolution kernel size. When it comes to picture cataloguing, CNN can learn to recognize limits in the original pixels in the initial layer. Then, in the second layer, they can use these edges to identify basic shapes. From there, they can use these forms to identify more complex characteristics, like the shapes of faces. Classifiers that make advantage of these sophisticated characteristics are given this layer. The majority of natural language processing jobs do not use picture pixels as input but rather matrix representations of words and texts. A vector of one word is represented by each row. This vector is often referred to as a vector. The word vector's dimension in natural language processing is a convolution kernel; in other words, the filter's width is equal to the input matrix's width. While the exact dimensions could vary, you should expect a sliding window that displays no less than two to five words at a time. To apply the aforementioned data to text sentiment analysis, RNN makes advantage of sequence information. For many natural language processing (NLP) applications, such sentence-level word prediction, the standard neural network's method in which inputs are considered independently is inadequate. Here, researchers knew that knowing the prior work was crucial for predicting the following word in context, therefore they developed RNN. On the natural language processing job, RNN performed well. RNNs may store data in seemingly endless sequences. The RNN network architecture is seen in Figure 2.3.

Time series modeling has made extensive use of recurrent neural networks (RNNs), a kind of DNN. In order to get a compressed and low-dimensional semantic representation, RNNs are used for sentence embedding. This is achieved by repeating analysing each term in an expression and plotting it to a low-dimensional vector. The following is the result that may be obtained using a basic RNN for calculation:

$$o_{t=}\sum\iint n\sqrt{f}(W_O H_T)+\frac{H}{H}+\frac{W}{W}$$

$$H_T=\frac{H}{H}+\frac{W}{W}+\sqrt{N(W_H H_T+W_X X_T}$$

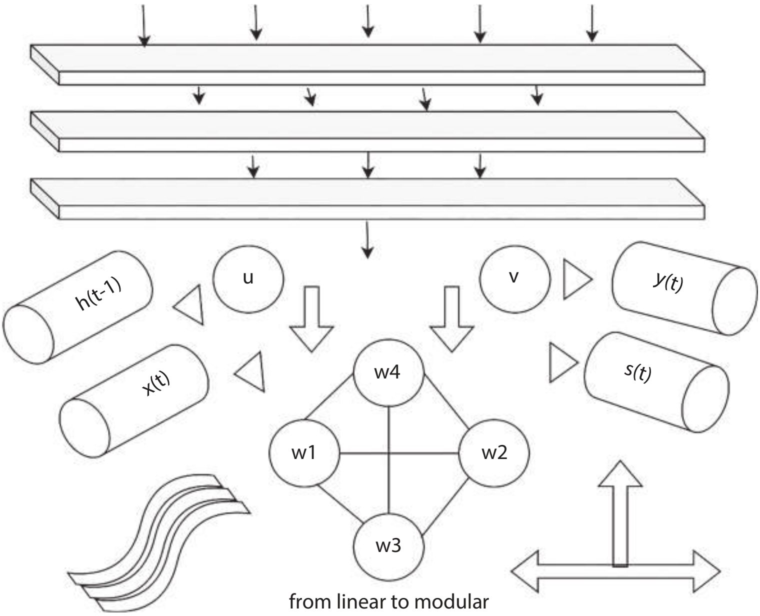


Figure 2.3 The RNN network architecture.

in where W0, Wh, and Wx are the neural network's parameter matrices; Xt stands for the neural network's input; and Ht−1 denotes the neural network's state at a single instant. Both the current input and the state at time t – 1 are connected to the state at time t, according to Equation (2). By studying word connections, RNN increases the text's temporal complexity while preserving the semantics of all prior manuscripts in a fixed-size secreted layer. In order to detention the semantics of lengthy text, RNN captures the high-level, relevant statistics. In RNN, the most recent words are given greater weight than the ones that came before them since it is a biased model. Using this to record details about the full document could be wasteful. The LSTM model is therefore created to circumvent RNN's shortcomings. Finding a sentence's long-term reliance is the focus of this article, which employs LSTM. Figure 2.4 displays the architecture of the LSTM network.

2.2.2 Standard CNN Model

The user's text is a single period. This chapter presents a comparative analysis of the four network models, focusing on three key aspects model mechanism, benefits and drawbacks, and recommendations for implementation.

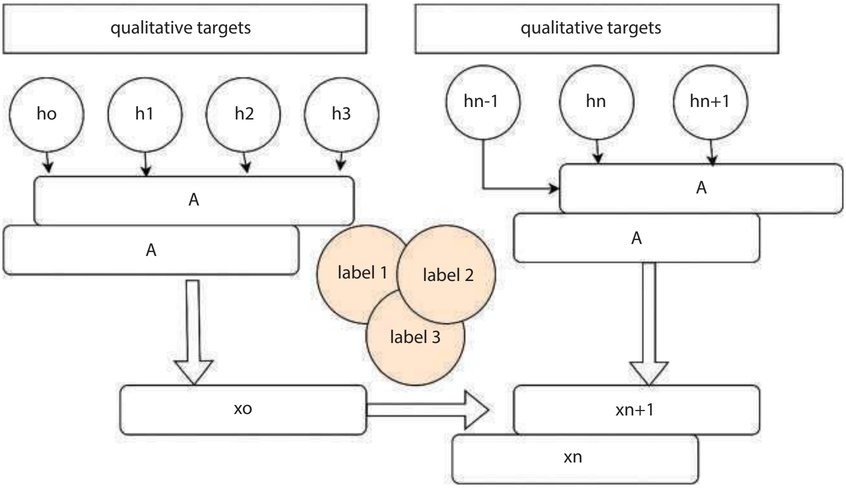


Figure 2.4 A structure of the LSTM network.

The research reveals that the selection and optimization of network models should be based on the specific features of the network in order to effectively use them, given their diverse methods.

2.2.3 Model of Stacking Structure

When additional topological structures are absent from a network model, the one that results from just stacking network layers is called the stacking structure model. Models for early neural networks, such as VG GNet, MSRANET, and ZF Net, continue used. Net networks use the RELU activation purpose as a nonlinear unit based on Le net and add Dropout and LRN to stop network overfitting. The Great Normalization of Alex stacks them to form network models, such as Alex Net, which uses Le net as its foundation and CNN as its basic computing unit. A freight structure model was divided and trained cooperatively on two GPUs since early-stage GPU video memory was limited. The creation of a hardware platform will allow for the use of a single GPU stage for network training, eliminating the need to partition the model structure.

2.2.4 Structure Model for Networks Within Networks

A network model called the Network in-network structure model is created by linking the outcomes of the operations of various branches of neural networks. Since it employs a minimal amount of limits to accomplish the Alex Net effect, the NIN model introduced in NIN has a significant influence. In most cases, input characteristics used for classification problems are not linear. Compared to stack structures, NIN networks are better at abstracting the characteristics of individual local blocks since they produce micro networks in each convolutional layer. Adding a micro network to every convolutional layer deepens the network, widens it, and improves its capacity for specific expression.

2.2.5 Model for the Attention Mechanism

The output of a residual structure model is given by the linear principle of superposition of the input and its nonlinear change. This model structure incorporates a short circuit mechanism. As the number of layers in the CNN model increases from 8 in Alex Net to 19 in GNet and 22 in Google Net, the model's performance improves and the model becomes deeper, allowing for better nonlinear expression and a better fit of complex features. Direct usage of the identity mapping is not possible due to inconsistencies between the quantity of channels in the residual element and the output. Nevertheless, the information flow between them will be hindered if additional input channels are created using the 1 × 1 convolution layer. Dares Net stays put in the construction of the residual unit. In the residual route, the channel is filled with zero environments after the input feature channel is directly added to the output channel.

2.2.6 Model for Free-Motion Learning

Using free-motion learning, the attention mechanism model learns which characteristics need attention while blocking out the structure of elements that aren't relevant. A lot of effort goes into improving the spatial dimension performance of the previously introduced model, but SENET, a channel attention model, can independently determine the importance of each channel feature, give more weight to channels with useful features, and disable channels with useless ones. Its improved feature expression capabilities and lightweight design make it an ideal candidate for incorporation into any existing CNN model architecture, and its combination of channel and spatial attention mechanisms makes it superior to models using a single attention mechanism. A neural network may overcome information overload by training itself to concentrate on a subset of input features with the help of an attention mechanism.

2.3 Construction and Simulation of Hybrid CNN-LSTM Language Processing Models

2.3.1 Language Dispensation Model Construction: Hybrid CNN/LSTM

The model is built around a foundation of recursive and convolutional neural networks. The model's design loads a convolutional neural network with the word vector as input so that the network may learn high-dimensional feature extraction. Next, information passes through a language model that incorporates a classifier layer after receiving its output from a cyclic neural network with short- and long-term memory. Figure 2.5 of this article depicts an architecture that combines a CNN with a LSTM:

- Word vector: the first network layer assigns semantic information to each term in the emotive text, transforming it into a term direction. Word strings are fed into the model. For this experiment, the maximum phrase length at 100. The value zero is used to fill in any spaces when the sentence length falls short of the limit. A matrix of size 100 by 128 may be used to represent each emotional statement. Data

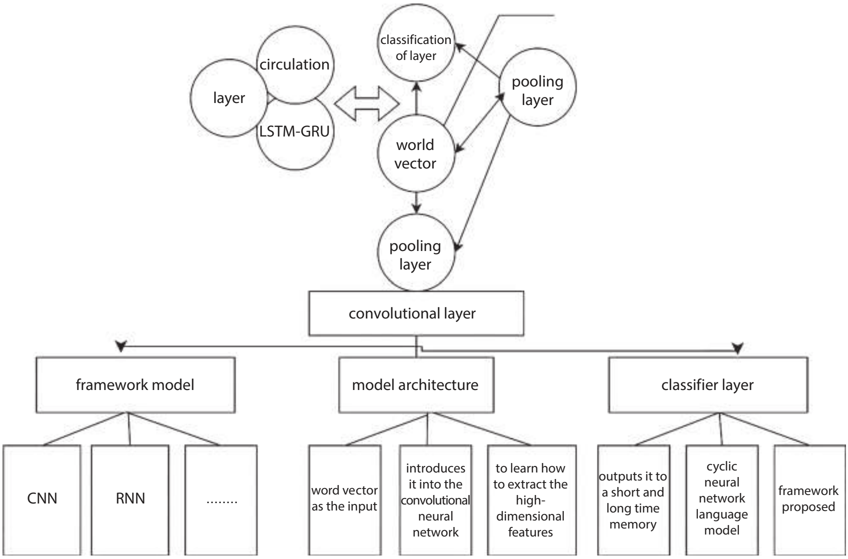


Figure 2.5 The article suggests a framework that combines LSTM with Hybrid CNN.

about the meaning and location of words are stored in the matrix.

- Word vectors, which are created by splicing basic word vectors, are used in the convolution layer of the network's first layer structure to represent each phrase as a 100^128 matrix. The extraction of high-dimensional characteristics from text is accomplished using a convolution kernel. Image convolution uses a convolution kernel size that is twice the word vector length, while text convolution allows the user to arbitrarily choose the size of the kernel. In this test, the convolution kernel's values are 4 ∗ 128, 3 ∗ 128, and 5 ∗ 128 each. The following convolved phrases are produced by this process: 98 ∗ 1, 97 ∗ 1, and 96 ∗ 1. To make a 98 ∗ 3 matrix, these features are merged and then filled with zeros. This matrix is then input into the model of short- and long-term memory.

2.3.2 An LSTM and CNN-Based Hybrid Model for Language Processing is Being Simulated

Recent developments in convolutional neural networks (CNNs) have revolutionized image classification. CNNs incorporate a new functional unit structure into their feature extraction processes, allowing them to tackle a wide range of learning tasks with greater depth and accuracy than before. As a result, CNNs are quickly replacing more antiquated methods. Target identification, semantic segmentation, white language processing, and related areas have recently made it a research focus. The article details the process of structure a model for language processing. Evaluative indices such as Precision, F-score, and Recall are computed using the given formulae. In this case, the variables HW, Hb, and Fn represent the number of correctly considered results, number of errors, and total amount of erroneously confidential results in the dataset for this kind of sample, respectively. As the harmonic value that takes accuracy and recall rate into full account, the F score shows the whole effect of a model:

$$Precision = \varnothing(\omega + h + b)\frac{HW - BW}{HW + BW}$$

$$recall = \varnothing(w + h + b)\frac{hw}{hw + bw}$$

$$F - score = \sqrt{\pi(whht - 1 + wxxt) + \frac{2 \times precistion \times recall}{precistion + recall}}$$

To improve the text's semantic and emotional information, the Hybrid CNN and LSTM model takes text characteristics as input data and adds linguistic knowledge, such as part of speech and emotive terms. By manipulating the quantity and mix of input features, this experiment evaluates the model's performance in sentiment classification. generated POSV, WV, SWV, WV + POSV, WV + SWV, WV + POSV, and WV + POSV + SWV by combining the word vector, the part-of-speech vector, and the emotional word vector. Figure 2.6 shows the untried results of the feature fusion model using the Hybrid CNN and LSTM classification layer. The optimal outcome for emotional classification is attained by integrating word features, part-of-speech characteristics, and affective word features, which is POSV + WV + SWV. Multiple permutations using one or two parts are shown in Figure 2.6. The three feature combinations outperformed the others with positive category F values of 92.8%, negative category F values of 93.2%, and macro average F values of 93.0%. It follows that the classification impact might be improved by using numerous structures in the fusion model of the organization layer. The results show that external language information, including part-of-speech and emotion word structures from text characteristics, may significantly increase the performance of the CNN sentiment organization model. Figure 2.7 shows the Hybrid CNN

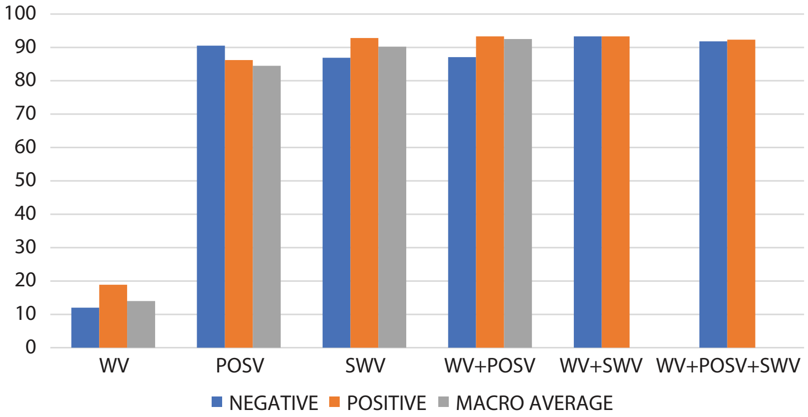


Figure 2.6 Hybrid CNN-LSTM classification layer fusion model feature fusion findings.

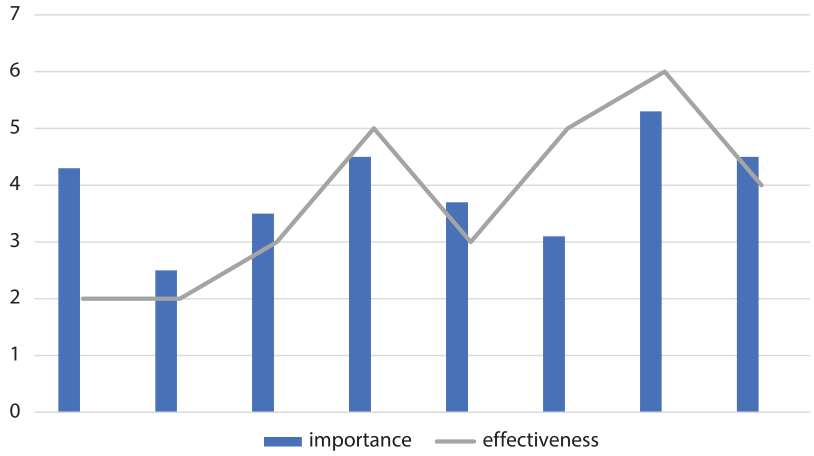


Figure 2.7 Hybrid CNN/LSTM fitting curve.

and LSTM fitting curve. With such a little space between phases and an often flat slope, the approach seems to be quite stable.

Backpropagation is used to compute the gradient, and stochastic incline parentage (CSGD) is used to train the system. It is possible to decrease the number of convolutional layers needed to detention dependence over time by substituting a loop layer for a pooling layer in the model. The recursion and layer of convolution are combined in this model for that reason. Reducing the number of layers that use convolution and pooling in the network is one of the design goals in order to keep fine-grained local information as intact as possible. Accordingly, the suggested model necessitates a convolutional layer activation function including RELUs, a hidden state measurement of d = 128 and an LSTM-based recursive layer. There is a range of five to twenty training cycles for the two datasets. The model is evaluated in comparison to several deep learning and conventional approaches, including techniques that use word embedding and convolutional architecture. Following consideration of the settings for regulation, knowledge rate, and refusal rate, judgement features are extracted using the convolutional layer. proposed method in a number of domains. All systems are go with this model, which has a convolutional and cyclic layer learning rate of 0.01 and a loss rate of 0.5. In this article, we demonstrate that Dropout is an efficient method for regularizing deep neural networks. We made it more consistent by sandwiching the Dropout component between the Hybrid CNN and LSTM layers, limiting the weight vector to 12 norms, and stopping the hidden units from responding to each other. The accuracy of organization algorithms is typically higher than that of machine learning algorithms; the margin of error between the two approaches is only 1.7% at most; and the model combining BOW and CNN is surpassed by the support vector machine model that uses hybrid convolutional neural networks and long short-term memory. Possible causes include the fitting problem, the BOW + CNN model's lack of optimal parameters during training, etc. This study proposes the best course of action, as seen in Figure 2.8. Over time, the classification algorithm accuracy of the paper's proposed model beats that of the CNN+ BOW strategy. The resulting model is also more stereoscopic, intuitive, and has a reduced error rate. Among the components. Uyghur does not contain natural word segmentation markings as Chinese does, which makes feature extraction more difficult. Spaces are used as separation indicators among words in Uyghur. The experimental feature extraction techniques employed were Unigram and Bigram, respectively. categorization findings showed that Bigram feature extraction had a far stronger impact on Uyghur text sentiment categorization than Unigram. Uyghur is another language where the approach suggested in this study performs well. The support vector machine technique is outperformed by 6.4% and the LSTM-CNN with a pooling layer by 1.3%. The theoretical and experimental justifications for the suggested approach are presented. Due to the local nature of convolution and pooling, CNNs need several layers of convolution in order to detention long-term relationships, despite CNNs' learning ability to extract locally invariant high-dimensional features.

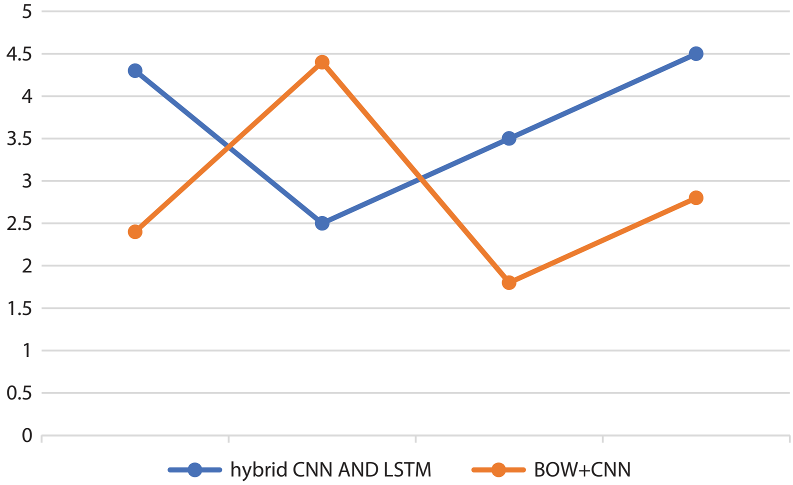


Figure 2.8 Hybrid CNN, LSTM, and BOW + CNN display models and classification accuracy comparison.

The longer the input sequence, the worse this issue becomes. A deep network with several convolutional layers is necessary in the end. In order to get around this issue and decrease the amount of design parameters, this paper suggests a novel framework that can collect word information in phrases. Combining convolutional and cyclic neural networks is the foundation of this framework's operation, which is based on the input word vector. It is possible to keep the sorting information even with a single recursive layer. Hence, a loop layer, as opposed to a pooling layer, is more efficient at capturing long-term relationships and assumes less damage of detail in local information. The way that has been suggested CNNs with LSTM yields better results than competing approaches on both datasets, allowing for competitive classification accuracy. A smaller design may reach the same degree of classification performance, according to experimental data. This method has the potential to improve future research in the fields of data retrieval and machine translation.

2.4 Conclusion

The performance and scalability of convolutional neural networks have been enhanced via research. This research examines a well-performing hybrid convolutional neural network model that uses CNNs and LSTMs. In terms of picture identification and classification, the standard model of convolutional neural networks has achieved outstanding results. Among its primary components are technologies that expand the breadth and depth of a network's structure and combine the attention mechanisms of the station and spatial domains. Artificial disturbances, such as adding certain sounds to the original picture, may easily lead the neural network model to misclassify the image. We need to figure out a way to fix this and make the model better at generalizing. In addition, the training cost rises steadily when the neural network's depth and breadth are increased. The training pace of the model may be significantly enhanced by including past knowledge about particular challenges into its development. Not to mention that structural studies of convolutional neural networks still have a lot of room to grow. Because convolutional neural networks rely on trial and experience to establish its super parameters, and because increasing model performance requires more realistic network structure design, parameter analysis is a difficulty for these networks. Despite the current level of interest in convolutional neural networks, a complete mathematical description and demonstration of these networks is still lacking. Continuing the relevant theoretical research is crucial for improving existing convolutional neural networks and creating new ones that overcome the shortcomings of the existing network architecture. Additionally, this paper's special models provide additional suggestions for improving the standard convolutional neural network model:

- A lightweight architecture is required for typical models of convolutional neural networks. Scholars have long ignored the particular deployment platform of a convolutional neural network's model application in Favor of studying the network's algorithm architecture. The primary focus of model structure design research is the development of hardware-friendly model structures, which in turn will aid in the enhancement of model performance.
- Because it is more in line with how humans think, unsupervised learning is increasingly popular in the white domain and helps to reinforce the structure of poorly supervised or unsupervised convolutional models. While both weakly supervised learning and unsupervised learning have achieved some success with picture identification, supervised learning remains much superior when compared to semi-supervised learning and unsupervised learning.
- A multi-input convolutional neural network model is built using multi-information input. This model uses the source data to express features implicitly, leading to improved recognition with lower training cost. When the network topology utilized for recognition is shared, recognition efficiency may be further improved. Also, worth looking at is the efficient redundant feature generation approach, which can produce more feature graphs with fewer model parameters. This might lead to a more effective feature production process overall.
- The long short-term memory model and hybrid convolutional neural network are fundamental to this field of research. Use cases include standard convolutional neural network models, intelligent healthcare, white-knuckle driving, wearable technology, mobile payments, and virtual reality. When it comes to the future of science, technology, and the AI sector, the creation of models for deep neural networks is crucial. More research may be done to find solutions to the current difficulties and to find practical uses for them.

References

1. Kaliyar, R.K., Fake news detection using a deep neural network. *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, IEEE, 2018.
2. Tran, L., *et al.*, A Survey on Password Guessing, arXiv preprint arXiv:2212.08796, 2022.
3. Fatma, D., Chaya, K., Mohammed, S., Class, A., *Sentiment Analysis for Product Recommendation System using Machine Learning Algorithms*, 2021.
4. Bilal, S., Fake news detection using machine learning techniques. *AIP Conf. Proc.*, 2624, 1, 050076, AIP Publishing, 2023.
5. Bian, Q., An inertial sensor-based motion capture pipeline for movement analysis, Diss., University of Birmingham, 2023.
6. Li, M., *et al.*, What can be learned from the historical trend of crude oil prices? An ensemble approach for crude oil price forecasting. *Energy Econ.*, 123, 106736, 2023.
7. Santhadevi, D. and Janet, B., HSDL-based intelligent threat detection framework for IoT network. *J. Intell. Fuzzy Syst.*, 44, 3, Preprint, 1–16, 2023.
8. Dutta, S. and Bandyopadhyay, S.K., A crucial Psychological Analysis of Mental Anxieties of Job-seekers and Working Classes in Society. *ScienceOpen Preprints*, 2020.
9. Dinler, O.B., Şahin, C.B., Abualigah, L., Comparison of Performance of Phishing Web Sites with Different DeepLearning4J Models. *Avrupa Bilim Teknoloji Dergisi*, 28, 425–431, 2021.
10. Vigil, A., *et al.*, A combined neural network mechanism for categorizing the normal and cancer cells. *J. Intell. Fuzzy Syst.*, 45, 5, 7191–7203, Preprint, 1–13, 2023.
11. Thaweephol, K. and Wiwatwattana, N., Long short-term memory deep neural network model for PM2. 5 forecasting in the Bangkok urban area. *2019 17th International Conference on ICT and Knowledge Engineering (ICT&KE)*, IEEE, 2019.
12. Yadav, D.C. and Pal, S., Analysis of heart disease using parallel and sequential ensemble methods with feature selection techniques: heart disease prediction. *Int. J. Big Data Anal. Healthcare (IJBDAH)*, 6, 1, 40–56, 2021.
13. Lin, T., Guo, T., Aberer, K., Hybrid neural networks for learning the trend in time series. *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, 2017.
14. Wermter, S. and Sun, R., *An Overview of Hybrid Neural Systems*, Springer, Berlin Heidelberg, 2000.
15. Yaghini, M., Khoshraftar, M.M., Fallahi, M., A hybrid algorithm for artificial neural network training. *Eng. Appl. Artif. Intell.*, 26, 1, 293–301, 2013.
16. Perrone, M.P. and Cooper, L.N., When networks disagree: Ensemble methods for hybrid neural networks, in: *How We Learn; How We Remember: Toward*

An Understanding Of Brain And Neural Systems: Selected Papers of Leon N Cooper, pp. 342–358, 1995.

17. Hernández, G., *et al.*, Hybrid neural networks for big data classification. *Neurocomputing*, 390, 327–3405, 2020.
18. Zhao, R., *et al.*, A framework for the general design and computation of hybrid neural networks. *Nat. Commun.*, 13, 1, 3427, 2022.
19. Osowski, S. and Linh, T.H., ECG beat recognition using fuzzy hybrid neural network. *IEEE Trans. Biomed. Eng.*, 48, 11, 1265–1271, 2001.
20. Tsai, C.-F. and Lu, Y.-H., Customer churn prediction by hybrid neural networks. *Expert Syst. Appl.*, 36, 10, 12547–12553, 2009.
21. Dubey, A.K., Kumar, A., García-Díaz, V., Sharma, A.K. and Kanhaiya, K., Study and analysis of SARIMA and LSTM in forecasting time series data. *Sustain. Energy Technol. Assess.,* 47, 101474, 2021.
22. Kristjanpoller, W., Fadic, A., Minutolo, M.C., Volatility forecast using hybrid neural network models. *Expert Syst. Appl.*, 41, 5, 2437–2442, 2014.
23. Swarna, S.R., Boyapati, S., Kumar, A., Study of Game Theory Mechanism for Effective Sentimental Analysis using Natural Language Processing. *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC),* Palladam, India, pp. 1013–1017, 2020, doi: 10.1109/I-SMAC49090.2020.9243555.
24. Kumar, A., Rathore, P.S., Dubey, A.K., *et al.*, Correction to: LTE-NBP with holistic UWB-WBAN approach for the energy efficient biomedical application. *Multimed. Tools Appl.*, 82, 39813, 2023, https://doi.org/10.1007/s11042-023-15604-6.

3

An Approach to Ensure the Safety of Industry 4.0 Mobile Robots

P. Balaji Srikaanth[1]*, Rajeshwari M. Hegde[2], Ramachandra V. Ballary[3], Poornachandran R.[4], R. Senthamil Selvan[5] and Amandeep Kaur[6]

[1]Department of Networking and Communications, SRM Institute of Science and Technology, Kattankulathur Campus, Chennai, India
[2]Department of Electronics and Telecommunication Engineering, BMS College of Engineering, Bengaluru, India
[3]Department of Computer Science and Engineering, AGMR College of Engineering And Technology, HUBLI, India
[4]V.S.B Engineering College, Karur V.S.B College of Engineering Technical Campus, Coimbatore, India
[5]Department of Electronics and Communication Engineering, Annamacharya Institute of Technology and Sciences, Tirupati, India
[6]Dept. of CSE, Chandigarh University, Punjab, India

Abstract

This research suggests that human–robot cooperation in industries might boost efficiency and productivity. Nevertheless, this innovation goes against the grain of conventional safety protocols by placing human and robot workplaces in distinct places. Safety regulations for industrial robots have evolved over the last two decades. An emerging field of study focuses on avoiding the negative impacts of robots and mitigating their hazards and drawbacks. This study presents an examination of well-known safety systems that are designed and used in engineering robotic environments. These mechanisms help to ensure that people and robots can work together safely. In addition, a review and new ideas have been introduced under the present rule. This article presents a multidisciplinary approach to protecting humans with smartphone robots in enterprise 4.0, utilizing CPS and SDN with a GMM-GM ML system. The approaches cover a range of topics, including injury estimation and evaluation, impact detection systems, software and mechanical tools to reduce human-robot impacts, and collision prevention strategies.

**Corresponding author*: Balajis7@srmist.edu.in

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (33–48) © 2025 Scrivener Publishing LLC

Keywords: Human-robot cooperation, industrial robots, GMM-GM machine, robotic environments, SDN and CPS

3.1 Introduction

A more efficient, autonomous, and user-friendly industrial revolution has emerged in response to the ever-increasing number of new issues [1, 2]. Industry 4.0, a novel idea in the business world, describes the present tendencies in data sharing and automation in production as they pertain to the development of a "smart factory." Internet of Things (IoT) and software-defined networking (SDN) advancements in processing power form the backbone of Industry 4.0. Figure 3.1 shows Industry 4.0's nine weak pillars. The CPU satisfies the software and hardware requirements by remotely associating the robotic operations to connect the machines, it is necessary to provide intelligence services to the Mobile Robot [3, 4]. Industry 4.0's current application is shown in Figure 3.1.

Here, an integrated control-based direction-finding solution is proposed that links the machines and keeps an eye on the robotics' navigation. Automatic mobile robot navigation entails four stages: planning, positioning, design, and execution [5, 6]. Hence, it is advisable to be familiar with the environment map before beginning the design step [7]. The planning and execution of robots will not be affected by this difficulty since there are many other ways to work with robots. In conclusion, the research paper's architecture combines elements of numerous prevailing systems; It adheres

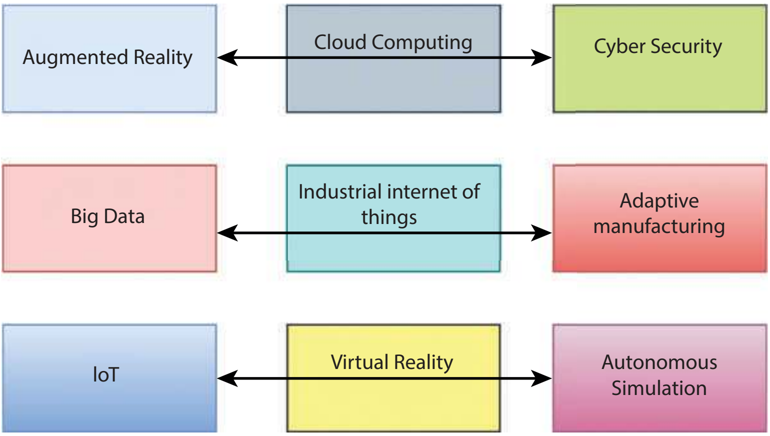


Figure 3.1 Industry 4.0's nine elements.

to the frequently recognized categorization of navigation architecture in the literature and is primarily built with subtle basics for managing unexpected changes in the environment and problems [8, 9]. The absence of concurrent encryption that is susceptible to cracks and security breaches, as well as the nonexistence of an established substructure that can accommodate Industry 4.0, are among the several concerns addressed in the framework for developing Industry 4.0 [1, 10]. It will need infrastructure upgrades over the next several years to close this security gap [11].

This area of study encompasses a range of disciplinary methods, including damage assessment and evaluation in the case of a human-robot collision, impact detection systems, mechanical and software tools for minimizing such impacts, and tactics for preventing and mitigating such collisions. Due to the industry's fast evolution and the primary goal of improving infrastructure and resources, the Internet of Things (IoT) and Industry 4.0 blueprint are essential. Both methods restore the reliability of industrial wireless sensor networks in the event of interference, according to research.

3.2 Methodology

Production functions may be expanded and the controller can be modified. To increase the workshop's assembly capacity, modular manufacturing units suggested a modular-adaptive autonomous robotic island [12, 13]. Cognitive robots were suggested to be physically integrated into production through cyber vertical integration and coordination with the manufacturing system, in addition to improved flexible manufacturing skills that are controlled and organized by an integrated management framework [14, 15]. AI-powered manufacturing machines are capable of autonomously detecting information ambiguity, modifying production schedules, and adjusting production behavior to tackle complicated manufacturing problems. Thus, it is crucial to have intelligent robot units. Here are a few pointers on how to construct manufacturing units that use modular components [16]. The interaction heterogeneity should also be taken into account. It is critical to have an acceptable or ideal combinatorial system as the purposes of several modular manufacturing components for a certain product could be redundant. Every manufacturing facility may meet product specifications and improve intelligent plant efficiency via self-organization intelligent manufacturing, an impasse can occur if small amounts of different goods are delivered into the fabrication system in an unorganized manner. One of the most discussed areas of study in Industry

4.0 is how to keep mobile robots safe while allowing for flexible production processes. As a result, the manufacturing unit would be able to respond swiftly to variations in operating circumstances. Utilize available software components to enhance robot functionality and adapt to diverse work situations. In addition, a state-of-the-art controller design for robots has been shown, which may be used by future robot installations [17]. Because standardization and universality may facilitate reconfiguration operations, component models of control unit elements have been established. In doing so, the manufacturing unit would be able to respond swiftly to variations in operational parameters. Modify the robot's performance and adapt it to new environments by using the software components for disposal. Revolutionary Instructions to enhance human security with Mobile Robots in Industry 4.0 can use a new distributed multiagent controller system compatible with an intelligent and reconfigurable numerical controller for the next generation of robots [18].

The following are the precise procedures for execution:

Procedures 1: Open the list to set the initial point.
Procedures 2: The search will fail if the open list is null.
Procedures 3: The least-value node Present node in the Open List.
Procedures 4: Search terminates if the node is the target point
Procedures 5: Lengthen the branch Circular and straight paths should be followed.
Procedures 6: If there is a difficulty, go to enter 5.
Procedures 7: Modify the motion if the precise kind of motion cannot be determined.
Procedures 8: Go to the next node in the open list.

3.2.1 Mobile Robots IN Smart Enterprises

Intelligent factories use mobile robot (MR) systems that can operate in tandem with human robots to sensors placed in designated safe zones. Mobile Robots (MRs) provide several benefits over more conventional types of industrial robots [19, 20]. With these robots, humans have a secure space to retreat to, and they can even make way for more conventional robots. When humans go too close to a robot, proximity sensors slow it down, limiting forces to keep people and the environment safe, and managing human intent and actions appropriately. The prevailing belief is that Mobile Robots (MR) pose no threat to humans and that regulated acceleration and force may shield humans from damage. Controlled separation, guided hand movements, and controlled stop-office procedures to ensure safety.

Vision and CAD robot planning and control allow for foregoing manual programming of robots over extended periods. Principles of dynamic motion for parameterization Learning may be facilitated by mobile robot (MR) mobility, which highlights the need for manual programming [21–23]. In cases when, say, production shifts need rearranging the plant's interior, this will furthermore provide more adaptability.

3.2.2 Cyber-Physical Systems

In Industry 4.0, cyber-physical systems (CPS) play a key role. Because of cyber-physical systems (CPS), the virtual and physical worlds are inseparable, since they are linked to one another via the internet of things. Everything from computers and equipment to apps, software, routines, and analytical operations may be found in cyber-spaces. The communication network connects these cyber-spaces to the actual world. In the context of software-actuator and sensor linkages, the CPS is crucial [24]. Maximizing system efficiency is central to CPS, which aims to boost output rates. Developing smart systems is given a lot of significance, and methods have increased, with the advent of the idea of Industry 4.0. Failures in nonoptimized procedures may be handled by smart and varied designs. In several fields, including medicine, farming, and banking, CPS is helping to make IoT-based production systems more adaptable. Construct the core idea of

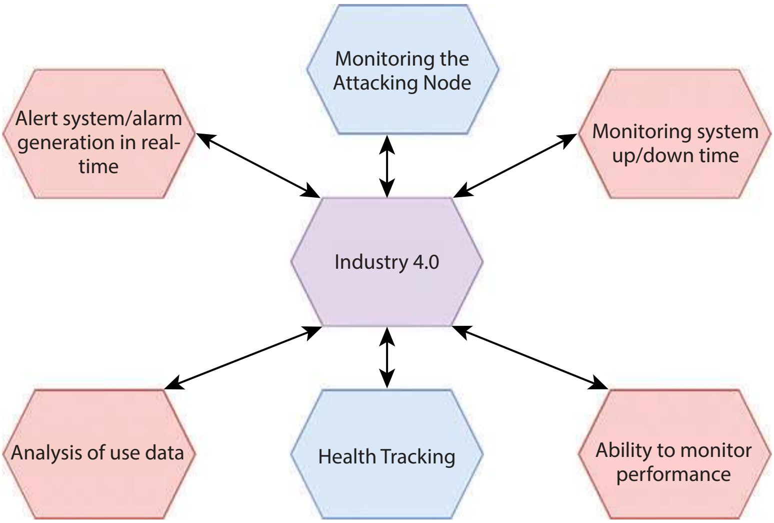

Monitoring the Attacking Node
Alert system/alarm generation in real-time
Monitoring system up/down time
Industry 4.0
Analysis of use data
Health Tracking
Ability to monitor performance

Figure 3.2 Digital–physical product development integration.

Industry 4.0 with the assistance of the updated human-computer interaction system. In Figure 3.2, one can see both digital and physical production are integrated throughout product development.

3.2.3 Internet of Robotic Things

A more accurate and consistent representation of the robot environment may be achieved by integrating IoRT information from many sources. Overall, it is believed that only approximate environmental models should be used to incorporate IoRT data. Data fusion plays a crucial role in monitoring interactions, for instance, when choosing impedance values or figuring out which dangerous Mobile Robot (MR) control points to leave for the relevant person. Close-by Internet of Things devices may include pan-tilt-zoom, stereo, or deep-camera-base cameras, as well as audio/video feedback systems. Traditional robotic systems may benefit from force limiting and power adjustment features, such as high speed, large payload, and substantial. A certified safety sensor, stereo cameras, 3D Lidar cameras, and regular old cameras are all part of the remote interaction sensor set. Integrated force/torque or grip proximity sensors are another option for audio/video recovery on HRC sensors.

3.2.4 Using SDN to Improve Cyber-Physical System Security for Mobile Robotics Industry 4.0

The production, results, and business simulation processes are all profoundly affected by the many changes brought forth by the conceptual Industry 4.0. As a result, there will be increased adaptability, productivity, flexibility, manufacturing speed, and product quality. With mass adaptation, even very small numbers may be produced since equipment can be easily adjusted to meet client demands and additions. Fast prototyping of new goods or services without extensive retooling or the creation of brand-new manufacturing lines promotes innovation. Industry 4.0 technologies allow it to reduce inventories, allowing it to make one product with several versions. As a bonus, the product can now be produced more quickly because of digital methods, virtual manufacturing models, and less time between designs when it comes to delivery time and time to market, data-driven supply chains in India can reduce manufacturing times by 50% and time to market by 50% again. A decrease in costs and an increase in competitiveness are both facilitated by improved quality. Eliminating flaws would save money that would otherwise go into destroying or reprocessing faulty items. It is also possible to boost the efficiency of the many impacts of Industry 4.0. Improved machine uptime and less downtime are outcomes of predictive maintenance programs that use advanced analytics. Several businesses are considering setting up shop in homes where autonomous robots are made in complete darkness. An example of the way robotics is influencing the economy is via the development of self-driving mobile robots. These machines can traverse environments that are crowded with obstacles to their sensors and feedback systems. Reduced wear in different industrial components and increased production productivity are the goals. The Reformist Framework for Improving Human Security with Mobile Robotics in Industry 4 offers decreased floor traffic, customizable flooring, dependability, and self-regulation. To operate the autonomous mobile robots, the following minimum requirements must be met:

1) Using a method that boosts efficiency and output without human intervention
2) Automation of material handling
3) Safety and security are automatically enhanced, minimizing stress and danger.
4) Creatively managing repetitious tasks
5) Minimizing industrial traffic under challenging circumstances

Improving human security using Mobile Robots in Industry 4.0 factories to completely link production is a critical vision of production control, which is a key component of Industry 4.0. Unfortunately, this capacity is still in its infancy due to the lack of a single framework that connects factory systems. Once implemented, it would enable the optimization and

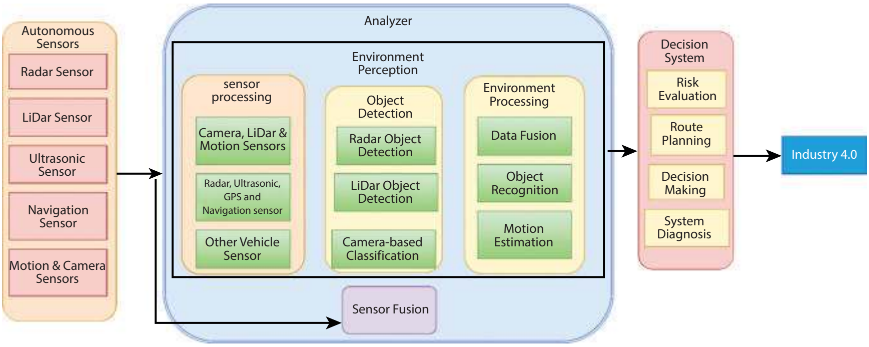


Figure 3.3 Security data analysis for smart robots in Industry 4.0 to protect humans.

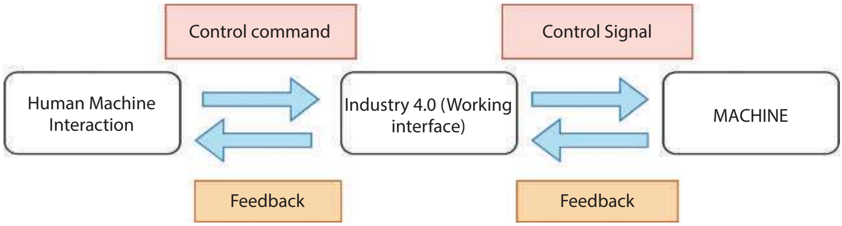


Figure 3.4 Procedure for human–machine interaction.

testing of whole production lines across several locations. It is a long-term objective for many businesses, and getting there will require better data handling security, an upgraded information infrastructure, and, as one firm put it, "automating automation" through the use of mobile robots in Industry 4.0. Figure 3.3 depicts the autonomous sensors with a decision-making system in action, and it demonstrates the analysis of assault data to strengthen human security using Mobile Robots in Industry 4.0.

Humans and robots are unlikely to benefit from efficient human-robot interaction in the workplace as shown in Figure 3.4:

3.3 Proposed Real-Time Attack of Data Classification

They suggested an EM-based Gaussian Mixture Model (GMM). By constantly recalculating its route, the Internet of Things (IoT) can avoid obstacles and get where it's going. Motion planning is the process of predicting and avoiding obstacles when a robot moves from one location to another. The three axes of motion (distance, speed, and acceleration) of the manipulator must be perpendicular to one another for path and trajectory planning to work. Information about the time axis is carried by the trajectory as shown in Figures 3.5 and 3.6. Trajectory planning is necessary for human-robot interaction. Their expertise is in unchanging settings where route planning is minimal, such as a randomly sampled, fast-spreading random tree. The majority of common robot job paths involve online tracking control. Operator workstations are protected by safety cages. Everything is always changing, and neither the surroundings nor the issues are static. Online collaborative trajectory planning using sensor inputs, which is done in real-time, has supplanted offline planning. They are making route and path predictions using environmental data from throughout the world, such as grid maps and random spanning trees. Nevertheless, they aren't up to the task of online obstacle avoidance planning because of

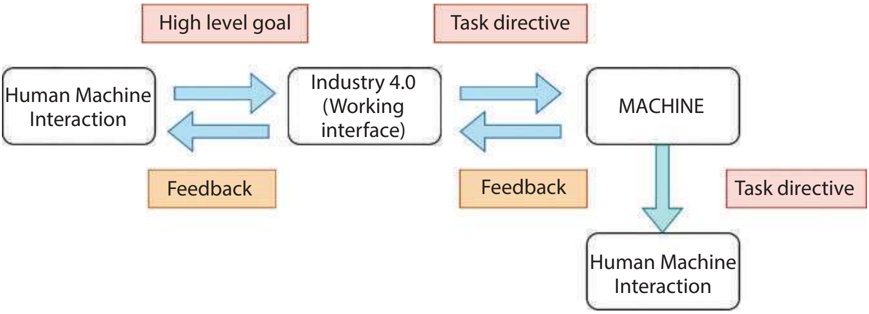

High level goal
Task directive
Human Machine Interaction
Industry 4.0 (Working interface)
MACHINE
Feedback
Feedback
Task directive
Human Machine Interaction

Figure 3.5 Contact between humans and their tasks.

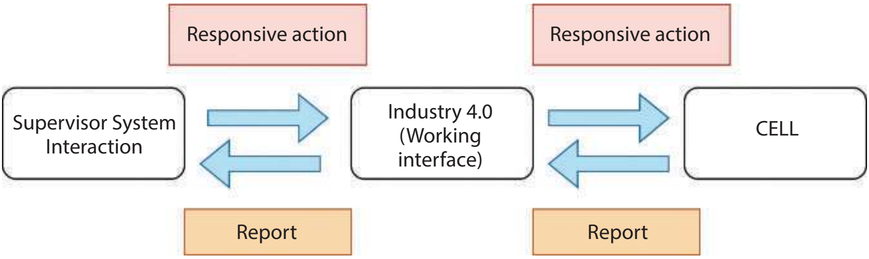

Responsive action
Responsive action
Supervisor System Interaction
Industry 4.0 (Working interface)
CELL
Report
Report

Figure 3.6 Involvement of supervisor systems.

how slowly they solve problems. Making advantage of local environmental data, it plans the online course on robot arms to avoid obstacles. Null space projection, feedback control, and potential fields are used by the online planning of the trajectory system of the operating arm. Certain individuals work hard to forestall cyberattacks. The Gaussian Mixture Model with expectation maximization may be tested using an N-point dataset that includes joint angles, hand trajectories, or hand object distance vectors. Probability density is present in a mixed model including K components.

3.3.1 Auto-Manufacturing IORT and COBOTS

Expanding the workforce using COBOTS and IORT that enhance the performance of workers, safety, and retention may help firms deal with falling talent pools, increasing labor costs, and greater competitiveness. Humans are engaged in a variety of tasks, including screw driving, low-value transportation, palletizing, and repair of machinery. Productivity and quality are both enhanced with an IORT. There is more time for people to think creatively about problems, enhance lean procedures, and solve difficult production issues. Employee happiness is enhanced when they can provide higher value without being subjected to repetitive and risky tasks. No human body can handle strenuous lifting, bending over too far, or doing the same actions over and over again. The dispute led to mishaps, injuries, and quality issues. The end of manufacturing can be the time when a cyber-physical assault reveals product defects. Products run the risk of failing if they are not tested for quality after manufacture. The market might be exposed to structural problems that have not been properly addressed. Inadequately structured, this shows up as design mistakes that the people making the product may not see. This is why post-production analysis requires a new perspective. Scanning technologies like computed tomography (CT), Raman spectroscopy (Raman), and others might detect defects in finished goods before they leave the factory. An IORT bird's-eye perspective of expanding information networks was made possible by SDNs and cyber-physical systems. It offers a fresh and encouraging solution to the present networking problem—the separation of hardware and network services, similar to software-defined networking. Network decoupling, according to their argument, necessitates the technology behind the network being abstracted. The whole network is considered in this case. An SDN's central controller is like the brains of a software-defined network.

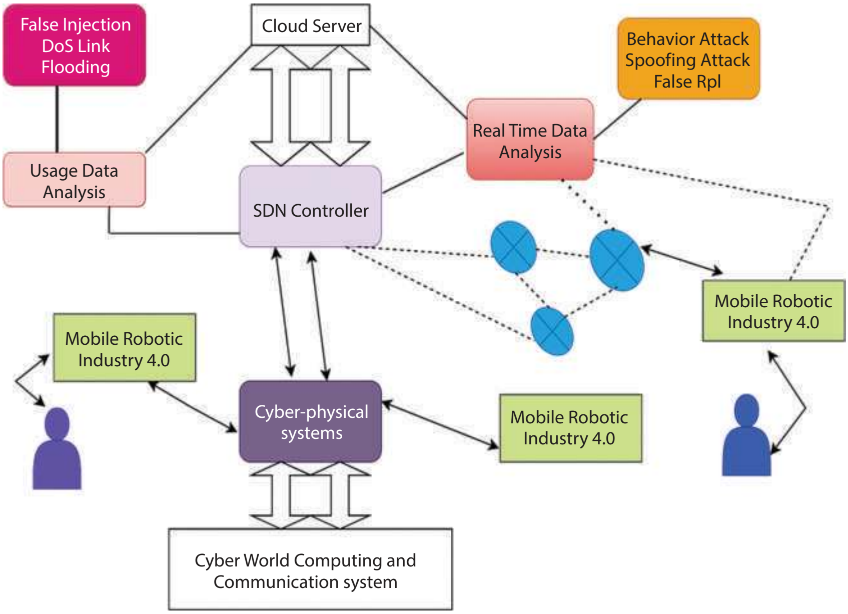


Figure 3.7 Mobile robots with a self-driving architecture are safe for humans.

The software used to manage a network also records the connections made by each device in the network. It is impossible to have an SDN network without them. Very separate from one another are firewalls, switches, and routing. Individual progress is possible once a network is divided into these three parts. The administration plane consists of firewalls, routing, and policies. Efficiency and cost savings may be achieved by load dispersion. Hardware or software, packets are sent to network switches via the forwarding plane. Checking packet headers, sending them to controllers, and managing network ports are all responsibilities of the data plane. It is possible to reject or redirect packets sent to devices that aren't listed in the flow tables. Robots that can navigate themselves are a boon to people's safety, as shown in Figure 3.7.

Businesses in the industrial sector are seeking a practical and effective networking solution for connecting their equipment will find what they need with the IORT with SDWNs.

3.3.2 Attacking Node Termination for Human Security

Enhanced human-machine interactions with distinguishing cyber-physical components raise new security and interoperability issues. Workplace integration is a goal of many industrial clients that use automated and semiautomatic manufacturing processes. Concerns in the industrial sector may differ in detail, but they always center on the need for protection. The mitigating mechanisms of SDN-CPS cannot detect and react to cyberattacks on the Internet of Things (IoT) after the cyber-layer has been compromised. At this stage of development, figure out how to manipulate the target system (CPS) so that the penetration can be increased. The attack strategies, intended victims, potential outcomes, and interdependencies between the various levels of the IoRT with SDN-CPS-security are shown on the right. Attacks on multiple levels of the system are possible with SDN-CPS security due to the many system layers. Unanticipated consequences are probable outcomes of the high level of interconnection among SDN-CPS-security components at various levels. These events may take place at different. Employees and their computers still need security, even if the cyber layer of the system is compromised. Another problem is covering the robot's surfaces so they cannot be touched by humans. When it comes to safety distance, camera systems are crucial. There is a larger safety gap when human workers are faster. Very little has been done about ineffective CPS attacks and IoRT. Several technological concerns arise from the selection of SDN-CPS-security technologies. The hardest part is figuring out how far away from the humans in the manufacturing cell the robot can safely operate. As far as technology is concerned, the limit will be very small sensor safety distances. To avoid sophisticated cyberattacks in a networked setting needs dependable parts and a strategy to counteract cyber threats. It is possible to get inaccurate results due to issues with network connection or the latency rates of individual sensors' data. Problems with delays caused by employing a wide variety of sensors from different manufacturers may be solved with the help of IoRT. A correlation between worker speed and recommended safe distance was found. These dangers have the potential to disrupt the system and endanger people [25].

3.4 Results

The most secure and effective machine learning models for classifying attack data on IoRT networks were determined by evaluating multiple models and utilizing a variety of performance metrics. These models included Decision Tree (DT), Random Forest (RF), support vector machine (SVM), and backpropagation neural network (BPNN).

Two intriguing facts, along with the confusion matrices for GMM and EM. One issue is that there is insufficient data and unequal representation of the classes. There may be an impact on the results due to the absence of test data demonstrates that when compared to RF, SVM, SVMG-RBF, and BPNN, all the other models perform badly, except DT (Figure 3.8).

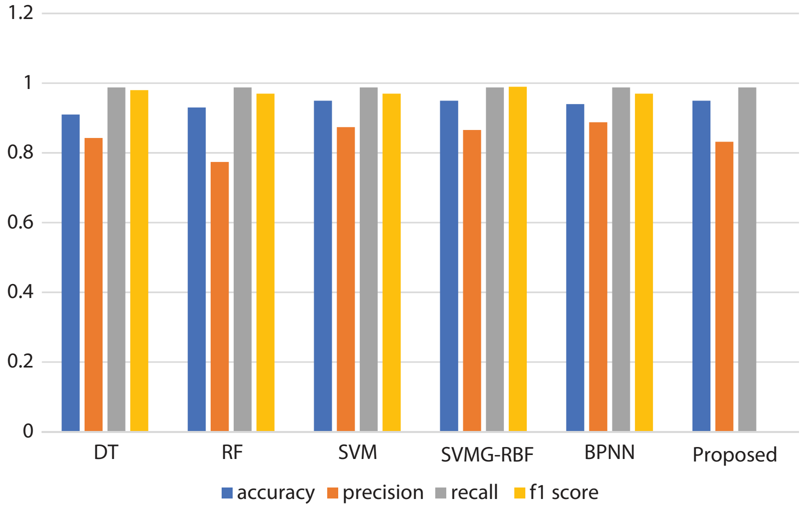


Figure 3.8 Classification of node data using a machine learning technique.

See how increasing the test data impacts the scores of the RF, SVM, SVMG-RBF, and BPNN models. increasing the amount of data attacked by the IoRT network. A 30% increase in the amount of test data enhances recollection at the expense of accuracy. Both recall and accuracy are lost by the model beyond 50%. The results produced by the model differ substantially because of the lack of data.

3.5 Conclusion

This study outlined the overarching strategy for the Internet of Robotic Things (IoT) steering in controlled environments, where the needs and actions of mobile robots are met automatically. To provide a versatile supervisory controller that uses SDN and CPS to ensure the robot's precise direction-finding even when faced with unpredictable obstacles. The suggested approach offers a generic framework for displaying requirements via parts that are wholly reliant on the tasks that automation accomplishes. To enhance human security with mobile robots in the context of Industry 4.0 architecture, provide a Gaussian Mixture Model with expectation maximization tussles. This model would be useful for doing research in close conjunction with many robots. Include the models developed in this study into an intrusion detection system (IDS) prototype shortly and test it with a variety of data and dangers to validate its multiclass capacity.

References

1. Singh Rajawat, A., *et al.*, Reformist framework for improving human security for mobile robots in industry 4.0. *Mobile Inf. Syst.*, 2021, 1–105, 2021.
2. Fankhauser, P., *et al.*, Kinect v2 for mobile robot navigation: Evaluation and modeling. *2015 International Conference on Advanced Robotics (ICAR)*, IEEE, 2015.
3. Birk, A., *et al.*, Safety, security, and rescue missions with an unmanned aerial vehicle (UAV) aerial mosaicking and autonomous flight at the 2009 European land robots trials (ELROB) and the 2010 response robot evaluation exercises (RREE). *J. Intell. Rob. Syst.*, 64, 57–765, 2011.
4. Simanek, J., Reinstein, M., Kubelka, V., Evaluation of the EKF-based estimation architectures for data fusion in mobile robots. *IEEE/ASME Trans. Mechatron.*, 20, 2, 985–990, 2014.
5. Crowley, J., Navigation for an intelligent mobile robot. *IEEE J. Rob. Autom.*, 1, 1, 31–41, 1985.

6. Savci, I.H., *et al.*, Improving navigation stack of a ros-enabled industrial autonomous mobile robot (AMR) to be incorporated in a large-scale automotive production. *Int. J. Adv. Manuf. Technol.*, 120, 5-6, 3647–3668, 2022.
7. Magalhães, H., *et al.*, Towards fast plume source estimation with a mobile robot. *Sensors*, 20, 24, 7025, 2020.
8. Stoyanov, T., *et al.*, Comparative evaluation of range sensor accuracy for indoor mobile robotics and automated logistics applications. *Rob. Auton. Syst.*, 61, 10, 1094–1105, 2013.
9. Palazzo, S., *et al.*, Domain adaptation for outdoor robot traversability estimation from RGB data with safety-preserving loss. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2020.
10. Uhm, T., *et al.*, Multi-modal Sensor Based Localization and Control Method for Human-Following Outdoor Security Mobile Robot, in: *Distributed, Ambient and Pervasive Interactions: 7th International Conference, DAPI 2019, Held as Part of the 21st HCI International Conference, HCII 2019, Orlando, FL, USA, July 26–31, 2019, Proceedings 21*, Springer International Publishing, Florida, 2019.
11. Jung, B. and Sukhatme, G.S., Real-time motion tracking from a mobile robot. *Int. J. Social Rob.*, 2, 63–78, 2010.
12. Lilienthal, A.J., *et al.*, A statistical approach to gas distribution modelling with mobile robots-the kernel dm+ v algorithm. *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2009.
13. Guo, P., Detection and Prevention: Toward Secure Mobile Robotic Systems, The Pennsylvania State University, 2018.
14. Townsend, N.W. and Tarassenko, L., Neural networks for mobile robot localisation using infra-red range sensing. *Neural Comput. Appl.*, 8, 114–134, 1999.
15. Brunner, M., *et al.*, Design and comparative evaluation of an iterative contact point estimation method for static stability estimation of mobile actively reconfigurable robots. *Rob. Auton. Syst.*, 63, 89–1075, 2015.
16. Suzuki, S., Hasegawa, S., Okugawa, M., Warning system for a crawler-type mobile robot with passive sub-crawler. *2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, IEEE, 2013.
17. Richarz, J., *et al.*, A monocular pointing pose estimator for gestural instruction of a mobile robot. *Int. J. Adv. Rob. Syst.*, 4, 1, 17, 2007.
18. Liu, J.NK., Wang, M., Feng, B., iBotGuard: an Internet-based intelligent robot security system using invariant face recognition against the intruder. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, 35, 1, 97–105, 2005.
19. Belzile, B., *et al.*, From safety standards to safe operation with mobile robotic systems deployment, 2021.
20. Gao, M. and Marius Zöllner, J., Local contextual trajectory estimation with a demonstration for assisting mobile robot teleoperation. *2017 European Conference on Mobile Robots (ECMR)*, IEEE, 2017.

21. Bui, T.T.Q., Widyotriatmo, A., Hong, K.-S., Sonar-based collision avoidance and probabilistic motion model for mobile robot navigation. *2009 7th Asian Control Conference*, IEEE, 2009.
22. Kristoffersson, A., Coradeschi, S., Loutfi, A., A review of mobile robotic telepresence. *Adv. Hum. Comput. Interact.*, 2013, 3–3, 2013.
23. Kumar, A., Chatterjee, J.M., Rathore, P.S., Smartphone confrontational applications and security issues. *Int. J. Risk Contingency Manage. (IJRCM)*, 9, 2, 1–18, 2020.
24. Kumar, A., Kumar, P.S., Agarwal, R., A face recognition method in the IoT for security appliances in smart homes, offices and cities. *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, IEEE, 2019.
25. Sasubilli, S.M., Dubey, A.K., Kumar, A., Hybrid security analysis based on intelligent adaptive learning in Big Data. *2020 International Conference on Advances in Computing and Communication Engineering (ICACCE)*, IEEE, 2020.

4

Feature Extrusion and Categorization of Disease by Hybrid Neuro-Fuzzy Computing

Manideep Yenugula[1]*, K.S. Chakradhar[2], Makhan Kumbhkar[3], D. Victorseelan[4] and Rupinder Karur[5]

[1]Department of Commerce, Chickasaw Nations, Neptune, USA
[2]Department of ECE, Mohan Babu University, Tirupathi, India
[3]Department of Computer Applications, ICAR-Indian Institute of Soybean Research, Indore, India
[4]Department of Statistics, Bharathiar University, Coimbatore, India
[5]Dept. of CSE, Chandigarh University, Punjab, India

Abstract

Medical illness categorization using machine learning algorithms encounters difficulties due to insufficient, unclear, and erroneous data. The performance of classification models is affected by the availability of data. The research in this article classifies illnesses using medical data using a model called Linguistic Neuro-Fuzzy Feature Extraction (LNF-FE). To deal with uncertainty, the first model uses linguistic fuzzification to derive membership values. While increasing membership values may not have a major effect on the system, it will increase the number of aspects, which means more time is needed for training. To address this issue, the Neuro-Fuzzy (NF) model employs a combination of Feature Extraction (FE) algorithms to determine and extract the most valuable properties for the network. The artificial neural network (ANN) method is used for categorization with these decreased features. They compare the proposed model's performance to that of existing models and test and verify it using eight benchmark datasets. Statistical methods like Friedman and Holm-Bonferroni were used to verify the accuracy of the findings. The results of these experiments demonstrate that, when applied to real-world issues, the proposed approach performs better than competing methods.

**Corresponding author*: manideep.sre@gmail.com

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (49–68) © 2025 Scrivener Publishing LLC

***Keywords*:** Machine learning, medical disease classification, LNF-FE, artificial neural network, benchmark datasets

4.1 Introduction

Performing accurate medical data evaluation is crucial for illness prediction, identification, and analysis [1–3]. A machine learning classification algorithm is used for accurate and efficient illness detection and diagnosis [4, 5]. Computationally demanding technologies for precise medical data analysis have revolutionized machine learning in recent years. Several clinical concerns, including accurate, reliable, and fast decision models, need more attention to help clinicians effectively diagnose illness [6–8]. Medical datasets often include distracting, duplicated, and missing data, which may hinder classification model performance. Medical data quality and classification models impact the effectiveness of the classifier (disease prediction) [9, 10]. Proper analysis of sensitive medical data using classifiers is crucial for accurate illness prediction and diagnosis. Pattern recognition and machine learning use classification to learn from real-world issues [11, 12]. A model is created to properly forecast desired class levels using data. Artificial neural networks (ANNs) and other standalone classification algorithms have several drawbacks, including poor adaptability for complicated situations, sluggish convergence, and a tendency to encounter local minima. As a result, their accuracy may suffer [13, 14]. Although it takes additional time to forecast the outcome owing to its enormous paralleled arrangement, ANN is a very computationally parallel model with self-adaptive and self-learning capabilities [15, 16]. Uncertainty concerns may emerge at any point in the classification process with ambiguous and imprecise data, and ANN is not designed to deal with such problems [17, 18]. This issue is addressed by using fuzzy logic (FL) to translate the numerical input characteristics into their equivalent language phrases (low, medium, and high) [19, 20]. By applying linguistic qualities like medium, low, and high, this fuzzy technique converts every input feature into its equivalent membership values [21–23]. This process also extracts all linguistic characteristics from the input features, up to a maximum of three times the amount of features. Finding the membership value to distinct linguistic concepts is another way FL might solve the uncertainty issue. Adaptive Neuro Fuzzy Inference System (ANFIS) is one of the hybrid models that combine the best features of ANN and FL [24, 25]. ANN can self-adapt its network architecture based on what it learns from data, but it cannot understand the data-derived knowledge. When it comes to contracts, FL is not good at learning from data, but it has no trouble understanding the language used instead of numbers [26, 27]. Words that are often linked to membership level are linguistic variables. In the ANFIS, a function for Gaussian membership converts crisp values to fuzzy ones [28, 29]. The rule-driven ANN model attached to the inference model receives this membership value. Use defuzzification to transform linguistic output to crisp values [30, 31]. Generating appropriate rules is crucial for accurate output prediction. To achieve this, the model should be trained more slowly due to the rule-based approach [32]. To handle uncertainties and imprecise input information, the Neuro-Fuzzy model (NF) combines the benefits of ANN and FL [33, 34].

This study's objective is to present a novel hybrid method for medical condition categorization termed Linguistic Neuro-Fuzzy with Feature Extraction (LNF-FE). It does this by combining the NF with FE methods. This suggested model simplifies and strengthens the model by removing the unnecessary fuzzy features [35, 36]. Fuzzified membership values are calculated using a linguistic fuzzification technique in this model to handle uncertainty issues. This method of linguistic fuzzing increases the dimensionality. Unfortunately, not every membership value will have a major impact on the model. To provide accurate illness predictions, FE algorithms that remove unnecessary fuzzy information are crucial. To identify which characteristics are really useful for building the model, this study employs FE methods including PCA and integrated component analysis (ICA). For illness classification, the ANN model receives these much reduced fuzzified features from the FE techniques. The model is made more basic and robust by removing all except the most important fuzzy characteristics. Removing superfluous information improves the accuracy of illness predictions and decreases the computing cost of the model.

The rest of the article is structured as follows: Part 2 lays forth the foundational ideas, including the verbal fuzzy principal and component analysis (PCA). In Section 4.3, the basic operation of the suggested LNF FE method for illness categorization using the Pima Indian disease dataset is detailed. Section 4.4, provides the results of the experiment and their analysis. All of the models' statistical analyses are in-depth in Section 4.5. The study concludes in Section 4.6 with a discussion of the research effort's future scope.

4.2 Methods and Materials

This section explains the procedure by which the subsequent parts, including principal component analysis and the linguistic fuzzification process, function in detail.

4.2.1 Procedure for Linguistic Fuzzing

This fuzzification procedure involves mapping each input pattern feature to a membership value according to one of three language membership functions (low, medium, or high). To calculate the membership values of each input pattern feature, the Π-type member function is used in this case. As a result of this fuzzy linguistic expansion, the original input characteristics are tripled in number to correspond to corresponding linguistic membership values, which helps with the uncertainty problem. This section explains the fuzzy-logic technique in depth.

Given a data D, the i^{th} outline of all features is shown in Eq. (4.1).

$$P_i = \left[F_{i,1,}F_{i,2,} \ldots, F_{i,n,}\right] \tag{4.1}$$

Using Eq. (4.2), the value of membership of the j^{th} feature found in the i^{th} patterns is given as $F_{i,j}$. To deduce linguistic qualities from input information, the Π-type function for membership is utilized. This membership function uses linguistic qualities like low, medium, and high to translate input features into fuzzified values. In the same way, according to Eq. (4.3), every feature in the i^{th} pattern has 3n fuzzy features. Here, n is the input features in the data. In this context, the phrases "medium," "low," and "high" stand for the value of membership of the i^{th} pattern that defines the j^{th} characteristic about the linguistic attributes, which may be described as low, medium, and high accordingly.

$$F_{i,j} = \left[\mu_{low}\left(f_{i,j}\right), \mu_{medium}\left(f_{i,j}\right), \mu_{high}\left(f_{i,j}\right)\right] \tag{4.2}$$

$$\begin{aligned} P_i = \Big[&\mu_{low}\left(f_{1,i}\right), \mu_{medium}\left(f_{1,i}\right), \mu_{high}\left(f_{1,i}\right), \mu_{low}\left(f_{2,i}\right), \mu_{medium}\left(f_{2,i}\right), \\ &\mu_{high}\left(f_{2,i}\right), \ldots \mu_{low}\left(f_{n,i}\right), \mu_{medium}\left(f_{n,i}\right), \mu_{high}\left(f_{n,i}\right) \end{aligned} \tag{4.3}$$

Lastly, using the three l fuzzy linguistic parameters, such as Medium, Low, and High, all of the input characteristics are translated into

Table 4.1 Fuzzified 5 mammographic mass dataset.

Features	Values	State 1	State 2	State 3	State 4	State 5
BIRADS	Minimum	1.1	1.1199	2	1.1088	1
Age	Middle	2.1112	1.7779	1.2223	1.7779	1.2223
	Highest	1.9979	1.8912	1.9979	1.8912	1.9979
	Minimum	1.1738	1.9095	1.4599	1.983	1.0459
	Medium	1.9797	1.7424	1.9988	1.1316	1.63
Form	High	1.8262	1.0907	1.5403	1	1.9543
	Minimum	1.1088	1.83	1	1.83	1.83
	Middle	1.7779	1	1	1	1
Border	High	1.8912	1	1.83	1	1
	Minimum	1	2.93	1	1.83	1
	Middle	1	1	1	1	1
Density	High	1.83	1	1.83	1	1.83
	Minimum	1.1088	1.1088	1.1088	1.1088	1.1088
	Middle	1.7779	1.7779	1.7779	1.7779	1.7779
	High	1.8912	1.8912	1.8912	1.8912	1.8912

fuzzified matching membership values. The membership values of the Mammographically Masses data, which have been enlarged and fuzzified, for five instances, are calculated using appropriate type Π functions in language and are shown in Table 4.1. Correspondingly, using the linguistic qualities of any input pattern, it is possible to model any dataset in order to get the fuzzified membership value.

4.2.2 Principal Component Analysis

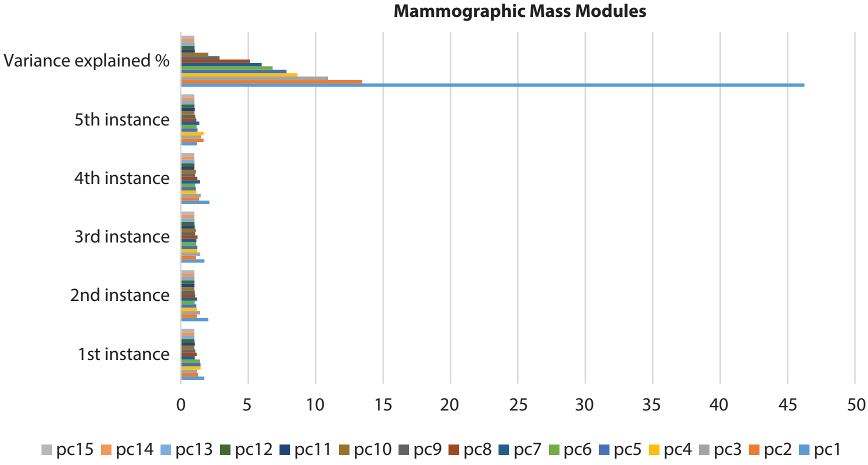
Following the linguistic fuzzification procedure, the input characteristics are enlarged to three times their original size. Fuzzy language expansion of the input characteristics raises the model's complexity. This problem is solved by using principal component analysis (PCA) to identify the elements that are strongly influencing the model. The fuzzified features that are not adding anything useful to the model are just discarded, and only the features that are useful are sent on. In this section, we have covered the basic idea of principal component analysis (PCA), which is used to extract important fuzzy characteristics from the initial input information. Presently have an extended and fuzzy feature matrix, "F," with 'n' samples and'm' features. It finds the average of all the features in the dataset. Subtracting this mean from each feature is the next step. After that, the correlation or covariance matrix is calculated. The eigenvalues and eigenvectors are then calculated from the covariance matrices. Eq. (4.4) and Eq. (4.5) demonstrate the use of eigen decomposition to measure the PCA.

$$E = (F^t \times F)_{m\times m} \tag{4.4}$$

$$T = F \times E \tag{4.5}$$

The primary components are arranged by the eigenvalues (λ) and are shown in each column of the eigenvector E. The feature vectors are formed by sorting the main components according to the eigenvectors in declining order of the eigen standards. Due to space constraints, only five occurrences of the calculated main components of the Mammographically Mass data's fuzzy matrix are shown in Table 4.2.

Table 4.2 Main modules of the mammographically mass.



The initial step is to find the 'r' columns in Ematrix and then choose the corresponding Ermatrix. Included in it are the first 'r' main components of matrix E, which play a crucial role in the model. People can find out what 'r' is by looking at the PCA's explained component. All of the PCs that are making up the model are evaluated according to the components that are described. Table 4.2 displays fifteen of the personal computers used in this example (the Mammographic Mass dataset). Each component's explained variance is as follows: 46.246, 13.455, 10.918, 8.659, 7.838, 6.809, 5.996, 5.127, 2.868, 2.038, 1.035, 1.014, 1.008, 1, and 1. Every PC's model relevance is defined by this explained variance. Using the explained variance score, PCs 11, 12, 13, 14, and 15 have less significance in this case.

As a result, removing these PCs is crucial for making the model more reliable and reducing the computational expense.

$$T_r = F \times E_r \tag{4.6}$$

The artificial neural networks method for disease predictions is fed this converted matrix, which comprises the reduced characteristics that are extremely substantially donating to the net. Here, the ANN model only takes into account PCs 1–10 for analysis since they account for 98.945% of the total data. It skips over PC11–PC15 and goes straight to the model of ANN for illness categorization processing.

4.3 Features Extraction Model-Based Linguistic Neuro-Fuzzy

This division proposes a novel LNF-FE, hybrid method, to use medical data for illness prediction. This model combines feature extraction methods with the linguistic neuro-fuzzy model. Figure 4.1 shows the three steps of this LNF-FE model's operation: (1) fuzzy logic, (2) feature extraction, and (3) artificial neural network. At the outset, this method generates the linguistic values that match to the input characteristics. The decision-making process could not always benefit from all of these language values. The second step is to use feature extraction algorithms to take the enlarged features and pull out the important ones, also called reduced features. One last step in using an ANN model for illness prediction is to provide reduced features to it.

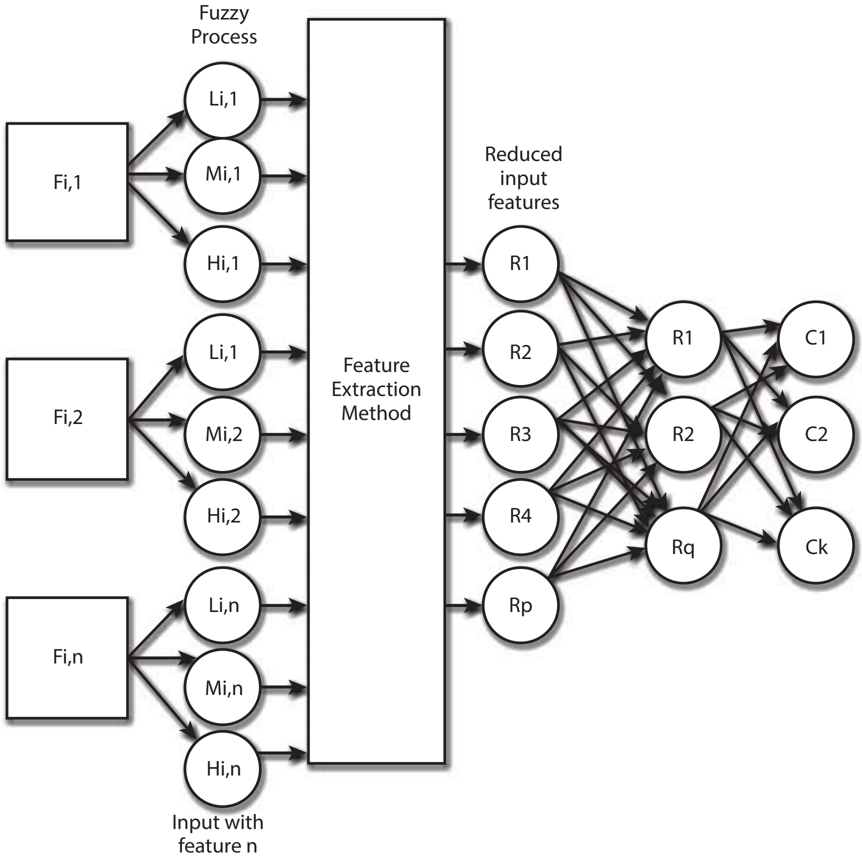


Figure 4.1 LNF-FE model's operation.

Using a number of benchmark biological datasets, this research examines a number of disorders, including heart disease, cancer, thyroid issue, liver disease, cardiovascular disease, and diabetes mellitus. Each patient is classified into one of two disease states, benign or malignant, for the purposes of cancer disease analysis. Abnormal development or alteration in the breast tissues is often seen in benign tumors, which are not malignant. Although it is more common in women, males may also get this illness, which presents with symptoms that are similar to those of breast cancer. In a similar vein, breast cancer cells may metastasize to other parts of the body over time, creating a malignant tumor. Beyond this, mammography is the gold standard for detecting breast cancer. A suspicious lump, which might be malignant or not, can sometimes be found within a human body.

Surgical intervention and malignant findings are common consequences of a biopsy with a high positive value. Also included for analysis is the survival rate of breast cancer patients who receive surgery. The human lung is another essential organ. Although heavy smoking is the most common cause of lung cancer, which is a malignant tumour of the lungs, the illness may also strike nonsmokers. As a result of the abnormal development of lung tissues brought on by lung cancer, a person's capacity to breathe will be diminished. A persistent cough, blood in the mucus, decreased appetite, chest discomfort, and difficulty breathing are the hallmarks of this illness. Similarly, constriction or narrowing of blood arteries may cause cardiovascular disease, which in turn can cause heart disease, stroke, and other cardiac conditions. If the heart is unable to circulate enough blood throughout the body, organs such as the kidneys, brain, and heart might all fail. When a blood clot blocks the flow of blood to the heart, it may cause a myocardial infarction. A hormonal imbalance is another prevalent cause of thyroid illness. Metabolic rate may be accelerated by the hormone secreted by the thyroid glands. Hyperthyroidism and hypothyroidism are the main issues with thyroid disorders. When a person has hyperthyroidism, their thyroid hormone is overactive and flows into the bloodstream in excess; when they have hypothyroidism, their thyroid hormone is inactive and flows into the bloodstream in low amounts. The liver is an extra essential organ that helps the body eliminate toxins from the blood as well as store vitamins and hormones. Nevertheless, hepatitis and liver disorders are linked to this organ. In order to analyze liver disorders that may develop as a result of heavy alcohol intake, the term "liver disorder disease" is utilized. Hepatitis, another infection-related inflammatory liver disease, is similar. The process is disrupted and several complications are caused by hepatitis. Diabetes mellitus, another prevalent condition, is characterized by consistently elevated blood glucose levels.

This work applies the LNF-FE to the Pima Indian Diabetes (PID) dataset. All of the data have been modeled for analysis as well. There are several records in the PID dataset that include eight variables: gestational age (A), blood pressure (BP), skin thickness (ST), insulin (I), body mass index (BMI), diabetes pedigree function (DPF), and pregnancy status (P) (see Figure 4.2). Using the linguistic fuzzification method, LNF-FE transforms the initial attributes into corresponding linguistic membership values, allowing it to forecast diabetic symptoms. The computational price of the model is increased due to the extension of linguistic membership values. By removing irrelevant characteristics, principal component analysis (PCA) lowers the dimensionality of the linguistic fuzzy matrix, allowing

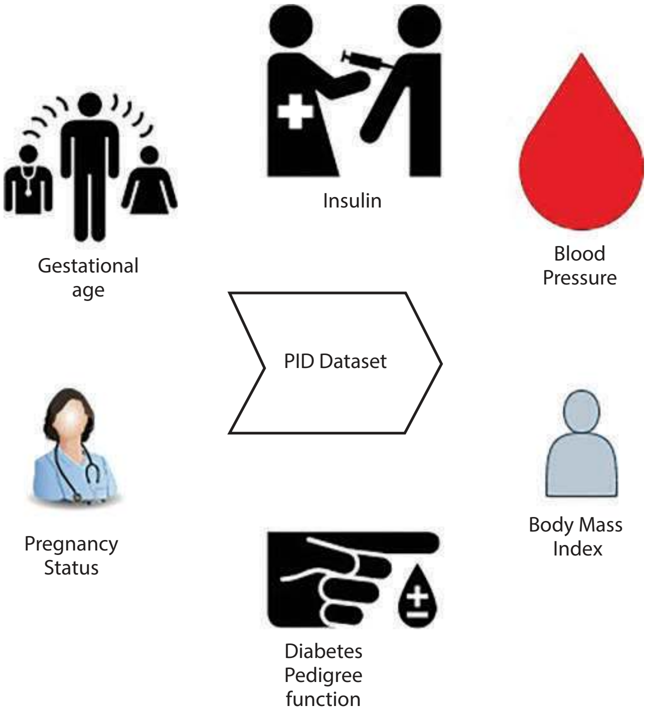


Figure 4.2 PID datasets variables.

us to bypass this problem. After that, the ANN model is fed this reduced matrix once again to forecast the occurrence of diabetes.

4.4 Results

Based on medical datasets, this section displays the results of six different classification models: ANN-ICA, ANN, ANN-PCA, LNF-ICA, LNF, and LNF-PCA. The medical datasets were sourced from the UCI machine learning library and Kaggle. Data cleansing and transformation are part of the pre-processing that makes these medical records usable. There is a possibility that medical records include incomplete or irrelevant data. Data cleaning techniques, such as missing value imputation, are used to address these difficulties by filling in the data that is missing. As part of the data cleansing procedure, this study has included the attribute's mean to fill in the missing values in this trial. In a similar vein, medical records may have characteristics with many ranges that the models struggle to handle.

They have transformed these medical data into a certain range and normalized them using the MinMax normalization approach in this experiment. Outliers are removed if the Z-score is either more than three or lower than 3, which is the Z-score approach used for detection. The datasets used in this experiment are split into a training set consisting of 80% cases and a testing set consisting of 20% instances. Python 3.6.5 is used to implement these categorization methods. A computer system with 8 GB of RAM and an Intel Core i53360M CPU running at 2.80 GHz is used for the tests. In order to analyze the findings of this experimentation, and a small number of hyper parameters are taken into account. The amount of characteristics in the dataset determines the amount of input neurons. There is a direct correlation between the amount of class labels and the amount of output neurons. All models have a learning rate of 1.69. The models will continue to iterate until either the error has not changed or a certain amount of iterations have passed. Figure 4.3 displays the classification accuracy.

As illustrated in Figure 4.4(a–f), the error plots include medical data's and six methods. For the PID dataset, the matrix value of the LNF PCA method is true positive, false negative, false positive, and true negative, in that order. Experimentation and careful observation formed the basis of the findings given here. Every dataset is run through these six models, and the average outcome is given. No change in error or maximum iterations are the terminating criterion of all the models mentioned above. The maximum iteration has been specified in this experiment. In machine learning, over fitting is a prevalent issue that may arise in real-world scenarios.

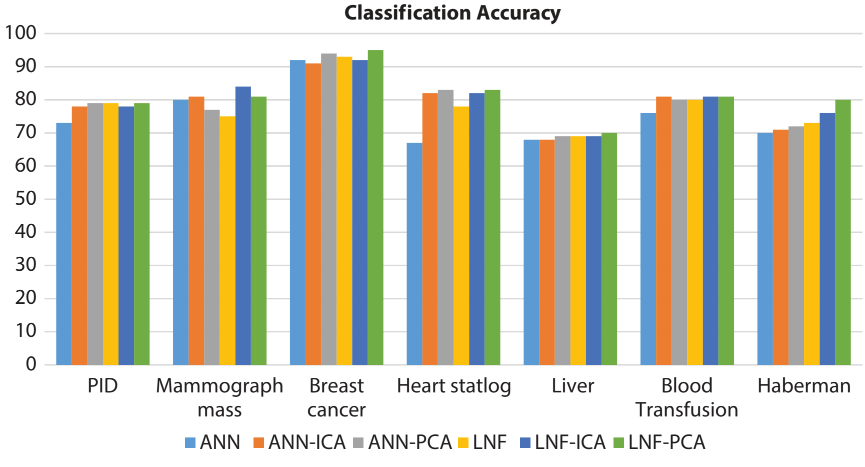
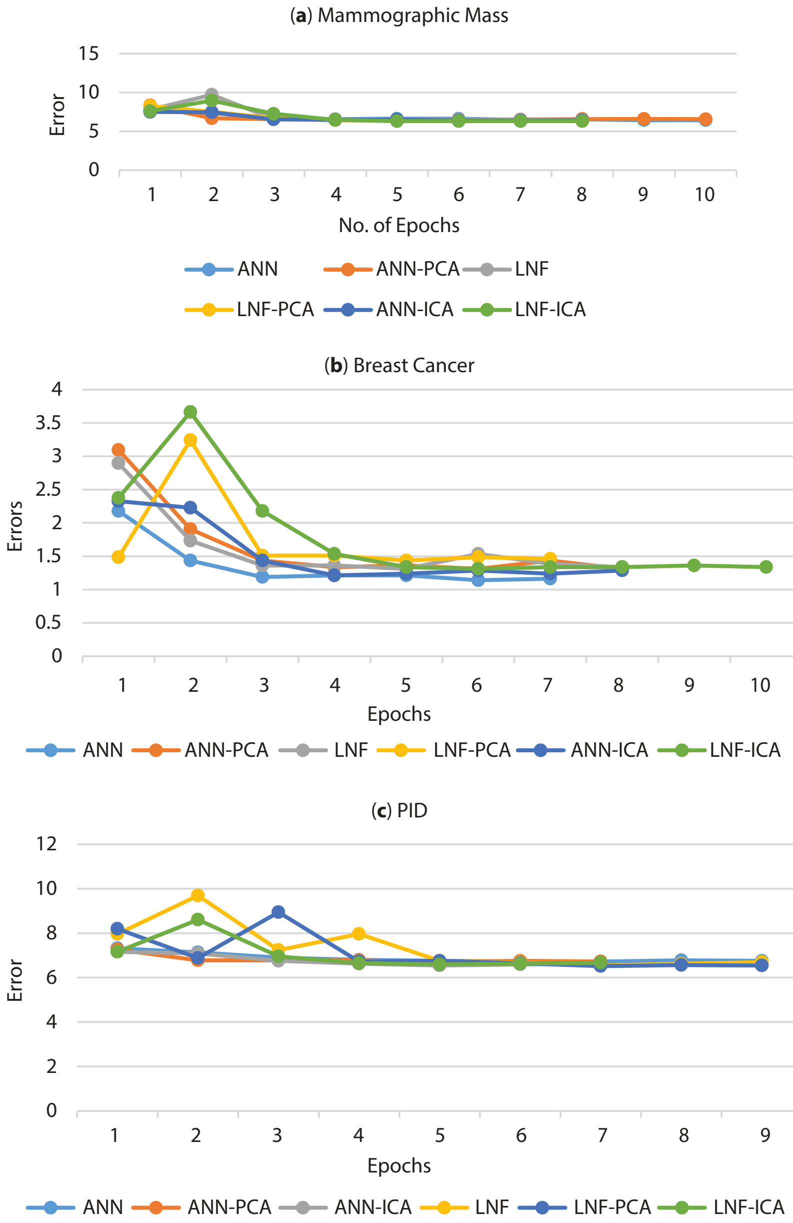


Figure 4.3 Six-model classification accuracy.



(**a**) Mammographic Mass

(**b**) Breast Cancer

(**c**) PID

Figure 4.4 (a–f) Error plots include medical data and six methods. (*Continued*)

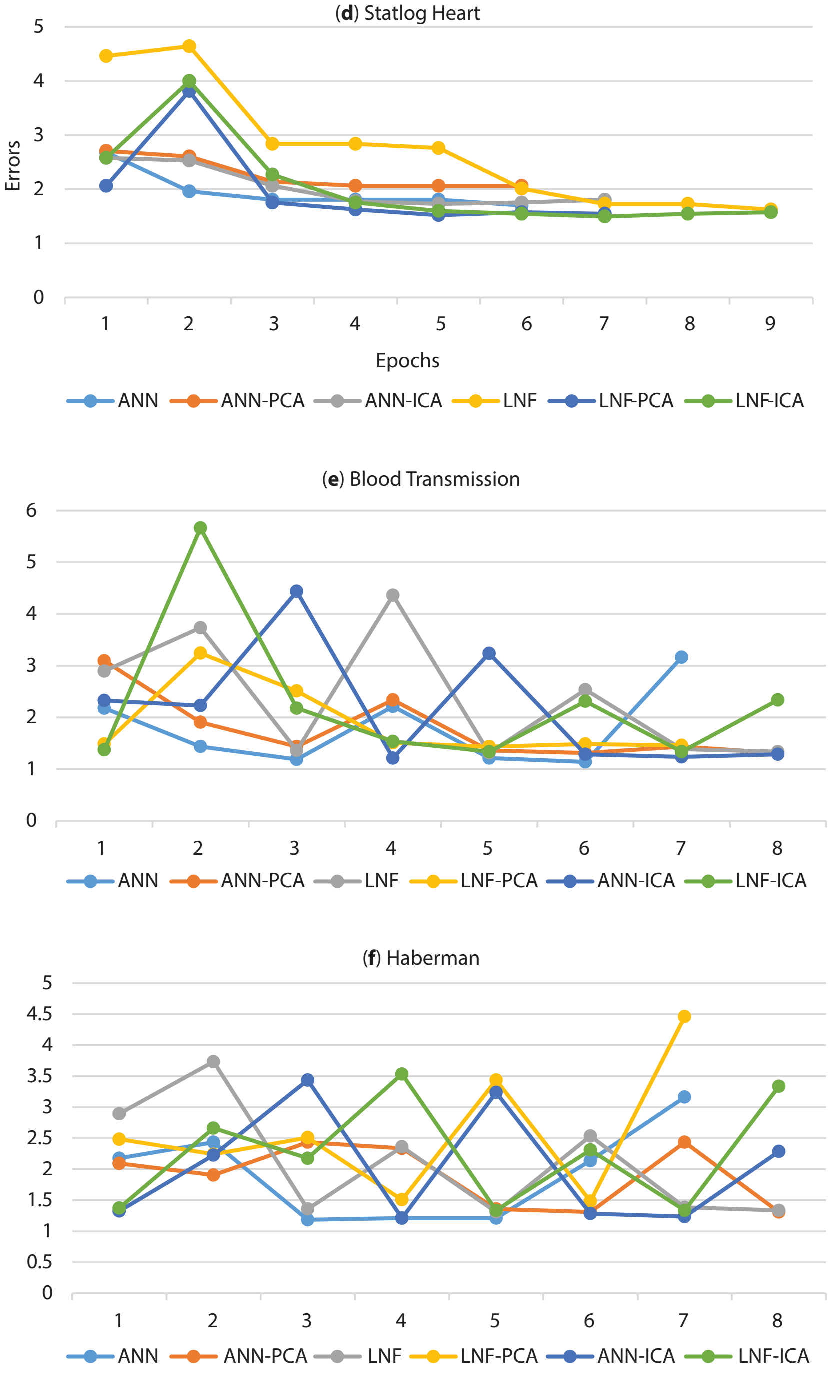


Figure 4.4 (Continued) (a–f) Error plots include medical data and six methods.

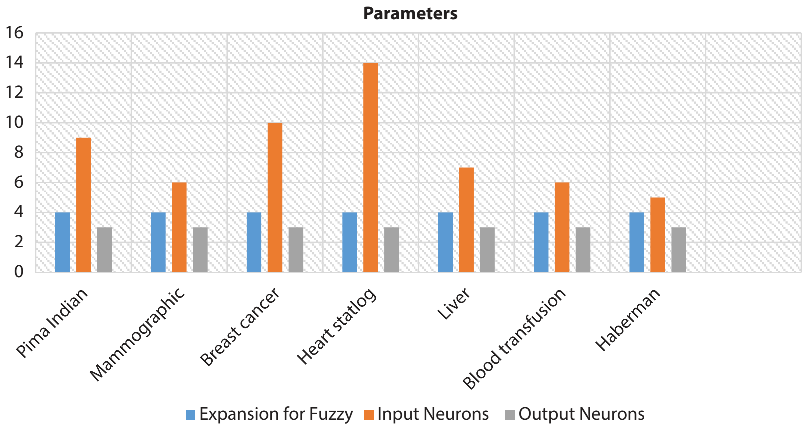


Figure 4.5 Data used by the models.

To reduce the discrepancy between the two sets of data, the back propagation algorithm is used. However, the trained model tends to over fit after a certain point in the optimization process, when further reduction of error no longer impacts performance. Feature reductions and regularization are two approaches used to prevent over fitting in these cases. To eliminate superfluous fuzzy input features, the feature reduction approach is used. The over fitting problem is avoided by removing these superfluous fuzzy features during the regularization procedure prior to execution in the ANN model.

A small number of hyper parameters were taken into account while the models were being constructed in this experiment. Figure 4.5 displays the eight datasets together with their corresponding fuzzy expansions, input neuron counts, and output neuron counts. In all the models, the total of neuron in the layer that is hidden is determined, and there is only one hidden layer employed. All of the models have a learning rate of 1.73. From forty percent to ten percent of the original data is used to reduce the main component dimensions in the FR process. This range is dependent on the datasets. The datasets differ, however, in a range of 10% in terms of the dimensions reduced from the original data.

4.5 Data Analysis Using Statistics

Analyses of statistical significance confirm that the suggested method outperforms its predecessors. The data's importance and kind are also defined in relation to various models via this process. When comparing the different classifiers on different datasets, Damsar presents the statistical tests that were employed. The suggested LNF PCA is evaluated alongside other categorization models, including ANN-ICA, ANN, LNF, ANN-PCA, and LNF ICA. It has also been confirmed by a battery of statistical tests, including the Holm method and the Friedman test. Based on their performance, the classifiers in the Friedman test are given rankings, as shown in Figure 4.6. By using Eq. (4.7), calculates the normal rank of all the categorization, including ANN-ICA, ANN, LNF, ANN-PCA, LNF-ICA, and LNF-PCA.

$$R_j = \frac{1}{P}\sum_i r_i^j \tag{4.7}$$

The six models' average ranks are calculated and shown as follows: R6 = 6.626, R5 = 5.376, R4 = 5, R3 = 4.26, R2 = 3.626, and R1 = 2.126. The null hypothesis cannot be accepted since the rankings of the classifiers do not coincide. This finding indicates that the null hypothesis is rejected. The Friedman statistic X_F^2 is determined to be 39.5 using the rankings 'R_j' of the classifiers and the equation (4.8). Here, P represents the number of data as well as q represents the number of classifications.

$$X_F^2 = 12P/q(q-1)[\sum_j R_j^2 - \frac{q(q+1)^2}{4} \tag{4.8}$$

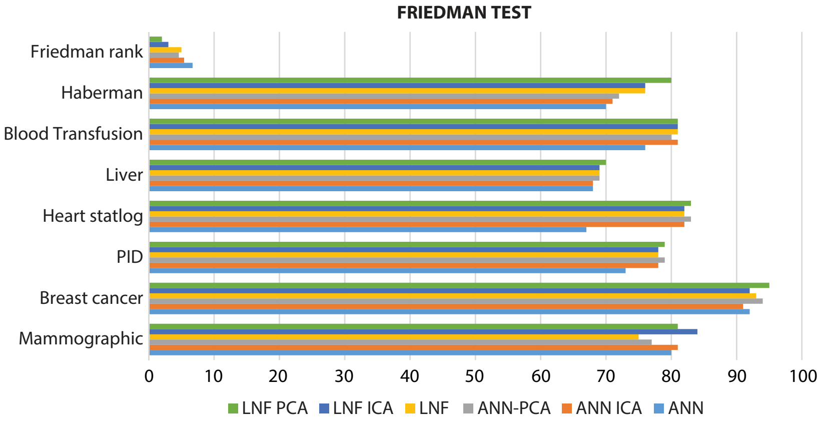


Figure 4.6 Classifiers in the Friedman test rankings.

Applying X_F^2 with 5 degrees of freedom, as shown in Eq. (4.9) yields a Friedman statistic (F_F) of 167. By setting α to 1.02, the critical value of 4.58 is calculated using the Friedman statistics F_F with 5 and 5×7 degrees of freedom. Since the calculated critical value of 4.58 is less than the actual F_F statistics value, they may reject the null hypothesis (H_0).

$$F_F = \frac{(P-1)X_F^2}{P(q-1)-X_F^2} \tag{4.9}$$

When comparing five or more classifiers, the Friedman test could be appropriate. While the Friedman test results do show that there is a difference between the classifiers, they do not identify which classifiers are unique. The purpose of post-hoc analysis is to determine which classifiers stand out from the others. Finding out which classifiers are substantially different from others is done by analyzing the outcomes of the experimental data using the post hoc test. Figure 4.7 displays the density plot.

Using the p-value and z-value, the post-hoc test compares the performance of each classifier against the other classifiers using the Holm technique. Equation (4.10) is used to get the z-value, and the normal distribution chart is used to derive the p-value from the z-value.

$$z = \frac{R_i - R_j}{\sqrt{\frac{q(q+1)}{6P}}} \tag{4.10}$$

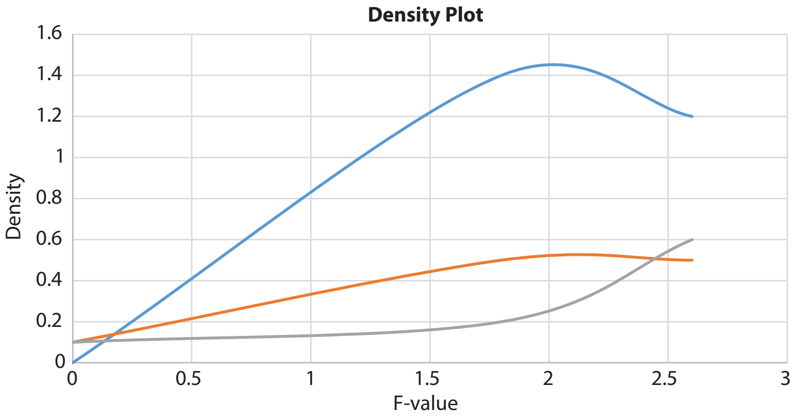


Figure 4.7 Density plot.

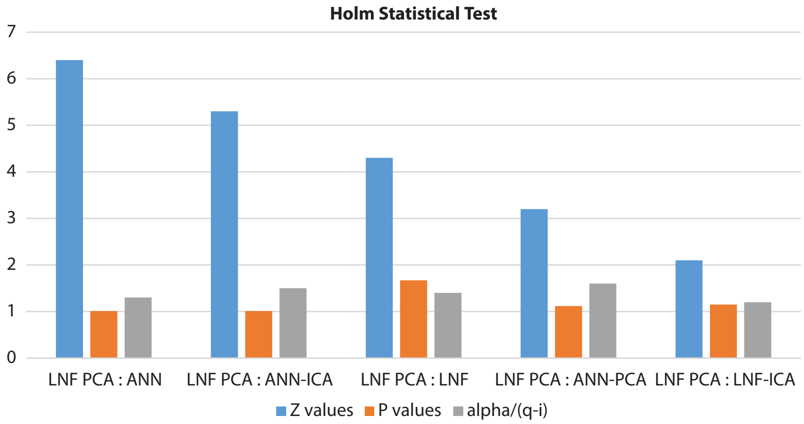


Figure 4.8 Holm statistical test results.

A dataset's count, the z-score, and the total number of classifiers are all represented by the letters "q," "z," and "P," respectively. R_i Stands for the normal ranking of the ith classifier and R_j for the jth categorizer. Figure 4.8 shows the results of comparing six models using z-value, p-value, and $\frac{\alpha}{(q-i)}$, where 'i' is the number of the model and α is less than or equal to 0.01. The majority of the time, the p-value is lower than the $\frac{\alpha}{(q-i)}$ value. In almost every instance, it shows that the null assumption is rejected. With the exception of LNF ICA and ANN PCA the results demonstrate that the suggested LNF PCA model outperforms the competition and is statistically significant. Since there is no statistically significant difference between these classifiers, it follows that LNF-PCA outperforms both ANN-PCA and LNF-ICA.

4.6 Conclusion

This study introduces the LNF-FE framework for medical condition categorization, which uses a linguistic membership function to fuzzily the input characteristics and deal with uncertain and imprecise data. The model becomes more complicated as a result of this fuzzy expansion, which in turn increases the training time. To ensure that the FE model extracts just the aspects that are useful and make a substantial contribution, these extended fuzzy values are provided to it. Once again, the ANN-based model is fed for these reduced characteristics in order to classify diseases. The experimental results show that compared to other models like ANN-ICA, ANN, LNF, ANN-PCA, and LNF-ICA, the LNF-FE model performs better when it comes to illness prediction. When compared to the other models, the suggested LNF-PCA model stands out and performs better statistically, according to analyses conducted using tools like the Holm procedure and the Friedman test. The model's flaw is its fuzzification procedure, which does not take class label properties into account. Picking the right function for membership is also no easy feat. To further improve these models' performance in a wide range of diverse applications, future work will include hybridizing novel feature extracting and feature selection techniques.

References

1. Butdee, S., A Prediction Approach for Aluminum Extrusion Processing Using Neuro-Fuzzy Based Decision Making. *International Scientific-Technical Conference MANUFACTURING*, Springer International Publishing, Cham, 2022.
2. Rajasekaran, S. and Vijayalakshmi Pai, G.A., *Neural networks, fuzzy logic and genetic algorithm: synthesis and applications (with cd)*, PHI Learning Pvt. Ltd., Coimbatore, India, 2003.
3. Elbaz, K., *et al.*, Optimization of EPB shield performance with adaptive neuro-fuzzy inference system and genetic algorithm. *Appl. Sci.*, 9, 4, 780, 2019.
4. Mazzetti, C., *et al.*, Partial discharge pattern recognition by neuro-fuzzy networks in heat-shrinkable joints and terminations of XLPE insulated distribution cables. *IEEE Trans. Power Delivery*, 21, 3, 1035–1044, 2006.
5. Anifowose, F.A., Labadin, J., Abdulraheem, A., Prediction of petroleum reservoir properties using different versions of adaptive neuro-fuzzy inference system hybrid models. *Int. J. Comput. Inf. Syst. Ind. Manage. Appl.*, 5, 413–426, 2013.
6. Ramesh, G., Logeshwaran, J., Rajkumar, K., The smart construction for image preprocessing of mobile robotic systems using neuro fuzzy logical system approach. *Neuro Quantology*, 20, 10, 6354–6367, 2022.
7. Ravi, S., *et al.*, Design of Synthetic Optimizing Neuro Fuzzy Temperature Controller for Twin Screw Profile Plastic Extruder Using Labview. *Intell. Autom. Soft Comput.*, 20, 1, 50–51, 2014.
8. Rajasekaran, S. and Vijayalakshmi Pai, G.A., *Neural networks, fuzzy systems and evolutionary algorithms: Synthesis and applications*, PHI Learning Pvt. Ltd., 2017.
9. Agor, C.D., Mbadike, E.M., Alaneme, G.U., Evaluation of sisal fiber and aluminum waste concrete blend for sustainable construction using adaptive neuro-fuzzy inference system. *Sci. Rep.*, 13, 1, 2814, 2023.

10. Wong, Y.J., *et al.*, Comparative study of artificial neural network (ANN), adaptive neuro-fuzzy inference system (ANFIS) and multiple linear regression (MLR) for modeling of Cu (II) adsorption from aqueous solution using biochar derived from rambutan (Nephelium lappaceum) peel. *Environ. Monit. Assess.*, 192, 1–20, 2020.
11. Mewada, S., Saroliya, A., Chandramouli, N., Rajasanthosh Kumar, T., Lakshmi, M., Suma Christal Mary, S., Jayakumar, M., Smart Diagnostic Expert System for Defect in Forging Process by Using Machine Learning Process. *J. Nanomater.*, 2022, 2567194, 8, 2022, https://doi.org/10.1155/2022/2567194.
12. McClinton, J.T., White, S.M., Sinton, J.M., Neuro-fuzzy classification of submarine lava flow morphology. *Photogramm. Eng. Remote Sens.*, 78, 6, 605–616, 2012.
13. Ravi, S., Sudha, M., Balakrishnan, P.A., Design of intelligent self-tuning GA ANFIS temperature controller for plastic extrusion system. *Modell. Simul. Eng.*, 2011, 1–85, 2011.
14. Jinjri, W.M., Keikhosrokiani, P., Abdullah, N.L., Predicting Cardiovascular Disease Using a New Optimized Adaptive Neuro Fuzzy Inference System-Dragonfly Algorithm.
15. Fragiadakis, N.G., Tsoukalas, V.D., Papazoglou, V.J., An adaptive neuro-fuzzy inference system (anfis) model for assessing occupational risk in the shipbuilding industry. *Saf. Sci.*, 63, 226–2355, 2014.
16. García, V., *et al.*, Using regression models for predicting the product quality in a tubing extrusion process. *J. Intell. Manuf.*, 30, 2535–25445, 2019.
17. Liu, X., *et al.*, Optimized adaptive neuro-fuzzy inference system using metaheuristic algorithms: Application of shield tunnelling ground surface settlement prediction. *Complexity*, 2021, 1–155, 2021.
18. Gotlib, B., Tarasyan, V., Vakalyuk, A., Application of the hybrid controller for isothermal extrusion process control. *Acta Mech. Autom.*, 2, 4, 16–18, 2008.
19. Admuthe, S. and Chile, R., Neuro–fuzzy–based hybrid controller for stable temperature of liquid in heat exchanger. *Int. J. Comput. Sci. Eng.*, 10, 1-2, 220–230, 2015.
20. Eivani, A.R., *et al.*, A novel approach to determine residual stress field during FSW of AZ91 Mg alloy using combined smoothed particle hydrodynamics/neuro-fuzzy computations and ultrasonic testing. *J. Magnesium Alloys*, 9, 4, 1304–1328, 2021.
21. Hashim, R., *et al.*, Selection of meteorological parameters affecting rainfall estimation using neuro-fuzzy computing methodology. *Atmos. Res.*, 171, 21–305, 2016.
22. Bhagya Raj, G.V.S. and Dash, K.K., Comprehensive study on applications of artificial neural network in food process modeling. *Crit. Rev. Food Sci. Nutr.*, 62, 10, 2756–2783, 2022.
23. Gao, J., *et al.*, A novel machine learning method for multiaxial fatigue life prediction: Improved adaptive neuro-fuzzy inference system. *Int. J. Fatigue*, 178, 1080075, 2024.

24. Tsoukalas, V.D., An adaptive neuro-fuzzy inference system (ANFIS) model for high pressure die casting. *Proc. Inst. Mech. Eng., Part B: J. Eng. Manuf.*, 225, 12, 2276–2286, 2011.
25. Afifi, F., *et al.*, DyHAP: Dynamic hybrid ANFIS-PSO approach for predicting mobile malware. *PLoS One*, 11, 9, e0162627, 2016.
26. Ofosu, R.A., *et al.*, Speed Control of an Electrical Cable Extrusion Process Using Artificial Intelligence-Based Technique. *J. Nasional Teknik Elektro*, 18, 1, February 2021, 75–94, 51, 2023.
27. Lu, Q., Lee, S., Chen, L., Image-driven fuzzy-based system to construct as-is IFC BIM objects. *Autom. Constr.*, 92, 68–87, 2018.
28. Shahriari-Kahkeshi, M. and Moghri, M., Prediction of tensile modulus of PA-6 nanocomposites using adaptive neuro-fuzzy inference system learned by the shuffled frog leaping algorithm. *e-Polymers*, 17, 2, 187–198, 2017.
29. Abdullah, F.A., Leaf wetness duration modelling using adaptive neuro fuzzy inference system, Diss., Auckland University of technology, 2016.
30. Gite, A.V., Bodade, R.M., Raut, B.M., ANFIS controller and its application. *Int. J. Eng. Res. Technol.*, 2, 2, 2013.
31. Singh, R.K. and Ou-Yang, F., Neuro-fuzzy technology for computerized automation, in: *Computerized Control Systems in the Food Industry*, pp. 119–178, CRC Press, 2018.
32. Tofigh, A.A., *et al.*, Application of the combined neuro-computing, fuzzy logic and swarm intelligence for optimization of compocast nanocomposites. *J. Compos. Mater.*, 49, 13, 1653–1663, 2015.
33. Jawad, L.H., *et al.*, Prediction of centrifugal compressor performance by using adaptive neuro-fuzzy inference system (ANFIS). *Int. Rev. Modell. Simul. (IREMOS)*, 5, 2012.
34. Kumar, A., *et al.*, Correction to: LTE-NBP with holistic UWB-WBAN approach for the energy efficient biomedical application. *Multimedia Tools Appl.*, 82, 39813, 2023, https://doi.org/10.1007/s11042-023-15604-6.
35. Ilyas, B., *et al.*, Blockchain-enabled IoT access control model for sharing electronic healthcare data. *J. Sens. Actuator Netw.,* 13, 1, 13, 1–22, 2024.
36. Bhargava, N., *et al.*, Adaptive Clustering Algorithm for Big Health Data Classification. *2023 3rd Asian Conference on Innovation in Technology (ASIANCON)*, Ravet IN, India, pp. 1–4, 2023, doi: 10.1109/ASIANCON58793.2023.10270544.

5

AI Based Neuromorphic Vision to Control the Robotic Drilling Machine

Venkat Namdev Ghodke[1]*, Rajeshwari M. Hegde[2], Ramachandra V. Ballary[3] and R. Senthamil Selvan[4]

[1]Department of Electronics and Telecommunication Engineering, AISSMS Institute of Information Technology, Pune, India
[2]Department of Electronics and Telecommunication Engineering, BMS College of Engineering, Bengaluru, India
[3]Department of Computer Science and Engineering, AGMR College of Engineering and Technology, Varur, Hubli, India
[4]Department of Electronics and Communication Engineering, Annamacharya Institute of Technology and Sciences, Tirupati, India

Abstract

A critical perceptual technology that allows these industrial robots to execute accurate activities in unstructured settings, machine vision is now causing a paradigm change in the manufacturing sector due to their extraordinary deployment. Traditional vision sensors are very sensitive to changes in illumination and fast motion, which limits the efficiency and dependability of assembly lines. Neuromorphic visualisation, a relatively new skill with promising features such a high temporal resolution, low latency, and a wide dynamic range, has enormous potential to address limitations of conventional vision. In this work, people introduce a ground breaking controllers for robotic machined applications based on neuromorphic vision. It will allow for quicker and further reliable process. Additionally, they showcase a whole robotic structure that can drill with sub-millimetre accuracy. Two perception phases tailored to the asynchronous results of neuromorphic cameras allow us to suggest a technique to precisely pinpoint the intended work piece in three dimensions. The first step involves estimating the work piece's posture using multi-view reconstruction; the second uses circular hole detection to refine this estimate for a particular portion of the work piece. Next, the robot uses a mix of image-based and position-based visual serving to

**Corresponding author*: venkatghodke@aissmsioit.org

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (69–86) © 2025 Scrivener Publishing LLC

accurately position the drilling end-effector before drilling the desired hole on the work piece. Testing the suggested method on nutplate holes drilled into randomly positioned work pieces in an unstructured setting with unregulated illumination confirms its efficacy. Results from experiments demonstrate that this method operates, with positioning errors of no more than 0.2 mm, and that neuromorphic vision can overcome the speed and illumination restrictions of regular cameras. In response to the demands of the latest industrial revolution, this paper's results highlight neuromorphic vision as a potential technique that might strengthen and speed up robotic manufacturing procedures.

***Keywords*:** Industrial robots, neuromorphic vision, dynamic range, hole detection, robotic manufacturing

5.1 Introduction

Automating cyber-physical manufacturing and machining processes with a high degree of accuracy is a key component of the fourth industrial revolution. The performance, efficiency, productivity, and safety benefits of automating such operations far outweigh the risks of structural damage, rework, and health concerns that come with human operation. Academics and practitioners have devoted a great deal of time and energy to studying drilling and other processes because of their ubiquitous usage in many industrial operations, particularly in the aerospace and automotive sectors. All of these sectors rely heavily on high-precision drilling since it directly affects the machined structures' performance and fatigue life [1–5].

Drilling and related machine-tool automation has long relied on Computer Numerical Control (CNC) systems due to their repeatability and accuracy. On the other hand, CNC machines need a large initial investment in hardware and software, and they have limited workspace and capabilities. Because of their adaptability to changes in the environment and the positioning of work pieces, their large range of functions, their cost-effectiveness, and their capability to operate on huge workspace volumes, industrial robots have recently emerged as a potential substitute for CNC machines in machining applications. Repeatability is still the biggest problem with robotic machining, even though there are a lot of good examples of robots being used for industrial machining. Errors in robotic machining can be caused by the work piece not being perfectly positioned or by the joints on the robot not being very stiff. Closing the loop and using real-time guidance based on sensory input and metrology technologies may help reduce these mistakes. Such methods have been used in a number of published publications to provide exact control over orientation, position, and force within the context of robotic machining [6–10].

Principles of work piece localization and vision-based feature identification are also extensively used in a broad range of assembly-related activities. For instance, a visual guiding system was suggested for a peg-in-hole robotic presentation using four cameras: two for localizing the robotic tool and the others for aligning it with reference holes. The cameras are set up in an eye-to-hand formation. For sub-millimeter level precision in localizing target items in a pick-and-place framework, a multi-view technique was described. Mobile industrial robots' navigation, guiding, and calibration systems have found additional applications for vision systems due to their adaptability.

Using traditional cameras based on frames, which have issues with motion blur, latency, limited dynamic range, and low-light awareness, all of the previously described robotic production systems fail. By integrating the incident light throughout a set exposure period, frame-based cameras are able to produce intensity pictures. Blurring occurs when there is a lot of relative motion, and this intrinsic activity creates perceptual lag, especially with longer exposure times. Conversely, in low-light situations, the picture quality is significantly diminished by using short exposure times, and the depth of field is narrowed since wider apertures are needed. Robot operating speeds, workspace sizes, and ambient illumination conditions are all negatively impacted by the limitations of frame-based cameras, which in turn influence the reliability and efficiency of robotic manufacturing procedures. A lot of the relevant research in the field tries to fix these issues with traditional cameras by including more complicated and expensive supporting sensors [11–13].

New neuromorphic vision sensors, often called event-based cameras, may solve some of the problems with traditional machine vision. Neuromorphic cameras provide computationally efficient perception, lower latency, and a widespread dynamic range since its pixels work individually and react asynchronously to changes in input light in continuous interval. Autonomous driving, UAV management, recognizing objects and tracking, grip detection, mapping and localization, tactile sensing, and a host of other applications benefit greatly from the use of neuromorphic cameras due to their immunity to motion blur and resilience to different lighting conditions. Although neuromorphic cameras have great promise, they pose new problems when it comes to control algorithms and creating perception that work with their unusual and asynchronous output, rendering ineffective long-standing algorithms designed for frame-based cameras. The work demonstrated a basic 2D object avoidance controller based on neuromorphic vision, but it doesn't take the robot's positioning precision into account and can't conduct any machining or manipulation. The recent study shown that neuromorphic cameras are superior to traditional frame-based cameras in a robotics pick-and-place framework for high speeds and uncontrolled illumination operation. The positional mistakes were caused by the event camera's act-to-perceive nature, the lower resolution of the neuromorphic cameras, and the hypothesis of knowing depth. The sub-millimeter accuracy needed for robotic machining operations has not been achieved by any previous research in the field of neuromorphic vision-based control [14, 15].

To accomplish a robot drilling job with submillimeter near precision, design and implement a two-stage controller based on neuromorphic vision in this study. The primary step employs Position Based Visual Serving (PBVS) and a multi-view three-dimensional reconstruction method to precisely locate the target workpiece in 6DoF. In order to fix positional faults, the second control stage uses Image Based Visual Servoing (IBVS) in conjunction with a series of reference holes. The robot uses both control stages to put a clamp mandrel into the reference hole, even though there is less than clearance. Drilling nutplate connection holes on two sides of each reference hole is done by the robot while the clamp mandrel keeps the robotic tool in place. The neuromorphic camera is able to operate at greater speeds and withstand changes in ambient illumination thanks to its capabilities.

This paper is structured as follows for the rest of it. In Section 5.2, the configuration and set-up of the robotic drilling system that is being suggested are detailed. An explanation of how neuromorphic cameras operate and their practical benefits is provided in Section 5.3. This approach for event-based multi-view three-dimensional reconstruction and work localization is described in Section 5.4. Section 5.5 provides an overview of the pipeline for tracking and event-based circular hole identification. The vision-based controller that used both PBVS and IBVS in its two-stage implementation is detailed in Section 5.6. The benefits of using neuromorphic vision for accurate robotic operations are validated by both qualitative and quantitative experimental evaluations of the offered method shown in 7th section. Section 5.8 concludes the whole paper.

5.2 Setup for Robotic Drilling

The robotics nutplate hole drilling system's general configuration is shown in Figure 5.1. An industrial robot's end-effector, which includes a motor for drilling and a neuromorphic visual sensor for seeing and guiding, is the main component of the system. Using a collection of reference holes as a guide, the robot drills nutplate installation holes into a work piece. The following reference frames are defined for the purpose of this paper's discussion of robot control and guidance:

- F_B: The foundation coordinate frame of the robot.
- F_E: The frame that the robot's end-effector is located in.
- F_C: Vision sensor coordinate frame
- F_S: The coordinate frame for the split-pin.
- F_W: The synchronize frame of the work piece.
- F_{h^i}: The reference hole's coordinate frame.

The positioning of a single coordinate frame F_i in relation to alternative coordinate frame F_j is represented by the revolution matrix $j^{R_i} \in \mathbb{R}^{3\times3}$ space. The following expression represents the location of point b compared to fact a specified in the coordinated frame b, where b ∈ R3. The affine transformation matrix $j^{T_i} \in \mathbb{R}^{4\times4}$ is so defined in the following way:

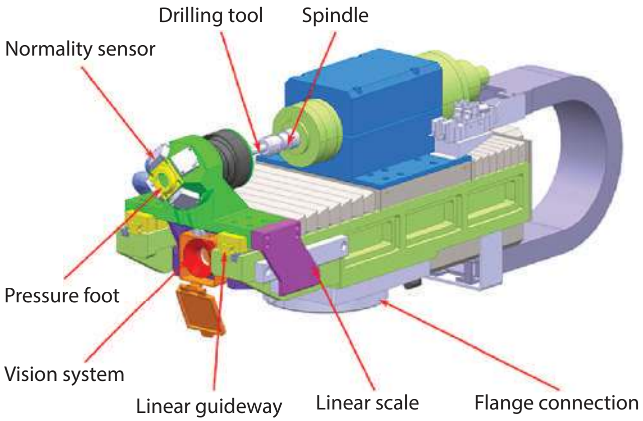


Figure 5.1 Robotics nutplate hole drilling system's general configuration.

$$j^{T_i} = \begin{bmatrix} j^{R_i} & j_j \overrightarrow{P_i} \\ 0^T & 1 \end{bmatrix} \tag{5.1}$$

The transformation matrix B^{T_E}, which changes vectors from $F_E \; to \; F_B$, is assumed to be known from solving the forward kinematics of the robot throughout the rest of this work.

$$B^{T_E} = g(\theta), \theta \in \mathbb{C} \tag{5.2}$$

The robot's configuration space is denoted by $\mathbb{C}$, the determined robot joint angles are denoted by θ, and b (θ) is a function that is nonlinear which represents the robot's kinematics. Also, as mentioned in Section 2.1, a geometrical calibration process may be used to get the constants E^{T_C} and C^{T_S}. Hence, by integrating B^{T_E} with the calibrated transformations, B^{T_O} and B^{T_S} may be simply calculated.

In the same way, the forward kinematics of the robot are used to determine the twist vector $\overrightarrow{V_E} \in \mathbb{R}^6$, which incorporates both the linear and angular components of the velocity, in the following way:

$$\overrightarrow{V_E} J(\theta)^{\dagger} \dot{\theta} \tag{5.3}$$

$$J(\theta) \triangleq \frac{\partial g}{\partial \theta} \in \mathbb{R}^{6 \times N_j} \tag{5.4}$$

In this case, the Jacobian matrices is J (θ), Nb is the total number of robot joints, and † is the Moore-Penrose reverse calculated using Singular Value Decomposition.

5.2.1 Geometrical Tool and Hand-Eye Calibration

The hand eye connection, which is needed by the vision-based controller, is the precise understanding of how the cameras coordinate frame changes in respect to the robot's end effector frame. The total drilling system's repeatability and accuracy are directly impacted by any errors in calculating this transformation. A Hand-Eye calibration procedure is used to approximate this connection.

5.3 A Sensor for Neuromorphic Vision

The neuromorphic visions sensor, which is also called an "event camera," interprets modifications in the visualized scene's illumination as a series of events denoted as $e_k =< u_k, v_k, t_k, p_k$. Here, (u_k, v_k) are the pixel directs of the change, t_k is the timestamp of the incident, and p_k is the polarity of the lights change, which can be either -1 or 1. Neuromorphic vision sensors deviate from traditional frame-based imagers in that they use asynchronous pixel operation to react microsecond-level to variations in logarithmic light rather than a set sampling rate. The disparities between traditional imagers, which produce frames at discrete intervals of time, and neuromorphic event based camera, which produce an event stream in almost unceasing time, are shown in Figure 5.2.

Neuromorphic camera provides several benefits over traditional image sensors due to their operating concept. As an example, neuromorphic cameras are able to detect changes in the scene quickly because of their short latency (on the order of microseconds) and excellent temporal resolution (i.e., they do not have motion blur). Furthermore, unlike frame-based cameras, neuromorphic cameras do not suffer from the exposure timing issues that impact other types of cameras, and their large dynamic limit (>120 dB) is a result of the self-sampling nature of their independent pixels. Because of this, neuromorphic cameras may provide strong vision in low-light situations as well as others. The capacity to see through very tiny apertures, resulting in a broader depth of field, is another practically useful feature of neuromorphic vision that follows from the aforementioned abilities. This feature may eliminate the requirement for a focusing system, which needs extra hardware and introduces uncertainty into the camera projection model, which is problematic in the case since the camera is supposed to receive information over a variable depth. In addition to reducing signal redundancy and using little power, neuromorphic vision has the added benefit of transmitting only useful data in the form of events.

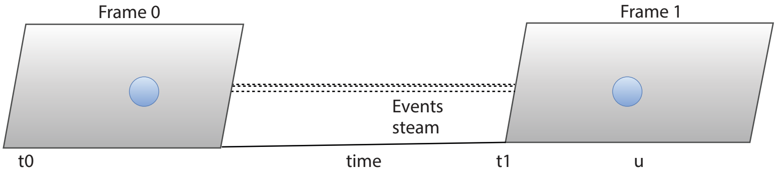


Figure 5.2 Visualization of conventional and neuromorphic camera output.

Neuromorphic cameras are capable, but they need new computer vision algorithms to handle their fundamentally different output, as these cameras cannot be used directly with traditional algorithms for frame-based imaging. The significant price difference between this new technology and traditional cameras is another obstacle to its widespread use. The non-recurring engineering expenditures are mostly responsible for the high price of neuromorphic cameras, thus it is anticipated that the rate will decrease substantially if the skill is mass-produced.

5.4 Multi-View Neuromorphic Event-Based Work Piece Localization

The 6-Dof work piece localization approach, which is based on event based multi-view three dimensional renovation, is presented in this section. Find the position of each reference hole in the work piece by analysing the relevant stream of events from numerous camera perspectives and utilizing the projective geometry of the cameras in conjunction with the Direct Linear Transformation (DLT). Using the space-sweep method, they find that the asynchronous events are correlated with the reference holes. As a last step, they find the proper work piece orientation by fitting models.

5.5 Hole Detection with Neuromorphic Events

The drilling process's positional precision is directly impacted by the visual feedback's exact recognition of circular holes. One of the most well-known techniques in frame-based vision for identifying circular shapes in pictures is the Circle Hough Transform (CHT). The goal is to provide an event-based CHT version applicable to neuromorphic cameras' asynchronous output. Unlike frame-based cameras, neuromorphic cameras provide very different output, making direct usage of CHT with them impossible. Assuming that the edge points are collected from the same picture frame with a precise temporal match, CHT establishes correspondence between them.

Applying CHT to fake frames formed by concatenating events inside a specific time period would be a simplistic and ignorant approach. The pace of change in the visual scene, however, determines the formation of events. Therefore, it would be difficult to decide a universally applicable duration for concatenation event. The findings show that this hypothesis is

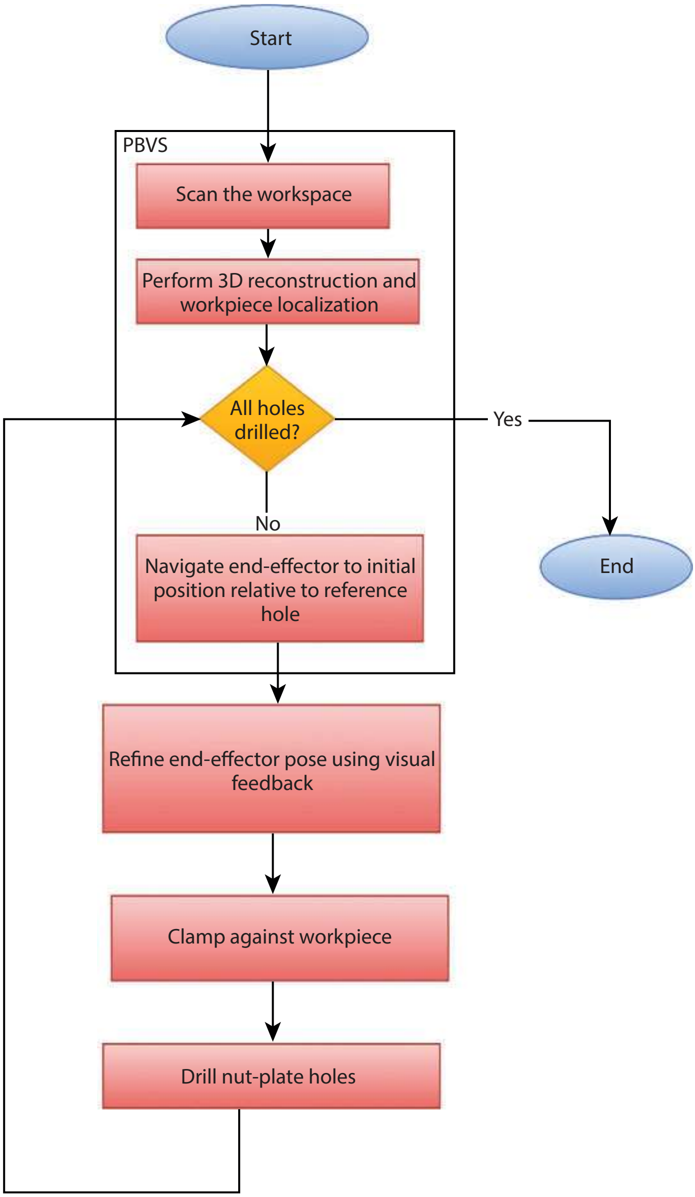

Start
PBVS
Scan the workspace
Perform 3D reconstruction and workpiece localization
All holes drilled?
Yes
No
End
Navigate end-effector to initial position relative to reference hole
Refine end-effector pose using visual feedback
Clamp against workpiece
Drill nut-plate holes

Figure 5.3 Two consecutive position based robotic control (PBVS) and image based robotic control (IBVS) phases.

correct; when ego motion velocities vary, CHT performance is contradictory regardless of the pace of event grouping. In order to overcome these obstacles, they develop a variation of CHT that is event-based and uses a Bayesian framework, allowing neuromorphic cameras to maintain their asynchronous nature.

5.6 Robotic Vision Controller

The perception algorithms used to control the robot's movement throughout the drilling process, and this section illustrates how the vision-based logic for control applies those algorithms. This controller, shown in Figure 5.3, has two consecutive position based robotic control (PBVS) and image based robotic control (IBVS) phases. In order to get the end-effector started in the right direction, the PBVS stage uses the six DoF posture estimation from the multiple view recognition to steer it towards the work piece's reference holes. With the help of the event-based hole detecting algorithm, IBVS achieves end-effector alignment precision down to the millimeter range.

5.7 Results and Experiment Validation

5.7.1 Protocol and Preparation for Experiments

The presentation of event-based robotic drilling techniques was evaluated using the apparatus shown in Figure 5.4 (a). Since it offers an accuracy of 0.1 mm, the UR10 from Universal Robots was the principal manipulator of choice. The Neobotix MPO-500 robotic base was modified and a manipulator was attached to it. The mobile robot can tour the industrial settings independently and position the manipulator near the work piece using two Sick S300 LIDARs as well as the ROS Navigating Stack. As a result, the robot can move autonomously throughout a vast work area and do other drilling tasks. Figure 5.7 (b) showcases the setup of the end effector, which includes the drilling motor with the cameras. The camera DAVIS346 from Inivation is utilized for visual perception. It offers both traditional frame-based intensity pictures and a neuromorphic event stream, thanks to its 346 × 260 spatial resolution. The event stream provides a 120-dB dynamic range, 20 μs latency, and 12×10^1 events/s bandwidth. Evaluate the performance and benefits of event-based perception using intensity pictures

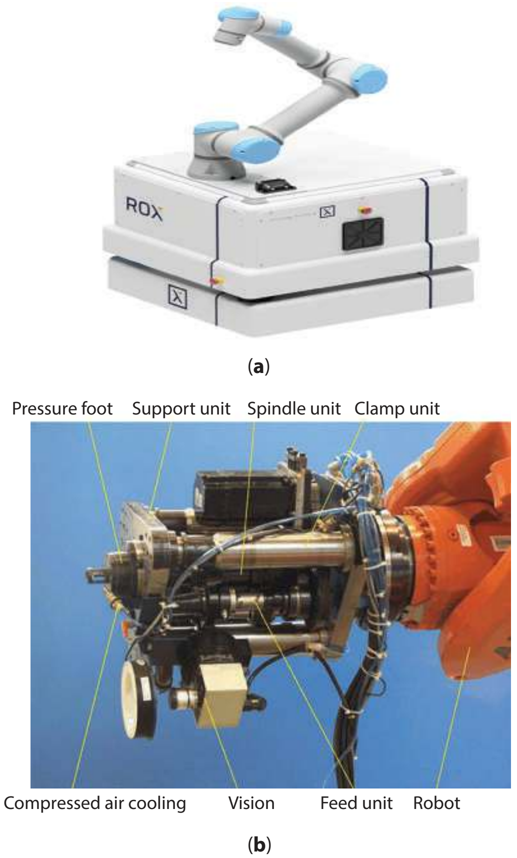


Figure 5.4 (a) Mobile manipulator robot. (b) Visual sensor and drill motor end-effector arrangement.

as a baseline, even if all operations are executed exclusively utilizing the event stream. The necessary calculations are carried out by means of an integrated computer equipped with an i7-5530 CPU with four gigabits of RAM.

Drilling nutplate holes with a certain degree of positioning inaccuracy allows us to measure the system's drilling performance. Here, mean the discrepancy between the intended and actual locations of the drilled holes as the Euclidean distance. Everything is specified in relation to the closest reference hole, including measurements and mistakes. Figure 5.5 shows the precise position of the drilled hole is determined using the Helmel Phoenix Coordinate Measurement Machine.

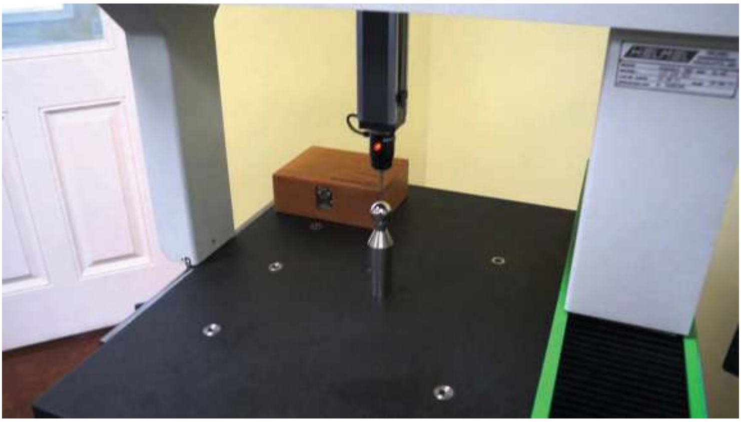


Figure 5.5 Helmel Phoenix coordinate measurement machine.

5.7.2 Localize 6-DOF Work Piece

In this part, they measure how well the suggested event-based multi-view reconstruction method, which gives preliminary guesses of the work piece and reference hole positions, performs. A quartet of ArUco fiducials are used to get ground truth records. From a resting robot position, the fiducials are seen via the DAVIS346's intensity picture output, and the 6-DoF posture of every fiducial is approximated with the help of Open CV's ArUco package.

Using conventional intensity pictures as a baseline, compare the multiple view localization findings acquired from the neuromorphic stream event. Apply the same multi-view technique to conventional photos after extracting features from each one using the typical canny edge detector. The benefits of neuromorphic vision may be more accurately evaluated by doing the multi-view localization studies in a range of illumination conditions and with varying scanning rates throughout the workspace. Figure 5.6 shows a summary of the findings for the conventional event-based and traditional image-based techniques, broken down by position error across all experimental conditions.

Figure 5.6 shows that both conventional and neuromorphic vision multi-view localization provide comparable accuracy under ideal illumination and lower speeds. With an increase in the robot's operating speed or in low-light situations, the benefits of neuromorphic vision became obvious. Under these circumstances, it becomes difficult to perceive with traditional cameras because of the motion blur and significant latency caused by the extended exposure time. More accurate and trustworthy 3D localization

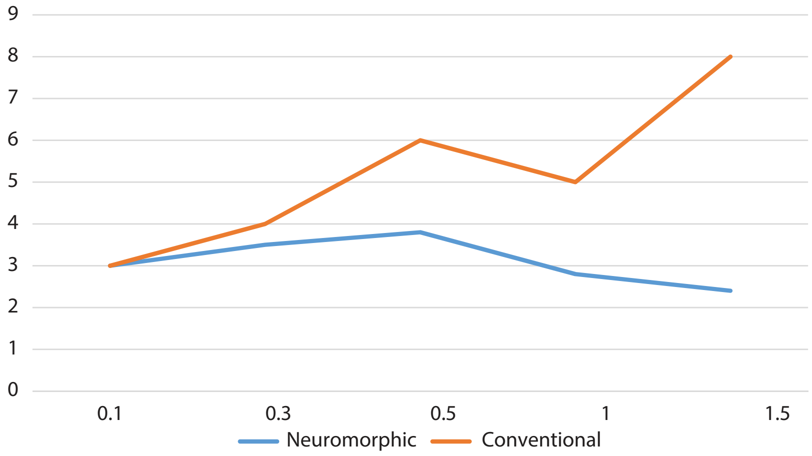


Figure 5.6 Conventional and neuromorphic vision multi-view localization.

findings are produced by neuromorphic cameras as they do not experience these limitations. Confirming the theory behind neuromorphic vision's higher performance.

5.7.3 Finding Neuromorphic Holes

In comparison to the traditional CHT, the circular detection based on the event and tracking approach is assessed in this section. Using intensity photographs and false image frames created by merging events at a predefined time interval, they evaluate the standard CHT. Using varying degrees of illumination and rates of movement, they compare the methods' capacity to follow the work piece's round holes. Outstanding to motion blur in the graphical world, which is induced by the operating principle of traditional pictures that depend on the temporal combination of input light, the usage of intensity pictures results in unpredictable detection at greater ego motion speeds. When working with event frames and traditional CHT, the exact duration in which events are gathered becomes important. A featureless picture is produced at slower rates by using a short concatenation period; an overpopulated image is produced at high speeds by using a big period, making precise feature extraction impossible. By considering the event stream as asynchronous, the event-based CHT takes use of neuromorphic vision. Because of this, it produces accurate findings even when the scene is in motion, and it avoids the problems associated with latency and motion blur that are present in traditional cameras.

Figure 5.7 (a) and (b) displays the experimental findings that support the benefits of neuromorphic vision in conjunction using the suggested CHT based on the event. These tests compare the tracking capabilities of several approaches by progressively increasing the camera's speed. To determine how well each approach held up, conducted experiments under varying light levels. In order to identify circular holes in intensity images, the famous Kanade-Lucas-Tomase (KLT) tracker was combined with traditional CHT. Contrary to expectations, intensity picture-based tracking degrades with increasing speed owing to motion blur and completely misses the hole in low-light conditions despite longer exposure times, all because the image is too blurry. In contrast, these differences do not affect neuromorphic vision-based perception.

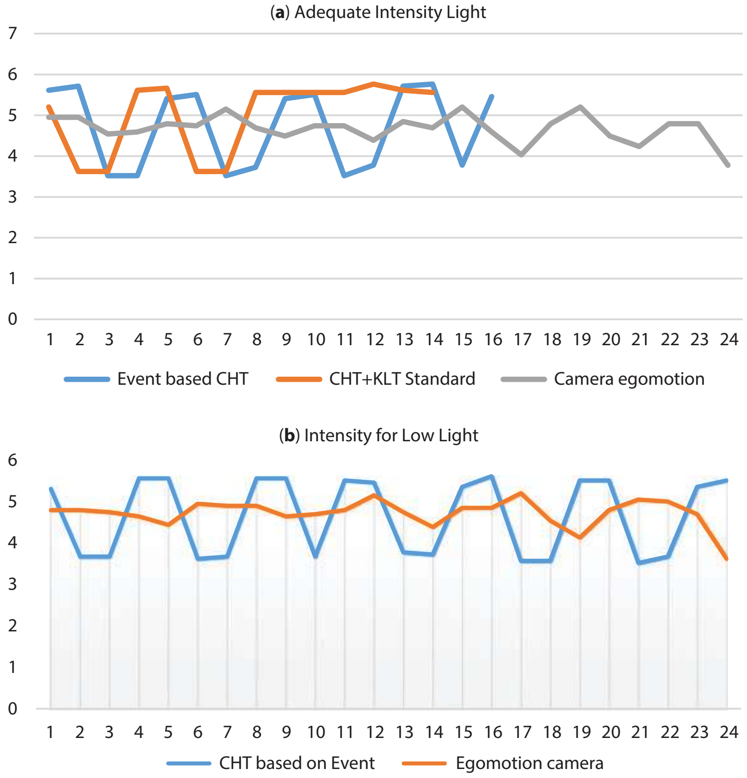


Figure 5.7 Neuromorphic vision in conjunction with the proposed event-based CHT.

5.7.4 Performance Drilling Nutplate Holes

This section presents the results of the entire procedure of drilling the nutplate holes. With five work pieces positioned in various ways throughout the environment, tests were carried out utilizing the arrangement. Before controlling the tool to align the robotic drill end effect with the intended reference holes on the work piece, the innovative neuromorphic vision pipeline ensures that the mobile robot independently negotiates to the front side of each work piece. After the drilling bit is in line with the mark, the clamp mandrel is put into the mark with a smaller gap. After a certain amount of force has been applied to the workpiece, the pressure foot will clamp down on it. Torques applied to the robot's joints provide an approximation of these contact forces. Using the robot manipulator's inherent compliance, this two-sided clamping eliminates normalcy mistakes and provides stability throughout the drilling operation. Drilling holes for nutplate fitting on both sides of the reference hole is the next step for the robot after initiating the drill motor.

The quantitative findings demonstrate that the neuromorphic vision-based method suggested may accurately place the robot's drilling end-effector within a certain range. The findings are in line with the stringent accuracy standards of several procedures in the aerospace and automobile production sectors. This proves that neuromorphic vision is useful for accurate industrial operations and shows that neuromorphic cameras might make automated production more dependable and quicker. The suggested methods successfully use neuromorphic cameras' benefits while mitigating some of their drawbacks, such as their unusual data output and limited resolution, as shown by the findings. The drilling procedure, which involves placing the clamping mandrel in a pilot hole, reduces positional errors by aligning the end-effector with the reference hole through manipulator compliance. However, the clamp mandrel inserting procedure can still lead to errors.

5.8 Conclusion

New neuromorphic vision technology has just emerged, and in this work we introduced the first system to use it for robotic milling. In instance, by combining two successive perception and control phases, we have created a comprehensive visual guiding system that places the robot in relation to the target work piece with sub-millimetre precision. In the first step, the robot's end-effector is aligned using PBVS and a multi-view three-dimensional reconstruction method.

Using an innovative event-based hole identification method in conjunction with IBVS, the second stage simultaneously controls any remaining defects. Together with an innovation neuromorphic camera, a bespoke end-effector, and an integrated robot manipulator, they have conducted experimental validation of the system for a nutplate hole drill presentation. The quantitative findings demonstrate that the given neuromorphic guiding system based on vision can accurately place the drilling end-effector of a robot within a wide range of errors. Additionally, these tests confirm that neuromorphic cameras are superior to traditional frame-based cameras in terms of overcoming issues with illumination, speed, and motion blur. The findings show that neuromorphic cameras may be used in precision manufacturing to make production lines quicker and more dependable.

The absence of a normalcy measurement and control mechanism is one of the existing system's limitations. Due to the fact that the multi-view reconstruction stage is the only means by which work piece orientation is measured, any orientation errors that may arise as a consequence of this step cannot be controlled. Problems with inconsistent hole alignment, as occurs with non-planar work pieces, would make these limits more obvious. The system's ability to perform a wider range of manufacturing operations is limited without a more precise and reliable method of normal alignment, even though the collaborating robot's complying and two-sided clamping can passively drive the end-effector toward greater work piece routine.

References

1. Ayyad, A., *et al.*, Neuromorphic vision based control for the precise positioning of robotic drilling systems. *Rob. Comput. Integr. Manuf.*, 79, 1024195, 2023.
2. Salah, M., *et al.*, High speed neuromorphic vision-based inspection of countersinks in automated manufacturing processes. *J. Intell. Manuf.*, 35, 3067–3081, 2024.
3. Rast, A.D., *et al.*, Behavioral learning in a cognitive neuromorphic robot: An integrative approach. *IEEE Trans. Neural Netw. Learn. Syst.*, 29, 12, 6132–6144, 2018.

4. Konstantinidis, F.K., *et al.*, A technology maturity assessment framework for industry 5.0 machine vision systems based on systematic literature review in automotive manufacturing. *Int. J. Prod. Res.*, 11, 4, 1–37, December 30, 2023.
5. Diesing, G., How AI and Machine Vision Intersect. *Quality*, 61, 2, 14–14, 2022.
6. Zhang, J. and Tao, D., Empowering things with intelligence: a survey of the progress, challenges, and opportunities in artificial intelligence of things. *IEEE Internet Things J.*, 8, 10, 7789–7817, 2020.
7. Reuther, A., *et al.*, Survey of machine learning accelerators. *2020 IEEE high performance extreme computing conference (HPEC)*, IEEE, 2020.
8. Faris, O., *et al.*, Design and experimental evaluation of a sensorized parallel gripper with optical mirroring mechanism. *Mechatronics*, 90, 1029555, 2023.
9. Camarillo, D.B., Krummel, T.M., Salisbury Jr., J.K., Robotic technology in surgery: past, present, and future. *Am. J. Surg.*, 188, 4, 2–15, 2004.
10. Feng, G., *et al.*, Retinomorphic hardware for in-sensor computing. *InfoMat*, 5, 9, e12473, 2023.
11. Naeini, F.B., *et al.*, Event augmentation for contact force measurements. *IEEE Access*, 10, 123651–1236605, 2022.
12. Mei, B., *et al.*, Positioning accuracy enhancement of a robotic assembly system for thin-walled aerostructure assembly. *J. Ind. Inf. Integr.*, 35, 1005185, 2023.
13. Etienne-Cummings, R. and Van der Spiegel, J., Neuromorphic vision sensors. *Sens. Actuators, A*, 56, 1–2, 19–29, 1996.
14. Shah, U.H., *et al.*, On the design and development of vision-based tactile sensors. *J. Intell. Rob. Syst.*, 102, 1–275, 2021.
15. Zaid, I.M., *et al.*, Elastomer-based visuotactile sensor for normality of robotic manufacturing systems. *Polymers*, 14, 23, 5097, 2022.

6

Design and Development of AI Neuromorphic to Control the Autonomous Driving System

J. Balamurugan[1]*, Mohammed Mahaboob Basha[2], Mamatha Bai B. G.[3], J. A. Jevin[4], Rakesh Bharti[5] and R. Senthamil Selvan[6]

[1]Department of Master of Business Administration, St. Joseph's College of Engineering, OMR, Chennai, India
[2]Department of ECE, Sreenidhi Institute of Science and Technology, Hyderabad, India
[3]Department of Computer Science and Engineering, Nitte Meenakshi Institute of Technology, Bangalore, India
[4]Department of CSA, KL University, Vaddeswaram, India
[5]Department of Physical Education, Lovely Professional University, Phagwara, Punjab, India
[6]Department of Electronics and Communication Engineering, Annamacharya Institute of Technology and Sciences, Tirupati, India

Abstract

One of the most defining features of AI is autonomous driving. By using energy-efficient computing frameworks based on spiking neural networks, neuromorphic (brain-inspired) control has the potential to make a substantial contribution to autonomous behaviour. Neuromorphic versions of four well-known independent driving controllers—Stanley, Pursuit, PID, and MPC—were investigated in this study utilizing a physics-aware simulation framework. The models' performance was compared with that of traditional CPU-based implementations and conducted thorough evaluations using a wide range of intrinsic characteristics. Provide instructions for constructing control-oriented neuromorphic structures and highlight the significance of the tuning parameters and neural resources that make them tick. The findings indicate that a small number of neurons—100 to 1,000—would be sufficient for the majority of models to achieve peak performance.

**Corresponding author*: drjbalamuruganpdf@gmail.com

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (87–104) © 2025 Scrivener Publishing LLC

As was proposed with the MPC controller, they similarly emphasize the implication of hybrid conventional and neuromorphic systems. In this case, the MPC controller, also shows how important it is to combine conventional and neuromorphic architectures. Mostly at higher speeds, where they incline to deteriorate quicker than in traditional enterprises, this research also shows the limits of neuromorphic implementations.

Keywords**:** PID, MPC, spiking neural networks, neural resources, neuromorphic

6.1 Introduction

Autonomous driving systems rely heavily on path and motion planning. Methods from classical control theory and machine learning are also included in ADSs. The three primary parts of an ADS are the following: route planning, path tracking, and environment sensing and localization. ADSs usually need substantial energy and computing resources, and they are implemented via problem formulation and optimization criteria like vehicle and safety. A potential replacement for traditional system control that is both accurate and fast is neuromorphic brain-inspired controller schemes built on tightly linked spiking neural networks (SNNs). The proposal here is to include four existing path-tracking control models for independent driving into a computational framework that takes physics into account in a neuromorphic fashion. Suggested ADSs employ a LiDAR sensor to approximatively determine the vehicle's track position. Use LiDAR data to produce a reference path [1]. (1) Used neuromorphic methods and the following controllers to keep track of paths: unrestrained yearning A well-liked controller for tracking paths that follows a point on a reference trajectory geometrically by adjusting the steering angle; (2) reducing heading and cross-track error with steering is autonomous driving's Grand Challenge. An angle among the vehicle and trajectory heads, and a distance among the front axle and the nearest reference path point; (3) the proportional-integral-derivative (PID) controller, a popular controller that is used to continually reduce the CTE; (4) Model predictive control (MPC) optimizes the control strategy based on an evaluation of the system's future state [2–4]. These control models are commonly used in reliable strategies and offer comfortable, explainable, safe, and interpretable motion control, in difference to conventional artificial neural network-based controllers. The former optimizes policies in the long run, but the latter can have harmful, unexplained consequences in the short term [5–10].

The neuromorphic implementations use the neural engineering outline, a popular neuromorphic computation framework. NEF includes mathematical constructs that allow spiking neurons to encode, decode, and transform numerical values, which makes it easier to implement functional large-scale neural networks [11–15]. From visual processing to perception and robotics control, NEF has been used in the proposal of several neuromorphic systems. In addition, the framework has been shown to work on well-known digital neuromorphic hardware architectures, such as TrueNorth, the SpiNNaker, the NeuroGrid, and the Loihi. It has also been used on dedicated analogue circuits to translate functional descriptions from high-level languages to low-level neural models, and it is based on NEF principles [10–16].

Evaluating neural approximation approaches for ADS control and providing guidance for neuromorphic control strategy design are the goals of this study [17–24].

6.2 Methodology

The neuromorphic designs are based on NEF, which is introduced here. This section introduces the kinematic bicycle model (KBM) that was used to represent the vehicle, as well as NEF, the theoretical basis for the neuromorphic designs. The KBM utilized to simulate the vehicle is also introduced. Next, describe in depth the research's simulated setting and path-tracking controllers. With the goal of path straggling supervisors being to qualify an independent vehicle to adhere to an orientation route with a minimum of mistakes and maximum performance, the simulation environment is designed to evaluate and compare the different managers' abilities (Figure 6.1).

6.2.1 An Architecture for Neural Engineering

Representation, transformation, and dynamics are the three tenets laid forth by the NEF for the construction of neuromorphic spiking neural networks.

6.2.1.1 First Principle—Image

The representation of a real number vector that changes over time may be achieved by a network of neurons using nonlinear encoding and linear decoding. Representative numerical structures as spikes are the

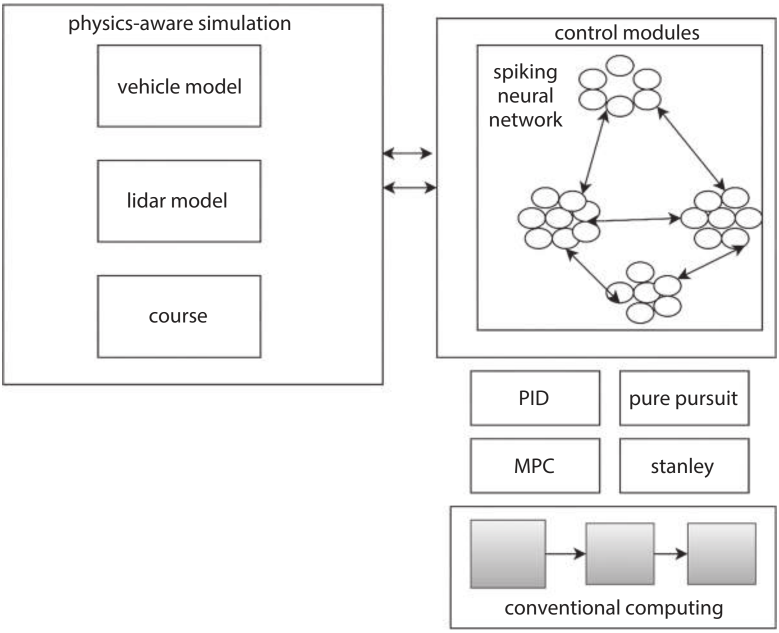


Figure 6.1 Systems schematic. With LiDAR (green lines), Air Sim's SUV perceives the FST Driverless Environment. A control signal is calculated by the module.

responsibility of the encoder. The equation (1) gives the training of an input vector x as: $\delta i(x) = Gi\ h\alpha ih\varphi\tilde{}i, x(t)i + Jibiasi$. Ji is a bias secure current, φ˜ is the neuron's favoured incentive, α is a gain term, and Gi is a nonlinear purpose that reflects a neuron model. With the help of a linear decoder, the predicted represented vector xˆ may be revealed.

$$x\hat{}(t) = 6i\big(hi(t) * \delta i(t)\big)dix$$

The linear decoders, represented by di, were tuned using least squared optimization to accurately replicate x. The filter, represented by hi, convolved with δi to reflect the spiking movement.

6.2.1.2 *Principle 2—Metamorphosis*

The cryptographer's di may be tuned by least-squared optimization to accurately recreate any given function f(x). Similarly, in the second case, the function fˆ(x) may be decoded using:

$$_{f}\,f\hat{}\left(x(t)\right)=6i\left(hi(t)*\delta i(t)\right)di$$

The calculation of f(x) involves the use of a collection of weighted synaptic connections wij, which provide a link between two neuronal ensembles A and B.

$$f(x)=wij=di\cdot ei$$

The decoders of ensemble A are represented by di, whereas the encoders of ensemble B are represented by ej. This enables the neurocomputational approximation of any function inside a spiking neural network (SNN).

6.2.2 Kinematic Bike Model

To simulate the steering of the four-wheeled vehicle, use the KBM. When describing a vehicle's motion, the KBM a simplified model is often utilized in control and robotics applications. It gives the impression that the car is a rigid body with a set wheelbase connecting the front and back wheels. Here, x and y stand for the car's location, Θ and δ are the heading angle and steering angle, respectively, and KBM expresses the automobile's state as [x, y, δ, θ]. The pair [v, ϕ] represents the vehicle's speed and steering rate, respectively, and is used as input to the model. In two-dimensional space, the location of the vehicle was shown concerning either the front axle centre, the rear axle centre, or the centre of gravity. The vehicle's current condition may be determined by referring to the centre of the rear axle and then doing the following:

$$x(t+1)=x(t)+v\ cos\left(\theta(t)\right)1t$$

$$y(t+1)=y(t)+v\ sin\left(\theta(t)\right)1t$$

$$\theta(t+1)=\theta(t)+v\tan\delta(t)1t,\delta(t)=\delta(t)+\phi 1t$$

The vehicle's current condition, measured from the center of the front axle, is:

$$\theta(t+1)=\theta(t)+v\ sin\ \delta(t)1t,\delta(t)=\delta(t)+\phi 1t$$

Extensive explanation, including a rundown of the model's benefits and drawbacks.

6.2.3 Detectors of Paths

Essential to autonomous vehicle systems, path-tracking controllers provide orders for the vehicle's direction-finding and accelerator to follow a reference route. Here, takes a look at the Pure Pursuit, MPC, Stanley, and PID controllers. The design, settings, and performance of each controller is distinct from the others. Below, provide a quick overview of each controller and then detail how they are implemented meromorphically.

6.2.4 Virtual Setting for Simulation

To recreate the feel of racing a remote-controlled car around a real track, turn to Microsoft's Air Sim simulator, an open-source, cross-platform framework built on the physics-aware Unreal Engine. the FST Driverless Environment as a foundation for the racing course and modified it by adding solid, wall-like sides that ran along a road that was fifteen meters wide (Figure 6.2). With the help of Air Sim's realistic SUV model and a LiDAR sensor, were able to cover an area of 180 degrees with a resolution of half a degree, a variety of 40 meters, and a scanning rate of 40 scans per second.

Python was used to build the traditional CPU-based controllers, while the Nengo library was used to create the Neuromorphic NEF-based controllers. To facilitate the transfer of control and LiDAR measurements, develop an adaptor that synchronizes communication between the Nengo and the Air Sim environments. To simulate 200 control signals per second, scenarios were performed in synchronization with a 5-millisecond interval. A reference trajectory is necessary for driving policies. In this

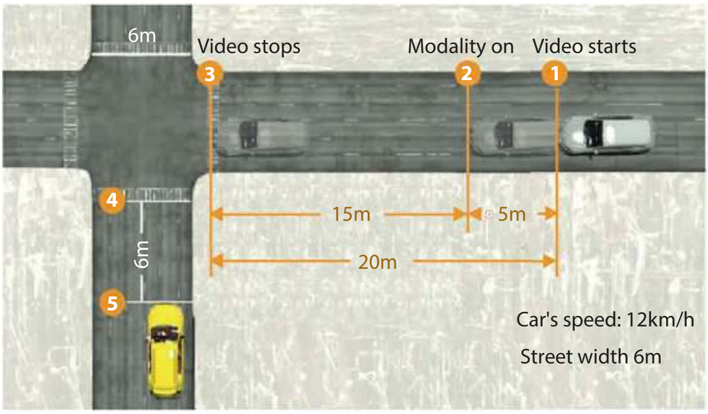


Figure 6.2 An aerial picture of the autonomous environment at FST.

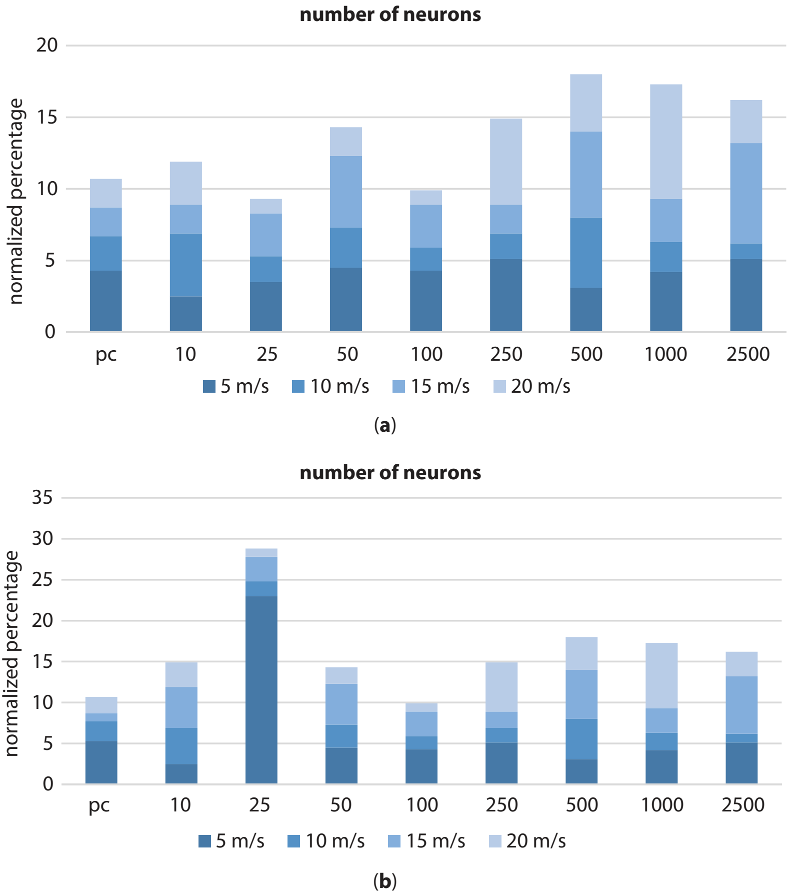
case, refer to the race sequence's mid-line. The route was shown using a third-degree multinomial and was produced in real time based on LiDAR data, which detected the walls along the racing course. The centre point amongst the borders was calculated at regularly spaced intermissions to obtain this polynomial. repeat each experiment ten times and maintain a fixed seed worth for the accidental machine to minimize any non-deterministic influence. With SNNs being inherently time-dependent, the architecture also had to deal with time-space organization between the Nengo and Air Sim simulants. So that Nengo and Air Sim can communicate in real-time, built a software adaptor layer. This adapter allows for the two-way transmission of control signals, which include the standards of the car's navigation and accelerator, and LiDAR data. To simulate 200 control signals per second, timed these situations so that they run simultaneously with a minimum spacing of 5 milliseconds. For each 5 milliseconds, one simulator will run while the other stays still; this procedure essentially alternates between the two simulators. An astronomy delta time of 0.001 s was used to calibrate the Nengo simulator, guaranteeing an incredibly precise simulation. The use of the Air Sim API, which is essentially based on a TCP socket connection, allowed Nengo and Air Sim to communicate with each other.

6.3 Results

The findings of this study apply to a wide variety of traditional and neuromorphic models. The pure detection, MPC, Stanley, and PID organizers were constructed conservatively using a CPU and neuro morphically using ensembles of spiking neurons. Four matrices were used to assess the performance of each model: average velocity, collision-free drives %, collision-to-finish percentage, and percentage of completed drives. To get a better understanding of what resources are needed for proper performance, test the neuromorphic implementations with different numbers of neurons. The controllers' efficiency changes when the goal velocities change. The findings of this study apply to a wide variety of traditional and neuromorphic models. The MPC, pure pursuit, and Stanley controllers were constructed utilizing either neuromorphic ensembles of spiking neurons or the more traditional usage of a central processing unit (CPU). Four matrices were used to assess the performance of each model: average velocity, collision-free drives %, collision-to-finish percentage, and percentage of completed drives. To get a better understanding of what resources are needed for proper performance, test the neuromorphic implementations with different numbers of neurons. The controllers' efficiency changes when the goal velocities change.

6.3.1 Controller for Pure-Pursuit

The results showed that in every test situation, the CPU-based model successfully finished the race sequence. The neurological design could likewise finish a lap in every situation, with the exclusion of a few efforts to drive the car at an elevated velocity of 20 m/s (Figure 6.3a).



(**a**)

(**b**)

Figure 6.3 Findings from a controller that does nothing but pursue. Results from experiments that managed to finish the lap, including their CTE (RMS) and average velocity. As a point of comparison, the scattered line represents the regular outcome of the CPU operation. (a) Drives that have been finished. (b) Roads devoid of collisions.

(*Continued*)

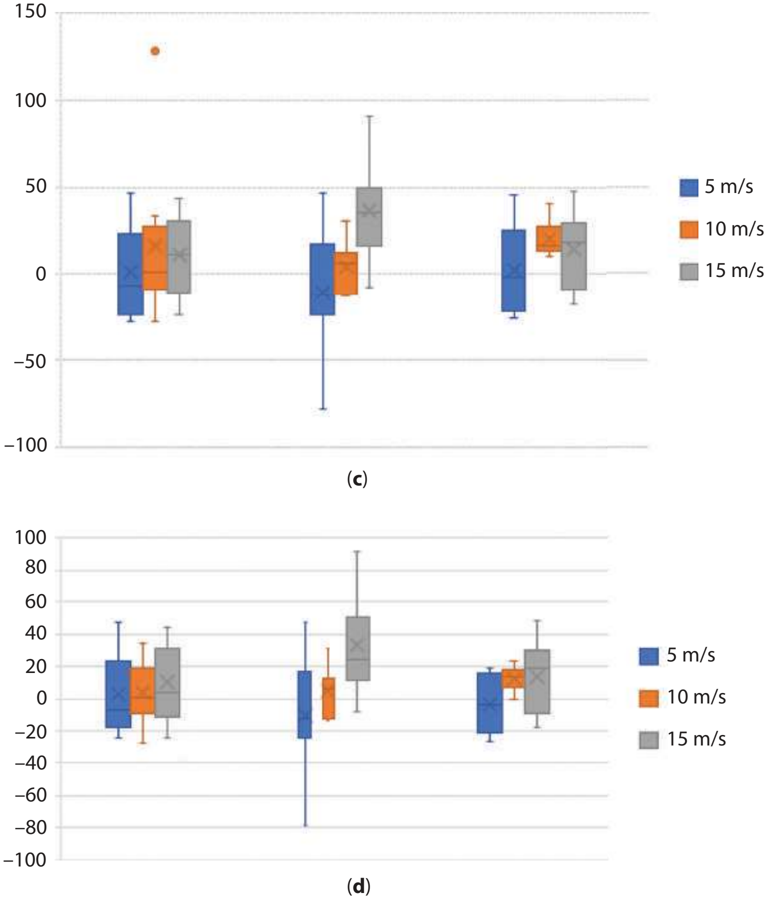


Figure 6.3 (Continued) Findings from a controller that does nothing but pursue. Results from experiments that managed to finish the lap, including their CTE (RMS) and average velocity. As a point of comparison, the scattered line represents the regular outcome of the CPU operation. (c) RMS stands for cross-track error. (d) The mean speed.

The controller's use of synaptic time constants on the millisecond scale is consistent with this outcome. Both the neuromorphic and CPU-based systems collide with the roadside barriers at high speeds. The findings show that the neuromorphic-controlled vehicle was able to contact the walls at lower speeds with less than 100 neurons, highlighting the significance of neural resources for directing the vehicle at high speeds (Figure 6.3b). According to the findings, in both CPU and neuromorphic applications, the root-mean-square (RMS) of the CTE increases with increasing speed. It seems that there is an optimal allocation of neural resources and efficiency, as the neuromorphic implementation's CTE performance converged at 100 neurons and more neurons had little effect on its performance. At slower speeds and with enough neurons, the neuromorphic model was just as effective as the CPU-based approach (0.12 to 0.23 m). Figure 6.3c shows that the neuromorphic implementation's CTE performance surged to 0.66 meters at higher speeds of 15–20 m/s. The CPU-based resolution realized its goal, with the exclusion of a high 20 m/s board speed, but the neuromorphic design was less accurate (Figure 6.3d).

The designated synaptic time constant (τ) is a crucial part of the neuromorphic architecture. In addition, used a 100-neun setup with a goal speed of 15 m/s and temporal constants ranging from 2 to 1,100 ms to evaluate the model's performance. It is shown in Figure 6.4 that the vehicle was unable to react quickly enough to effectively finish the race course when τ > 10 ms. Figure 6.4 shows that the CPU-based solution had a lower CTE (1.68 m) than the neuromorphic version (2.02 m), with a time continuous of 5 ms and a goal speed of 15 m/s. The high CTE is caused by the pure-pursuit perfect violently routing at greater speeds in both the CPU and neuromorphic applications. At slower rates, however, the neuromorphic version can hold its own against the CPU implementation, all with only 100 spiking neurons.

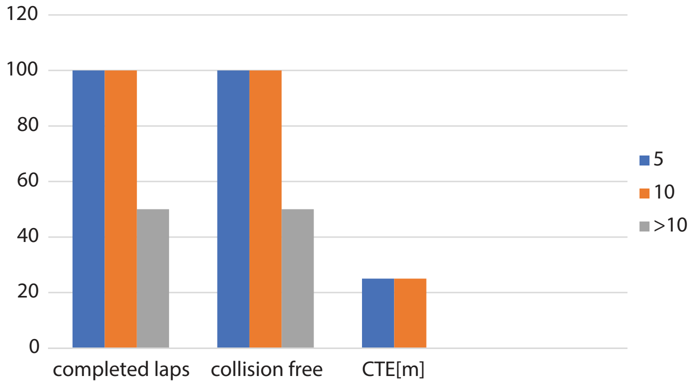
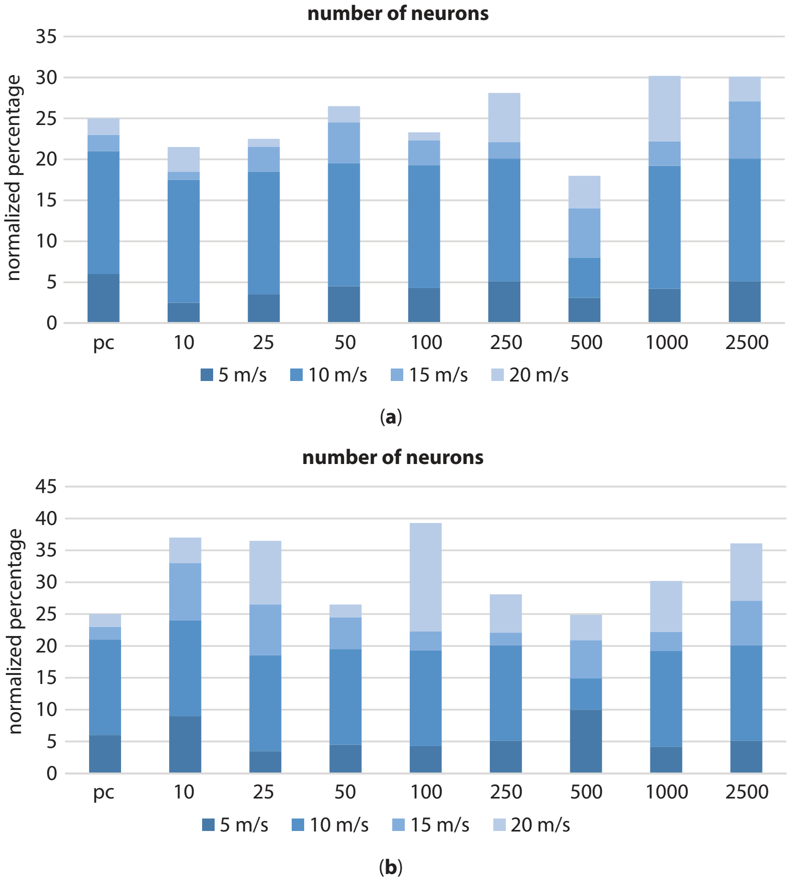


Figure 6.4 1 results for pure-pursuit neuromorphic implementations under various τ values.

6.3.2 Stanley Controls

A neuromorphic PID controller is used to find the target speed of the vehicle, and a single neuron ensemble is used to solve this equation in the neurons Stanley controller architecture, much as in the pure pursuit model. The output synapse time constant is typically set at 10 ms.

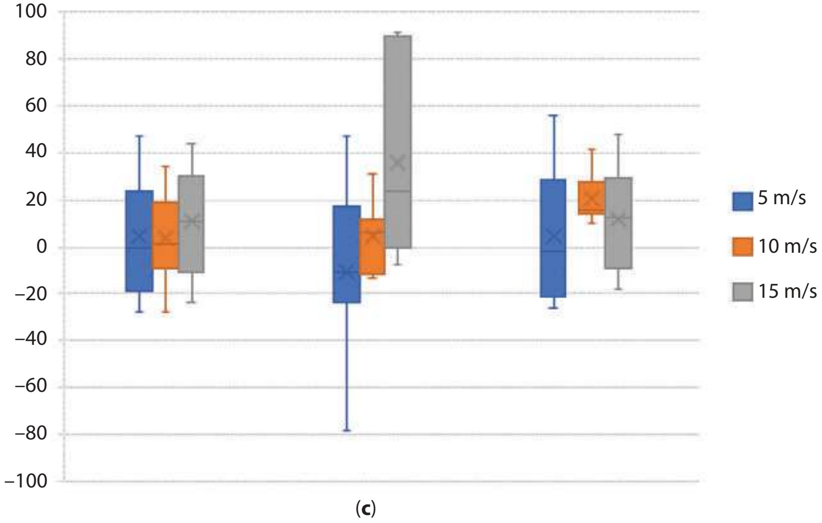
Figure 6.5a shows that in every test situation, both the CPU-based solution and the neuromorphic architecture completed the race course. In all trials, the neuromorphic operation did touch the limits at 20 m/s, but the CPU model finished every lap without a hitch. On the other hand, when



(**a**)

(**b**)

Figure 6.5 Data from the Stanley controller. The average speed and CTE (RMS) of the experiments that finished the lap. As a point of comparison, the spotted line represents the average outcome of the CPU application. (a) Runs taken and finished. (b) Road trips without accidents. (*Continued*)



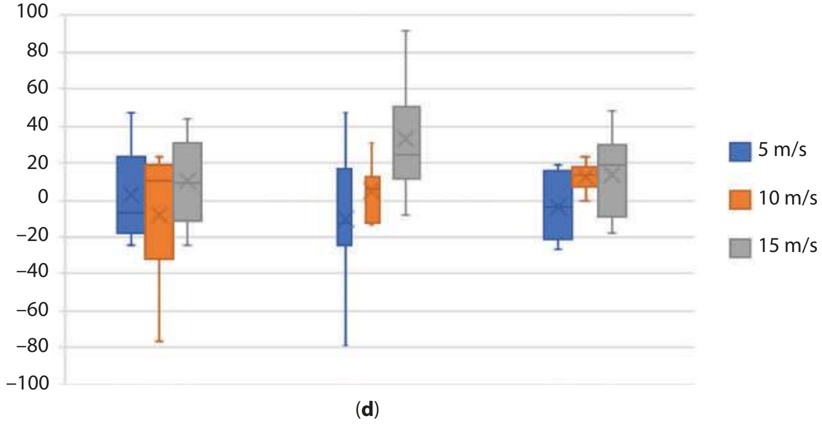


Figure 6.5 (Continued) Data from the Stanley controller. The average speed and CTE (RMS) of the experiments that finished the lap. As a point of comparison, the spotted line represents the average outcome of the CPU application. (c) RMS = cross-track error. (d) The average speed.

driven by more than 1,000 neurons, it managed to avoid collisions at 15 m/s (Figure 6.5b). It was not surprising that CTE performance improved with increasing speed in both neuromorphic and CPU implementations. The neuromorphic controller with 1000 neurons ran at a slow 5 m/s, but its CTE of 0.90 m was much better than the average CPU effectiveness of

0.92 m. With each additional neuron added to the neuromorphic architecture, the performance gap between it and the original CPU's CTE widened (Figure 6.5c). To maintain the reference trajectory, the automobile drove forcefully with rapid twists and Zig-Zag movement patterns, even though it had a low number of neurons, as can be seen by closely examining the vehicle's dive course. This is because it can't properly implement the Stanley driving equation. By 10,000 neurons, the driving route of the car was far more refined, resembling the implementation shown in the CPU. Nevertheless, there was a noticeable departure from the reference course. Figure 6.6 shows that the organizer could move the car pretty easily with modest Zig-Zag designs using 2,500 neurons, according to the computed trajectories.

In most cases, the findings indicate that implementations based on neuromorphic architectures or central processing units were able to keep their goal speed. The accuracy of our neuromorphic systems, nevertheless, was somewhat lower. The vehicle achieved an average speed of 19 m/s with more than 1,000 neurons, and a goal speed of 15 m/s with the same number of neurons (Figure 6.5d).

In addition, examine how the model's presentation was affected by the synaptic time constant τ using 1000 neurons and a goal speed configuration of 15 m/s. The relevance of a quick reaction time in neuromorphic systems is shown by the fact that the controlled car failed to successfully finish the race course when using a neuromorphic organizer with a

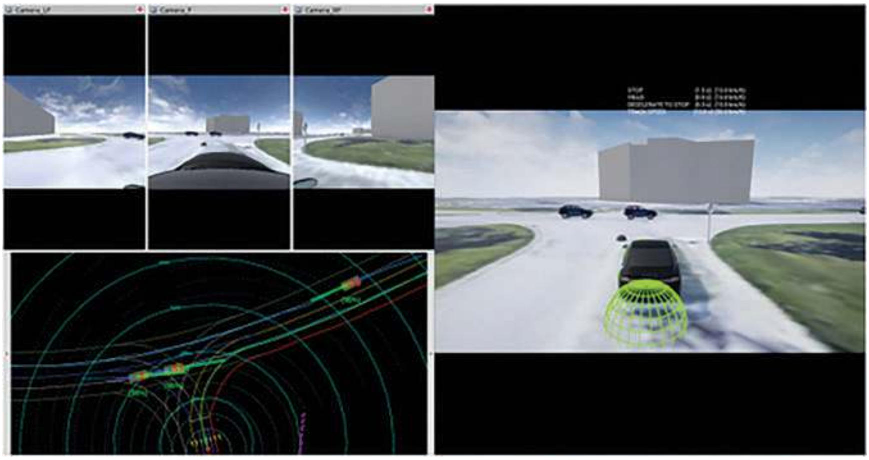


Figure 6.6 Neuromorphic Stanley controller with 2,500 neurons introduces Zig-Zag patterns.

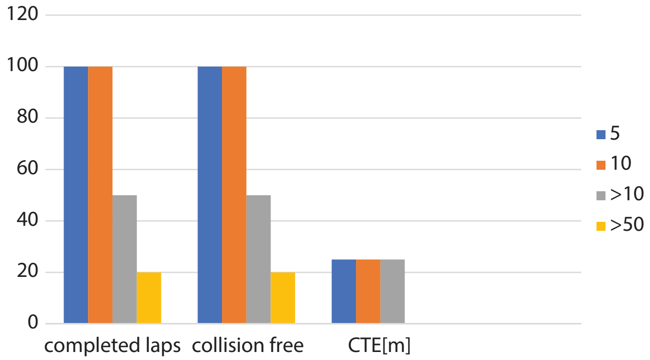


Figure 6.7 Results for Stanley neuromorphic implementation with varying τ values.

τ > 50 ms. When the time continuous was 60 ms, 80% of the creativities were collision-free. However, when the time constant was less than or equal to 10 m, all of the initiatives were collision-free (Figure 6.7). With a minimum of 2,500 neurons needed for smooth driving, the neuromorphic Stanely controller performed well overall, but with reduced accuracy at 20 m/s.

6.4 Discussion

Neuromorphic control shows promise for autonomous driving, according to the tests. Demonstrate that, especially at slower speeds, the neuromorphic versions of the Pure pursuit, PID, MPC, and Stanley controllers could compete with their CPU-based equivalents. A more energy-efficient alternative to traditional methods in autonomous driving might be neuromorphic control systems, as seen below. The results of the studies indicate that there seems to be an optimal allocation of neuronal resources and efficiency, where limits such as the synapse time constant and the total number of neurons are fine-tuned to bring about the best possible results. As an example, the findings reveal that the Stanley controller needs over a thousand neurons to achieve convergence, in contrast to the neuromorphic Pure-pursuit organizer, whose performance joins at 100 nerve cell. In contrast to PID and MPC controllers, which are susceptible to time-constant modifications, they use fewer neurons (10 neurons per ensemble). Furthermore, the findings show that the controller's responsiveness to changes in the environment may be greatly affected by the choice of synaptic time constant. At target velocities between 0 and 15 m/s, all neuromorphic models worked well; but, at greater speeds, their performance suffered. The study also heavily emphasizes the usage of hybrid neuromorphic-CPU controllers, such as the MPC application. An exciting new direction for autonomous driving study is the development of hybrid organizers that combine the best features of neuromorphic and conventional computer systems.

Although there were some encouraging findings, there are still several questions that need answering. At higher speeds, in particular, research shows that neuromorphic systems have their limits. Neuromorphic controller performance degrades more rapidly than CPU implementation at high target velocities, leading to less precise velocity control and more cross-track errors. When using rate-coded neuromorphic representation, this outcome is expected. Therefore, it is important to investigate architectural lessons from either traditional neural circuits or more modern neuromorphic designs that use different representation modalities, including spike time. To enhance the vehicle's capacity to adapt to changes in its surroundings, additional input from visual sensing and other sensing modalities might be used.

Other issues, such as hardware compatibility and safety, may become apparent when real-world vehicle performance is deduced. The models' scalability and power performance might also be evaluated by testing them on actual neuromorphic hardware and physical autos. First and foremost, software hardware conformance is crucial for hardware deployment. New evidence from a variety of learning methodologies suggests that the neural architecture might be fine-tuned for hardware optimization.

Finally, research sheds light on the possibilities of neuromorphic control in autonomous vehicle systems. The neuromorphic implementations' competitive performance when compared to CPU-based alternatives shows that this technique has potential. The performance and energy efficiency issues with neuromorphic controllers operating at high speeds need to be addressed in future studies, and the possibility of hybrid neuromorphic-CPU systems should be investigated.

References

1. Piñero-Fuentes, E., *et al.*, Autonomous driving of a rover-like robot using neuromorphic computing. *International Work-Conference on Artificial Neural Networks*, Springer International Publishing, Cham, 2021.
2. Chen, G., *et al.*, Event-based neuromorphic vision for autonomous driving: A paradigm shift for bio-inspired visual sensing and perception. *IEEE Signal Process. Mag.*, 37, 4, 34–49, 2020.

3. Frenkel, C.P., Bol, D., Indiveri, G., Bottom-up and top-down neural processing systems design: Neuromorphic intelligence as the convergence of natural and artificial intelligence, arXiv.org 2106.01288, 2021.
4. Patton, R., *et al.*, Neuromorphic computing for autonomous racing. *International Conference on Neuromorphic Systems 2021*, 2021.
5. Kim, J., New neuromorphic AI NM500 and its ADAS application. *AETA 2018-Recent Advances in Electrical Engineering and Related Sciences: Theory and Application*, Springer International Publishing, 2020.
6. Shalumov, A., Halaly, R., Tsur, E.E., LiDAR-driven spiking neural network for collision avoidance in autonomous driving. *Bioinspiration Biomimetics*, 16, 6, 066016, 2021.
7. Yang, J.-Q., *et al.*, Neuromorphic engineering: from biological to spike-based hardware nervous systems. *Adv. Mater.*, 32, 52, 2003610, 2020.
8. Chen, G., *et al.*, NeuroIV: Neuromorphic vision meets intelligent vehicle towards safe driving with a new database and baseline evaluations. *IEEE Trans. Intell. Transp. Syst.*, 23, 2, 1171–1183, 2020.
9. Viale, A., *et al.*, Carsnn: An efficient spiking neural network for event-based autonomous cars on the loihi neuromorphic research processor. *2021 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2021.
10. Lee, K.J., *et al.*, The development of silicon for AI: Different design approaches. *IEEE Trans. Circuits Syst. I Regul. Pap.*, 67, 12, 4719–4732, 2020.
11. Khacef, L., Abderrahmane, N., Miramond, B., Confronting machine-learning with neuroscience for neuromorphic architectures design. *2018 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2018.
12. Schuman, C., *et al.*, Evolutionary vs imitation learning for neuromorphic control at the edge. *Neuromorph. Comput. Eng.*, 2, 1, 014002, 2022.
13. Yu, Z., *et al.*, An overview of neuromorphic computing for artificial intelligence enabled hardware-based hopfield neural network. *IEEE Access*, 8, 67085–670995, 2020.
14. Vitale, A., *et al.*, Event-driven vision and control for UAVs on a neuromorphic chip. *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2021.
15. Liang, D., *et al.*, Neural state machines for robust learning and control of neuromorphic agents. *IEEE J. Emerg. Sel. Top. Circuits Syst.*, 9, 4, 679–689, 2019.
16. Roy, K., Jaiswal, A., Panda, P., Towards spike-based machine intelligence with neuromorphic computing. *Nature*, 575, 7784, 607–617, 2019.
17. Sun, B., *et al.*, Synaptic devices based neuromorphic computing applications in artificial intelligence. *Mater. Today Phys.*, 18, 1003935, 2021.
18. Lee, H.E., *et al.*, Novel electronics for flexible and neuromorphic computing. *Adv. Funct. Mater.*, 28, 32, 1801690, 2018.
19. Basu, A., *et al.*, Low-power, adaptive neuromorphic systems: Recent progress and future directions. *IEEE J. Emerg. Sel. Top. Circuits Syst.*, 8, 1, 6–27, 2018.

20. Ambrose, J.D., *et al.*, Grant: Ground-roaming autonomous neuromorphic targeter. *2020 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2020.
21. Kumari, S., Misra, A., Wahi, A., Rathore, P.S., Quality of Red Wine: Analysis and Comparative Study of Machine Learning Models. *2023 5th International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, pp. 769–772, 2023, doi: 10.1109/ICIRCA57980.2023.10220857.
22. Kanhaiya, K., Naveen, Sharma, A.K., Gautam, K., Rathore, P.S., AI Enabled-Information Retrival Engine (AI-IRE) in Legal Services: An Expert-Annotated NLP for Legal Judgements. *2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS)*, Trichy, India, pp. 206–210, 2023, doi: 10.1109/ICAISS58487.2023.10250733.
23. Wani, S., Ahuja, S., Kumar, A., Application of Deep Neural Networks and Machine Learning algorithms for diagnosis of Brain tumour. *2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, IEEE, 2023.
24. Batta, P., Ahuja, S., Kumar, A., A hybrid framework for secure data transfer for enhancing the Blockchain Security. *2023 Seventh International Conference on Image Information Processing (ICIIP)*, Solan, India, pp. 645–650, 2023, doi: 10.1109/ICIIP61524.2023.10537655.

7

Design of Brain-Computer Interface System to Develop Humanoid Robot

R. Raffik[1]*, K. Senthilkumar[2], A. Sakira Parveen[3], K. Akila[1], B. Sabitha[1] and P. Magudapathi[1]

[1]*Department of Mechatronics Engineering, Kumaraguru College of Technology, Coimbatore, India*
[2]*Department of Mechatronics Engineering, Coimbatore Institute of Engineering and Technology, Coimbatore, India*
[3]*Department of Electronics and Communication Engineering, SNS College of Technology, Coimbatore, India*

Abstract

The restoration of efficient individuality to undertake activities of daily living (ADL) is a key component in improving the quality of life for individuals with severe motor paralysis. In telepresence, a branch of robotics known as the "robotic-assisted route," a human operator provides sensory input to an assistance robot while also issuing high-level commands. However, traditional forms of engagement may not work for those who are completely paralyzed, as is the case with those who are severely motor-impaired. Integrating a telepresence controlled by brain framework with multimodal controller capabilities requires a new outline that combines a BCI method, as well as a humanoid robot. A BCI method generates the higher-level cognition commands required to conduct vital ADLs, while programs by demonstration (PbD) models implement the low-level control. Decoding attention-modulated neural responses evoked in brain electroencephalographic data in real time and producing numerous control instructions form the basis of the system that is given here. In this way, the user can communicate with an android via the system while also getting visual and audio cues from the robot's sensors which tested the solution in a real-world setting with few participants. The technique may be used to create a teleconferencing robot with good BCI decoding capabilities, according to the experimental findings.

**Corresponding author*: raffik.r.mce@kct.ac.in

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (105–122) © 2025 Scrivener Publishing LLC

Keywords**:** Robotic-assisted route, brain-computer interfaces, programming by demonstration, electroencephalographic data, teleconferencing robot

7.1 Introduction

Brain–computer interface (BCI) has made significant advances over the past period in its attempt to control external apparatus by deciphering the electroencephalogram (EEG) impulses of the brain. For example, BCI technology has shown promise in controlled wheelchair operations, robotic manipulators, computer cursors, and humanoid control situations in experimental demonstrations [1–6].

People whose movement is limited because of paralysis may find that BCI technology enhances their quality of life. Much research has shown that paraplegic persons can successfully use BCI techniques to operate manipulators and prosthetic limbs [7–10].

Telepresence with humanoids makes more sense than with robots that manipulate BCIs in certain contexts, as people are more naturally attracted to engage with them on a psychological level.

Various BCI technologies have been used in human telepresence research. Findings from studies like this stress the significance of selecting the appropriate paradigm when building a telepresence system. SMR-based BCIs are user-friendly as they generate control signals without relying on external stimuli. As the lengthy training time, limited work possibilities, and high concentration required, these systems might be difficult for disabled folks to operate. Furthermore, BCI systems powered by external inputs are often "Telepresence uses this technology." research. For example, the Potential for Visual Evoking Steady-State Even without instruction, BCIs could possess a high data transmission rate (ITR) and function effectively. The restricted alternatives for SSVEP-based interfaces caused by the effective frequency range mean that these advantages aren't free, either. Flashing stimuli in SSVEP BCIs lays a lot of strain on the eyes, which might lead to serious problems with vision and the eyes themselves. An external humanoid robot may be controlled for telepresence using a brain-computer interface (BCI) paradigm that is created on event-related potentials detected by electroencephalography (EEG) recordings in healthy persons. Attention-regulated brain reactions to external targeted visual/auditory events, ERPs differ from background events in EEGs in terms of spatiotemporal resolution and inherent qualities. What sets it apart is the P300 component, a positive wave that appears 300 to 400 ms after the triggering event begins and is seen when the user pays attention to an uncommon stimulus. Deciphering various ERP activity designs generated by a user on an experimental basis and converting them into suitable directives for a teleconferencing automaton is one of the primary objectives of the current work [11–15].

Human action performance consists primarily of two components: execution and decision-making. A decision is simply a deliberate and overarching selection of potential courses of action, whereas carrying out a task is an unconscious process. For example, every human being possesses a collection of acquired abilities that can be utilized in suitable circumstances. To gain an advantage from the present circumstance, pertinent abilities and actions are selected by the circumstances. Action implementation is a complex procedure requiring the participation of numerous controls, timing, trajectory creation, and other systems. Given that even humans do not consciously decode body motions at the mechanical and dynamical levels, low-level regulators may be unnecessary like a pipeline. This is because the ITR of BCI systems impedes complete robot supervision. Conveniently instructing robots instinctively through enforced demonstrations and imitation is programming by demonstration. Instruction is akin to instructing an infant; the trainer is not required to have any prior knowledge of programming or robotics [16–25].

The study's authors propose a reliable brain-computer interface (BCI)-to-telepresence system that would allow people with severe paralysis to interact more socially with humanoid robots. This system aims to enhance the mobility and social interaction abilities of the disabled, leading to advancements in their overall health.

7.2 Methodology

7.2.1 Proposed BCI Telepresence System Structure

Figure 7.1 shows the system architecture of the physical telepresence robot, which is connected to the closed loop BCI real-time method. The architecture is based on the client server model. A Buffer Server facilitates communication between the many clients that make up the BCI to the telepresence system, including those responsible for data gathering, processing signal, stimulus presentation, information storage and exoskeleton control. The "Experiment Control" client, which houses the primary GUI and is in charge of processing scheduling and defining the sequences of actions throughout an experimental session, is responsible for controlling all of these clients. For the experimental paradigm and interprocess

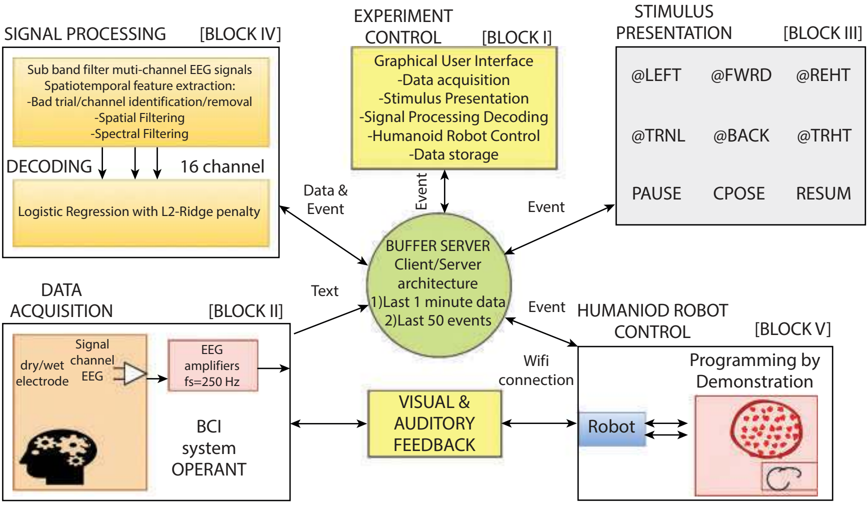

SIGNAL PROCESSING [BLOCK IV]
Sub band filter muti-channel EEG signals
Spatiotemporal feature extraction:
-Bad trial/channel identification/removal
-Spatial Filtering
-Spectral Filtering
DECODING
16 channel
Logistic Regression with L2-Ridge penalty
EXPERIMENT CONTROL [BLOCK I]
Graphical User Interface
-Data acquisition
-Stimulus Presentation
-Signal Processing Decoding
-Humanoid Robot Control
-Data storage
STIMULUS PRESENTATION [BLOCK III]
@LEFT
@FWRD
@REHT
@TRNL
@BACK
@TRHT
PAUSE
CPOSE
RESUM
Data & Event
Event
Event
BUFFER SERVER
Client/Server architecture
1)Last 1 minute data
2)Last 50 events
Text
Event
DATA ACQUISITION [BLOCK II]
dry/wet electrode
Signal channel EEG
EEG amplifiers fs=250 Hz
BCI system OPERANT
HUMANIOD ROBOT CONTROL [BLOCK V]
Wifi connection
Programming by Demonstration
Robot
VISUAL & AUDITORY FEEDBACK

Figure 7.1 A high-level description of the telepresence robots integrated closed-loop real-time BCI system's design. Collecting data, analyzing signals, controlling the robot, presenting stimuli, and storing data are the core components of a BCI to a humanoid robot. These clients communicate with one another via a Buffer Server that follows a client-server architecture.

communication to flow, every client produces events that determine the program's flow using an event-driven programming paradigm. Every client produces events that determine the program's flow using an event-driven programming paradigm. A client's ability to trigger and listen to proceedings is shown in Figure 7.1. An example use case would be when a user in Block I clicks on the GUI to initiate EEG data gathering; this action triggers an event in the Buffer Server, which is then handled by the event listener in Block II. The data is temporarily stored in the Buffer Server for up to 60 seconds before being saved permanently, with a maximum of 50 events. A scenario that may be handled by a robot controlled by a BCI is shown in the following flow diagram: Each client's event receivers and the buffering server are started in BLOCK I, the first block of the method. The second block is where data collection starts. The presenting stimulus software is debuted in Block III. The boot-construct interface (BCI) is activated in Block IV. Block V processes and sends the processed EEG data.

7.2.2 Participants

In this research, ten healthy persons (ranging in age from 22 to 35 years) were included. None of the participants had a history of neurological, physical, or mental disease. Everyone who took part in the study was an innocent bystander: a BCI user who had never been in an experiment like this before. No participant in the research was unable to provide their informed consent. The research project has received approval from the Institutional Research Ethics Committee at Nazarbayev University.

7.2.3 Electroencephalography

A 256-Hz sampling was used to collect electronic brain waves (EEG) from the scalp using 16 channels of active Ag/AgCl electrodes. Positioning of the EEG electrodes was done by the International 10–20 method. For the ground electrode, utilize the individuals' right earlobes, and for the reference electrode, use the spot on the brain known as FCz (see Figure 7.2).

7.2.4 Calibration Session

Figure 7.3 shows the result of implementing the Farwell & Donchin style speller using an LCD monitor to display a 3 × 3 grid of alphanumeric characters. A comfortable chair was provided for each participant, and they were positioned to face the LCD with about 60 cm of space between them. Every individual had a single session when their electroencephalogram signals

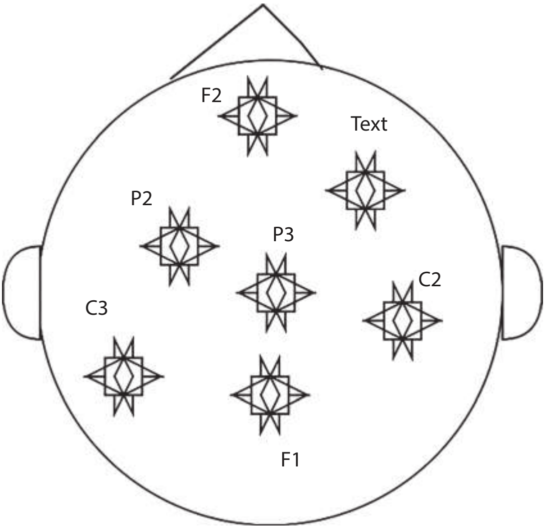


Figure 7.2 This investigation made use of a particular electrode montage. The normal ERP waveforms are depicted on all channels.

@LEFT	@FWRD	@REHT
@TRNL	@BACK	@TRHT
PAUSE	CPOSE	RESUM

Figure 7.3 A picture that triggers P300 and gives the NAO robot specific instructions.

were captured. The participants were asked to quietly count to ten whenever the target sign appeared in a flash row or column. With a two-second interval between each sequence, there were a total of five that each participant had to pay attention to. The stimulus rows and columns were repeated three times per sequence. A whole set of twelve random flashes over all six rows and columns (a total of fifteen total) comprised one repeat (or trial) for each target character. Both the stimulus duration (the amount of time a row or column is highlighted) and the inter stimulus intermission were set to 100 ms. The target-to-target interval (TTI), the bare minimum of time that must pass before two target letters are highlighted, was set at 600 ms. Onset asynchrony between stimuli was configured at 250 ms. See Figure 7.4 for an illustration of the stimulus-on-demand (SOA) timer in action.

7.2.5 Feedback Session

Next, segment and process the continuous EEG data according to the procedures outlined in Section II-F. Then, a classifier system on this data

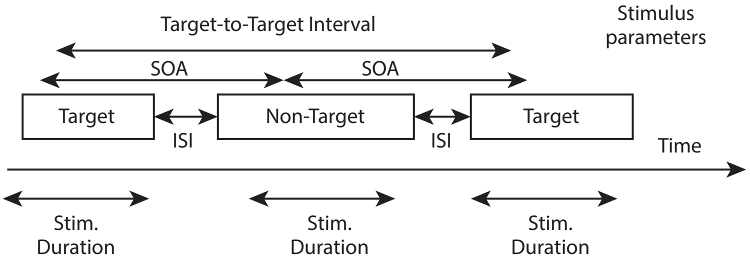


Figure 7.4 The settings for the displays of visual stimuli. ISD: time between sequences, Inter-stimulus interval (ISI) Time between targets (TTI), Asynchrony of stimulus onset (SOA).

without removing the EEG cap. On average, this very quick step took 70 seconds from start to finish. Subjects were then taught to direct the autonomous robot according to their preferences once the BCI classifier model was acquired. The sixteen instructions shown on the stimulus grid were open-ended, so respondents could employ them in whatever sequence they liked to complete the task. To call someone from far away, for example, the user may use these commands:

- Mobility, as in going forward, veering to the side, etc.
- “Hi,” “How are you?” “Shake hands,” and “Good-bye” are all examples of basic social interactions.

Figure 7.5 shows the customer interacting using the telepresence robot using BCI, and it also shows a simplified illustration of the architecture. In addition to interacting with the humanoid robot in response to cues received from other users in the interaction environment, BCI system users may see their surroundings in real time via the robot’s sensors. Additionally, participants had the option to mentally pick the “pause” command or notify the researcher to end the session at any point. Figure 7.1 depicts the whole pipeline of the created telepresence robot that is based on brain-computer interfaces. Because robot control was a separate buffer customer that was not in sync with the BCI pipeline, the humanoid robot may be taught fresh tasks at any point during the period. Any time, regardless of the session state, they may change the three slots for PbD training activities. As inputs to the Task-parameterized Gaussian mixture model

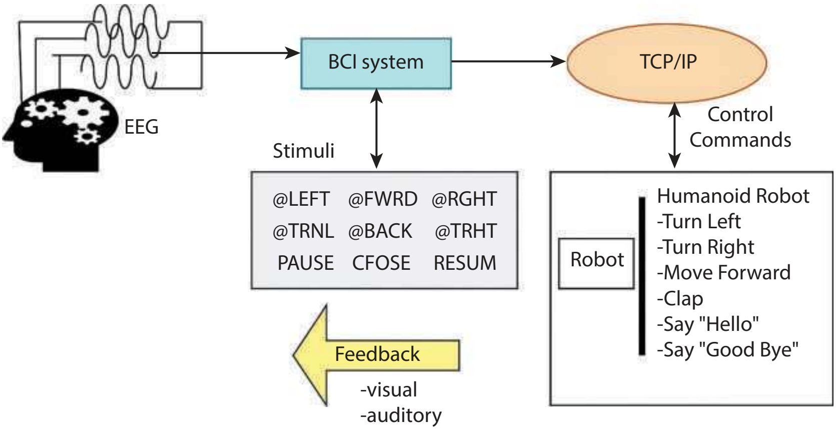


Figure 7.5 A streamlined perspective on the structure of the BCI-to-telepresence system.

(TP-GMM) method, landmarks of NAO with NaoQi built in functions for object detection, recognition, and localization as task parameters (TPs). A seven-dimensional vector, the reproduction trajectory includes three values for the end-effector's location, orientation, and time.

7.2.6 EEG Signal Filtering

Here are the techniques that were used to prepare the signals for ERPs that were detected in a single trial.

- After the onset of stimulus event markers, the continuous EEG data was segregated into 600 ms target and non-target trials. Removing trends: Deducting the overall mean from each channel eliminated data offsets that were not related to the analysis. This process eliminates the potential for perspiration and inadequate sensor-to-head contact to cause slow drifts, a kind of noise, to be present.
- Removing a failed trial to eliminate any inaccurate data caused by severe movement artefacts, the EEG data underwent artefact editing by an arithmetical thresholding approach. To remove trials using values exceeding three deviations from the mean, the procedure entailed computing the absolute value of each trial. This method served as the criterion for identifying poor trials.
- Removing bad channels: Electrodes that are contaminated with high noise due to incorrect connection to the participant's scalp were located by analyzing all channels across trials. To identify channels with abnormally high power, calculate the variance and mean of network control, as well as the total power for all epochs of each channel. To make room for the common averaged reference channel, eliminate all channels with a power greater than three standard deviations.
- Geographical Restrictions: To reduce the amount of noise caused by volume conduction and source mixing, a spatial bleaching filter was used. The whitening filter creates a new space with unit power and no correlation between the sensors by linearly re-weighting the electrodes and mapping the raw signals to this new space.

- Filtering the spectrum: The electroencephalogram (EEG) data has been filtered using a Fourier transform within the frequency range of 0.5 to 12 Hz. After the signal underwent Fourier transformation, it was weighted to inhibit and eliminate frequencies that did not fall within the rate of recurrence range of curiosity. Obtaining the filtered signal required opposite Fourier transformation of the weighted signal.

Emphasize that the aforementioned procedures were taken based on research that experimentally confirms a near-optimal strategy, even if there are several pre-processing approaches for ERP detection in the literature, include all the techniques in the MNE toolset.

7.2.7 Demonstration-Based Programming

1) **Localizing Objects**
   By strategically positioning NAO Landmarks on the worktop, ascertain the TPs. They were identified by the use of an in-built feature of the NaoQi program, which provides a wealth of information on the discovered landmark. The coordinates of the landmark's centre, its angular size, and its theoretical size, as well as the values of the head's orientation (yaw, pitch, and roll), are necessary parameters of the output for geometric landmark localization in 3D robot body space.
2) **Collecting Information**
   To enable force-guided movement, humanoid arms are programmed to operate in zero stiffness mode. Before the demonstration phase, the robot is positioned in front of its designated workstation, with all markers precisely located. The task demonstrator directs the robot's hands during the demonstration stage. The forward kinematics routine calculates and stores the final-effects device's 6D orientation and position vectors in the demonstration matrix at the highest achievable frequency, which is around 10 data points per second. The task conditions were varied and many demos (ranging from three to five) for each task.

3) **Processing Signals**

   When it comes to data points, all demos must have the same length so that matrix operations and manipulations are consistent. In addition, to replicate more uniform trajectories, the demonstration sounds were removed.

   - Interpolation: To ensure that all demos are of uniform size, a method called cubic Hermite spline is used. A third-degree polynomial given in Hermit form is applied to each slice of the raw data. It evenly spaces data points and forecasts additional information points to boost the number of samples obtained.
   - An adjusting filter: So that there is more information to process, every axis of position and orientation was subjected to the Savitzky—Golay filter. Finding and applying least-squares convolution coefficients to demo pieces allows for assessing new smooth data trajectories, which are appropriate given that data points are evenly spaced.

4) **TP-GMM**

   In recent years, TP-GMM Deterministic Gaussian Mixture Model) has become a popular approach for encoding robot movements. This model is an improved version of the original Gaussian mixture system and takes into account the locations and orientations of items in the environment as task parameters (TP).

5) **FINDING FRAMES OF REFERENCE THAT AREN'T RELEVANT**

   When starting, training stage 5the TP-GMM treats all the TPs as valid source points. Every frame of reference will have its trajectory and covariance matrix evaluated using GMR, and the experimental trajectory constructed using the initial estimated model. For every time step, the frame significance is determined by the standardized determining factor of the accuracy matrix, which is the inverse of the covariance matrix. When training a new TP-GMM system with updated TPs, any frames whose relevance is below the controlled threshold (0.1 in the example) are removed from the task dataset. The model will not be updated if all frames are essential. Generate trajectories in novel contexts using the revised TP-GMM model.

7.3 Results

Before the calibration step, the human experimenter instructed the android robot. The participants were then given the task of freely controlling the NAO robot in a self-paced mode after the calibration and BCI training phases. Walking about on a 2D vertical plane was their favorite method of greeting people and carrying out instructed activities. A vertical plane of 2D containing objects landmark seen as task constraints allowed for the seamless execution of PbD pre-trained tasks. The experiments consisted of a total of three PbD exercises, each of which taught the humanoid its left hand to follow various bench mark. In effect, remove from the database of the assignment any extraneous landmarks that were not submitted. Case studies of the execution of a particular task are illustrated in Figure 7.6. In Figure 7.7, the results of the reproduction trials for the identical job together with the frame significance plot are seen. Starting from the bottom left of the screen, make it to the bottom right of the screen to complete the trajectory. Since no training demo ever reached the highest landmark, the TP-GMM algorithm disregarded it as a meaningless TP. The red and green frames at the start as well as finish of the assignment, respectively, are crucial, as shown in the significance graph at the bottom of Figure 7.7. Nevertheless, the significance of the blue frame was consistently below the 0.1 limit.

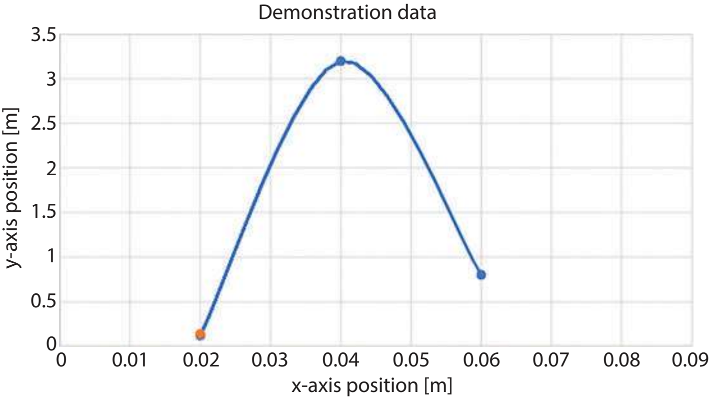


Figure 7.6 An example of a demonstration for one of the jobs, shown as the left-hand end-effector's trajectory superimposed over a two-dimensional vertical vector. The TPs are represented by the plus signs that are red, color coded green, and blue separately.

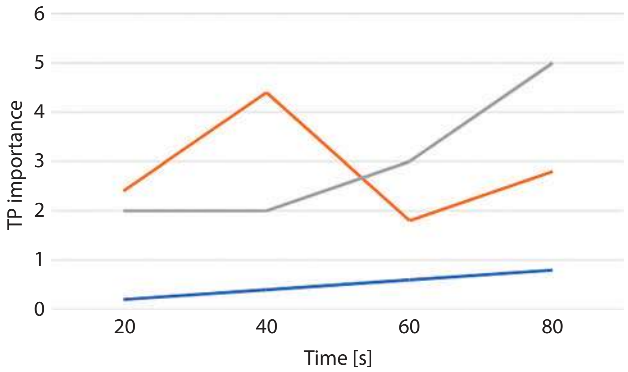


Figure 7.7 The top figure displays many instances of the left-hand end-effector's reproduction trajectories superimposed on a 2D vertical plane. The predicted significance of TP over time is shown in the graph at the bottom. At the outset, the green TP is paramount, the red TP acquires significance toward the conclusion, and the blue TP has zero bearing. The reproduction step disregards blue TP and removes it from the model database.

The chart displays the progression of Subject 10's grand average sixteen channel ERPs throughout time. Topographic distributions of target ERPs across time points are shown in the upper panel. The amplitude values of the representation target ERPs are shown in the bottom left panel using color coding. Analogous amplitude values for non-target ERPs are shown in the bottom right panel using colored representations. Brain reaction patterns associated with events become quite apparent between 300 and 400 ms. Discriminative patterns in the temporal distribution of targeted and non-target ERPs may be seen, especially between 300 and 400 ms. The number of ERP events utilized to train subject-specific classifiers for online BCI is shown in Figure 7.8. This quantity includes both target and non-target ERP events for each subject.

Figure 7.9 displays the area under the curve (AUC) score that represents the BCI system's real-time performance. It is noteworthy to note that all subjects exhibit BCI performance with an AUC better. The telepresence robot could be controlled and interacted with by an average of seven control commands per minute from all users. It is worth noting that the developed BCI system achieves overall performance near to the state-of-the-art described in the BCI literature, even though improving BCI decoders is not the main objective of this study.

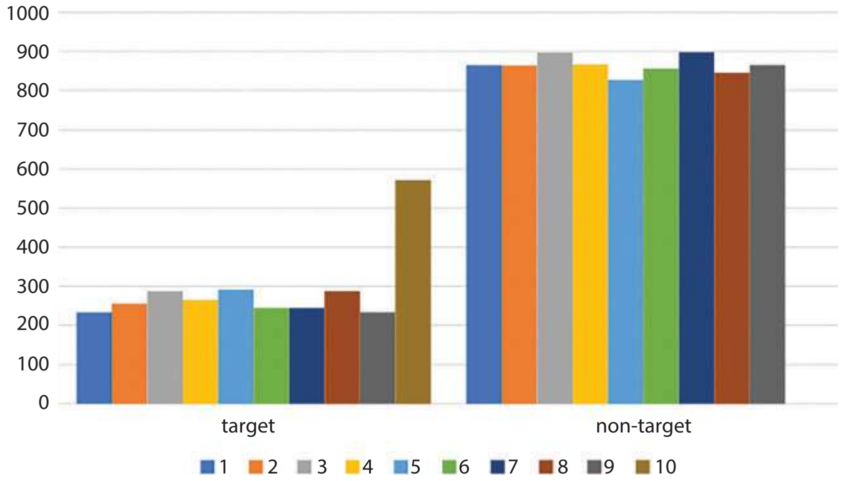


Figure 7.8 Individuals' total number of training sessions.

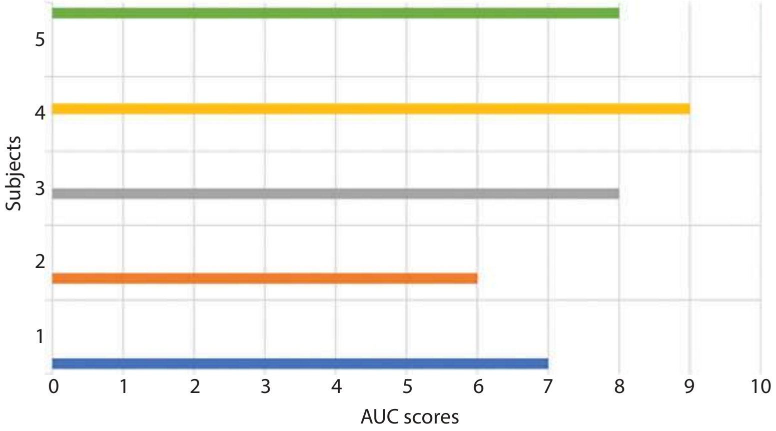


Figure 7.9 Accuracy of the BCI system as measured by AUC in real-time. Throughout five activities, the participants piloted an android. Forward, backwards, right, and left movement, as well as the instruction to "say Hello," were the only controls available.

7.4 Discussion

The telepresence robot can only be made more versatile and useful in more environments if BCIs and PbD methods are combined. While BCI provides for high-level control of the robot, PbD allows it to be pre-trained to execute critical everyday tasks. The end user, who may be a paraplegic individual, would utilize a brain-computer interface (BCI) to direct the remote robot's actions and choose tasks. Following the principles of cognitive decision-making and unconscious human learning, this situation can be characterized as a shared-control strategy in which the operator causes a high-level decision using the BCI scheme and PbD methodologies regulate the low-level implementation complexities. Caregivers of handicapped people might play a key role in the robot learning process, even if they may lack expertise in programming or robotics. Disabled persons would have more independence in their ADLs if they could utilize the robots that these folks trained using the PbD.

It becomes significantly more challenging to handle BCI events from one activity to another when the tasks include a broad variety of things to do. This section explains what event-driven programming is and how it works with clients that generate events (see Figure 7.1). To express the current state of the situation, including the status of the telepresence robot, the server aggregates all events that occur within the framework of a unified control mechanism. The method is asynchronous, meaning that it processes new event-based situations in response to an influx of new events. The client receives commands from the server and executes them via the shared control as soon as they are supplied. This study expands upon the shared-control technique mentioned and enhances the pure BCI-actuated Humanoid robot. An eventual convergence of telepresence robots and people in need may be possible with the help of the suggested PbD-based BCI system, which simplifies the operation of such a device. We argue that people with disabilities can engage with their surroundings more effectively when BCI and PbD devices are combined. In addition, the telepresence robot may pre-learn certain duties by watching a human execute the relevant function.

7.5 Conclusion

This research was inspired by the challenges that individuals with motor impairments encounter when attempting to engage in social interactions and carry out even the most basic activities. Utilizing cutting-edge technologies like brain-computer interfaces and robotics to enhance their quality of life is the primary outcome of the work presented. Supporting individuals with severe motor disabilities in remotely interacting with loved ones and carrying out physical everyday tasks is the goal of the proposed telepresence system enable telepresence through a remote humanoid robot, researchers validate a new use of program by demonstration (PbD) learning methods in conjunction with Brain-Computer Interfaces (BCI). The primary objective of the proposed telepresence system is to enable individuals with severe motor disabilities to engage in remote interactions with their loved ones or companions while carrying out routine physical tasks. The next reasonable step is to test the system on populations affected by motor paralysis, even though this research has only included healthy people. Building a new brain-computer interface (BCI)-to-telepresence system to operate a humanoid robot is one of the main contributions of the suggested research. This robot is anticipated to enhance the social elements of life for those with severe paralysis. Because these health advancements improve people's mobility and capacity to engage, the research is legitimate in its emphasis.

References

1. Li, W., Jaramillo, C., Li, Y., Development of mind control system for humanoid robot through a brain-computer interface. *2012 Second International Conference on Intelligent System Design and Engineering Application*, IEEE, 2012.
2. Bell, C.J., *et al.*, Control of a humanoid robot by a noninvasive brain–computer interface in humans. *J. Neural Eng.*, 5, 2, 214, 2008.
3. Chamola, V., *et al.*, Brain-computer interface-based humanoid control: A review. *Sensors*, 20, 13, 3620, 2020.
4. Li, W., Jaramillo, C., Li, Y., A brain-computer interface based humanoid robot control system. *Proceedings of the IASTED International Conference on Robotics*, 2011.
5. Choi, B. and Jo, S., A low-cost EEG system-based hybrid brain-computer interface for humanoid robot navigation and recognition. *PLoS One*, 8, 9, e74583, 2013.
6. Kucukyildiz, G., *et al.*, Design and implementation of a multi-sensor based brain-computer interface for a robotic wheelchair. *J. Intell. Rob. Syst.*, 87, 247–263, 2017.
7. Esfahani, E.T. and Sundararajan, V., Using brain–computer interfaces to detect human satisfaction in human–robot interaction. *Int. J. Humanoid Rob.*, 8, 01, 87–101, 2011.
8. Escolano, C., Antelis, J.M., Minguez, J., A telepresence mobile robot controlled with a noninvasive brain–computer interface. *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, 42, 3, 793–804, 2011.

9. Aznan, N.K.N., *et al.*, Using variable natural environment brain-computer interface stimuli for real-time humanoid robot navigation. *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, 2019.
10. Bogue, R., Brain-computer interfaces: control by thought. *Ind. Rob. Int. J.*, 37, 2, 126–132, 2010.
11. Rakshit, A., Konar, A., Nagar, A.K., A hybrid brain-computer interface for closed-loop position control of a robot arm. *IEEE/CAA J. Autom. Sin.*, 7, 5, 1344–1360, 2020.
12. Alimardani, M. and Hiraki, K., Passive brain-computer interfaces for enhanced human-robot interaction. *Front. Rob. AI*, 7, 125, 2020.
13. Nisar, H., Khow, H.-W., Yeap, K.H., Brain-computer interface: controlling a robotic arm using facial expressions. *Turk. J. Electr. Eng. Comput. Sci.*, 26, 2, 707–720, 2018.
14. Batula, A.M., Kim, Y.E., Ayaz, H., Virtual and actual humanoid robot control with a four-class motor-imagery-based optical brain-computer interface. *Biomed Res. Int.*, 2017, 107, 2017.
15. Stankevich, L. and Sonkin, K., Human-robot interaction using brain-computer interface based on EEG signal decoding, in: *Interactive Collaborative Robotics: First International Conference, ICR 2016, Budapest, Hungary, August 24-26, 2016, Proceedings 1*, Springer International Publishing, Cham, Switzerland, 2016.
16. Cincotti, F., *et al.*, Non-invasive brain–computer interface system: towards its application as assistive technology. *Brain Res. Bull.*, 75, 6, 796–803, 2008.
17. Zhang, B., Wang, J., Fuhlbrigge, T., A review of the commercial brain-computer interface technology from the perspective of industrial robotics. *2010 IEEE international conference on automation and logistics*, IEEE, 2010.
18. Lance, B.J., *et al.*, Brain–computer interface technologies in the coming decades. *Proc. IEEE*, 100, Special Centennial Issue, 1585–1599, 2012.
19. Mason, S.G. and Birch, G.E., A general framework for brain-computer interface design. *IEEE Trans. Neural Syst. Rehabil. Eng.*, 11, 1, 70–85, 2003.
20. Zhang, W., *et al.*, A review of EEG-based brain-computer interface systems design. *Brain Sci. Adv.*, 4, 2, 156–167, 2018.
21. Chae, Y., Jeong, J., Jo, S., Toward brain-actuated humanoid robots: asynchronous direct control using an EEG-based BCI. *IEEE Trans. Rob.*, 28, 5, 1131–1144, 2012.
22. Stephygraph, L. and Arunkumar, N., Brain-actuated wireless mobile robot control through an adaptive human–machine interface, in: *Proceedings of the International Conference on Soft Computing Systems: ICSCS 2015, Volume 1*, Springer India, New Delhi, India, 2016.
23. Thobbi, A., Kadam, R., Sheng, W., Achieving remote presence using a humanoid robot controlled by a non-invasive BCI device. *Int. J. Artif. Intell. Mach. Learn.*, 10, 41–45, 2010.

24. Güneysu, A. and Levent Akin, H., An SSVEP-based BCI to control a humanoid robot by using portable EEG device. *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2013.
25. Wani, S., Ahuja, S., Kumar, A., Application of Deep Neural Networks and Machine Learning algorithms for diagnosis of Brain tumour. *2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, IEEE, 2023.

8

AI-Based Neural Network Used to Enhance the Decision-Making System to Improve Operational Performance

G. Naga Rama Devi[1]*, Manthena Swapna Kumari[2], Vijaykumar S. Biradar[3], Manish Maheshwari[4], Subramanian Selvakumar[5] and Jenita Subash[6]

[1]Department of CSE: Data Science, Sreyas Institute of Engineering and Technology, Hyderabad, Telangana, India
[2]Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana, India
[3]Department of Electrical Engineering, N. B. Navale Sinhgad College of Engineering, Sholapur, Maharashtra, India
[4]Department of Computer Science and Applications, Makhanlal Chaturvedi National University of Journalism and Communication, Bhopal, India
[5]Department of Electrical & Computer Engineering, Bahir Dar Institute of Technology, Bahir Dar University, Bahir Dar, Ethiopia
[6]Department ECE, Cambridge Institute of Technology, Bengaluru, India

Abstract

This study analyzes the intended alignment of presentation and information technology (IT) objectives, providing a framework for decision makers in operations and production to improve operational performance. A unique decision-making framework was developed using the integrated methodologies, which were based on a thorough literature assessment. Using information gathered from 242 managers across different sectors, test the hypothesized correlations in an SEM model. To determine if the combined tactics are optimum, a decision-making framework is fed data from artificial neural networks (ANN), which is an AI-based approach. The findings show that (a) marketing strategy has a favourable effect on performance via IT strategy and (b) organizational structure moderates this effect. The results show that the suggested framework yields better results than the current techniques when applied to the extracted strategies. This work adds to the existing

**Corresponding author*: dr.g.nagaramadevi@sreyas.ac.in

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (123–138) © 2025 Scrivener Publishing LLC

body of knowledge by posing the question of how marketing strategy mediates between IT strategy, performance, and operational decision-making and conducting empirical tests to evaluate this hypothesis. Manufacturing other complex businesses might benefit from a new three-stage decision-making framework that makes use of AI processes to boost operational efficiency, insight, and decision accuracy when faced with strategic-level difficulties. Effective decision-making by operations executives may be aided by this.

Keywords**:** ANN, IT, decision making, AI, operational efficiency

8.1 Introduction

Research in the areas of production, information systems, planned management, and process management all revolve around the idea of alignment. While the industrial and supply chain has been the subject of some production studies, other studies have concentrated on product design. Another field that Bullinger and Schweizer looked at was product economy (2006). Examining the impact of organizational strategies in marketing and information technology (IT) on product quality and, by extension, company success, this training aims to fill a break in the literature by concentrating on route three of production research. Incorporating marketing strategy into manufacturing planning is well-known in production research for lowering overall costs and considerably improving profitability. To gain and keep clients' long-term preferences, loyalty, and business, relationship marketing the backbone of every marketing strategy strives to establish mutually beneficial connections with suppliers, distributors, and customers. With the exponential growth in processing power and network throughput in the last few decades, many businesses have begun to use IT strategies aimed at lowering operating expenses. Data redundancy may be reduced and operational efficiency can be increased by implementing IT stratification, which in turn pushes companies to optimize worldwide interconnectedness and data exchange. Strategic alignment is a must for all organizations in today's complicated market and rapidly evolving technology landscape. To achieve its goals, the company places a premium on coordinating its many strategies, including those for production, marketing, technology, and operations, with the priorities of its various business and functional units. The literature has stressed the significance of a company's strategy being in sync with its internal capabilities and the possibilities and risks presented by the external environment. Both external and internal fit categories describe this idea, which is prevalent in the field of operation management. Internal fit, on the other hand, describes the degree to which practices and tasks are compatible with one another. Aligning functional strategies with the overall company strategy is crucial for external fit, as it leads to a focused and relentless pursuit of corporate goals.

According to the research, businesses will suffer when their strategies are not in sync with their external environments or when their chosen strategies are not well implemented. This is known as "misalignment" and it is a set of symptoms or factors that organizations may face. Operations management literature has placed a premium on the effect of such a link on performance in a production environment. It is difficult for organizations to properly describe their IT plans upfront in response to environmental dynamics, making alignment in this setting particularly tough. Both marketing and IT strategy are crucial to an organization's success, and the research suggests that they have a substantial impact on company performance. Nevertheless, there is a dearth of research that takes external factors like environmental dynamism into account when studying the strategic links between marketing and IT, even though there are signs that, operationally, a closer relationship between those strategies will significantly affect company performance.

Various phases of production and operations management may benefit from AI. The use of AI in manufacturing, quality control, and packing, for instance, may increase the efficiency of these processes. Its analytical viewpoints may likewise be put to use in transportation and storage. Before releasing a new product, for instance, they estimated two Bass model parameters using machine learning. In comparison to earlier, more conventional models, they demonstrated that the AI-based estimate performed better. The accuracy of AI analyses will determine how widely it is used as a tool for analysis, using data mining and machine learning to sift through the little information gathered during a new product's experimental phase before mass manufacturing, as an example. Consequently, this analytical method should be used more often to discover the best advertising and IT plans for operational effectiveness. In the next issue. This study seeks to seal the gap in marketing-IT strategic position theory.

There are several ways in which this article advances the field of operations management. By the suggestion made by Andrews *et al.* (2009), this study fills a vacuum in the current literature by using structural equation modelling (SEM) to investigate the mediating and moderating impacts of the relationship with organizational structure. The second issue is that current alignments at both strategic functions have not been well studied or conceptualized. The third part of this study is devoted to developing an ANN-based strategy selection framework and analyzing the optimal simulation structure for ANN construction. Each industry's performance, like the manufacturing sector's, is forecasted using the data used to train the neural network, which in turn generates potential plan scenarios. Last but not least, the suggested technique retrieved the optimal strategy for every industry. Using predictive analytic approaches to analyse research model findings, adds to the literature. Those in charge of production and operations management may put the study's recommended method to work. The use of an ANN as an intelligent decision-making tool and for discovering the correlations between variables based on existing data is a natural consequence. The suggested method has several potential applications in the production and supply chain, even though applied to strategy selection.

Using SEM and ANN as our bases, do a comprehensive literature analysis covering topics such as marketing, IT strategic alignment, AI, operations management, and more. Our analysis of the theoretical arguments leads us to formulate three assumptions, which are then used to establish a research model. After collecting data from 242 participants, the model the tested using real-world examples. Section 8.5, provides the outcomes of selecting the AI-based decision-making framework. Wrap up by summarising our results and discussing their theoretical, methodological, and practical consequences.

8.2 Methodology

The majority of the prior research that combined SEM and ANN for different purposes focused on prediction. Consistent with earlier research, the present work validates its methodological contribution by taking into account the hybrid technique for prediction and decision-making in choosing the optimal scenario. Not previously examined in this context are IT and marketing tactics, which are the subject of the present research. Three distinct parts make up the present investigation. In the first stage established and validated the idea that, via environmental dynamics and organizational structure, marketing and IT strategies impact marketing success. Phase two included building an ANN that could forecast industry performance using the chosen marketing and IT strategies, after the extraction of an effective neural network from the collected data from different industries. The last step was to construct every conceivable scenario for every industry based on the predictions made by the trained ANN. Then, for every industry, the optimal case was retrieved. As a last step, confirmed the model's efficacy and identified the industries' targeted strategies by administering a follow-up questionnaire. Figure 8.1 depicts the primary stages of the suggested technique.

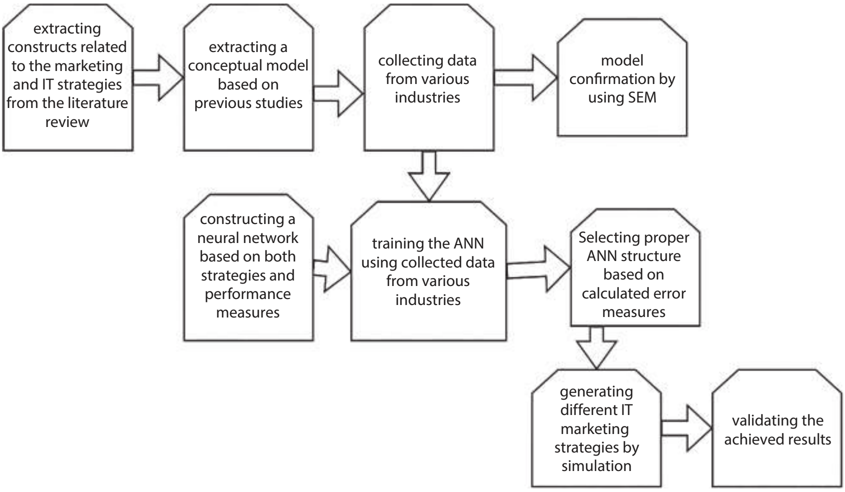

extracting constructs related to the marketing and IT strategies from the literature review
extracting a conceptual model based on previous studies
collecting data from various industries
model confirmation by using SEM
constructing a neural network based on both strategies and performance measures
training the ANN using collected data from various industries
Selecting proper ANN structure based on calculated error measures
generating different IT marketing strategies by simulation
validating the achieved results

Figure 8.1 A research approach map for demonstration purposes.

8.3 Conceptual Model

Research and theory building using SEM and ANN are the main topics of this section. Before delving into the findings, the theoretical models are introduced.

8.3.1 A Model for SEM Research

When looking at the link between strategy and operational performance, moderating effects are crucial, and several research have used mediation to examine interactions between different organizational strategies. Using new forms of mediation and moderation, this study expands upon previous research by bringing four factors into harmony.

To provide a more inclusive understanding of the deliberate alignment between marketing and IT, this method explains how marketing strategy influences IT strategy, with the moderating effects of environmental dynamics and organizational structure. The impact of this planned alignment on active performance will be investigated using this multi-stage technique, which combines theory with experiment. This is a fresh strategy, as far as are aware. When a company is market-oriented, its strategy may be more clearly defined and focused, which in turn boosts its performance. Some empirical studies with somewhat solid findings provide evidence (total and relative) for the presence of a positive connection between components, even if the results of this correlation are problematic. Possible foundations of competitive advantage that rivals face significant obstacles to match include resources that aid in value development, such as a focus on the market. This leads us to hypothesize that marketing strategy may have an impact on performance:

H1a: There is a favourable correlation between marketing strategy and performance.

Evidence suggests that strategic information systems may help businesses gain a competitive edge. By differentiating services and products, increasing market part, and decreasing transaction and operating costs, IT has enhanced performance in certain business scenarios. However, the question of whether or not a competitive advantage can be maintained, as well as the concept of competitive advantage itself, has been the subject of much debate. Conceptual and theoretical immaturity, methodological problems in assessing IT and performance, and uncontrollable, opaque factors are to blame for the indefiniteness of earlier studies. Results from case studies are also not easily generalizable or comparable. It is still up to empirical evidence to determine the impact of IT on presentation. Therefore, the following hypothesis is based on the belief that performance is directly affected by IT strategy:

H1b: There is a favourable correlation between IT strategy and performance.

At its core, customer relationship management (CRM) is a marketing approach. This is in line with the core principle that successful CRM implementations begin with well-considered marketing strategies. Instead of focusing on technical excess, marketing and IT managers should work together to transform the organization. This will allow them to make crucial practical additions while still using IT's capabilities. This synthesis of perspectives summarizes the previous research on CRM completion. Therefore, the following hypothesis is based on the belief that IT strategy directly impacts marketing strategy:

H1c: There is a favourable correlation between IT strategy and marketing strategy.

Both information technology and marketing have a favourable impact on company success, as the research has shown. There is some evidence that links business and IT alignment to improved company success, according to several studies. According to Zhu and Nakata (2007), there seems to be a significant impact on company performance in cases when there is a close relationship between IT and marketing, although not always at a planned level. The current research postulates a similar outcome for the strategic-level relationship between information technology and marketing. Therefore, the following hypothesis is derived from the belief that performance is indirectly affected by the mediating function of marketing strategy:

Marketing strategy favourably mediates the influence of the IT approach on performance (H1d).

The relationship between conservational dynamism and IT strategy has been the subject of a great deal of research. for instance, discovered that companies encountered intense competition when deciding to adopt a growth-oriented strategy to maximize their primary assets and achieve better competitiveness. This leads one to the succeeding theory on the impact of environmental dynamism on IT scheme:

H2a: IT strategy has a favourable correlation with environmental dynamism. Discovered that corporate success, as measured against strategic environmental value and decisions, is positively correlated with executive certainty. implied that this improved company performance in highly competitive environments and showed a favourable correlation between company success and market concentration. Environmental dynamism is thought to have an indirect influence on performance via its moderating function and a direct effect on performance itself, which leads to two hypotheses:

H2b: Performance is favourably correlated with environmental dynamism.

H2c: The impact of environmental dynamic on the medium via which marketing strategy influences performance.

The first to be researched was the decentralization-centralization dilemma. There is, however, no conclusive empirical evidence supporting this assumption at this time. By streamlining the flow of information across all organizational levels and departments, IT is thought to facilitate the decentralization of control and decision-making authority. By requiring the reinforcement of suitable representations of object systems and result processes, many IT applications might point to expanded formalization. By increasing the location of experts required to carry out a process, control activities, and systems progress, IT use may foster structural complexity, which is a more distinct and specialized structure. The possibility of opposite causality also exists. It is believed that decentralized companies would implement a decentralised IT department and use distributed hardware and software. More complex information support and information resource management are required by more formalized organizations since they use more management techniques including financial analysis, quality control, inventory control, and project management. A more intricate structure necessitates a framework that can be enhanced or supported by IT since it suggests a higher level of harmonization, communication, and control devices. Following the belief that organizational structure directly impacts IT strategy, the following hypothesis is put forth:

H3a: IT strategy is positively correlated with organizational structure.
If you want significant improvements in performance and efficiency thereafter, you should explain the regulating structures. For instance, to reap the rewards of new IT duties, specific subunits must be formed, a competent team must be appointed, and tools must be developed to coordinate their activities. Similarly, the growth of IT use may be replaced and made easier to control by appropriate structures. For example, the tactics required to overcome the risks and gain the competitive benefits associated with end-user computing are better suited to more organizationally sophisticated companies. This leads to the following theory, which is based on the assumption that organizational structure has an impact on performance:

H3b: There is a positive correlation between organizational structure and performance.
A distinctive and decentralised construction with non-routine processes technology was found to be moderated by high-performing organizational functions, according to researchers using the moderating technique. Although there is no mention of such outputs in the IT literature, comparable assumptions might be made. Accordingly, the following hypothesis is derived from the belief that the moderating influence of organizational structure has an indirect effect:

H3c: The relationship between marketing strategy and performance is moderated by organisational structure.
Figure 8.2 shows the research model, which includes all hypothesized linkages. Six components and the interactions between them make up the suggested model. The effect of strategic alignment on operational performance is investigated in this research. Results show that operational performance is positively affected by strategic alignment. Because of this, it is worth wondering if any other significant effects affect operational performance, such as whether or not marketing strategy alignment modifies the association between IT plan and operative performance. In a similar vein, this research looks at the possibility that organizational structure and environmental dynamics might moderate the link between marketing strategy and operational success.

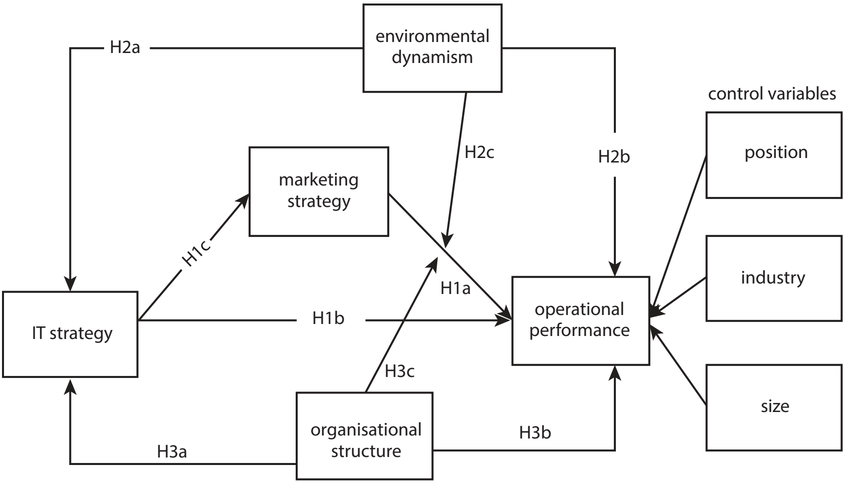


Figure 8.2 Model for SEM inquiry.

8.3.2 Artificial Neural Network Studies

An ANN with twenty-two inputs and five outputs was built once the constructions were taken into account. The ANN underwent training using training information to determine the optimal model weights (W) and bias (b) using an iterative approach. Next, the accuracy of the ANN predictions for the performance factors was evaluated using test data. Figure 8.3 shows the model that is based on ANNs.

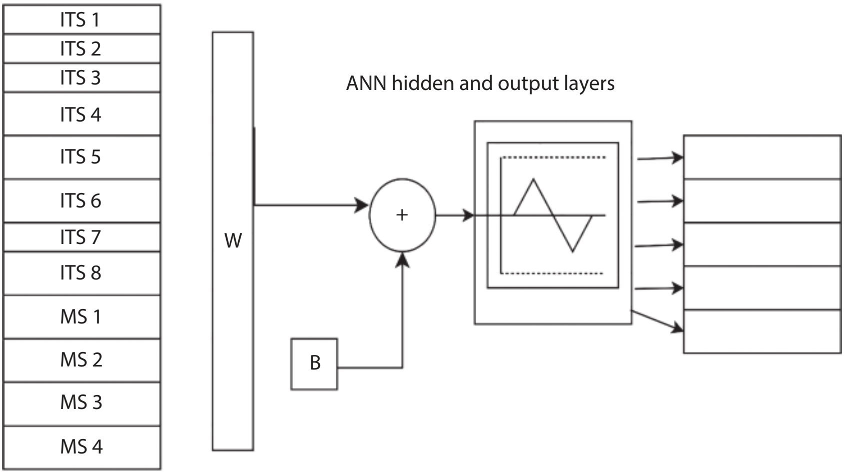


Figure 8.3 Structured equation modelling using an ANN model.

8.4 Results

Applying SEM and ANN, this part interprets the 242-questionnaire data collected from different sectors.

8.4.1 Data Gathering and Sample

A questionnaire survey was administered to private industries listed in the MENA region's Ministry of Industry and Trade. The participants were recruited at random and included marketing and IT managers, who helped to eliminate the potential bias in self-reported data. This allowed for an analytical assessment of the hypotheses. The findings may be applied to a broader population since the sample included a range of sectors, providing respondents with a rather consistent setting.

8.4.2 ANN Implementation

The 242*5 matrix with 242 samples and 5 output variables was used for this study. The input variable quantity was ED1-3, ITS1-8, MS1-8, and OS1-3. The ANN fitting allowed us to construct a correct ANN, as shown in Figure 8.4, with 22 input variables, 5 output variables, 10 hidden layers, and 1 output layer. used 170 samples for exercise, 36 samples for authentication, and the remaining 36 models for testing to create the network. This allocation corresponds to 70% for training, 15% for validation, and 15% for testing, based on the available data set of 242 acquired samples, respectively. Literature on strategic and operational management often makes use of this utility function.

Number of Hidden Layers: Varied to determine the optimal depth for better model performance. Experiments with single-layer networks and deeper networks were compared. Tested 12 different learning algorithms and explored networks with varying numbers of hidden layers to find the

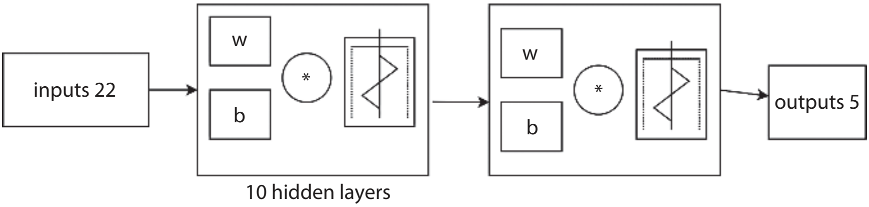


Figure 8.4 The OP neural network architecture.

optimal configuration: Algorithm of Levenberg-Marquardt and BFGS Close to Newton, Regularization using Bayesian methods, Reinforcement learning, Powell-Beale resumes in a conjugate gradient, three types of conjugate gradients: scaled Fletcher-Powell, and one-step secant. The gradient of the Polak-Ribiére conjugate, Using momentum for gradient descent, Two methods of gradient descent: mutable learning rate and gradient descent. evaluated all possible training methods for every possible number of hidden layers in the network and used performance metrics to choose the best one. Due to the nature of the output variable quantity, had to come up with an alternative performance metric other than the mean squared error (MSE). This new metric was determined using the following equation:

$$\text{e e ggrgrror} = \sum_{i=1}^{n} \sum_{j=1}^{j} \left| o_{ij-T_{ij}} \right|$$

where Oij is the predicted target value for the jth variable and Tij is the output of the ith sample can see the rounding function in the ||||. Once the experiments are finished the neural network with the fewest hidden layers and the most effective Bayesian regularization achieved the best results. The optimal neural network architecture was thought to be this. Given any value as an input, the extracted network could predict the performance.

Then, took a look at the twelve sectors picked manufacturing, education, transportation, banking and finance, electronics, retail, service, healthcare, communications, and others and extracted several scenarios based on what may happen in their business environments. Then, for each industry, the optimal approach was retrieved using a simulation-optimization technique. Figure 8.5 shows the schematic of the suggested process. The suggested method included training an artificial neural network (ANN) and then utilizing it to determine the best course of action for each industry, whether its performance was poor, medium, or high. Every industry may benefit greatly from implementing appropriate, well-planned strategies, according to the findings of the optimum strategy extraction process. Consider the marketing and advertising business. The research suggests that in a competitive advantage environment, a mix of IT strategy, customer, and competitor-based strategies, and a hierarchical structure would lead to better performance and effectiveness. This method extracts the finest IT and marketing strategies in many domains with more variables, and it also defines optimal plans for distinct company sectors in the studied region.

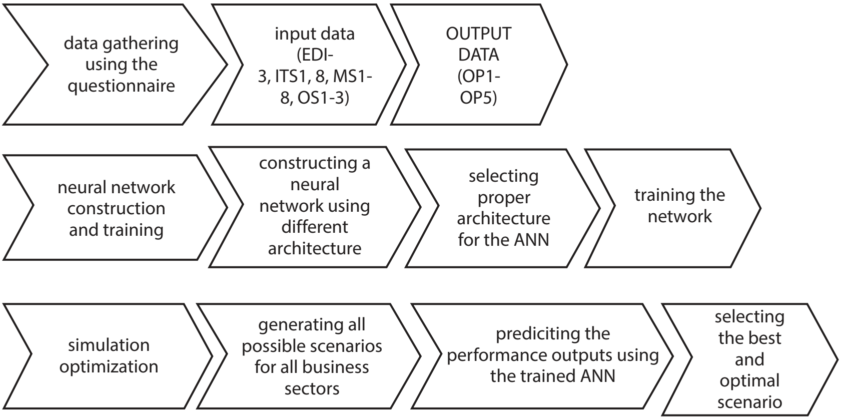

data gathering using the questionnaire
input data (EDI-3, ITS1, 8, MS1-8, OS1-3)
OUTPUT DATA (OP1-OP5)
neural network construction and training
constructing a neural network using different architecture
selecting proper architecture for the ANN
training the network
simulation optimization
generating all possible scenarios for all business sectors
prediciting the performance outputs using the trained ANN
selecting the best and optimal scenario

Figure 8.5 Neural network-based schematic flowchart of operations to extract the best strategy for each company sector.

8.5 Conclusion

The research utilizes a three-stage decision-making framework to identify optimal marketing and IT strategies, considering organizational constraints and environmental dynamics. Examined current research on operations management to resolve the issue of marketing and IT strategy position. The overarching goal of this research was to find out how operations and production decision makers may utilize AI to boost operational performance via marketing and IT initiatives. Research on the effects of marketing and IT strategy alignment on operational success is scarce. The impact of moderators and mediators on placement and its relationship to operational performance were examined in this research. Among the few studies that provide a model to decision makers in production and operations for choosing the best marketing and IT strategy in the face of organizational structure restrictions and environmental dynamism, this one stands out. The best marketing and IT strategies across industries may be uncovered using a three-stage decision-making framework that offers theoretical, methodological, and practical insights.

8.5.1 Contribution to Theory

There are three areas where our work adds theoretical value to the existing literature. The first way it helps with operations and strategic management is by adding to the body of knowledge on selecting the best possible strategy. Second, some studies have focused on how businesses and IT departments should work together strategically; this one added to that body of work by defining and then implementing a framework for how marketing and IT departments should work together to adapt to changing external conditions and internal organizational dynamics. Third, following the advice of previous research, this study took into account other aspects, namely, organizational structures and environmental dynamism. With the current organizational structure and the rapid evolution of the business landscape in mind, the model's developers have placed a focus on the significance of coordinating marketing and IT strategies.

8.5.2 Methodological and Empirical Contributions

The research drew on several existing methods and provided new empirical evidence. The first stage was thinking about how to include the interplay effects of moderation and mediation. Findings from the structural equation modelling (SEM) analysis support the previous assertion that marketing and information technology (IT) strategies individually improve operational performance, and they also prove that hypotheses H1a and H1c are correct. This research provides evidence that the mediation effect supports the link between marketing and IT strategies and operational success, demonstrating the need for strategic alignment. It has also been shown that theorized favourable effects of organizational structure and environmental dynamics on operational effectiveness exist. As a new contribution to the literature, this study provides empirical evidence that organizational structure moderates the link between marketing strategy, IT strategy, and operational performance. This indicates that organizational structure mediates and moderates the impact of marketing and IT initiatives on output performance metrics. Even though not all of the proposed hypotheses held up to empirical scrutiny, the majority demonstrated that the structural model is supported by the data gathered from certain sectors.

Second, building an ANN model becomes much easier after obtaining and validating the hypothesized model. An artificial neural network (ANN) was built and trained using data from a training set based on the effective extracted elements. To improve the network's performance, the right artificial neural network (ANN) architecture was chosen, which included the right number of layers and an effective learning algorithm. By isolating the most efficient input variables, the SEM results decreased the tuning time of the ANN. A rise in ANN accuracy is another outcome of this combination approach.

Third, the best methods for each industry were extracted using a staged approach. This work makes a methodological contribution by achieving the aforementioned findings through the integration of a multi-phased analytical strategy that uses SEM and ANN applications. Machine learning may be used to extract the correct model for system behaviour, and the analytical skills of the suggested AI-based technique can be applied to various areas of operations management. Researchers will have a fresh outlet to maintain and develop this streamlining across both approaches. Given access to this decision-support technology, decision makers will be able to make stronger judgments in less time. Optimal decision-making in various phases of operations management is possible with the suggested AI-based decision support system, but we used it for strategy selection.

References

1. Al-Surmi, A., Bashiri, M., Koliousis, I., AI-based decision making: combining strategies to improve operational performance. *Int. J. Prod. Res.*, 60, 14, 4464–4486, 2022.
2. Bal Beşikçi, E., *et al.*, An artificial neural network based decision support system for energy efficient ship operations. *Comput. Oper. Res.*, 66, 393–4015, 2016.
3. Mumali, F., Artificial neural network-based decision support systems in manufacturing processes: A systematic literature review. *Comput. Ind. Eng.*, 165, 107964, 2022.
4. Andronie, M., *et al.*, Artificial intelligence-based decision-making algorithms, internet of things sensing networks, and deep learning-assisted smart process management in cyber-physical production systems. *Electronics*, 10, 20, 2497, 2021.
5. Gupta, S., *et al.*, Artificial intelligence for decision support systems in the field of operations research: review and future scope of research. *Ann. Oper. Res.*, 1–60, 2022.
6. He, Z., Guo, W., Zhang, P., Performance prediction, optimal design and operational control of thermal energy storage using artificial intelligence methods. *Renew. Sustain. Energy Rev.*, 156, 111977, 2022.
7. Abdolrasol, M.G.M., *et al.*, Artificial neural networks based optimization techniques: A review. *Electronics*, 10, 21, 2689, 2021.
8. Wamba-Taguimdje, S.-L., *et al.*, Influence of artificial intelligence (AI) on firm performance: the business value of AI-based transformation projects. *Bus. Process Manage. J.*, 26, 7, 1893–1924, 2020.
9. Tsai, J.-M. and Hung, S.-W., Supply chain relationship quality and performance in technological turbulence: an artificial neural network approach. *Int. J. Prod. Res.*, 54, 9, 2757–2770, 2016.

10. Lim, A.-F., *et al.*, Unfolding the impact of supply chain quality management practices on sustainability performance: an artificial neural network approach. *Supply Chain Manag.: Int. J.*, 27, 5, 611–624, 2022.
11. Huang, Z., *et al.*, Credit rating analysis with support vector machines and neural networks: a market comparative study. *Decis. Support Syst.*, 37, 4, 543–558, 2004.
12. Nica, E., *et al.*, Artificial intelligence decision-making in shopping patterns: Consumer values, cognition, and attitudes. *Econ. Manage. Financ. Mark.*, 17, 1, 31–43, 2022.
13. Massaro, A., Implementation of a decision support system and business Intelligence algorithms for the automated management of insurance agents activities. *Int. J. Artif. Intell. Appl. (IJAIA)*, 12, 3, 137, 2021.
14. Ali, M.H., *et al.*, Autonomous vehicles decision-making enhancement using self-determination theory and mixed-precision neural networks. *Multimed. Tools Appl.*, 82, 1–24, 2023.
15. Nica, E. and Stehel, V., Internet of things sensing networks, artificial intelligence-based decision-making algorithms, and real-time process monitoring in sustainable industry 4.0. *J. Self-Gov. Manage. Econ.*, 9, 3, 35–47, 2021.
16. Taherdoost, H., Deep Learning and Neural Networks: Decision-Making Implications. *Symmetry*, 15, 9, 1723, 2023.
17. Kumar, T.S., Data mining-based marketing decision support system using a hybrid machine learning algorithm. *J. Artif. Intell. Capsule Netw.*, 2, 3, 185–193, 2020.
18. Azadeh, A., *et al.*, An integrated artificial neural network and fuzzy clustering algorithm for performance assessment of decision-making units. *Appl. Math. Comput.*, 187, 2, 584–599, 2007.
19. Chen, S., *et al.*, Which product description phrases affect sales forecasting? An explainable AI framework integrating WaveNet neural network models with multiple regression. *Decis. Support Syst.*, 176, 114065, 2024.
20. Phillips-Wren, G., Intelligent decision support systems, in: *Multicriteria Decision Aid and Artificial Intelligence: Links, Theory and Applications*, pp. 25–44, 2013.
21. Efendigil, T., Önüt, S., Kahraman, C., A decision support system for demand forecasting with artificial neural networks and neuro-fuzzy models: A comparative analysis. *Expert Syst. Appl.*, 36, 3, 6697–6707, 2009.
22. Batta, P., Ahuja, S., Kumar, A., A hybrid framework for secure data transfer for enhancing the Blockchain Security. *2023 Seventh International Conference on Image Information Processing (ICIIP)*, Solan, India, pp. 645–650, 2023, doi: 10.1109/ICIIP61524.2023.10537655.
23. Kumari, S., Misra, A., Wahi, A., Rathore, P.S., Quality of Red Wine: Analysis and Comparative Study of Machine Learning Models. *2023 5th International Conference on Inventive Research in Computing Applications (ICIRCA)*, Coimbatore, India, pp. 769–772, 2023, doi: 10.1109/ICIRCA57980.2023.1022085.

24. Kanhaiya, K., Naveen, Sharma, A.K., Gautam, K., Rathore, P.S., AI Enabled-Information Retrival Engine (AI-IRE) in Legal Services: An Expert-Annotated NLP for Legal Judgements. *2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS)*, Trichy, India, pp. 206–210, 2023, doi: 10.1109/ICAISS58487.2023.10250733.
25. Kumar, A., Dubey, A.K., Segovia Ramírez, I., *et al.*, Artificial Intelligence Techniques for the Photovoltaic System: A Systematic Review and Analysis for Evaluation and Benchmarking. *Arch. Comput. Methods Eng.*, 138, 2024, https://doi.org/10.1007/s11831-024-10125-3.
26. Dubey, A.K., Kumar, A., García-Díaz, V., Sharma, A.K., Kanhaiya, K., Study and analysis of SARIMA and LSTM in forecasting time series data. *Sustain. Energy Technol. Assess.*, 47, 101474, 2021.

9

Simulation and Implementation of English Speech Recognition by NLP

K. Kavita[1]*, K. Suresh Kumar[2], Sridevi Dasam[3] and Kiran Sree Pokkuluri[4]

[1]Department of Mathematics, BVRIT Hyderabad College of Engineering for Women, Hyderabad, India
[2]MBA Department, Panimalar Engineering College, Varadarajapuram, Poonamallee, Chennai, India
[3]Department of English, Velagapudi Ramakrishna Siddhartha Engineering College, Vijayawada, India
[4]Department of Computer Science and Engineering, Shri Vishnu Engineering College for Women, Bhimavaram, India

Abstract

The use of hybrid deep learning-based voice recognition in oral English practice, in conjunction with multimodal natural language processing education, begins with an introduction to the fundamentals of speech recognition technology. An explanation of the hidden Markov model and its three essential algorithms is provided, followed by the realization of its simulation and use in voice recognition. The system's architecture and essential technologies are presented. First, the text delves into the use of deep learning in natural language processing and recording oral English instruction by specialized instructors. Each person has their own optimal reading time and preferred phrases. In all, there are several individuals. The phrases used are spoken English, so it would be advantageous to provide a course in spoken English to help people enhance their oral communication skills. The findings of the trial indicate a decrease in the accuracy of identification, but a tenfold increase in recognition speed. Another advantage is that the scoring system is equally accurate to the platform system. By validating the feasibility and effectiveness of this approach, it enhances the accuracy of instruction categorization. Attention mechanisms will be utilized to expand this strategy in the future.

**Corresponding author*: kavitha.k552017@gmail.com

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (139–158) © 2025 Scrivener Publishing LLC

***Keywords*:** Hybrid deep learning, english speech recognition, mechanisms, specialized instructors, communication skills, spoken english, accuracy of identification, natural language processing

9.1 Introduction

Growth in commercial openness and global economic integration have increased worldwide trade. Today, success in many industries requires fluency in many languages, especially English. To address rising English language teaching demand, language schools, pedagogical methods, and course materials have proliferated. Spoken English has traditionally been difficult for learners [1, 2]. This has two main causes: (1) Due to phonetic differences between English and their original language, speakers who learn English under substantial influence from their native tongue may make audible but invisible pronunciation errors. (2) Schools struggle to recruit skilled foreign language teachers. There is a severe lack of qualified English language instructors in even the most elementary and middle schools in major urban areas. The only way students learn general knowledge is in an individual setting, not in small groups. Oral instruction is ineffective because both instructors and pupils may use it [3–10].

Native speakers who can talk to pupils in other languages are few. Even in big and medium cities, elementary and intermediate schools lack native English speakers who can teach spoken English. General media education must be offered uniformly to all pupils. Since teachers and students can communicate, spoken education is ineffective.

The main aims of AI-powered language learning systems are word and grammar acquisition. Certain voice recognition algorithms can only give pupils a global pronunciation score. Self-scholars' expertise makes it hard for them to discover faults and correct pronunciation. Figure 9.1 shows how voice recognition technology enables the software to fix speech faults; this feature may teach pupils to fix their own mistakes and not make the same ones over and over again. Both society and businesses stand to gain substantially from efforts to boost the effectiveness of students' oral learning. Voice car navigation systems, intelligent robots, interactive items controlled by voice (echo), intelligent voice assistants for mobile phones, and voice input methods are just a few of the products that have recently made their way into the market, all made possible by speech recognition technology. In terms of raising people's level of life, these applications are

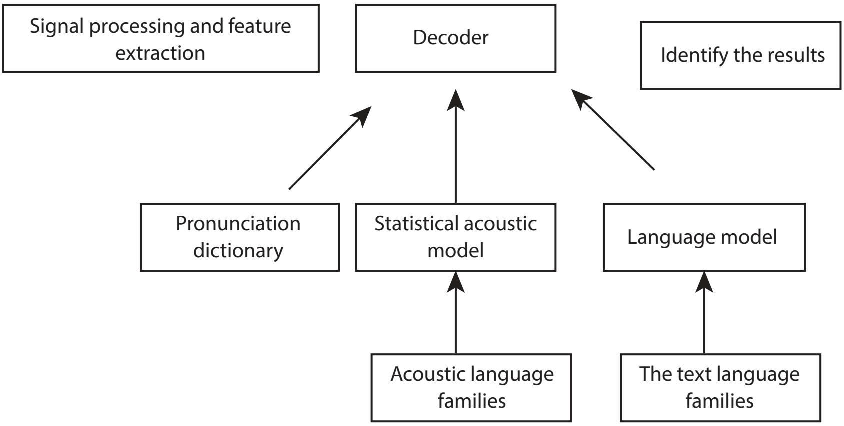

Signal processing and feature extraction
Decoder
Identify the results
Pronunciation dictionary
Statistical acoustic model
Language model
Acoustic language families
The text language families

Figure 9.1 Recognition of speech.

very consequential. Voice credit technology is now a research hot spot in society, serving as an essential avenue for artificial intelligence research [11–13].

One kind of neurolinguistic programming is NLP. A neuro-linguistic programming (NLP) system consists of three letters: nerve, language, and program. Studying the impacts of human conduct on others, discovering commonalities among highly successful people, and creating ways to imitate their activities are all areas of interest in natural language processing.

Their research primarily focuses on three competent psychotherapists: a pioneer of family therapy and a pioneer of hypnotherapy. To study these outstanding psychologists; to identify similarities in their thinking and language usage; and to distil this data into a curriculum that others might adopt to mimic their success. Psychotherapy and personal development are two of its many uses. "Imitation" is the central concept in natural language processing. First, goal setting (or defining the requirements); second, consistent affinity (or building affinity); third, sensory acuity (or making use of various senses); and fourth, behavioural flexibility (or the ability to utilize procedures in a variety of contexts) are the four pillars upon which neuro-linguistic programming (NLP) rests. The use of natural language processing (NLP) in English language instruction has several benefits, including raising students' motivation and enhancing their ability to learn, as well as assisting them in overcoming the mental challenges associated with speaking English fluently [14, 15].

9.2 Methodology

9.2.1 Practice of Oral English Using Speech Recognition

Phonetics is based on comparison. It is recommended to examine the model of functioning presentations with the design of presenting skills one by one to find the optimum fit Presignal communication is used in general recognition for eigenvalue extraction, training models, matching models for recognition, result determination, and recognition [16–18]. Human speech is generated when air is compressed and vibrates through the airways. Human speech is capable of producing three distinct sounds—voiced, unvoiced, and staccato—since it is stimulated in three distinct ways. Language relies on a smaller set of numerical symbols to express ideas, even though people are capable of producing an infinite variety of sounds [19]. There are typically just twelve phonemes in a language. The smallest unit of encoding in a system of communication, a phoneme may be thought of as a collection of finite characters. Two types of phonemes, "open" and "closed," are distinguished by the various speech and action states they represent [20]. The English language uses closed phonemes for consonants and open phonemes for vowels. A narrower pitch gives rise to a little fricative sound known as a semivowel, even if the vowels themselves are simple in tone. Since it is an analogue transmission, the amplitude of the speaker signal varies in real-time. Computers will be able to read and process it after digitalization [21–23]. Digitization of voice signals is a vital aspect of digital processing. Testing and quantification are part of the process of digitizing voice sounds [24]. This two-step procedure produces digital signals with varying amplitudes.

One step in digitizing speech is testing and quantifying it. Various amplitude digital signals are generated by this two-step process [25]. Therefore, speech signals' spectral properties may be quite transient, as they tend to remain relatively constant throughout lengths. In time-dependent processing, the most fundamental method is to intercept a voice signal and analyze it inside a restricted window sequence {w(m)}. The signal may be analyzed at any point around the window's east end. Follow this generic formula:

$$Q_n = \sum_{m=-\infty}^{\infty} T\left[x(m)\right] * w(n-m) \tag{9.1}$$

The input signal sequence is denoted by {x(m)}, and T[] stands for a specific operation. A unit stimulus of {w(m)} should be applied to the discrete signal T [x(m)] via an FIR low-pass filter, as seen in Figure 9.2, to

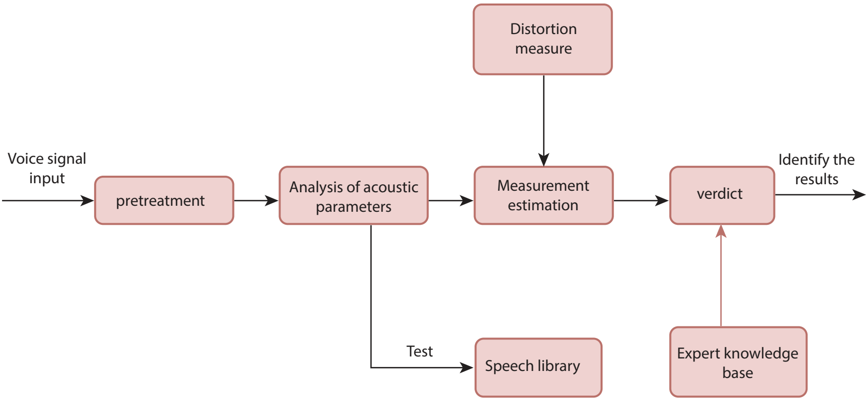


Figure 9.2 Speech-recognition flowchart.

understand Qn, as Equation (9.1) is in convolutional form. The filter that corresponds to this impulse response has low-pass characteristics since the window function is often assumed to be a flat function with big middle and minor ends of x(n).

The creation function chooses to determine the bandwidth and frequency responsiveness. The rectangular, Hamming, and Hanning windows are the three most common types of windows, and their definitions are as follows:

Window rectangle

$$w(n) = \left\{ \begin{matrix} 1 & 0 \le n \le L-1 \\ 0 & other \end{matrix} \right\} \tag{9.2}$$

Window hammer

$$w(n) = \left\{ \begin{matrix} 1 & 0.54 - 0.46\cos\left(\frac{2\pi n}{L-1}\right) \\ 0 & other \end{matrix} \right\} \tag{9.3}$$

Window Hanning

$$w(n) = \left\{ \begin{matrix} 1 & 0.5\left[1-\cos\left(\frac{2\pi n}{L-1}\right)\right] \\ 0 & other \end{matrix} \right\} \tag{9.4}$$

In this case the window functions exhibit low-pass properties and L is the window length. Hamming windows with low side steps are commonly used because they combat water leakage and have low characteristics. Rectangular windows with high side lobes are rarely used due to their drawbacks. Also, the average interference to the signal becomes greater with increasing window length, leading to increased signal resolution at the expense of resolution time. See Figure 9.3 for an illustration of why a shorter window length is preferable for intercepting files with varying speeds.

Voice recognition software identifies speech beginnings and ends Popular front-end endpoint detection approaches include multi-threshold and double-threshold. Since the front-end finding value may require several beginning zeros, both start and end-finding approaches are utilized for real-time removal. The approach reduces error despite its delay, which is unnecessary for managing time Strategies like Tiny Time Strength and Short Time Zero Trip Value allow the user to overcome limitations of the original search. The audible signal intensity changes visually over time. To employ the pitch, loudness, and other variables of voiceless speech, remember that it requires far less energy than speaking. This is the definition of the short-time energy for signal x(n):

$$E_n = \sum_{m=-\infty}^{\infty} \left[X(n) * w(n-m)\right]^2 = \sum_{N=0}^{N-1} s_w^2(n) \tag{9.5}$$

In certain situations, integration is disadvantageous because the square function of the signal affects the short energy period, resulting in high-low

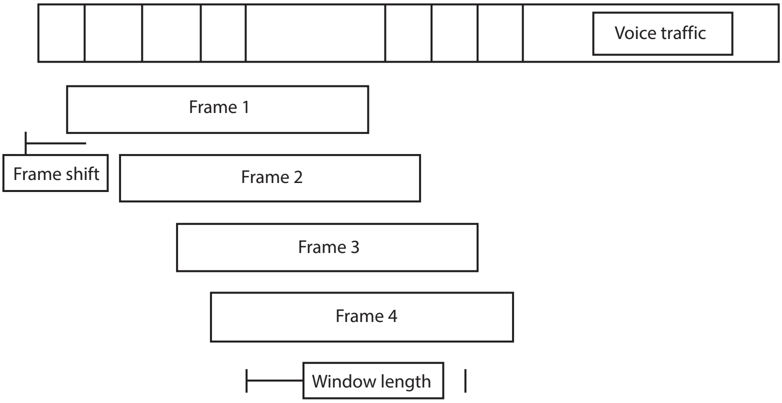


Figure 9.3 Rectangular window intercepts voice.

signal discrepancies. A temporary average amplitude to express energy change solves this problem in a simple formula:

$$M_n = \sum_{m=-\infty}^{\infty} \left[X(n)w(n-m)\right]^2 = \sum_{m=n}^{n+N-1} \left|x_w(m)\right| \tag{9.6}$$

The frequency of occurrences in each post when the signal crosses zero is known as the temporary unkind zero crossing, as the designation suggests. At the signal sampling point, the number of signal changes is proportional to the significance of the short-term mean zero crossing value with respect to the difference. Its primary usage is in providing a basic description of signals' spectral properties, but it also has two other significant uses. The second step is to determine the beginning and ending points of the speech in relation to the available short-term energy, also known as endpoint detection. This is done by calculating the zero-crossing rate of the signal {x(n)}.

$$Z_n = \sum_{m=n}^{n+M-1} \left|sgn\left[x(m) - sgn\left[x(m-1)\right]\right]\right| w(n-m) \tag{9.7}$$

where symbolic function is sgn []

$$sgn\left[x(n)\right] = \begin{Bmatrix} 1 & x(n) \geq 0 \\ -1 & x(n) < 0 \end{Bmatrix} \tag{9.8}$$

Window sequence w(n) is set to

$$w(n) = \begin{Bmatrix} \dfrac{1}{2N} & 0 \leq N-1 \\ 0 & other \end{Bmatrix} \tag{9.9}$$

In the window range, the 1/2N window amplitude averages zero crossing statistics. The window contains examples, and each sample occupies 2. There are several window options than right-angle. Noise in the quiet region may cause a high zero crossing rate, thus establishing a threshold is the first line of protection. If the difference between the signs of the present and next two examples exceed the threshold, the zero-crossing rate increases by one point.

The Mel frequency cepstral coefficient (MFCC) is used in speaker recognition systems to quickly characterize speech. Cestrum parameters are theoretically robust. Using cestrum settings has two evident advantages. Filtering and weighting the cestrum domain for spectrum processing is a benefit. Mel cestrum hypothesis is also widely applied. Mel frequency cestrum parameter analysis addresses the human auditory system and analyzes the speech spectrum based on hearing studies to produce a high recognition rate and outstanding noise resistance.

The basement membrane in the inner ear plays a vital role in regulating external impulses, which allows people to hear speech clearly even in noisy environments. This membrane moves at different frequencies in response to signals within the important bandwidth. By imitating the human ear, a band-pass filter bank can effectively decrease background noise while preserving speech signals. A crucial frequency band must first be specified. Subjective sense of loudness stays constant until the apparent volume of the sound pressure varies, as long as the noise remains within a particular frequency range. If the sound pressure is constant, the bandwidth's core occurrence, independent to the signal's occurrence distribution, is equal to a pure tone's loudness. A signal's loudness changes dramatically when its bandwidth crosses a threshold. In line with the fact that the perceived frequency increases as the occurrence varies. A rough approximation of the following formula explains the nonlinear connection between frequency and the way the human ear perceives frequency

$$Mel(f) = 11260.10471n\left(\frac{1+f}{700}\right), \tag{9.10}$$

$$Mel^{-1}(f_{mel}) = 600(e^{ef}\,mel^{1127.01047} - 1)$$

Many knowledge systems use template matching for pattern counting, as seen in Figure 9.4. Each model class generates one or more patterns during training using clustering or other methods following feature extraction and dimension reduction. To establish a model's acceptance class, calculate its feature vector similarity and identify it. Speech thankfulness may measure standard assurance comparisons using comparison models. However, here is an uncommon assembly time difficulty on one dimension.

A quasi-stationary signal is the one used to convey speech. In addition to accurately describing the statistically normal distribution of speech

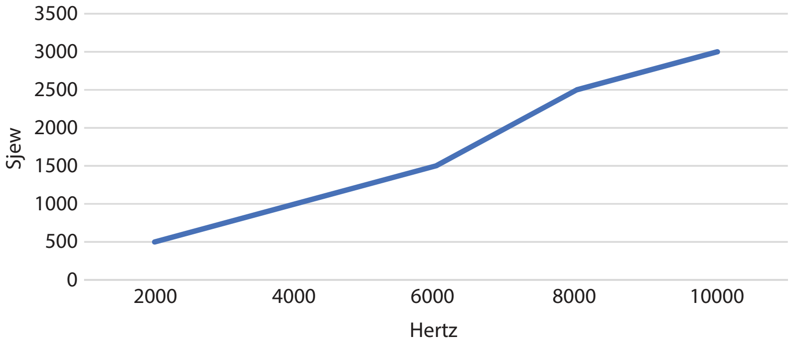


Figure 9.4 Mel–frequency relationship.

characteristics, HMM is a mathematical framework that can depict the dynamic changes of features in speech signals. Used for speaker identification and quasi-static time-varying analysis of speech signals, it is a powerful tool. The features of random processes may be characterized by this example, which arises out of nowhere. A chain was its original source. When it comes to phonetics, there are two types of knowledge: random phonetics and structured phonetics. It may take the form of a multiple vector sequence or a one-dimensional series of letters used for encoding or observation.

The various structures of secret Markov chains in HMM are determined by the parameters π and A, as can be shown in Tables 9.1 and 9.2. Figure 9.5 shows that the transition state in the left-to-right model with journey can only go after the left to the right, not the other way around. Due to the dynamic nature of speech signals, this model is well suited for simulating them, and it requires few computational resources. A simpler and more typical Markov chain is shown in Figure 9.5. It turns into a left-to-right model without crossing since it doesn't have any states.

Table 9.1 Parameter of HMM.

Parameter of model	**Justify it**
N	Number of model states
$A = \{a_{ij}\}$	Transition matrix state
$\pi = \{\pi_i\}$	Distribution for each state Probability starts
$B = \{b_j(o)\}$	Density function output probability

Table 9.2 Parameters for the HMM identification procedure.

Parameter	**Justify it**
O	Observe vector
M	number of Gaussian components per state
c_j	the first mixed Gaussian's J state weight
N	Gaussian normal probability density function
μ_j	J state's initial mixed Gaussian element mean vector
U_{ij}	J state's initial mixed Gaussian element covariance matrix

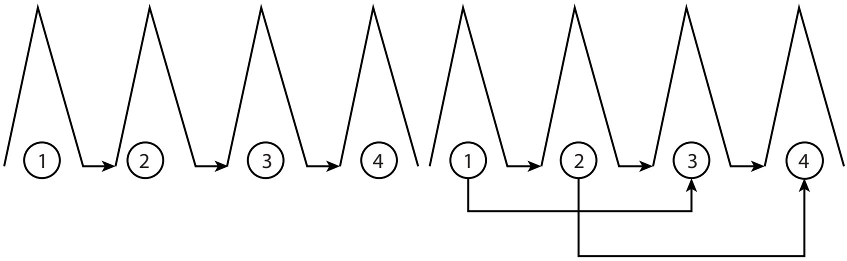


Figure 9.5 State diagram of an HMM.

9.2.2 Error Correction and Voice Scoring

A voice recognition-based spoken English learning system has been the subject of many study. Some address the most typical ways in which newcomers mess up their pronunciation, such using the wrong nasal sound or one of several similar vowels. Intonation, continual reading, and emphasis are just a few of the areas where certain courses concentrate on the specific challenges or strengths of English pronunciation. A different sort takes a system-wide approach to teaching English orally, making it more personable while still making the most of the computer's capabilities in accordance with the phonetic teaching technique. Achieving better English pronunciation is as simple as using speech recognition software, which is also essential for learning pronunciation. Based on this, a plethora of new research is required:

1. Try to find methods to assess how well words are spoken. Rhythm and other indicators of the performance of hyperphoneme speech vectors may be quantitatively measured using it, and it is also relevant to phoneme units. Pitch, tension, speed, and rhythm calculations, together with understanding the interrelationships of these four vocal acoustic structural elements, are the meat and potatoes of this challenge. Leave your rating for the how words, phrases, and sentences sound.
2. How to provide constructive criticism to students in a manner that is both helpful and courteous while they work to improve their phoneme-level pronunciation.

A computer-aided pronunciation learning system provides students with valuable feedback, one of which is their current pronunciation level. Consequently, the fundamental and central role of such a learning system is to automatically score English pronunciation. Measuring needs a point of reference or standard. They both rely on HMMs trained using references speech and reference corpus, which are the two most used approaches.

Eloquence evaluation has psychological, physiological, and sociological effects in addition to disciplinary ones (telephone, description, and environment). The great Speech Test offers key and target scores. Several scoring schemes exist presently. The exam's main components include test grouping decisions, intermediate score distortion, and test satisfaction decisions. A speech's first grade is based on its accuracy and effort. Test conditions and examinee material impact test results' reliability. In addition to environmental and human context, various application factors affect speech quality measurement tools Dynamic time warpage, HMM log purchases, HMM log succeeding findings, segmented distribution, long term, performance, and reliability time score probabilities are common. The speech model is used as a basis for several comparable computations in the aforementioned metrics.

Figure 9.6 shows an optimized algorithm that unevenly distorts and bends. The process of speech recognition involves finding the most similar path between two vectors, one being the speech signal that needs to be recognized and the other being the reference signal, over time. The goal is to obtain a regularized function with the smallest cumulative distance when the two vectors are matched. This is the earliest and most widely used method for speech recognition.

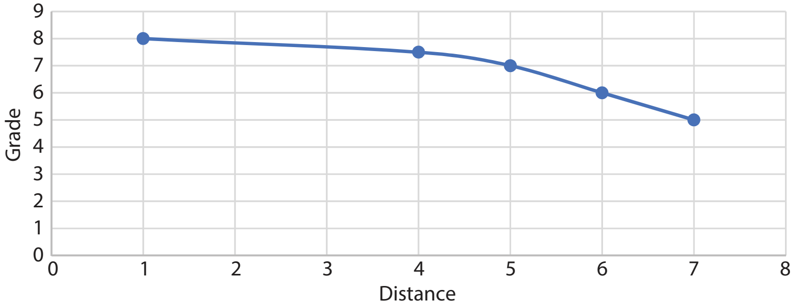


Figure 9.6 Graph showing the converted distance score.

The HMM scoring system is a method of evaluating speech by using a speech model. This model mainly considers the sound and tone of the speech and compares it to three distinct models: an acoustic model, a tone model, and an actual speech model. Figure 9.7 depicts marking scheme flow. Analyze the test speech and the model using speech recognition, then rate the difference. The trained acoustical and tone models will be the analytical baseline. During feature parameter extraction, the fundamental frequency trajectory and Mel cep-strum parameters are recovered for tone identification and sound recognition, respectively. Viterbi decoding divides the voice stream into monosyllabic segments, and the sound and tone models are compared for each syllable. The comparison test speech score is calculated using recognition results and a predesigned scoring method. Popular speech recognition systems in this scoring system include the hidden Markov the text describes various techniques such as model, tree net, and Viterbi algorithm for classifier design. It also explains the use of orthogonal expansion, K-means clustering, and Chebyshev approximation.

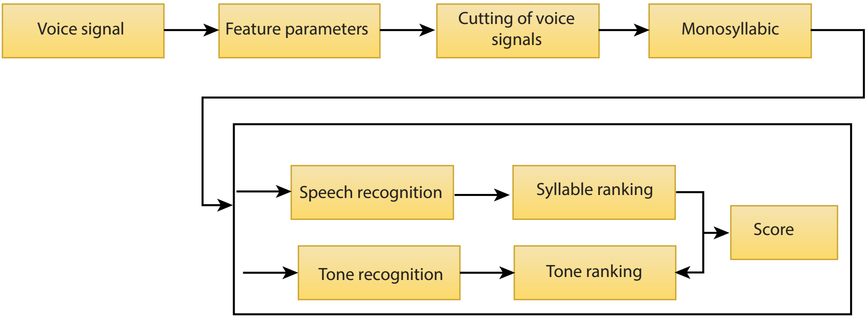


Figure 9.7 The scoring system's flowchart.

9.2.3 Deep Learning in NLP-Specific Applications

NLP relies on imitation where failure is avoided via imitation. Imitate others to gain beneficial resources. Kids mimic babbling. They imitate speech. Imitation helps English learners speak better. Similar to how the general population tends to mimic the actions of successful people, pupils who succeed in spoken English likewise have their every move mirrored. Strive to sound and act like native speakers of the intended language by mimicking their intonation and pronunciation. Mastering the art of mimicking speech intonation is essential and yields promising results. English pronunciation varies because the languages come from different families. The sounds of English consonants /f/ and /l/ are distinct despite their similarity. The lower lip between the top incisors creates a fricative sound gap for /f/. An elevated soft palate and a blocked nasal canal make this sound. A noise may be produced by forcing air to pass through the space between the tooth and its lip. The English phonetic system associates the friction consonant with the action of the lips and teeth with the sound /f/. A friction sound is produced when the lower lip lightly brushes against the top teeth when pronouncing. Air flows between the teeth and lips to create this sound. Stops /ş/ are one example of a sound that is absent from the phonology system compared to the English pronunciation system. It is important to focus on spoken English practice more while learning these phonetic symbols. To help students' muscles and organs adjust to the English system of pronunciation, practice this strategy that is distinct.

9.2.3.1 Application Process

Applying deep learning to natural language processing requires scientifically implementing gradient descent. Here is how the application procedure works in practice: (1) Create the appropriate model structure. Make sure the chosen neural network structure is reasonable, accomplish the goal of building the deep learning model framework, and combine it with the appropriate contents that need processing. (2) make sure the model is correct. Complete the method check, examine the existing gaps, and verify that they comply with the applicable rules by using the gradient descent approach in a reasonable manner. (3) Experience the impact of the model's initialization. After a thorough evaluation, fix any problems by scientifically improving the relevant models' parameters and optimizing the relevant models to account for any shortcomings. (4) make sure to update applicable models regularly. Model parameters that fail to satisfy the appropriate fitting requirements are gradually adjusted until they do so, all while making reasonable use of the regularization procedure.

9.2.3.2 Evaluation of Practical Metrics

1. Mark correctly the parts of speech and word segments.
   Depending on word segmentation needs, it may keep word order while merging it with an entirely novel word sequence. It is important to be precise when identifying parts of speech. Adjective, verb, etc., are all forms that this word may take. Can perform tasks such as named entity identification, semantic role tagging, and part of speech tagging by enhancing the use of deep learning methods to the problem.
2. Syntax for scientific parsing.
   A reasonable analysis of sentence grammar and its interrelationships is provided. Automated unit recognition of sentence syntactic components, connection sorting, scientific input of a given sentence, rational exploitation of grammar features, successful completion of the task of building a phrase framework tree, and effective handling are all outcomes of scientific applications of deep learning methods.
3. Thoroughly research word definitions.
   Prioritizing word meaning learning and using an appropriate unsupervised learning system are essential components of deep learning. Everyone should use this framework logically and experimentally the text when establishing the deep neural network model to get the most efficient expression form of word meaning, master phrase meaning vocabulary, and precisely analyze ambiguous words with the same name. The model optimization approach may be used to improve the expression accuracy and semantic richness of word vectors when there are many polysemy word vectors.
4. Advance the field of scientific emotion analysis.
   To effectively analyze emotions using deep learning, build an emotion analysis model, label relevant sentences, and use regulations and context characteristics to predict emotional characteristics. Examine document and phrase emotive colors. Deploying deep learning technologies may enhance the efficiency of processing natural language and enhance sophisticated emotional analysis.

9.3 Result

In order for a speech recognition engine to function, the training template relies on a corpus as its primary source of speech information. From a performance assessment standpoint, the validity and reliability of the findings are closely correlated with the corpus quality. Here are some things that a complete and standardized corpus should do:

1. Applicability: The material delves deep into a wide range of speech phenomena, providing ample coverage.
2. Representative: Speaking at a moderate pace, the speaker is representative of a broad range of ages, regions, and genders.
3. Reliability: The pronunciation material is congruent with the corpus markings, which are detailed.

As the system examines English pronunciation, its corpus recording personnel includes language teachers with classroom oral English teaching experience. Everybody requires time to read properly, and the material has sentences. The sentences include 1595 spoken English words. Someone must record the phrase time at the word level. The table below lists recorded gear and data. Later, it describes learning methods.

1. Choosing one's own class time: This technique ensures that learners have access to all course materials at all times, not only during scheduled class times.
2. Plan 30 days: Users will be able to learn in phases to this technique, which divides the learning material into 30 class times, with 30 phrases in each class time.
3. Review intensive: This method involves including fill-in-the-blank multiple-choice questions into the learning material. The goal is to enable users to identify the answers just by speaking them out.

This approach involves vigorous training for the software to address issues that have been known to cause errors in the past. The system's key component is HMM. There is a comparison of the two types of HMMs utilized in this system—continuous and semi continuous—and evaluates their recognition rates and recognition times. Figures 9.8–9.10 show the outcomes of the experiments.

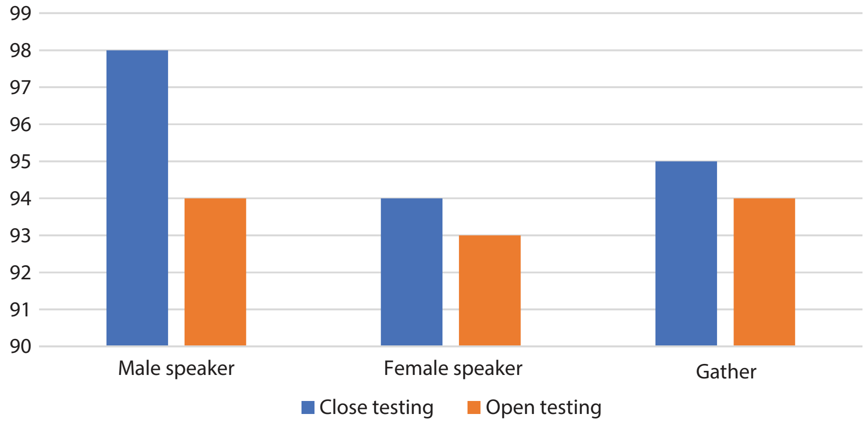


Figure 9.8 Tests of continuous HMM accuracy.

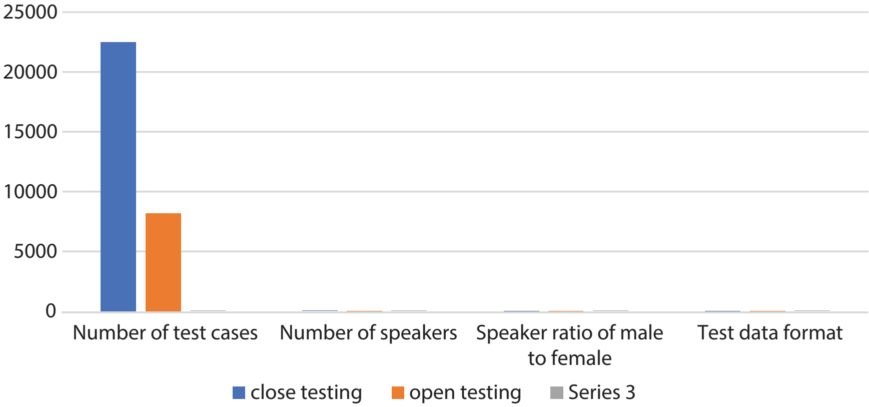


Figure 9.9 Results of continuous HMM correctness test.

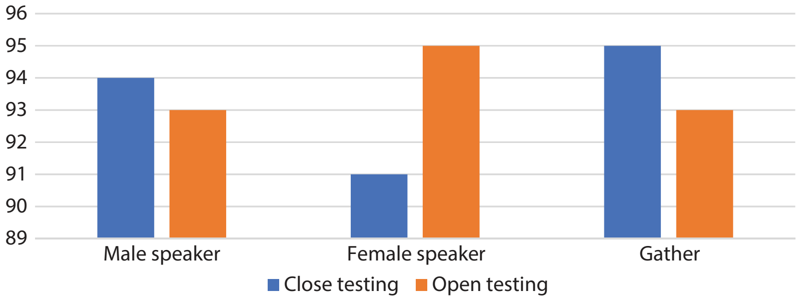


Figure 9.10 Semicontinuous HMM accuracy tests.

After each module had run its course, this test validated that the English language design model was functioning properly. The system's efficiency and ability to satisfy customer expectations are shown by studying system terms and contrasts and by assessing performance in real time. Tests and evaluations of speech recognition have been conducted, and the results are good, according to the data.

9.4 Conclusion

Incorporating natural language processing (NLP) into college spoken English courses improves upon current practices while giving instructors more leeway in how they structure their lessons. Flexible and interactive teaching activities that follow the four main neuro-linguistic programming concepts improve the confidence of learners in learning oral English, self-acceptance, goal-setting, and classroom participation. Improvements in students' oral competence are a side effect of natural language processing, which, when further refined and used, may help eradicate the phenomenon of "dumb English" and free oral English instruction from its present predicament. The goal of the native-speaker developers of the Spoken English Learning app for Android was to create a tool that would let people study and practice English at their own pace and in any environment. A user-friendly framework for learning and practicing English, including activities related to speech proficiency, speech measurements, broadcasting on radio, and oral communication, is provided by the system, which is designed by identifying certain essential tasks that must be done in the terminal.

The environment has a significant impact on speech recognition when using spoken English, and noise in the environment lowers the system's recognition degree. While this work does have some success in reducing Gaussian white noise using the final identification procedure, it falls short of completely eliminating the noise and has no discernible effect on speaking abilities. Improving his public speaking skills is another area he should focus on. The use of phonetic knowledge-based products has become more widespread in recent years. Research into speech therapy is receiving increasing funding from academic institutions. In the future, individuals will be able to express themselves more clearly, which will facilitate their daily lives and contribute to the expansion of human knowledge.

References

1. Kamath, U., Liu, J., Whitaker, J., *Deep Learning for NLP and Speech Recognition*, vol. 84, Springer, Cham, Switzerland, 2019.
2. Xu, J. and Li, T., Application of Multimodal NLP Instruction Combined with Speech Recognition in Oral English Practice. *Mobile Inf. Syst.*, 2022, 140, 2022.
3. Kim, C., *et al.*, End-to-end training of a large vocabulary end-to-end speech recognition system. *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, IEEE, 2019.
4. Suominen, H., *et al.*, Benchmarking clinical speech recognition and information extraction: new data, methods, and evaluations. *JMIR Med. Inf.*, 3, 2, e4321, 2015.
5. Montenegro, C., Santana, R., Lozano, J.A., Analysis of the sensitivity of the End-Of-Turn Detection task to errors generated by the Automatic Speech Recognition process. *Eng. Appl. Artif. Intell.*, 100, 104189, 2021.
6. Chantarutai, N., SpeechBuddy: English Learning Application with Automatic Speech Recognition and Natural Language Processing for Improving Students' Speaking Skill, 2018.
7. Shi, L., Application of big data language recognition technology and GPU parallel computing in English teaching visualization system. *Int. J. Speech Technol.*, 25, 3, 667–677, 2022.
8. Teller, V., *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, pp. 638–641, Upper Saddle River, New Jersey, 2000.
9. Gershov, S., *et al.*, Automating medical simulations. *J. Biomed. Inf.*, 144, 1044465, 2023.
10. Li, Y. and Fung, P., Language modeling with functional head constraint for code switching speech recognition. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.
11. Alharbi, S., *et al.*, Automatic speech recognition: Systematic literature review. *IEEE Access*, 9, 131858–1318765, 2021.
12. Kumar, Y. and Singh, N., A comprehensive view of automatic speech recognition system-A systematic literature review. *2019 International Conference on Automation, Computational and Technology Management (ICACTM)*, IEEE, 2019.
13. Yue, T., English spoken stress recognition based on natural language processing and endpoint detection algorithm. *Int. J. Electr. Eng. Educ.*, 60, 2_suppl, 416–426, 2023.
14. Bhatt, S., Jain, A., Dev, A., Acoustic modeling in speech recognition: a systematic review. *Int. J. Adv. Comput. Sci. Appl.*, 11, 4, 141, 2020.
15. Chen, X., *et al.*, Exploiting future word contexts in neural network language models for speech recognition. *IEEE/ACM Trans. Audio Speech Lang. Process.*, 27, 9, 1444–1454, 2019.

16. Calefato, F., *et al.*, An empirical simulation-based study of real-time speech translation for multilingual global project teams. *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, 2014.
17. Sun, W., The research and implementation feasibility analysis of an intelligent robot for simulating navigational English dialogue under the background of artificial intelligence. *Comput. Sci. Inf. Syst.*, 19, 3, 1533–1548, 2022.
18. O'Shaughnessy, D., Interacting with computers by voice: automatic speech recognition and synthesis. *Proc. IEEE*, 91, 9, 1272–1305, 2003.
19. Shan, C., *et al.*, Investigating end-to-end speech recognition for mandarin-english code-switching. *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019.
20. Dubey, A.K., Kumar, A., García-Díaz, V., Sharma, A.K., Kanhaiya, K., Study and analysis of SARIMA and LSTM in forecasting time series data. *Sustain. Energy Technol. Assess.*, 47, 101474, 2021.
21. Heigold, G., *et al.*, Discriminative training for automatic speech recognition: Modeling, criteria, optimization, implementation, and performance. *IEEE Signal Process Mag.*, 29, 6, 58–69, 2012.
22. Moore, R.C., Using natural-language knowledge sources in speech recognition, in: *Computational Models of Speech Pattern Processing*, pp. 304–327, Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.
23. DeVilliers, E.M., Implementing voice recognition and natural language processing in the NPSNET networked virtual environment, Diss., Naval Postgraduate School, Monterey, California, 1996.
24. Kumar, A., Chatterjee, J.M., Díaz, V.G., A novel hybrid approach of SVM combined with NLP and probabilistic neural network for email phishing. *Int. J. Electr. Comput. Eng.*, 10, 1, 486, 2020.
25. Kanhaiya, K., Naveen, Sharma, A.K., Gautam, K., Rathore, P.S., AI Enabled-Information Retrival Engine (AI-IRE) in Legal Services: An Expert-Annotated NLP for Legal Judgements. *2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS)*, Trichy, India, pp. 206–210, 2023, doi: 10.1109/ICAISS58487.2023.10250733.

10

Deep Learning-Based Neuro Computing to Classify and Diagnosis of Ophthalmology by OCT

D. Arul Pon Daniel[1]*, Santhana Sagaya Mary A.[2] and S. Chidambaranathan[3]

[1]Department of Computer Science, Jayarani Arts and Science College for Women, Nethimedu, Salem, India
[2]Department of Information Technology, St. Xavier's College (Autonomous), Palayamkottai, Tamil Nadu, India
[3]Department of Computer Applications, St. Xavier's College (Autonomous), Palayamkottai, Tamil Nadu, India

Abstract

The goal of this research was to find how to use deep learning models with optical coherence tomography (OCT) pictures to screen for retinal diseases and identify lesions. Ophthalmologists carefully labelled 37,138 OCT pictures taken from 775 individuals. To determine the kind and location of lesions or illnesses from photos, many deep-learning models were created, such as YOLOv3 and ResNet50. The models were tested using a group of independent patients who were not part of the study. There was a 98.5% performance accuracy, 98.7% sensitivity, 98.4% specificity, and 97.7% F1 score for the binary classification of oct pictures with or without lesions. The models successfully identified age-related macular degeneration and vitreomacular traction syndrome in multiclass multilabel disease classification with F1 scores of over 97% and 98%, sensitivity and specificity, and accuracy, respectively. Recalls for various kinds of lesions varied from 87.0% to 98.2% when it came to detecting their locations. Models built using deep learning might be a great resource for ophthalmologists when it comes to screening, classifying, and diagnosing retinal diseases.

**Corresponding author*: lcmapd16@gmail.com

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (159–174) © 2025 Scrivener Publishing LLC

***Keywords*:** OCT, YOLOV3, Fi score, accuracy, deep learning

10.1 Introduction

Half of the world's over 2.2 billion individuals with visual impairments might have their condition improved or averted with simple measures. Risk of blindness and technological degradation associated with untreated retinal disorders. OCT uses two-dimensional cross-sectional images to assess retinal health. These pictures may be utilized to examine retinal anatomy and disease on different levels. B-scan retinal lesions must be manually detected, which is time-consuming and specialist. Due to their backgrounds and skills, ophthalmologists diagnose retinal disorders differently. However, an automated illness screening system might provide accurate, fast, and timely findings. Therefore, this study aimed to construct an AI system to help ophthalmologists evaluate retinal disorders using OCT pictures.

Recently developed deep-learning methods have considerably boosted medical imaging. To improve image classification and learning performance, deep learning uses a multilayer neural net with convolutional layers. Several neural networks were trained to identify, malignant mesothelioma, breast cancer and coronary artery fibrous plaque in histological pictures image deep-learning models have been examined by macular visual function researchers. Predicted retinal lesions using electronic medical data and OCT images and built a deep learning model to pretend outcomes and recommend AMD therapies. Not only were deep learning models created for picture classification, but also for OCT image contour identification and layer segmentation. identified changes to photoreceptors caused by macular disorders using a segmentation model. Scientists have now created models that can detect lesions in photos and categorize them. A neural network with attending faculty called Lesion-Aware to identify and emphasize retinal lesions was created. While there has been progress in the area of automated, high-performance diagnosis of ocular diseases, most research has ignored lesion areas in OCT images in favor of disease categorization, leaving diagnostic information lacking. When it comes to ophthalmologists verifying and examining the prediction findings, however, the interpretability of the deep learning model is equally crucial. Class inactivation map (CAM)-based technology was employed in a few experiments to show where the model was focusing its attention, but this approach cannot distinguish between various kinds of lesions that appear on the same picture. A preferable method of interpreting the prediction findings would emphasize the lesion locations in addition to the type categorization; this would aid in clinical diagnosis. This work, aimed to address these issues by collecting optical coherence tomography (OCT) pictures and classifying them according to two disease categories' worth of lesion kinds. Classifying retinal illnesses and detecting lesion kinds and areas was accomplished via the development of an intelligent system. To improve the execution of deep learning models for predictions, transfer learning and ensemble learning approaches. The ability to allow model creation based on a minimal number of data is one key advantage of applying transfer learning. To outperform a single classifier, ensemble learning integrates reasoning consequence from multiple models to cast a vote for a final anticipation. Due to inherent bias and learning limitations in individual models, combining models may enhance overall accuracy and generalizability, reducing the likelihood of mistakes. In addition, YOLOv321 is used as the basis for an object identification algorithm that could forecast the kinds and locations of lesions. This work gives a thorough evaluation of several OCT-based models for eye disease prediction and detection.

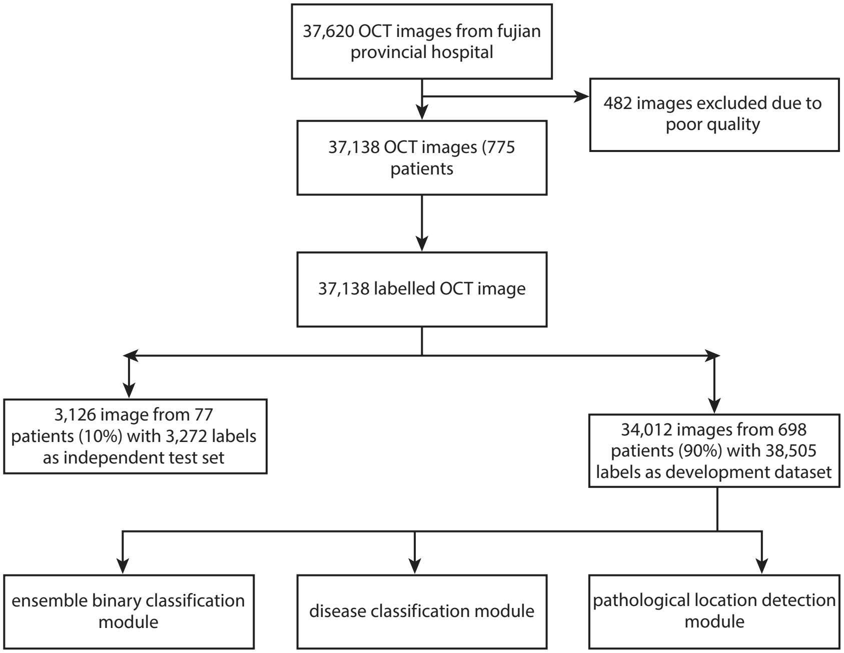

37,620 OCT images from fujian provincial hospital
482 images excluded due to poor quality
37,138 OCT images (775 patients
37,138 labelled OCT image
3,126 image from 77 patients (10%) with 3,272 labels as independent test set
34,012 images from 698 patients (90%) with 38,505 labels as development dataset
ensemble binary classification module
disease classification module
pathological location detection module

Figure 10.1 Flowchart for OCT image.

10.2 Methodology

In Figure 10.1, the whole process flowchart of this research is seen, from B-scan pictures from Fujian Administrative district Hospital. With the exclusion of 482 low-quality photos (1.3% of the total) with unclear retinal structures, the illustration size was reduced to 37,138 images (775 patients). The SPECTRALIS system from Heidelberg Engineering in Heidelberg, Germany was used to collect all of the OCT pictures, which were then saved in a PNG format with a resolution of 767 × 496.

10.2.1 The Training and Labeling of Images

Before the image tagging procedure, all patient identity information from the OCT pictures was appropriately deleted. Two board-certified ophthalmologists then used an in-house online program to analyze each OCT image. A senior ophthalmologist was consulted if the two had any disagreements about the labelling of the picture. The diagnosis was then finalized. Ophthalmologists used a rectangular box to indicate the kind and location of retinal pathological lesions when an OCT B-scan picture showed a lesion. It is important to keep in mind that an optical coherence tomography (OCT) picture might show a healthy control, several labels indicating different retinal abnormalities, or no lesions at all. The analysis of OCT images involves ophthalmologists identifying lesions related to two disease groups: vitreomacular traction syndrome (subretinal fluid, epiretinal membrane, macular pucker, full-thickness retinal prominence, retinal detachment) and age-related macular degeneration (outer retina atrophy, choroid atrophy, retinal atrophy, ocular hemorrhage, exudation, pigment epithelial detachment, retinal oedema). It randomly chose all images from 10% of patients to create a commutative model test set. Lesions were seen in 41.2% of 3,126 pictures from 77 individuals. A 4:1 ratio was used to randomly split the remaining photos from the expended 90% of patients into training and substantiation sets. We trained and validated the models using the training and validation sets, and then we used the independent test set to see how well they did. No data leakage happened while evaluating the model's performance on the autonomous test set since the patients in that set were distinct from the ones in the training and validation sets.

10.2.2 Progress in Intelligent System Development

Building an AI with three DL modules was the goal of this research (Figure 10.2). The first component is a multi-classifier ensemble deep learning module that can distinguish between healthy and unhealthy OCT pictures. These five deep learning models—Alex Net, Dense Net, InceptionV3, ResNet50, and VGG16—are built on convolutional neural networks and serve as binary classifiers. Deep convolutional layers, organization layers (feature dimensionality reducers), dropout, and batch normalization are all functional components of these deep learning classifiers that extract and handle complex visual information. Train each classifier using its data using a transfer learning procedure that began with pre-trained weights trained on ImageNet. The classifiers were learned using pre-trained

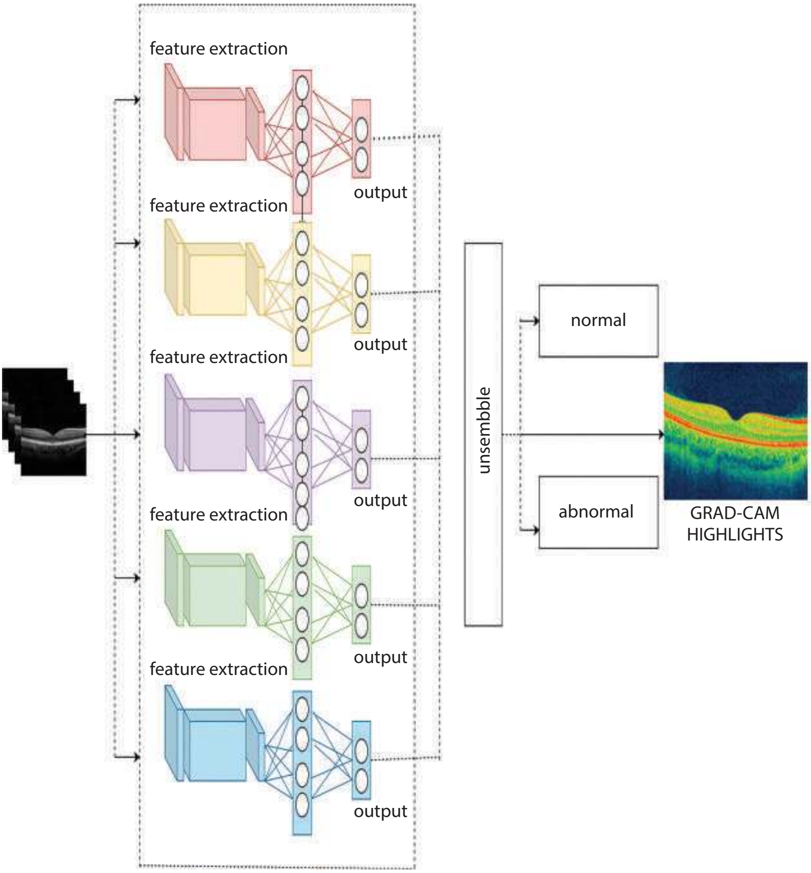


Figure 10.2 Deep learning system framework.

weights from ImageNet. Then these training results to data and used to train the models. This allowed us to train the models on a small dataset. Improved generalizability and prediction accuracy were achieved by the application of ensemble learning, which integrated all findings from five models to vote for a final prevision. To illustrate and clarify these models, CAMs for the test pictures were generated using Grad-CAM32. These CAMs emphasized the input areas that were crucial for the models to make predictions. Additional modules for pathologic allocation identification and multiclass multilabel disease classification were created because the binary classification module is unable to identify the illness or the location of a lesion on an image. After the ensemble binary classification model flagged any OCT pictures as potentially abnormal, the disease classification module determined whether the images belonged to the vitreomacular traction syndrome or AMD categories. Once again, transfer learning was used to train the illness categorization module, which was based on the ResNet50 model. To further distinguish between age-related macular degeneration and vitreomacular traction syndrome, two YOLOv3 models 21 were created for pathological site recognition. As the detection module finds the item in the shot, DarkNet53, an image feature extraction module in YOLOv3, builds characteristic maps.

In this research, the perception module used the COCO dataset and then adapted it to work with its data by using the pre-trained weights of DarkNet53, which were trained using ImageNet. Multiple diseases or pathologies might coexist in a single picture; hence, the illness categorization and pathologic site detection modules were developed with multiclass multilabel prediction in mind. After the final prediction, three independent modules' results were combined to provide comprehensive picture diagnoses. A server with four 1.90 GHz Intel Xeon Gold 6140 CPUs and four GeForce RTX 2080 Ti graphics cards runs Py Torch-based deep learning models. Using a batch size of 16, an impulse of 0.9, and a learning rate of 0.001. Early pausing may be used to supervise model preparation and avoid over fitting.

10.2.3 Evaluation of Performance

Quality, sensitiveness, specialness, F1 score, and area nether the receiving system operating symptomatic curve were some of the measures used to measure the classifiers' prediction ability. Also included as references were the confusion matrices. Recall and precision were used to evaluate the problematic location detection module.

10.3 Results

Table 10.1 displays the sociology data of the patients as well as the illness labels and lesion kinds. Among the 37,138 OCT photos taken from 775 individuals, 13,106 images (35.3%) were marked as having lesions, while the other images were considered normal. This represents 47.7% of the total. Atrophy of the outer retina was detected 403 times (1.0%), but motear pucker was detected 6,980 times (17.7%) less often.

Table 10.1 The OCT image data statistics.

Type	Dataset development	Test set independent	Total
Female (%)	401(57.2)	40(52.8)	441(56.8)
Images	35.3	3126	37,138
Labels	38,505	3272	41,777
Patients	698	77	775

Figure 10.3 displays the results of the binary categorization module's testing on an autonomous test set consisting of 3,126 OCT pictures from 77 patients. This set was used to distinguish between illness images and healthy control images. By combining the results of all five models into one, the musical organization model outperformed the individual models on every metric: area under the receiver operational characteristic curve (98.1%), accuracy (98.5%), sensitivity (98.7%), particularity+ (98.4%), and F1 score (97.7%).

Figure 10.4 displays the ensemble model's confusion matrix at the same time as all of the models' receiver operating characteristic curves. While making 3,126 predictions, the confusion matrix uncovered 47 erroneous instances, consisting of 34 false positives and 13 false antagonistic. Eight instances of macular pucker, two instances of haemorrhage and exudation, one instance of cystoid macular oedema, one instance of pigment epithelial withdrawal, and one instance of retinal dimension enhanced by oedema were among the 13 false negatives. The neural network models were visually explained using Grad-CAM, which helped to understand how the deep learning model made predictions. To show the areas that the model focused

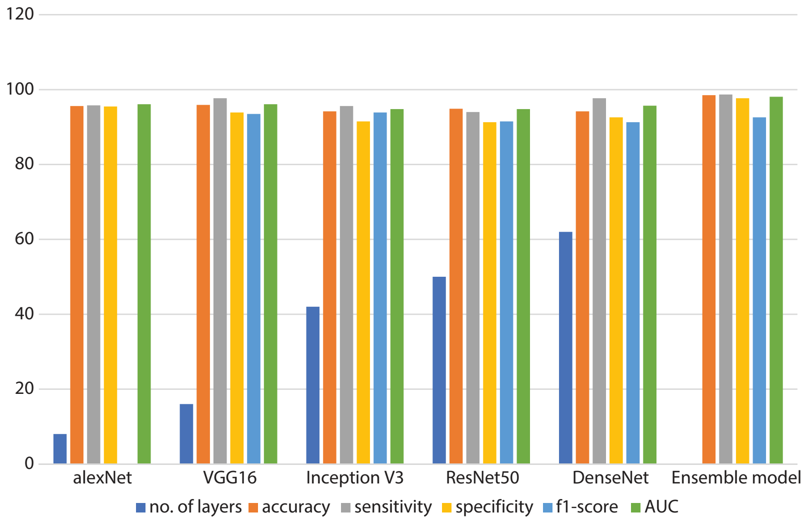


Figure 10.3 Deep learning binary classifiers' results on a controlled experiment.

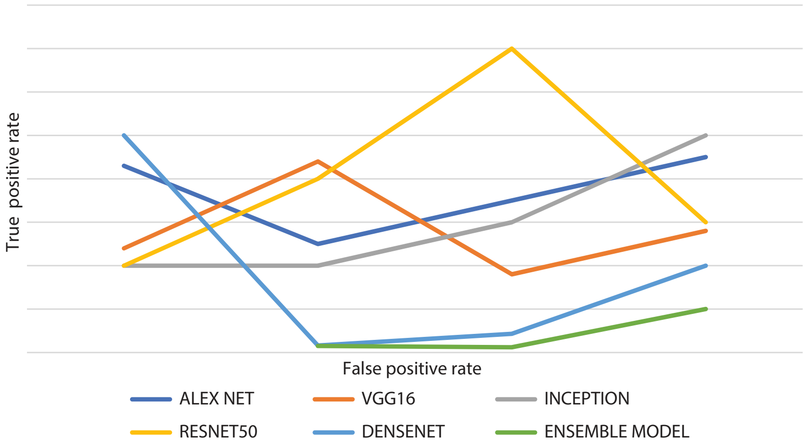
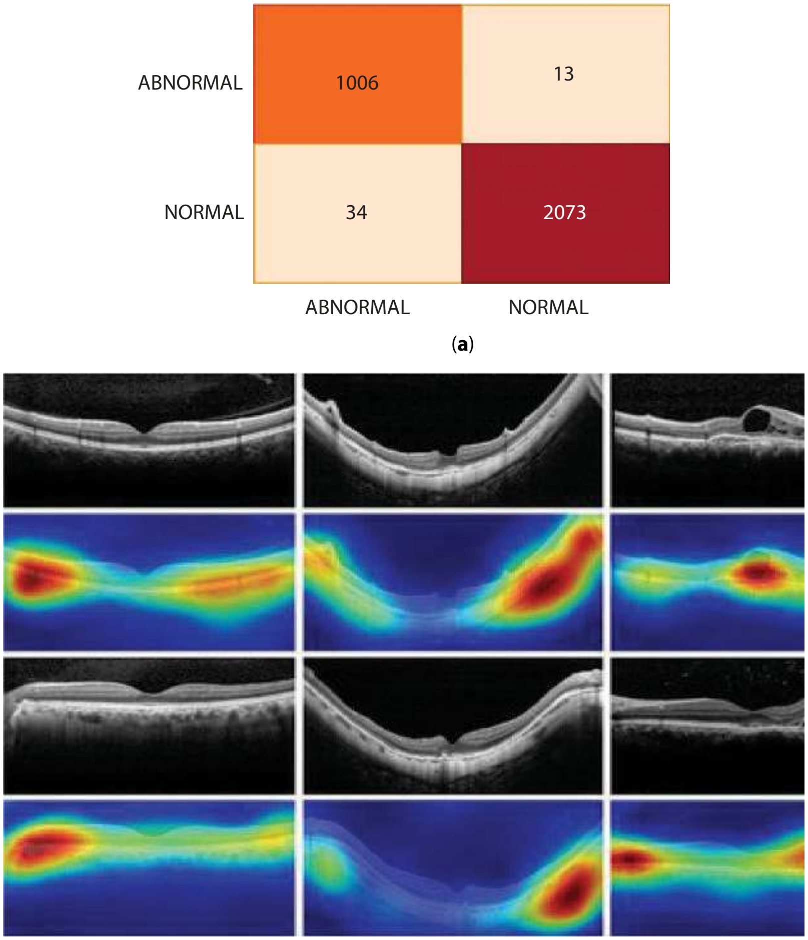


Figure 10.4 Evaluation of binary categorization module performance.

on, it created a heatmap to superimpose over the OCT picture. Two instances of accurate and inaccurate predictions are in Figure 10.5. The model correctly identified the lesion location on the OCT image and gave it more weight in Figure 10.5a, but in Figure 10.5b, it produced erroneous negative predictions and Grad-CAM emphasized healthy and normal regions.



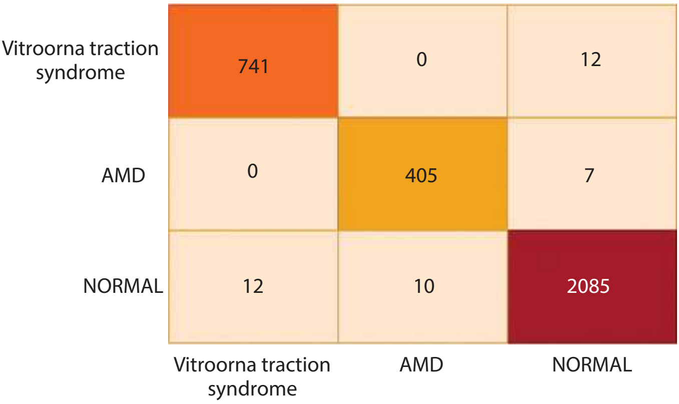
(**a**)

(**b**)

Figure 10.5 (a) OCT heatmaps by Grad-CAM, (b) OCT heatmaps by Grad-CAM.

To determine whether the suspicious sample falls within the category of AMD or vitreomacular traction syndrome, Figures 10.6 and 10.7 provide the disorder matrix and public presentation prosody for the illness categorization module on the autonomous test set.

For extracurricular traction syndrome, the prediction performance was 99.3% accuracy, 98.4% sensitivity, 99.5% specificity, and 98.4% F1 score; for related macular degeneration (AMD), the detection performance was 99.5% accuracy, 98.3% sensitivity, 98.3% specificity, and 97.9% F1 score. Total binary classification accuracy.



	Vitroorna traction syndrome	AMD	NORMAL
Vitroorna traction syndrome	741	0	12
AMD	0	405	7
NORMAL	12	10	2085

Figure 10.6 Module for illness categorization confusion matrix.

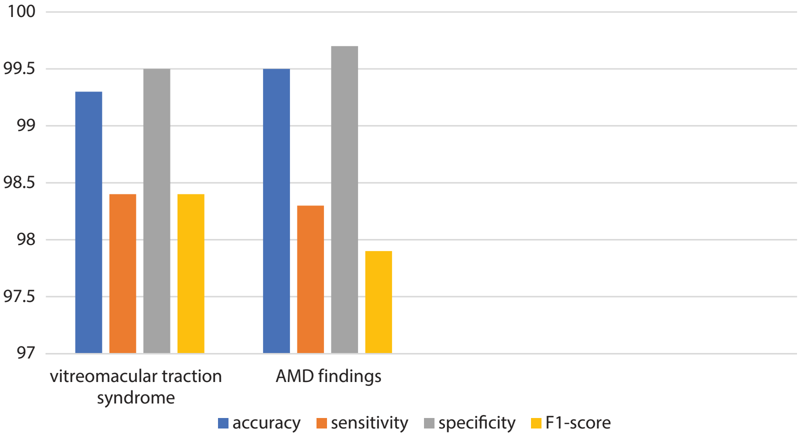


Figure 10.7 Results from the separate test set for the disease classification module.

Both AMD and vitreomacular traction syndrome have their unique pathology kinds and sites, which led to the development of two YOLOv3 models for detection. Results from the medical science detection on the separate test set, including the confusion matrix and performance indicators.

10.4 Discussion

Visual impairment, which may lead to permanent blindness and diminished quality of life, is more common with age and is associated with bad lifestyle choices. This highlights the vital importance of early discovery, intervention, and prevention. Due to medical resources and ophthalmologist shortages, many retinal patients wait for diagnosing and treatment. The current advances in AI and deep learning may solve the problem. Studies indicate AI systems categorize and segment pictures well. In clinical practice, AI-based diagnosis is still developing. These issues stem from prediction accuracy and the lack of high-quality annotated OCT image data. Recent advances in AI and deep learning are noteworthy. AI algorithms efficiently categorize and segment photos. Due to prediction accuracy difficulties and a scarcity of high-quality annotated OCT image data, AI-based diagnosis is still in its infancy in clinical practice. Collect 37,138 OCT pictures and mark each one with two illness categories and ten kinds of retinal diseases. This will allow for a thorough analysis of the research. Images of illnesses may be identified, disease types can be classified, and pathologies can be highlighted with the use of three submodules: Musical organization binary arrangement, disease classification, and pathology catching. Create this three-part system rather than a one-step model to reduce the effect of negative feedback and make the most of ensemble classification as the majority of the training data are photos of healthy controls (64.7%). Using the ensemble learning approach, combine the output of five separate models in the first binary grouping faculty. The cast of characters model achieved better results than any of the individual models, although it was more complex and took longer to train. A notable difference between the models with and without layers is that the ones with fewer layers performed better. Overfitting occurs more often on smaller datasets in models with higher levels of complexity, thus this discovery is not surprising. The previous study showed that using transfer learning and ensemble learning might reduce the impact of overfitting, bring about good impacts, and achieve greater performance than individual models, which helped alleviate this issue. Ophthalmologists also benefited from the visualization and highlighting of the model's focus at the lesion sites. A second disease classification module distinguished between AMD and vitreomacular traction syndrome, two retinal diseases, based on OCT images. The model has good accuracy and balanced false positives and negatives after adding prediction information. Additionally, the prediction categories were expanded into ten kinds of disorders in the final pathology detection module. The performance dropped compared to earlier modules and varied among various diseases, even though it was the most thorough study and prediction. This was because of the unequal distribution of pathology kinds. There may still be clinical significance, however, since its accuracy was still rather strong, falling anywhere between 87.0% and 98.2%. The model used three multistage deep learning modules to analyze OCT images. This allowed for stage-specific predictions and a single ResNet50 model sans the binary classification module to test the effectiveness of the multistage deep learning system over the single merged model for direct sickness classification. One YOLOv3 model, independent of the aforementioned categorization modules, can directly identify the locations of ten different retinal diseases. rely on the multistage system, whereby each module was fine-tuned for its unique job, the data imbalance's impact was reduced at each stage, and overall performance improved.

Accurate, understandable diagnosis. After differentiating choroidal neovascularization, diabetic macular oedema, drusen, and normal OCT images, divide retinal disease ocular fundus images into three groups for deep learning classification Screening for fundus abnormalities such AMD, DR, ERM, RVO, and likely glaucoma using deep learning. On the other hand, the pathology detection method may provide more useful insights than the GradCAM heatmaps employed in these experiments for model interpretation. Additionally, using a combination of pathological and thickness characteristics, they were able to use deep learning to identify fifteen different kinds of pathologies in OCT images, which is a significant increase over previous work. Multistage classification and lesion detection approach may only work with a small dataset, but it may nevertheless validate and supplement previous work. The research has several limitations. To start, the optical coherence tomography (OCT) data came from a single hospital, which may only be representative of a limited regional sample. It is also possible that there is an imbalance in the sample size and distribution of retinal diseases. Ophthalmologists may use the technology for routine duties since it can identify many serious retinal abnormalities. It is reasonable to doubt the model's ability to detect previously unidentified lesions or illnesses in light of the available data and the limitations of the model. To complete the diagnosis, it is necessary to consult with ophthalmologists. Improving the therapeutic usefulness of the model and decreasing its biases requires increasing the number of patients, sources, and pictures. This should be a major focus for future work. Also, employ rectangles to mark the areas of pathology in the samples; however, also consider using retinal layer segmentation, which might provide more accurate results for identifying individual lesions. Rectangle labelling is easier now, but segmenting the retinal layers will allow for more accurate and precise lesion diagnosis in the future. Finally, the method in a clinical setting.

The clinical translatability has not been evaluated, even though the dry trials showed that the model performed well. This technique will be useful in assisting ophthalmologists with their diagnostic work. Expert judgments are required to complete the diagnosis, yet the system is not intended to substitute ophthalmologists. A high-quality diagnosis and treatment suggestion needs more than just optical coherence tomography (OCT) pictures; other clinical data, such as a patient's medical history and results from other tests, are also essential. Adding optical coherence tomography (OCT) pictures to electronic medical records will allow for a more thorough diagnostic and recommendation system, in the future.

10.5 Conclusions

In this study, ophthalmologists classified 37,138 optical coherence tomography (OCT) pictures from 775 individuals into 10 pathology classes, two of which relate to diseases: AMD and vitreomacular traction syndrome. A multi-stage intelligent system screens OCT images for illnesses and pathologies. A binary classification module, illness classification module, and pathology site detection module can analyze the OCT picture at various degrees of resolution, identify diseases and pathologies, and locate them. The results from the independent test set were positive for all three modules. This system can detect and forecast eye illnesses using OCT pictures, which might be useful for ophthalmologists in their clinical practice.

Bibliography

1. Jena, O.P., *et al.*, (Eds.), *Machine Learning and Deep Learning in Efficacy Improvement of Healthcare Systems*, CRC Press, Boca Raton, Florida, 2022.
2. Nayak, J., *et al.*, Firefly algorithm in biomedical and health care: advances, issues and challenges. *SN Comput. Sci.*, 1, 6, 311, 2020.
3. Verma, S., *et al.*, Tele-Health Monitoring Using Artificial Intelligence Deep Learning Framework, in: *Deep Learning for Targeted Treatments: Transformation in Healthcare*, pp. 199–228, 2022.
4. Ibrahim, Y., *et al.*, A Systematic Review on Retinal Biomarkers to Diagnose Dementia from OCT/OCTA Images. *J. Alzheimer's Dis. Rep.*, 7, 1, 1201–1235, 2023.
5. Dande, P. and Samant, P., Acquaintance to artificial neural networks and use of artificial intelligence as a diagnostic tool for tuberculosis: a review. *Tuberculosis*, 108, 1–9, 2018.

6. Charles, J., S-075: PANEL: THE 10TH ANNUAL RAM BOWL. *Aerosp. Med. Hum. Perform.*, 89, 3, 287, 2018.
7. Shah, O.J., *et al.*, 22nd Postgraduate Presentation Programme 2018. *JMS SKIMS*, 1–84, 2018.
8. York, W.H., Experience and theory in medical practice during the later Middle Ages: Valesco de Tarenta (fl. 1382–1426) at the court of Foix, The Johns Hopkins University, 2004.
9. Juras, J.A., Peripheral and Central Glucose Flux in Type I Diabetes, 2022.
10. Farsana, W.F. and Kowsalya, N., Congenital Malformations Ultrasound-Based Fetal Prediction using a Computer-Aided Diagnosis System.
11. Hill, C.D., Segmentation of oral optical coherence tomography with deep learning, 2023.
12. Preethi, S., Retinoblastoma-demographic features, clinical presentations, treatment modalities and outcomes in a tertiary care centre, Diss., Aravind Eye Hospital and Postgraduate Institute of Ophthalmology, Madurai, 2006.
13. Nguyen, V., *et al.*, Clinical and social characteristics associated with reduced visual acuity at presentation in Australian patients with neovascular age-related macular degeneration: a prospective study from a long-term observational dataset. The Fight Retinal Blindness Project. *Clin. Exp. Ophthalmol.*, 2017 Aug 25, [Epub ahead of print]. *BMC Ophthalmol.*, 17, 1, 158, 2017.
14. Tsoutsanis, P.A. and Charonis, G.C., Congenital orbital teratoma: a case report with preservation of the globe and 18 years of follow-up. *BMC Ophthalmol.*, 21, 1, 1–7, 2021.
15. Maheshwari, A., *et al.*, Novel treatments in optic pathway gliomas. *Front. Ophthalmol.*, 2, 9926735, 2022.
16. Burri, S. R., Kumar, A., Baliyan, A., Kumar, T. A., Predictive Intelligence for Healthcare Outcomes: An AI Architecture Overview, in: *2023 2nd International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)*, Villupuram, India, pp. 1–6, 2023, doi: 10.1109/ICSTSN57873.2023.10151477.
17. Pouyeh, B., *et al.*, New horizons in one of ophthalmology's challenges: fungal keratitis. *Expert Rev. Ophthalmol.*, 6, 5, 529–540, 2011.
18. Bhatia, S., Nanoparticles types, classification, characterization, fabrication methods and drug delivery applications, in: *Natural Polymer Drug Delivery Systems: Nanoparticles, Plants, and Algae*, pp. 33–93, 2016.
19. Swarna, S. R., Kumar, A., Dixit, P., Sairam, T. V. M., Parkinson's Disease Prediction using Adaptive Quantum Computing, in: *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, Tirunelveli, India, pp. 1396–1401, 2021, doi: 10.1109/ICICV50876.2021.9388628.
20. Maurya, R.P., *et al.*, Pattern of Presentation of Retinoblastoma in North India: A Teaching Hospital Survey. *J. Sci. Res.*, 66, 1, 186–190, 2022.
21. Chua, J., *et al.*, Diabetic retinopathy in the Asia-Pacific. *Asia-Pac. J. Ophthalmol.*, 7, 1, 3–16, 2018.

22. Wright, S., *et al.*, Operational Assessment of Color Vision, p. 0029, 2016.
23. Kumar, A., *et al.*, IoT based arrhythmia classification using the enhanced hunt optimization-based deep learning. *Expert Syst.*, 40, 7, e13298, 2023.
24. Wani, S., Ahuja, S., Kumar, A., Application of Deep Neural Networks and Machine Learning algorithms for diagnosis of Brain tumour. *2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, IEEE, 2023.
25. Choudhary, N., Singh Rathore, P., Kumar, L., Rajaan, R., Sharma, A., Sinha, D., ResNet-50 Powered Masked Face Detection: A Deep Learning Perspective. *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, Pune, India, pp. 1–5, 2024, doi: 10.1109/I2CT61223.2024.10543563.

11

Deep CNN-Based Multi-Image Steganography: Private Key

S. Pavan Kumar Reddy[1]*, K. Suresh Kumar[2], Madhu G.C.[3] and Pavitar Parkash Singh[4]

[1]Department of AI & DS, BVRIT, Narsapur, India
[2]MBA Department, Panimalar Engineering College, Varadarajapuram, Poonamallee, Chennai, India
[3]Department of ECE, Mohan Babu University (Erstwhile Sree Vidyanikethan Engineering College), Tirupati, India
[4]Mittal School of Business, Lovely Professional University, Phagwara, Punjab, India

Abstract

Deep multi-image steganography using private keys is presented in this study. Many techniques based on deep convolutional neural networks (CNNs) have been suggested as a means to cover many hidden pictures less than a single-cover image. Unfortunately, traditional approaches generally decode all concealed information and do not offer access to individual secret images, making them vulnerable to secret information leaking. Private keys for secret images are introduced as a solution to the issue. By using a single-cover picture to encrypt several hidden images, this method creates a visually identical container image with encrypted secret information. Additionally, every hidden image's private key is produced concurrently. Limit each key to decipher one secret picture that protects the other hidden images and keys. The approach uses deep networks to conceal and reveal information. After receiving the cover image and secret photographs, the hidden network extracts upper-level features to build private keys. Next, create a container image with obtained properties and secret keys. Conversely, the revealing network looks at the container image, finds certain high-level characteristics, and then uses those features and a matching private key to decode a secret image. Achieving great security while efficiently hiding and revealing numerous secret images is shown experimentally by the suggested approach.

**Corresponding author*: pavankumar.s@bvrit.ac.in

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (175–190) © 2025 Scrivener Publishing LLC

***Keywords*:** Multi-image steganography, private key, CNN model, secret images, container image

11.1 Introduction

The goal of steganography, as an algorithm, is to make an item seem identical to its original form while simultaneously hiding private information [1]. The fundamental idea behind steganography is to ensure that only authorized clients may access secret information while keeping its existence and content hidden from others [2]. Confidential information has been securely sent via a variety of carriers, including tangible items, written words, audible noises, and data packets sent over a network. Modern digital steganographic algorithms often make use of digital images as carriers as shown in Figure 11.1. (i.e., image steganography) [3–5].

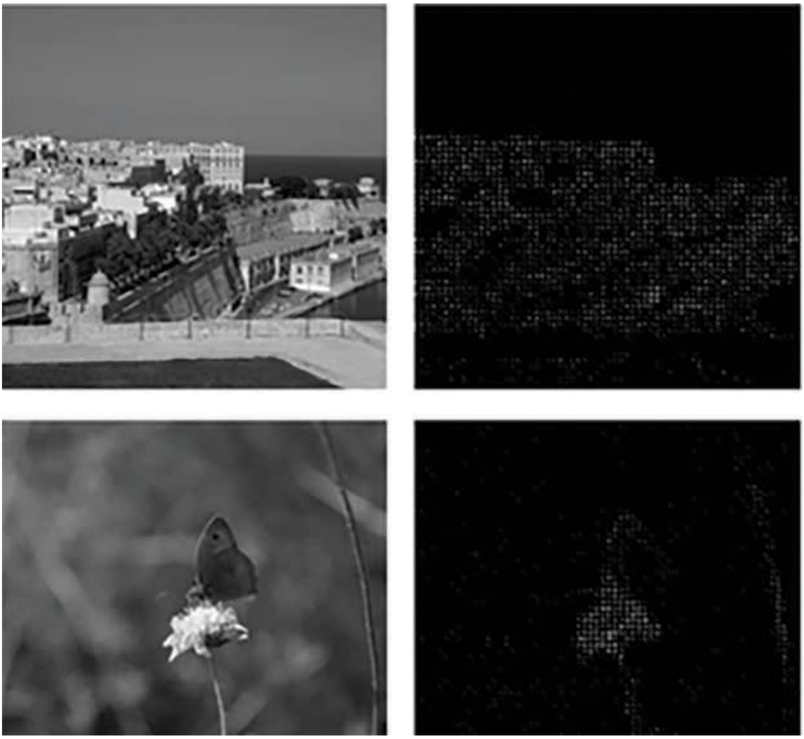


Figure 11.1 Example of steganography images.

Standard picture steganography techniques often seek to encrypt hidden messages inside a cover image. Countless investigations, including both spatial and frequency domain approaches, have been diligently carried out with impressive outcomes in pursuit of this goal [6–8]. Despite the impressive advancements in picture steganography, concealing massive amounts of data remains a challenge [9]. Several recent research has explored the use of deep convolutional neural networks (CNNs) to encase full-size hidden pictures inside cover images. In contrast to the standard ways of picture steganography, these techniques are novel [8, 10, 11]. A revealing network and a hiding network are the standard components of a deep learning-based steganography approach [12]. By enclosing the hidden image inside the cover image, the concealing network may transform the two images into a container image [13]. A concealed secret image is removed from the container images via the revealing network. It was shown in studies based on deep learning that full-size pictures may be hidden in single-cover images with little quality loss [14–16].

The majority of studies are constructed to conceal a single-cover picture [17]. The picture steganography model was expanded to conceal many

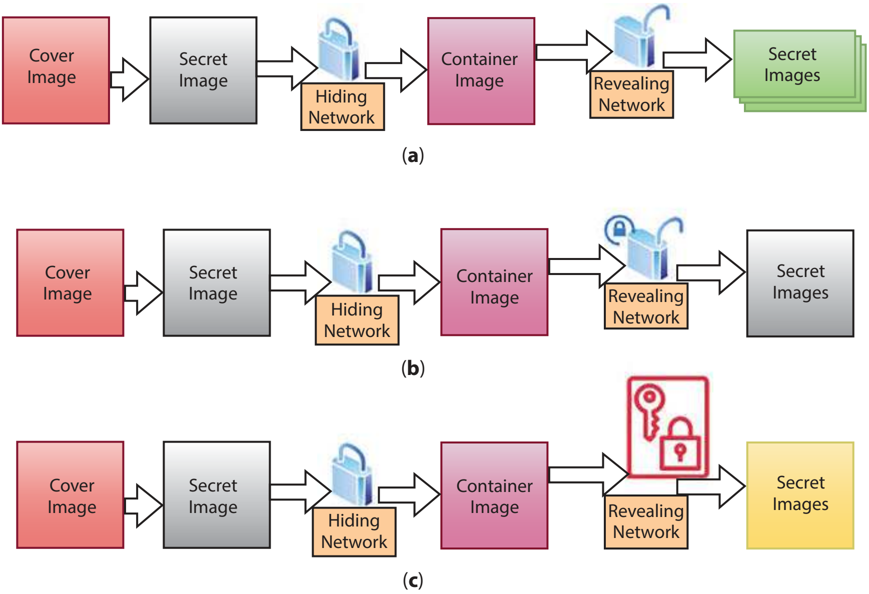


Figure 11.2 Contrast of image steganography methods. (a) The usual methodology does not offer particular access to every hidden picture. (b) The method uses several revealing nets to remove concealed pictures individually. (c) The methodology allows private key-based access to secret photos.

images concurrently, as seen in Figure 11.2 (a), by hiding. Regrettably, it can lead to a serious security issue because of its single reveal network [18]. More specifically, during decoding, the revealing network grabs all of the concealed pictures immediately. This means that different levels of protection cannot be applied to different sets of concealed pictures [19]. Preparing several reveal networks for each secret picture is one easy way to get around this problem as displayed in Figure 11.2 (b) but it increases memorial use as well as disclose how many hidden images need to remain concealed [20].

Using a private key, cover images may conceal secret messages, which can only be retrieved with the right private key. Regarding multi-image steganography, provides the idea of the private key [21]. Figure 11.2 (c) shows that the steganography model has two networks: one for concealing and one for revealing. The hiding network generates private keys for every hidden picture and merges them into a single-cover image during the concealing process [22]. A container image and its associated secret key are sent to an authorized account. During the revealing stage, the object's secret image may be acquired by providing the disclosing network with a private key and the container image [23]. As a result, the proposed steganography paradigm securely and efficiently hides several pictures behind a single-cover image. Results show that our photo steganography method is more adaptable and successful than previous methods [24]. According to everyone, this study is the first to employ private keys for the technique of image steganography, which delivers distinct images instead of messages.

Primary contributions are summarized below: For multi-image steganography, a Private Key is proposed. This technology allows only authorized people to see the concealed picture, unlike earlier image steganography methods. This approach extracts the examined secret image while concealing others [25].

11.2 Works in a Related Field

This section provides an overview of both traditional techniques of spatial and frequency domain steganography and more contemporary approaches that rely on deep learning.

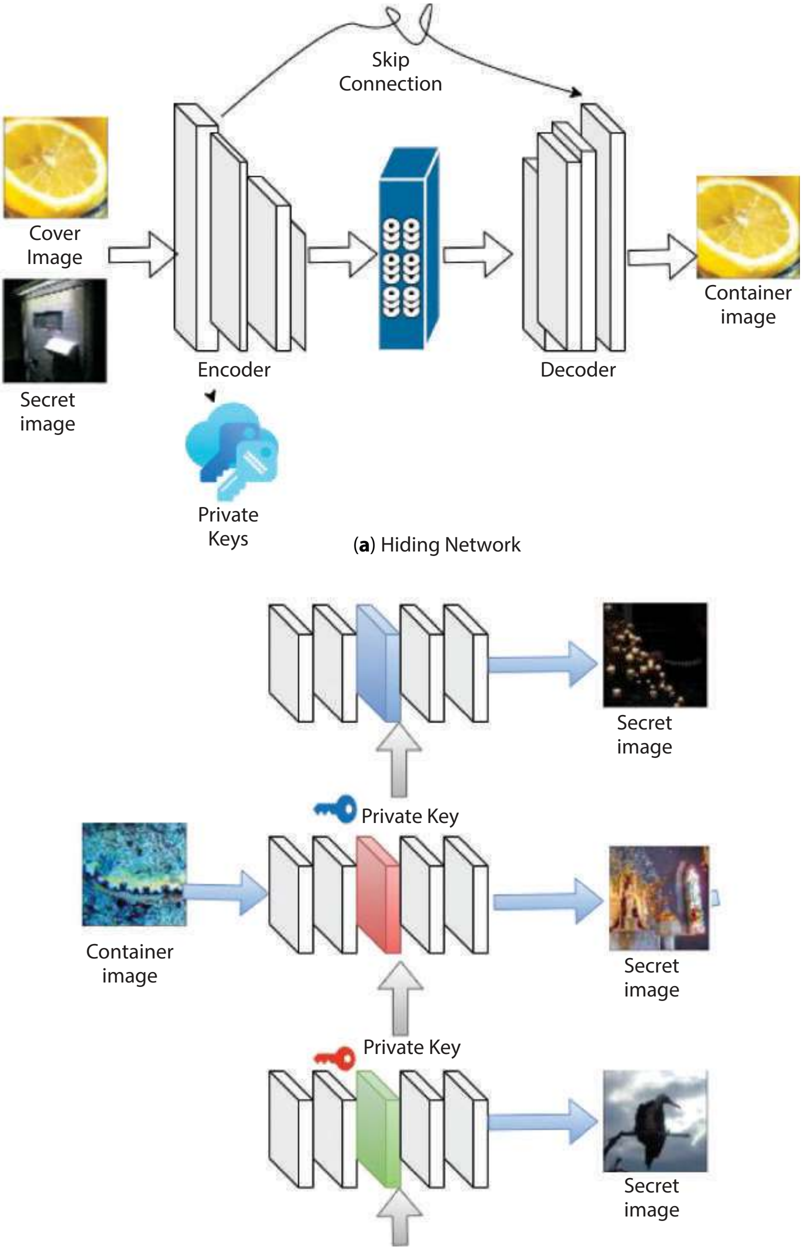
This research provides a technique based on the Least Significant Bit (LSB) that modifies the value of the cover image's spatial domain's least significant bit to conceal messaging. Develop a technique that uses PVD (Pixel Value Differencing) to inject hidden data based on differences in pixel values. Nevertheless, these approaches use well-planned steganalysis procedures. Consequently, several initiatives have been launched to use the local signal-to-noise ratio (LSB) of undetectable high-frequency components. These initiatives include HUGO (Highly Undetectable steGO), UNIWARD (UNIversal WAvelet Relative Distortion), as well as WOW (Wavelet Obtained Weights). In addition, there should be a steganography technique that inserts information into the middle-frequency region of the DCT blocks and a way to change the DWT (Discrete Wavelet Transform) coefficients.

Recent advances in deep learning have led to substantial research accomplishments in several computer vision domains. Recent efforts to use deep learning technology for steganography challenges have grown, however, they are very small compared to other domains, providing a paradigm for deep steganography that can create container images by enclosing full-size images in smaller ones. Using a reveal network, the concealed image may be retrieved from the container picture. A novel method for concealing images using a U-Net architecture has been put forth. Figure 11.2 (a) further shows all concealed pictures simultaneously. This method cannot retrieve just one secret query picture and hide the others. However, this approach concurrently retrieves all of the concealed pictures.

The circumstance when they want to recover a single concealed picture while hiding the others cannot be handled using this method. In the multi-image steganography problem, they provide a notion of private keys to circumvent the aforementioned limitation. It is feasible to retrieve a single concealed picture using this method without affecting or exposing the others.

11.3 Methodology

This section describes the steganography model, which includes a hiding and reveal network. Figure 11.3 depicts the suggested steganography model's pipeline.



(**a**) Hiding Network

(**b**) Revealing Network

Figure 11.3 Steganography pipeline overview.

11.3.1 Net Concealment

The concealing net uses an original image $C \in \mathbb{R}^{256\times256\times3}$ and N secret pictures $S = \{S_n\}_{n=1}^{N} \in \mathbb{R}^{256\times256\times3}$ as inputs to generate a container picture $C' \in \mathbb{R}^{256\times256\times3}$ and private key. For each secret picture S_n, $K = \{K_n\}_{n=1}^{N}$.

$$\{C', K\} = f(C, S; D_f) \tag{11.1}$$

where the factors of the hidden net f (·) that may be trained are represented by $\phi_f \cdot C'$, the container picture, contains the concealed details about the secret image S is visually identical to C, the inserted image. The U-Net structure is used for the hidden network without regard to the architectural designs. The inserted pictures share a single U-Net oriented concealing network. Both the encoder and the cryptographer are made up of repeating patterns. The encoder is made up of strided seven 4 × 4 convolutional layers, while the decoder is formed of seven 4 × 4 deconvolution layers, ReLU, and batch standardization. Additionally, at the very end of both the encoder and the decoder, there are ReLU and Sigmoid layers, correspondingly, rather than LeakyReLU and ReLU. By sending the secret image S and

Table 11.1 Hidden network's architecture.

Pattern	Map-Feature (W×H×D)	Outcome
4×4 Conv.+BN+LeakyReLU 4×4 Conv.+BN+LeakyReLU 4×4 Conv.+BN+LeakyReLU 4×4 Conv.+BN+LeakyReLU 4×4 Conv.+BN+LeakyReLU 4×4 Conv.+BN+LeakyReLU 4×4 Conv.+ReLU	128×128×64 64×64×256 32×32×512 16×16×512 8×8×512 4×4×512 2×2×512	2×2×512
4×4 TransConv.+BN+LeakyReLU 4×4 TransConv.+BN+LeakyReLU 4×4 TransConv.+BN+LeakyReLU 4×4 TransConv.+BN+LeakyReLU 4×4 TransConv.+BN+LeakyReLU 4×4 TransConv.+BN+LeakyReLU TransConv.+LeakyReLU	4×4×512 8×8×512 16×16×512 32×32×256 64×64×128 128×128×64 256×256×3	256×256×3

the cover image C, they may extract the visual properties of both photographs. The decoder receives an image C' that is the result of the encoder's visual feature extraction process, which involves concatenating all of the features along the channel dimension. Keep in mind that the private key K_n is based on visual characteristics of the encoder's secret picture S_n. Table 11.1 displays the hidden network's architecture details.

11.3.2 Network Reveals

The revealing net retrieves a hidden image for the enquiry's key that is private from the container's image during the revealing step.

$$S_n' = g(C', K_n; \phi_g) \tag{11.2}$$

This reveals network g (·) has trainable parameters denoted as ϕ_g. The revealing network, in particular, is made up of six 3 × 3 convolution blocks that do not use down sampling. As a result, traveling across the revealing network does not compromise the spatial dimension. The result of the third layer of convolution is used to enlarge the query's private key to an identical spatial size as indicated in Figure 11.3. Then, the intermediate activation of the container picture produced from the exposing network are concatenated with the key. Ultimately, the reconstruction process ends with a concealed secret picture $S_n' \in \mathbb{R}^{256\times256\times3}$, which corresponds to the query's private key K_n. Table 11.2 displays the architectural elements of the revealing net.

Table 11.2 Architectural elements of the revealing network.

Pattern	**Map-Feature (W×H×D)**	**Outcome**
3×3 Conv.+BN+ReLU	256×256×64	256×256×3
3×3 Conv.+BN+ReLU	256×256×128	
3×3 Conv.+BN+ReLU	256×256×256	
3×3 Conv.+BN+ReLU	256×256×128	
3×3 Conv.+BN+ReLU	256×256×64	
3×3 Conv.+Sigmoid	256×256×3	

11.3.3 Training

The training process for both the hidden and revealed networks is end-to-end. These are the parameters that make up the steganography model's loss function:

$$\mathcal{L} = E\left\| C' - C \right\| + \beta \sum_{n=1}^{N} \left| S'_n - S_n \right| \tag{11.3}$$

where β serves as a weighting coefficient to equalize two components in equation (11.3). In every trial, they maintained β = 0.76.

11.4 Results

For the purpose of training, they have selected a random subset of the MS-COCO train dataset, 1000 validation pictures, and 1000 test pictures from the MS-COCO test datasets and validation, respectively. Using bicubic interpolation, each image is downsized to 256 × 256. They used a Titan RTX GPU and Ubuntu 19.05 to build the method using PyTorch. The suggested model undergoes 100 iterations of training with an Adam optimizer at a learning value of 1 × 10–3. Once the loss trend stops getting better, then increase the learning rate by 0.3.

11.4.1 Analysis Model

The next challenge was to determine which feature layer would be most helpful when creating private keys. To do this, they evaluate the efficiency using private keys and finally open the features generated by the fifth $(K^p \in \mathbb{R}^{4\times4\times512})$ and final $(K^l \in \mathbb{R}^{2\times2\times512})$ layers of the encoder. To conceal pictures ranging from 1 to 5, they trained two networks, one for each secret key. The PSNR and SSIM of the container's and recovered photos are compared with the matching novel images using both randomized keys and correct in order to assess each network. K^p and K^l operate similarly with N = 3 irrespective size of the key, as seen in Figure 11.4. However, K^p continues to demonstrate dependable performance even as the number of concealed pictures increases, while K^l experiences significant performance loss. Since K^l should be able to reconstruct a similar amount of information with less private key information than K^p, this behavior is predicted. Using random keys also yields poor quantitative results, as seen in Figure 11.4. It demonstrates that the concealed picture can only be accessed by the approved private key.

With N = 3, the suggested algorithm's decryption and encryption results are shown in Figure 11.5. Regardless of the private key size, they are able to produce container photos with little distortion contrasted to cover images in the majority of circumstances. Every private key precisely recreates the original concealed picture when it comes to extracting secret photos. On account of the large capacity of keys that is private, K^p is able to extract all the hidden photos with superior quality related to K^l. It should be noted that decrypted photos using K^l often have rather fuzzy and noisy appearances. Visual artifacts originating from other hidden pictures are also often seen. Along with the container picture, it is also necessary to feed the disclosing network randomized private keys to ensure the method is resilient. The secret pictures cannot be appropriately retrieved using random private keys, regardless of their capacity.

To be more specific, the concealed information is rendered unidentifiable due to the recovered pictures' extreme case of mixed textures and sounds. This study's findings demonstrate that the suggested method successfully grants access to a certain concealed image to a designated user.

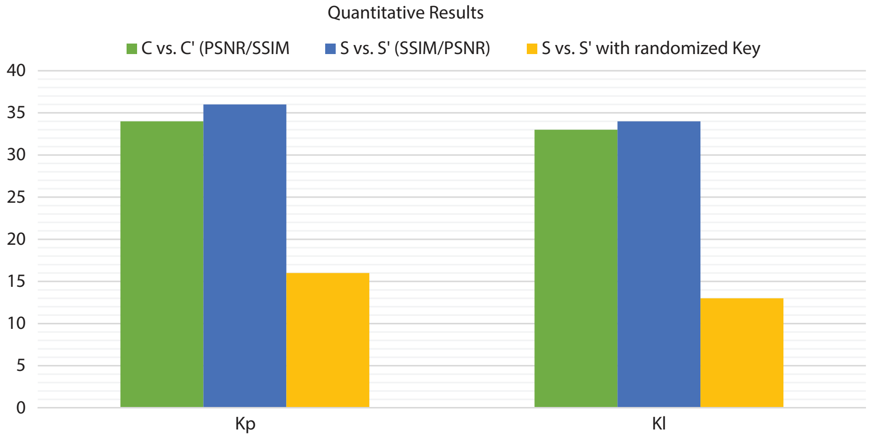


Figure 11.4 Quantitative findings for K^p and K^l based on N. Results include disclosed hidden pictures with random keys.

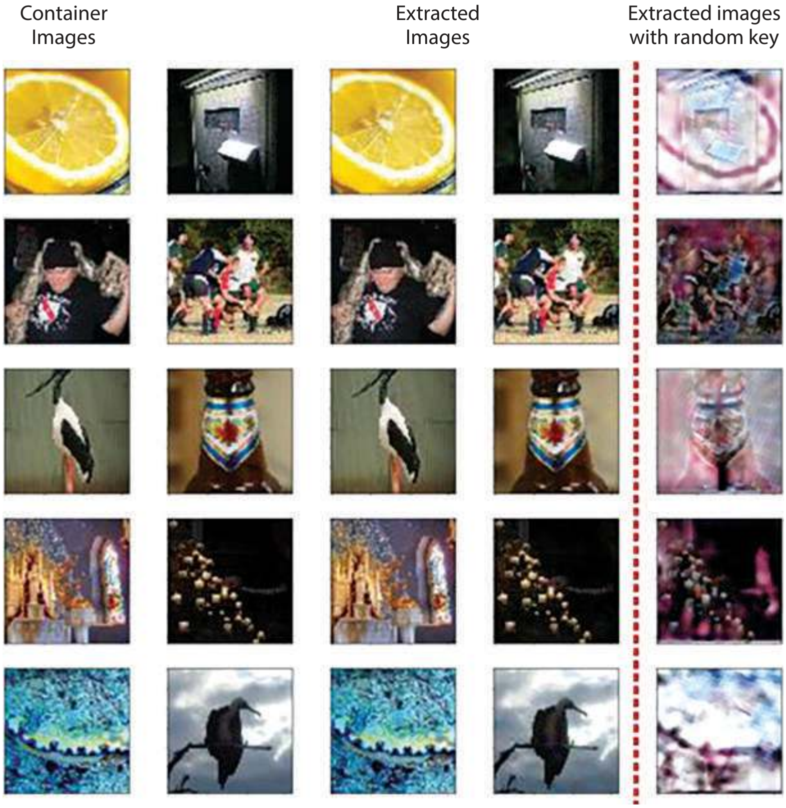


Figure 11.5 Decryption and encryption results for N = 3.

11.4.2 Steganalysis Robustness

LSB-based steganalysis is widely used, therefore they wanted to determine how well this method held up against it. If the current staganalysis approach fails to effectively differentiate between the container picture and the cover image, then the suggested method may be seen as effectively concealing the hidden images. A number of cover and container picture pairings were used in the research to generate receiver operating characteristic (ROC) curves with different levels. Figure 11.6 (a) shows that the curves of ROC with K^l and K^p are almost identical to a straight diagonal line, which represents a random estimate. Since the steganalysis's performance is comparable to random guessing, this indicates that the suggested approach effectively conceals hidden pictures. As shown in Figure 11.6 (b), the AUC values for K^p and K^l are also rather low when N is included. The experimental findings demonstrate that the suggested approach remains highly resistant to the widely used steganalysis model, irrespective of the total secret key size.

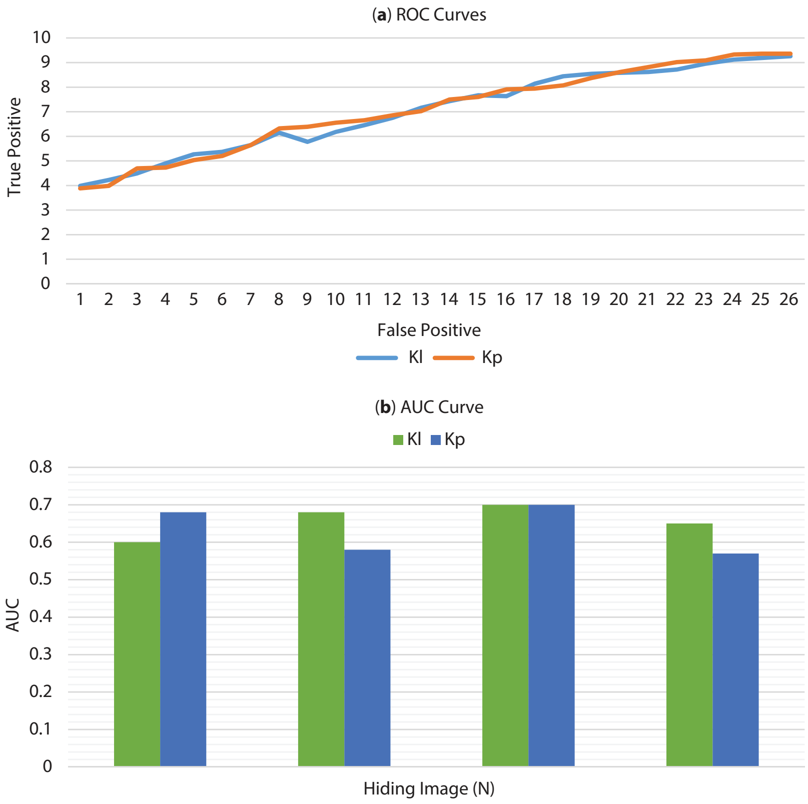


Figure 11.6 (a) ROC curves by key size and (b) calculate AUC values based on N.

11.4.3 Noise Effects

In order to confirm that the method is vulnerable to private key contaminants, they conduct experiments to test the revealing network's ability to extract concealed pictures when noise is introduced to the private key. By combining private keys with sounds, they were able to retrieve hidden pictures. Specifically, each of us contaminated private keys with salt-and-pepper (S&P) or Gaussian noises of varying intensities, and then retrieved secret pictures using these keys. Figure 11.7 shows that, quantitatively speaking, performance drops when noise levels reach higher levels. To be more specific, they can verify that K^p is further vulnerable to higher levels of noise in the presence of Gaussian noise as opposed to K^l. At low noise levels, there is little visual distortion, as seen in Figure 11.8. Although the private key is resilient to low levels of noise, recovering the original secret picture becomes challenging in cases with high levels of noise.

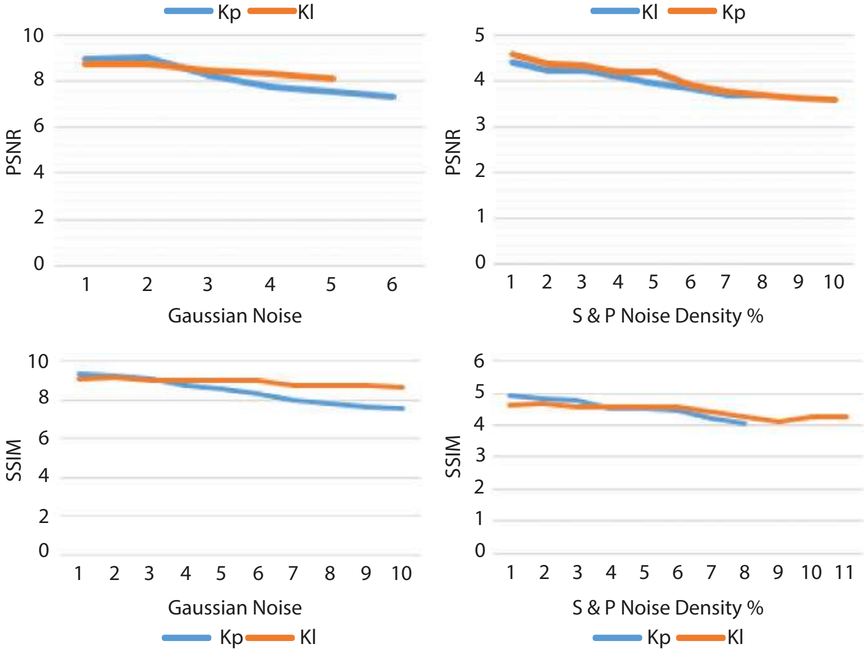


Figure 11.7 Average PSNR and SSIM metrics for noise-contaminated secret pictures reconstructed from private keys.

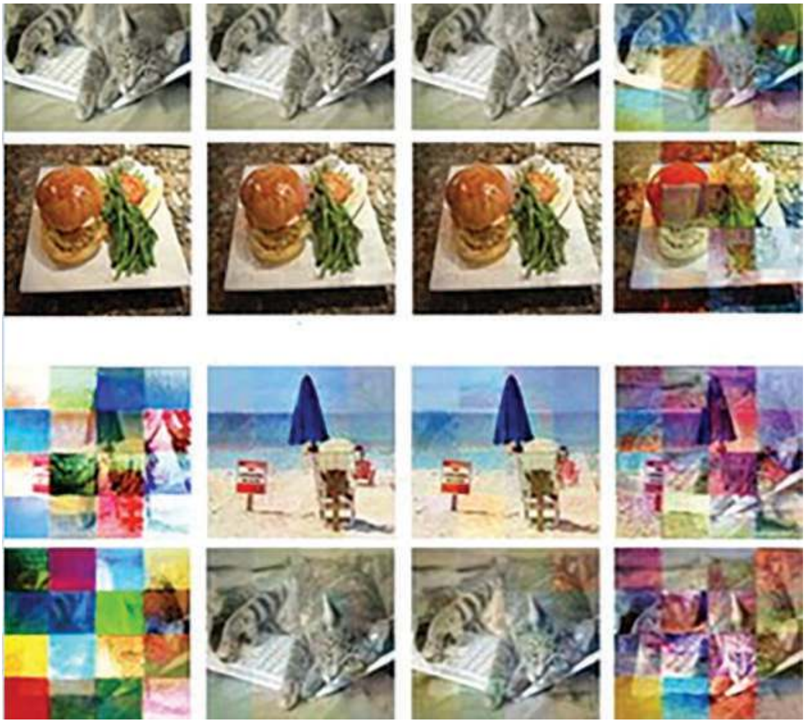


Figure 11.8 Noise-induced private key (K) insertion results for N = 3.

11.5 Conclusion

This study presents an extension of the private key idea to multiple image steganography, which incorporates the use of a single-cover picture to conceal numerous hidden images. For every secret picture, the steganography model generates the private keys and a container image after receiving a real image and a stack of secret photos as inputs. A private key is needed to extract a hidden images from the container images. The suggested model conceals all information on hidden pictures and only returns one when the correct private key is supplied. This method was tested extensively under several settings (for example, randomized key and noise key) to confirm its efficacy.

References

1. Kweon, H., *et al.*, Deep multi-image steganography with private keys. *Electronics*, 10, 16, 1906, 2021.
2. Telli, M., Othmani, M., Ltifi, H., An Improved Multi-image Steganography Model Based on Deep Convolutional Neural Networks. *International Conference on Intelligent Systems Design and Applications*, Springer Nature Switzerland, Cham, 2022.
3. Chambial, S. and Sood, D., Image steganography using CNN. *Int. Res. J. Eng. Technol.*, 9, 02, 176, 2022.
4. Bararia, A.R., Image Steganography on Cryptographic text using Neural Networks, Diss., National College of Ireland, Dublin, 2021.
5. Chinniyan, K., *et al.*, Image Steganography Using Deep Neural Networks. *Intell. Autom. Soft Comput.*, 34, 3, 178, 2022.
6. Sultan, L.R., Deep learning approach and cover image transportation: a multi-security adaptive image steganography scheme. *Smart Sci.*, 11, 4, 677–694, 2023.
7. Guan, Z., *et al.*, DeepMIH: Deep invertible network for multiple image hiding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45, 1, 372–390, 2022.
8. Telli, M., Othmani, M., Ltifi, H., Check for updates An Improved Multi-image Steganography Model Based on Deep Convolutional Neural Networks, in: *Intelligent Systems Design and Applications: 22nd International Conference on Intelligent Systems Design and Applications (ISDA 2022) Held December 12-14, 2022-Volume 3*, vol. 716, Springer Nature, Cham, Switzerland, 2023.
9. Zeng, L., *et al.*, Advanced Image Steganography Using a U-Net-Based Architecture with Multi-Scale Fusion and Perceptual Loss. *Electronics*, 12, 18, 3808, 2023.
10. Raj, R., Gupta, A., Shanmugam, S.G., Image Steganography Using CNN. No. 10240, EasyChair, 2023.
11. Kumar, A., Chatterjee, J. M., Díaz, V. G., A novel hybrid approach of SVM combined with NLP and probabilistic neural network for email phishing. *Int. J. Electr. Comput. Eng.,*, 10, 1, 486, 2020.
12. Agarwal, V. and Gupta, S., Image Steganography using Encoder-Decoder Architectures. *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON)*, vol. 1, IEEE, 2022.
13. Kumar Pandey, B., *et al.*, Encryption and steganography-based text extraction in IoT using the EWCTS optimizer. *Imaging Sci. J.*, 69, 1-4, 38–56, 2021.
14. Wu, B., *et al.*, A multi-party secure encryption-sharing hybrid scheme for image data base on compressed sensing. *Digital Signal Process.*, 123, 1033915, 2022.
15. Majumder, S., *et al.*, Enhanced and secured hybrid steganography model for hiding large data. Diss., Brac University, 2023.
16. Ahmad, S. and Abidi, M.R., RGB Based Secure Share Creation in Steganography with ECC and DNN, in: *Applications of Artificial Intelligence*

and Machine Learning: Select Proceedings of ICAAAIML 2021, pp. 237–250, Springer Nature Singapore, Singapore, 2022.

17. Panpaliya, R., *et al.*, Secret Communication Using Multi-Image Steganography for Military Purposes.
18. Zhu, X., *et al.*, Generative high-capacity image hiding based on residual CNN in wavelet domain. *Appl. Soft Comput.*, 115, 1081705, 2022.
19. Liu, L., Tang, L., Zheng, W., Lossless Image Steganography Based on Invertible Neural Networks. *Entropy*, 24, 12, 1762, 2022.
20. Yang, P., *et al.*, Hiding Image within Image Based on Deep Learning. *J. Phys. Conf. Ser.*, 2337, 1, IOP Publishing, 2022.
21. Palani, A. and Loganathan, A., Multi-image feature map-based watermarking techniques using transformer. *Int. J. Electr. Electron Res.*, 11, 339–344, 2023.
22. Pan, W., *et al.*, Seek-and-hide: adversarial steganography via deep reinforcement learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44, 11, 7871–7884, 2021.
23. Rathore, P.S. and Sarkar, M.K., Defending Against Wormhole Attacks in Wireless Networks Using the Twofish Algorithm: A Performance Analysis. *2024 IEEE 14th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, pp. 0583–0588, 2024, doi: 10.1109/CCWC60891.2024.10427928.
24. Kumar, A., *et al.*, Efficient Privacy Preserving Lightweight Cryptography for Multi-hop Clustering in Internet of Vehicles Network, 2023.
25. Kumar, P.S., *et al.*, Designing a Smart Cart Application with Zigbee and RFID Protocols. *Recent Adv. Comput. Sci. Commun. (Formerly: Recent Pat. Comput. Sci.)*, 15, 2, 196–206, 2022.

12

Automatic Classification of Honey Bee Subspecies by AI-Based Neural Network

B. Sai Chandana[1]*, Ravindra Changala[2], R. Sivaraman[3] and Anand Bhat B.[4]

[1]School of CSE, VIT-AP University, Amravati, India
[2]Department of Computer Science and Engineering, Guru Nanak Institutions Technical Campus, Hyderabad, India
[3]Department of Mathematics, Dwaraka Doss Goverdhan Doss Vaishnav College, Arumbakkam, Chennai, India
[4]Department of Electrical and Electronics Engineering, NMAM Institute of Technology, Nitte, Karkala Taluk, India

Abstract

The intention of this study was to observe the feasibility of using honey bee wing image recognition algorithms for subspecies discrimination using four models built on convolutional neural networks. Research using Inception Net V3, Res Net 50, Mobile Net V2, and Inception Res Net V2 was conducted on a dataset of 9887 wing pictures representative of seven categories and one hybrid. All models performed fitter than conventional morphometric analysis, and the accuracy scores for wing-by-wing categorization were more than 0.92. Across the board, the Inception models outperformed the competition in terms of accuracy, precision, and recall. Upon grouping wing photos according to colony, almost every wing in the samples from that group was assigned the same class. The results show that the European subspecies of honey bees can be consistently distinguished using machine learning and automated picture identification. This might be a great help when trying to quickly categorize subspecies for breeding and conservation purposes.

***Keywords*:** Machine learning, AI, conventional morphometric analysis, accuracy, Inception Net V3

**Corresponding author*: saichandana.bolem@vitap.ac.in

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (191–206) © 2025 Scrivener Publishing LLC

12.1 Introduction

The protection of honey bee biodiversity relies heavily on subspecies discrimination. Honey bee producers place a premium on maintaining pure framework and often seek formal authorization to verify that their bees are members of a certain category, therefore subspecies identification is also crucial in this field. The geographic distribution of honey bee subspecies has been elucidated via a multitude of morphometrically based investigations. This topic is best summarized in the monograph that deals with numerical taxonomy and its applications. Honey bee workers from different areas of the world were analysed using 36 different features, which included things like wing venation, colouration, pilosity, and body component size. Studies of geographic variation still primarily employ this collection of features, called "standard morphometry," as their reference approach. There are several morphometrical approaches used for subspecies identification nowadays. The amount of time and level of accuracy used in the analysis are different. While these approaches are adequate for discriminating a small number of subspecies, they have often been favoured for breeding and conservation Vol. initiatives because of their speed and lack of precision based on the measurement of a few features [1–10].

The morphometric identification technique has been worked on to make it more computerized and maybe even partly automated. By identifying venation connections, which produce angles and distances or serve as markers for shape analysis, the forewing may quickly and reliably provide numerous features, making it the ideal body part for computerized analyses. A scanner, a desktop computer, and data-collecting software tailored to this kind of analysis are necessities. Applying AI via neural network-based machine learning approaches represents a new frontier in picture identification and categorization. Problems in the biological sciences may be well-suited to these systems, which have lately shown remarkable outcomes across several areas. Since the advent and subsequent development of Convolutional Neural Networks 10 years ago, computer vision techniques have seen a rapid transformation. Convolutional neural networks (CNNs) have the potential to learn any computer image task with near-perfect precision and are very versatile in their problem-solving abilities. CNN can train to tackle subspecies identification from a series of tagged samples since the task is an image classification problem. Consequently, the current research set out to determine whether honey bee subspecies could be effectively distinguished from one another using wing image analysis and classification using current image recognition methods. The use of these methods to evaluate the health of honey bee comb cells is not novel and has shown promising results in the past. This research examines four convolutional neural network designs and trains them to identify honey bee race using a reference sample of 9887 fore-wing photos. The dataset includes seven subspecies from Europe and an intraspecific hybrid called Buckfast [11–25].

12.2 Methodology

12.2.1 Morphometrical Analysis, Colony Samples, and Wing Pictures

The honey bee colonies that provided the wing photos used in this research were tracked for the better part of four decades by the CREA-Research Centre for Agriculture and Environment. There was an entire of 508 honey bee colony examples utilized. The CREA-AA reference dataset for morphometric subspecies categorization includes 273 that were verified by routine morphometry analysis of different features. The remaining colony samples were obtained by beekeepers in Italy and other European countries for the autochthonous subspecies. The CREA-Research Centre for Agriculture and Environment followed the honey bee colonies that generated this study's wing pictures for 40 years. Samples from 509 honey bee colonies were used. The CREA-AA reference dataset for morphometric subspecies categorization includes 273 that were verified by routine morphometry analysis of different features. The remaining colony samples were obtained by beekeepers in Italy and other European countries for the autochthonous subspecies. After selecting these colony samples, they were compared to the CREA reference dataset using the Discriminant Analysis with Numerical Output technique to measure 30 wing parameters. Additionally, the third tergite pigmentation was checked, following the Italian procedure. Conformance to designated subspecies determined sample selection. There are seven different subspecies of honey bees from Europe and one intraspecific hybrid called "Buckfast" among the samples taken (Figure 12.1). The bees' right forewings are removed from a single colony, put on a microscope slide or camera slide, and then scanned at 3200 dpi to create numerical photographs.

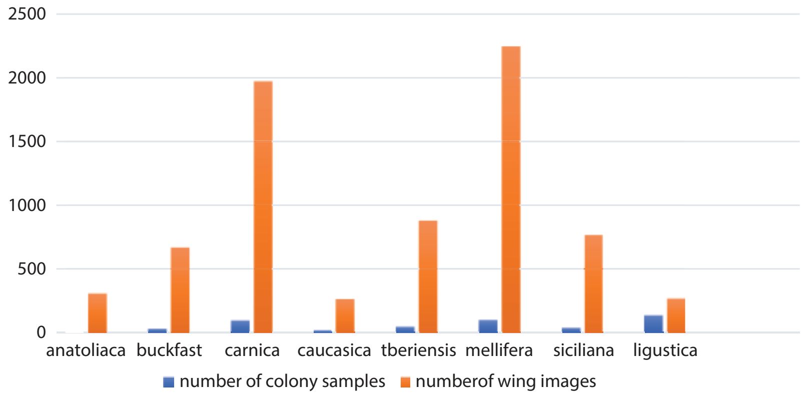


Figure 12.1 The number of colony samples per class and selected wing photos.

12.2.2 Utilizing AI for Image Processing

Using object detection, a Retina Net model with a Res Net 50 backbone was trained on an illustration of 22 extremity annotated photographs to extract single-wing pictures (thus referred to as "images") from the dataset mentioned earlier. The creation of PASCAL-VOC XML files allowed for annotation to be accomplished. After getting a dataset with eight classifications, the results were double-checked by hand to remove false positives. Pictures were reduced and white-pixel-padded to make them square, the standard for the neural networks we will be looking at.

12.2.3 Models for Recognition and Instruction

Inception Net V3, Mobile Net V2, Res Net 50, and Inception ResNet V2 are some of the well-established CNN models in the IT industry that were taken into consideration for the experiment. A variety of deep learning packages, including TensorFlow, PyTorch, and Keras, provide implementations of these models, and although none of them are state-of-the-art, they all offer relatively decent overall performance. Even the smallest model, Mobile Net V2, has around 3.4 million trainable parameters, demonstrating the enormous complexity of the models under consideration Optimizing the training process for excellent outcomes at acceptable time-frames is crucial. The researchers used a 10-fold stratified cross-validation approach (Figure 12.2).

This method has a solid reputation in the machine learning field and is iterative by design. A total of ten equal-sized subsets (folds) of the data were extracted from the original dataset. This ensured that the class with

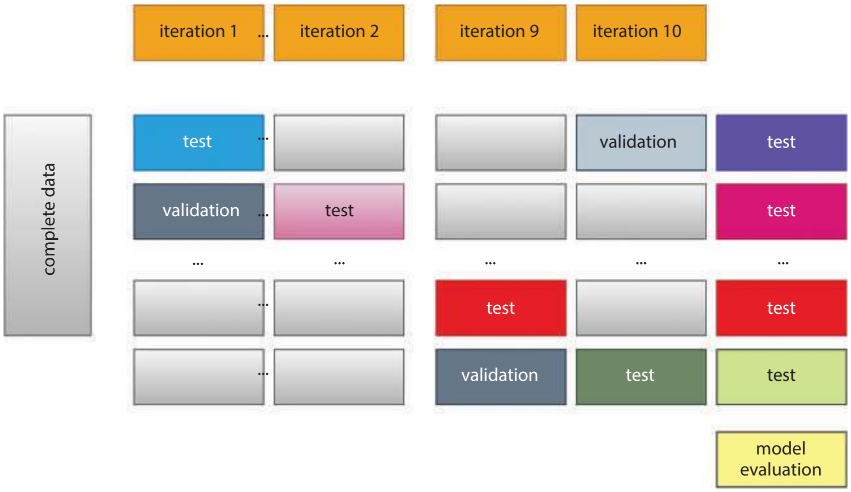


Figure 12.2 Visualization of 10-fold cross-validation with a validation split.

the highest representation stayed at the top across all 10-folds, while all other classes were represented at an equal rate. These data were randomly partitioned. The computation of folds was the first stage in an iterative process that included assembling the partitions created in the preceding phase into three bigger partitions, or splits, at each iteration: the validation set, the training set, and the test set. Except for two splits of onefold each, the data in the training set consisted of eightfolds. No matter how many folds were used, the class proportions of the unique dataset were preserved in each split. To generate a set of class forecasts for every picture in the subsequent split, a new model occurrence was trained on the training set and then tested on the test set for each iteration. After going through the process once, each fold served as a test set. That way, a model that was not trained with the image will predict every single photo in the dataset, run the model through its paces using these predictions for every measure in the dataset.

Data bootstrapping and early training stopping were added to the previously stated experimental methods to report two significant issues: overfitting and numerical bias. Bootstrapping the data used for the training split made it more regularly distributed, which helped to minimize statistical bias while training the model. A major bias toward more represented subspecies, like Siciliana, would have resulted from an imbalance in the distribution of classes, thus something was done to fix that. Each of the training split classes was resampled using replacement at random until there were 1600 photos in total, which is equivalent to 0.6 times the cardinality of the class with the highest number of images. To prevent any manipulation of the model selection and assessment processes, the test and validation data were not bootstrapped. An unbiased assessment requires a third split (Figure 12.3). There were three parts to the process: training the model with the bootstrapped data set, validating its progress using the validation set, and testing its performance with the test set. Train all of the models using a triangle learning rate scheduling protocol that includes stochastic gradient descent.

Training data must be processed several times, each termed an epoch, in this iterative technique as well. Because it is iterative, the training method can potentially continue indefinitely; it is the data expert's responsibility to terminate it once a good fit is achieved. The ideal number of training epochs cannot be known in advance, thus introduce the validation set to experimentally establish it. When the accuracy performance measure used on the validation set reached its maximum and no further improvement could be seen, the training method was terminated. For further information, go to the "Analysis" section. Since it offers a reasonable compromise between underfitting and overfitting, this maximum point might be thought of as the optimal fit. The data from the validation set were included in the trained model, even though it was not handled during training, since the number of training epochs was adjusted using this set. Therefore, a third split was necessary to conduct an unbiased review. Twenty epochs of online data augmentation training were used for all models. Making many versions of the same image to feed into the model was part of this process.

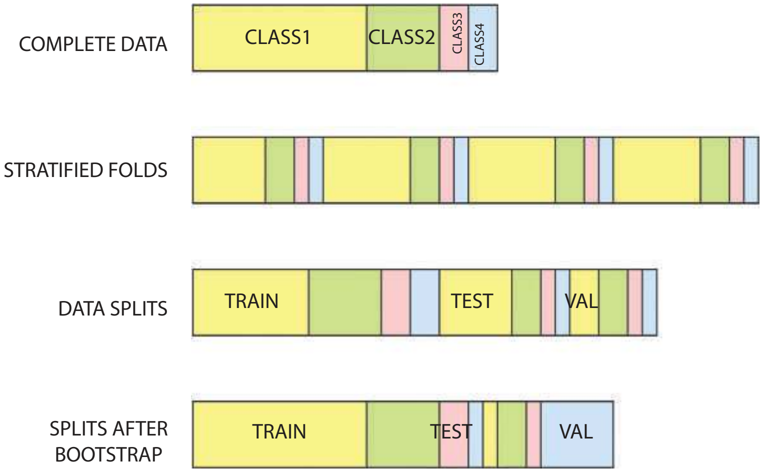


Figure 12.3 Experimental design fold layer and training data bootstrapping visualization.

This approach was more memory economical (fewer photos needed to be overloaded into the GPU memory) and provided a larger degree of randomness throughout multiple epochs compared to a pre-computed collection of perturbed images, enabling the perfect to reach a better tolerance toward suboptimal images. To make the training data more unpredictable, rotated, flipped, and adjusted the brightness of the photos. Use replication padding to make sure that the original pixel colour distributions are preserved. To increase the model's resilience against noisy data, the different transformations were done in a stochastically cascading fashion. This meant that a wing picture may be, for example, both rotated and flipped.

12.2.4 Evaluation

Model performance was assessed using three well-known classification metrics:

- Accuracy: the percentage of photos that were properly labelled. Its purpose is to measure the overall efficiency of the model.
- Precision: the percentage of positive values that are positive, sometimes called positive predictive value. To measure how well a model does relative to a certain class, it is used. It is a measure of the model's ability to prevent false positives.
- Recall: the percentage of positive samples that the system accurately identifies; it is also called specificity. To measure how well a model does relative to a certain class, it is used. As such, it is a proxy for the model's accuracy in detecting true negatives.

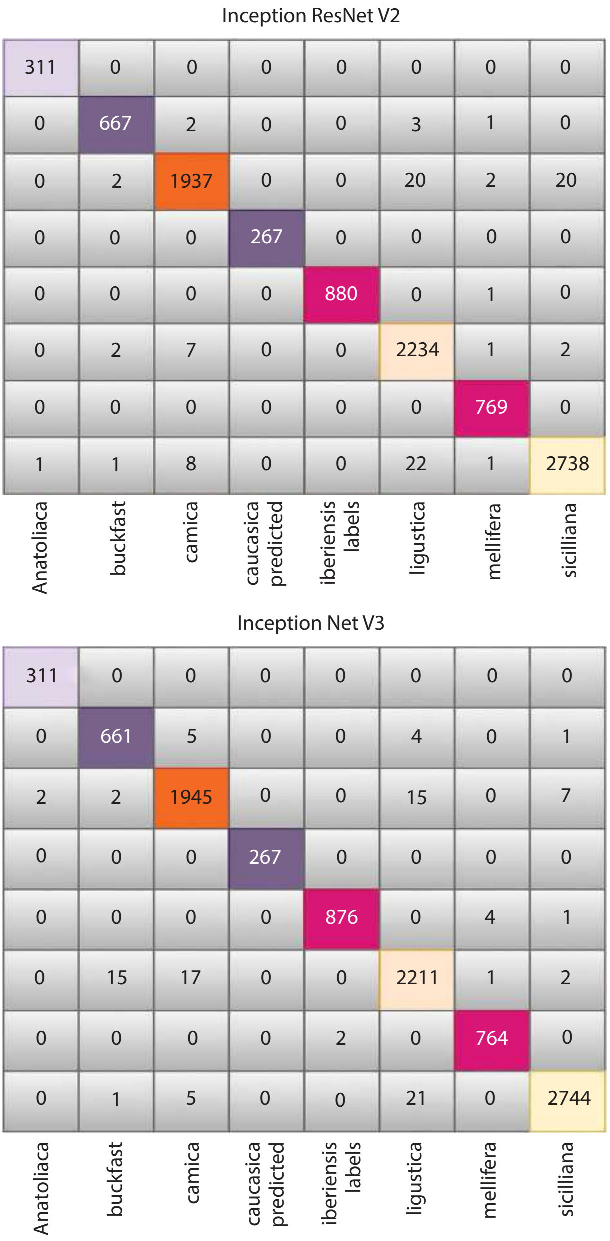
Accompanying accuracy and recall is their harmonic mean, also known as the F1 score; this is helpful when arresting a balance among the two is necessary, as they are complementary.

12.3 Results

12.3.1 Analysis of the Model

The precision outcomes for each of the four models under consideration the average value was determined after each test split was taken into account in the cross-validation technique. With superior average accuracy and reduced accuracy variance, the Inception Res Net and Inception Net designs seemed to outperform Res Net and Mobile Net. Utilizing pre-computed stratified partitions of the dataset, evaluate global metrics across all models and combine the predictions from the test splits. This is accomplished via the cross-validation technique. This enabled us to assert, as a result of experimental design, that every picture was present in the test screen of the information precisely once. Although some distributional information was due to this aggregation, to assess metrics on a sample size that was appropriate for all the classes that were taken into consideration and, most importantly, observed that Mobile Net, Inception ResNet, and Inception Net V3 performed fairly consistently across different folds. If you want a better understanding of how the two high-accuracy models fared relative to each other, you should look at the F1 values, recall, and precision over all of the test splits for each class. The Inception ResNet model seems to do somewhat better in the majority of classes, but both models seemed to perform similarly in all of them. The majority of scores were above average, and the two models had the best recall and accuracy when classifying the Iberians, Caucasic, and Anatomica classifications. Confusion matrices, in which the rows indicate milled truth values and the columns model predictions, were constructed using the forecasts of the tested models. This allowed us to conduct a more thorough analysis of classification mistakes. The confusion matrices (Figure 12.4), in line with the accuracy and recall metrics, were sparse and almost slanting, with few non-zero members beyond the diagonals. By comparing the two models' mistakes in the non-diagonal cells, you can see that they were almost identical; for example, Siciliana and Carnica were both mistaken for Linguistic most of the time. Furthermore, compared to the Inception Net V3 model, the Inception ResNet V2 model seems to be less prone to confusing Ligustica for either Buckfast or Carnica. Bee wings are often found in samples that comprise numerous wings from various individuals of the same colony. The analysis that has been done so far is based on models that are trained on images of a single wing. Sort the images into eight categories based on the colony name and then use the mode class of each wing's predictions to determine the colony's post-processing treatment. Because of this, to test the validity of the computer vision models in an actual environment.

Table 12.1 shows that the procedure yielded an accuracy range from 0.9921 to 1 for each colony. Based on these numbers, it seems that the mistake that happens when you categorize one wing picture is spread out throughout several samples of the same colony, and all four models that were taken into consideration were able to accurately identify most of the wings in the same colony. A confidence value is established for each colony as the percentage of wings in the mode class, which further illustrates this fact.



Inception ResNet V2

311	0	0	0	0	0	0	0
0	667	2	0	0	3	1	0
0	2	1937	0	0	20	2	20
0	0	0	267	0	0	0	0
0	0	0	0	880	0	1	0
0	2	7	0	0	2234	1	2
0	0	0	0	0	0	769	0
1	1	8	0	0	22	1	2738
Anatoliaca	buckfast	camica	caucasica predicted	iberiensis labels	ligustica	mellifera	sicilliana

Inception Net V3

311	0	0	0	0	0	0	0
0	661	5	0	0	4	0	1
2	2	1945	0	0	15	0	7
0	0	0	267	0	0	0	0
0	0	0	0	876	0	4	1
0	15	17	0	0	2211	1	2
0	0	0	0	2	0	764	0
0	1	5	0	0	21	0	2744
Anatoliaca	buckfast	camica	caucasica predicted	iberiensis labels	ligustica	mellifera	sicilliana

Figure 12.4 Model predictions from Inception ResNet V2 and Inception Net V3 are shown in confusion matrices. Columns make forecasts and rows show ground facts.

Table 12.1 Tested models obtained accuracy when grouping photos by group sample.

	Inception net V3	Inception ResNet V2	ResNet 50	Mobile net v2
Sample accuracy	0.9980	1.0000	0.9941	0.9921

The supply of these confidence values across studied colony samples is shown in Figure 12.5, for each tested model. The most prevalent scenario across all four models was the labelling of all wings to the same class; nevertheless, there was a noticeable disparity between the lower-scoring networks and the top-scoring ones. Top-scoring networks exhibited shorter tails of low assurance samples than lower-scoring networks, which had a non-negligible number of colony samples and wider tails down to 0.4 confidence and below.

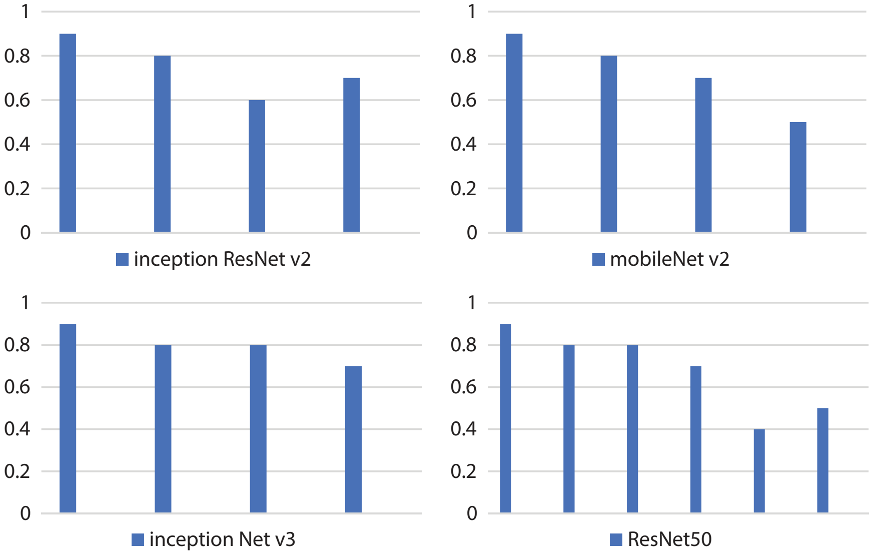


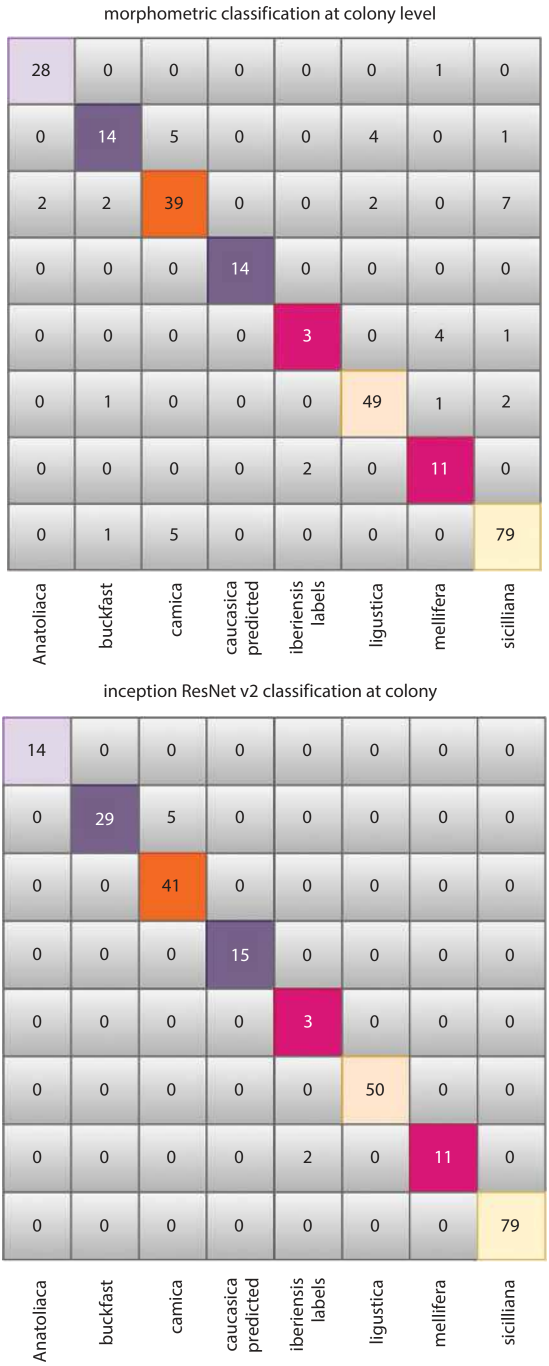
Figure 12.5 Distribution of sureness levels, defined as the proportion of mode-class wings on a subset of 242 colony models, for the four models investigated.

12.3.2 Evaluation Using the Morphometric Approach

The predictions from the test splits and evaluate all models using precomputed stratified partitions of the dataset. Check global metrics. This is by using a cross-validation strategy. Importantly, found that Mobile Net, Inception ResNet, and Inception Net V3 performed fairly consistently across different folds, even though lost some distributional information due to this aggregation. To evaluate metrics on a sample size that was suitable for all the classes that were considered. Examining the F1 values, recall, and precision over all test splits for each class will provide a clearer picture of how the two high-accuracy models performed in comparison to one another. The individual wing labelling job was also taken into account to provide more insights on the comparing performance. The confusion matrix and accuracy were produced by using the morphometric approach to classify the bee wings in the colony models separately. Results for the computer vision techniques' accuracy ranged when tested on the identical set of 242 colony models as shown in Figure 12.6. This is the Inception Net V3 model's confusion matrix. Inception Net and Inception ResNet perform almost identically when comparing the precision values achieved by the different models in the benchmark. Mobile Net and ResNet, on the other hand, seem to have much lower accuracy. Computer vision models get much better ratings across the board when accuracy is taken into account in class-wise measurements. Notably, for almost all classes, the computer vision pipeline took into account a much larger number of photos.

12.4 Discussion

Breeders and beekeepers are increasingly aware of honey bees' genetic origin due to environmental awareness and colony wins in the previous decade. Several variables, notably pesticides, have been widely identified as the primary causes of these losses. Consequently, a lot of people who maintain bees want to know whether the bees they are taking care of are members of the native subspecies. A rise in the stock's market value, admission to protected areas, or incentives based on local standards are all possible outcomes of subspecies designation. Consequently, the apicultural industry might benefit from a quicker and less expensive process. This study's findings provide credence to the hypothesis that honey bee wing image identification using Convolutional Neural Networks may outperform morphometric analysis when it comes to subspecies discrimination. Using 9887 wing photographs from seven subspecies and one intraspecific hybrid, ResNet50, Inception Net V3, Mobile Net V2,



morphometric classification at colony level

Anatoliaca	buckfast	camica	caucasica	iberiensis	ligustica	mellifera	sicilliana
28	0	0	0	0	0	1	0
0	14	5	0	0	4	0	1
2	2	39	0	0	2	0	7
0	0	0	14	0	0	0	0
0	0	0	0	3	0	4	1
0	1	0	0	0	49	1	2
0	0	0	0	2	0	11	0
0	1	5	0	0	0	0	79

predicted labels

inception ResNet v2 classification at colony

Anatoliaca	buckfast	camica	caucasica	iberiensis	ligustica	mellifera	sicilliana
14	0	0	0	0	0	0	0
0	29	5	0	0	0	0	0
0	0	41	0	0	0	0	0
0	0	0	15	0	0	0	0
0	0	0	0	3	0	0	0
0	0	0	0	0	50	0	0
0	0	0	0	2	0	11	0
0	0	0	0	0	0	0	79

predicted labels

Figure 12.6 Confusion matrices for a subset of 242 colony samples categorized using the Morphometric method and the best computer vision model.

and Inception ResNet V2 achieved discriminating precision of 0.92 to 0.99. This study's findings provide credence to the idea that honey bee wing image identification using Convolutional Neural Networks may outperform morphometric analysis when it comes to subspecies discrimination. Notably, analysed 9887 wing images to distinguish seven subspecies and one intraspecific hybrid. utilizing Mobile Net V2, ResNet50, Inception ResNet V2, and Inception Net V3 with 0.92 to 0.99 precision. The second model surpassed the others in accuracy and recall with F1 values over 0.98 across all classes.

Notably, most of the misclassifications occurred within an evolutionary branch or between subspecies that were geographically adjacent to one other. Another research that used genetic markers came to similar conclusions. The discriminating capacity of the computer vision approach was confirmed when utilising the morphometric system to categorize separate wings that were also identified by the CNN models. Observed that its precision was noticeably lower. While it is true that samples were not selected based on conformance to racial norms, prior research including five populations found that the traditional morphometric approach had a precision of up to 0.94 and as low as 0.33 for some classifications, including linguistic. Additionally, it is noteworthy to note that the morphometric approach and the computer vision method differed in the number of single-wing pictures that were analysed, with class numerosity discrepancies ranging from 7% to 60%. The two distinct processing processes at work here provide this effect: According to the morphometric method, a human expert would examine each image and select a subset of wings to classify. In contrast, the computer vision tube would process the images using the Retina Net model, which would then pass on to the classification model any image that it detected as a bee wing including those with noise, blur, or damage. Because the statistical approach utilizing morphometrical measurements was utilized for the categorization of the genus at the specific and subspecific level, standard morphometry parameters were deemed the primary or baseline method that accompanies other ways. As an additional benefit, morphometry can detect even the most intricate genetic patterns, providing a trustworthy and inexpensive substitute for initial population structure assessment. This approach has been tested using a unique collection of reference samples gathered over the previous 40 years at CREA-AA, and it successfully differentiates numerous subspecies as specified. In addition, for wing venation measuring techniques that have been suggested so far, the operator is still somewhat involved in locating and verifying vein crossings, even if the software has been created to streamline the process for geometric and classical morphometry.

In contrast, the suggested fast method refrains from processing a predefined geometric pattern and instead uses a computer vision system integrated with artificial intelligence software for wing recognition and classification, with no human intervention in the analytical process. Although molecular methods are advancing rapidly, beekeepers do not yet have easy access to them. In addition, although approaches derived from honey bee maternal lines using mtDNA sequencing provide fascinating patterns of mitotypes for phylogenetic investigations, they fall short when it comes to recognizing subspecies. To distinguish almost all subspecies, only cutting-edge methods of gDNA sequencing and SNP analysis were used. A recently developed and commercially available molecular method is based on single-nucleotide polymorphisms (SNPs) and uses over 4,000 individual honey bees as a reference. These bees were collected from samples where morphometric analysis confirmed the subspecies designation. Even though this instrument is a significant improvement, the typical beekeeper will still find the price of a representative colony sample to be too high. Since molecular tools are based on a single individualist per colony, a novel wing representation analysis approach could be a quick and inexpensive substitute for them. Alternatively, it could be viewed as a supplementary method that takes into account many individuals per colony.

Machine learning has made it possible to continuously train models on large quantities of data, which is driving rapid advancements in image identification technologies. Available in several popular software packages, production-grade models like the ones used in this research seem to be strong sufficient to attain a decent level of quality. Ultimately, this study's studies demonstrate that subspecies discrimination in honey bee wings is possible using automated image recognition and machine learning. Regarding wing samples, in particular, Inception Resnet V2 stood up as the most promising CNN model. The significance of image recognition in biological systems is highlighted by the findings. Here, is the groundwork for the eventual creation of a practical application for traditional informatic devices by demonstrating how picture recognition may be a leading-edge and helpful tool for honey bee beekeepers and breeders in terms of quick and inexpensive categorization.

References

1. De Nart, D., *et al.*, Image recognition using convolutional neural networks for classification of honey bee subspecies. *Apidologie*, 53, 1, 5, 2022.
2. Rodrigues, P.J., Gomes, W., Pinto, M.A., DeepWings©: automatic wing geometric morphometrics classification of honey bee (Apis mellifera) subspecies using deep learning for detecting landmarks. *Big Data Cognit. Comput.*, 6, 3, 70, 2022.

3. Karthiga, M., *et al.*, A deep learning approach to classify the honeybee species and health identification. *2021 Seventh International Conference on Bio Signals, Images, and Instrumentation (ICBSII)*, IEEE, 2021.
4. da Silva, F.L., *et al.*, Evaluating classification and feature selection techniques for honeybee subspecies identification using wing images. *Comput. Electron. Agric.*, 114, 68–775, 2015.
5. Gernat, T., *et al.*, Automated monitoring of honey bees with barcodes and artificial intelligence reveals two distinct social networks from a single affiliative behaviour. *Sci. Rep.*, 13, 1, 1541, 2023.
6. Dubey, A.K., Kumar, A., García-Díaz, V., Sharma, A.K. and Kanhaiya, K., Study and analysis of SARIMA and LSTM in forecasting time series data. *Sustain. Energy Technol. Assess.*, 47, 101474, 2021.
7. Kumar, A., Chatterjee, J.M., Díaz, V.G., A novel hybrid approach of SVM combined with NLP and probabilistic neural network for email phishing. *Int. J. Electr. Comput. Eng. Syst.,* 10, 1, 486, 2020.
8. Schurischuster, S. and Kampel, M., Image-based classification of honeybees. *2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, IEEE, 2020.
9. Kaya, Y., *et al.*, An automatic identification method for the comparison of plant and honey pollen based on GLCM texture features and artificial neural network. *Grana*, 52, 1, 71–77, 2013.
10. Al-Mahasneh, M., *et al.*, Classification and prediction of bee honey indirect adulteration using physiochemical properties coupled with k-means clustering and simulated annealing-artificial neural networks (SA-ANNs). *J. Food Qual.*, 2021, 1–9, 2021.
11. Kviesis, A. and Zacepins, A., Application of neural networks for honey bee colony state identification. *2016 17th International Carpathian Control Conference (ICCC)*, IEEE, 2016.
12. Ghanem, W.A.H.M. and Jantan, A., Training a neural network for cyberattack classification applications using hybridization of an artificial bee colony and monarch butterfly optimization. *Neural Process. Lett.*, 51, 905–946, 2020.
13. Alamri, N.M.H., Packianather, M., Bigot, S., Deep learning: Parameter optimization using proposed novel hybrid bees Bayesian convolutional neural network. *Appl. Artif. Intell.*, 36, 1, 2031815, 2022.
14. Moreira-Filho, J.T., *et al.*, BeeToxAI: An artificial intelligence-based web app to assess the acute toxicity of chemicals to honey bees. *Artif. Intell. Life Sci.*, 1, 1000135, 2021.
15. Almryad, A.S. and Kutucu, H., Automatic identification for field butterflies by convolutional neural networks. *Eng. Sci. Technol. Int. J.*, 23, 1, 189–195, 2020.
16. Tiwari, A., A deep learning approach to recognizing bees in video analysis of bee traffic, Diss., Utah State University, 2018.
17. Braga, D., *et al.*, An intelligent monitoring system for assessing bee hive health. *IEEE Access*, 9, 89009–890195, 2021.

18. Khalil, H., Identification of Insect Pollinator Species using Bioacoustics and Artificial Intelligence, Diss., University of York, 2021.
19. Bullinaria, J.A. and AlYahya, K., Artificial bee colony training of neural networks: comparison with back-propagation. *Memet. Comput.*, 6, 171–182, 2014.
20. Bustamante, T., Baiser, B., Ellis, J.D., Comparing classical and geometric morphometric methods to discriminate between the South African honey bee subspecies Apis mellifera scutellata and Apis mellifera capensis (Hymenoptera: Apidae). *Apidologie*, 51, 123–136, 2020.
21. Braga, D.M.M., Sistema de monitorização de colmeia de abelhas, Diss., 2020.
22. Bhargava, N., Kumar Sharma, A., Kumar, A., Rathoe, P.S., An adaptive method for edge preserving denoising. *2017 2nd International Conference on Communication and Electronics Systems (ICCES)*, pp. 600–604, Coimbatore, India, 2017, doi: 10.1109/CESYS.2017.8321149.
23. Choudhary, N., Rathore, P.S., Kumar, D., Comparative Evaluation of Marker-Controlled Method and Gradient Distance Transformation through Watershed Segmentation. *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, pp. 1–6, Pune, India, 2024, doi: 10.1109/I2CT61223.2024.10543576.
24. Bhargava, P.N., Rathore, P.S., Vaishnav, P., Rai, M., Utilizing Artificial Neural Networks and Multivariate Patient Data for Anemia Detection using WEKA based Approach for Diagnosis. *2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS)*, pp. 643–648, Pudukkottai, India, 2023, doi: 10.1109/ICACRS58579.2023.10404355.
25. Rathore, P.S., Kumar, A., García-Díaz, V., A holistic methodology for improved RFID network lifetime by advanced cluster head selection using dragonfly algorithm, Automatic Classification of Honey Bee Subspecies by AI-Based Neural Network, 193, 2020.

13

Acoustic Modeling and Evaluation of Speech Recognition by Neural Networks

Y. Ramadevi[1]*, K. Suresh Kumar[2], Venkata Pavankumar[3] and G.N.R. Prasad[4]

[1]Department of AIML, Chaitanya Bharati Institute of Technology, Hyderabad, India
[2]MBA Department, Panimalar Engineering College, Varadarajapuram, Poonamallee, Chennai, India
[3]Department C S E (DS), Siddhartha Institute of Engineering and Technology, Hyderabad, India
[4]Department of MCA, Chaitanya Bharati Institute of Technology, Hyderabad, India

Abstract

The most popular acoustic modeling approach for ASR with a big vocabulary is artificial neural networks. The multi-layer design of a traditional ANN requires enormous computational resources. Similar to biological neural networks, intelligence-inspired spike-based neural networks may operate on neuromorphic hardware with little power consumption. Investigating SNN's potential for speech recognition is motivated by their remarkable ability to process data quickly and efficiently while using very little energy. SNN is used for audio modeling and evaluated in numerous big vocabulary recognition situations. An appealing option for ASR applications operating locally on embedded and mobile devices is to combine the computational capability of deep SNN with neuromorphic computer hardware, which is energy efficient.

Keywords: ASR, ANN, SNN, biological neural networks, neuromorphic hardware

**Corresponding author*: yramadevi_cseaiml@cbit.ac.in

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (207–226) © 2025 Scrivener Publishing LLC

13.1 Introduction

Automated speech recognition allows smartphone and smart home devices voice control. The excellent presentation of ASR systems that employ artificial neural networks in acoustic models has made voice interface integration rapid. Artificial neural networks like feedforward and recurrent networks have been used to simulate speech-indicating acoustic data. The processing of incoming audio signals must be time-synchronous, which results in massive computing demands along with performance advantages. There are a variety of methods that aim to lessen the number of parameters needed for inference, which in turn reduces the computing burden and memory storage requirements of ANNs. Reduce processing burden with another typical solution: use a wake phrase or word to limit access to voice recognition services. In addition, slightly than using local on-device solutions, the majority of voice-controlled devices depend on ASR engines hosted in the cloud. Concerns about data security, processing speed, etc., arise with the requirement of online voice processing via cloud computing. This speech signal can be processed locally utilizing the computing power of mobile devices, and several attempts to build on-device ASR solutions were made. A lot of people have been paying more and more attention to event-driven methods, like spiking neural nets, that are modeled after the human brain. Incredibly sophisticated visual and cognitive processes are within the human brain's capabilities. Importantly, the light from a dim bulb is comparable to the amount of power that an adult's brain needs to tackle complicated problems. Though ANNs designed to mimic the brain have shown impressive performance on a variety of cognitive and perceptual tasks, these models still have a way to go before they can compete with real brains in terms of efficiency and computing load. In contrast to ANNs, SNNs emulate the computing paradigm seen in human brains via their asynchronous and event-driven data processing; in this model, energy consumption is proportional to the intensity of sensory inputs. Therefore, event-driven computation is far more efficient computationally than synchronous computation in artificial neural networks, even when dealing with temporally sparse data sent in the environment. A non-von Neumann computing paradigm, neuromorphic computing uses SNN on silicon to simulate the event-driven processing seen in organic brain networks. To facilitate spike-based data processing, the new neuromorphic computer designs make use of highly parallel, low-power computing units. Memory and processing units that are physically close together avoid the von Neumann bottleneck, which occurs when there is insufficient bandwidth between the two. Hence, for ubiquitous machine learning responsibilities for always-on submissions, an attractive approach might combine the algorithmic strength of deep SNNs with the amazing low-power consumption of NC hardware. In addition, there has been an upsurge in the pursuit of innovative non-volatile memory technologies for use in physiological synapses and neurons that function with very low power consumption.

Phone categorization and small-vocabulary voice recognition systems based on SNNs have been investigated in some exploratory studies. These SNN-based ASR schemes, nevertheless, pale in comparison to the sophisticated and widely used ANN-based ASR systems. The primary cause of this is the absence of a powerful software toolbox for artificial smart radar systems that use SNNs and fast training techniques for deep SNNs.

The robust error back-propagation approach is not immediately useful for training deep SNNs since spike production is discontinuous and non-differentiable. Much recent work has focused on this issue and the learning rules that have emerged fall into three main categories: tandem learning, back-propagation across time with substitute gradient, and SNN-to-ANN conversion. The use of deep SNN for large-vocabulary continuous ASR tasks has not been investigated, despite several successful efforts at large-scale picture categorization. Using a newly suggested tandem learning instruction to facilitate effective and fast implication, this study investigates an SNN-based acoustic model for LVCSR.

Overall, this work mostly contributes to three areas:

- SNN-Based Automatic Speech Recognition for Comprehensive Vocabulary.

Investigate the acoustic models based on SNN for automated voice recognition tasks with a big vocabulary. SNN-based ASR structures attained similar precision to ANN systems in phone gratitude, lower-resourced ASR, and larger-vocabulary ASR processes. It is the opinion this is an initial study to apply SNNs to LVCSR.

- Improving Speech Recognition Speed and Energy Efficiency.

An exciting possibility of fast inference and unparalleled energy efficiency from a neuron's method has been shown in the basic investigation for an SNN-based acoustic model. ASR Toolkit based on SNN.

The PyTorch-Kaldi Speech Gratitude Toolkit allows for the easy integration of SNN-based acoustic models, allowing for the successful construction of SNN-based ASR systems in a short amount of time.

This is the remaining structure of the paper: An introduction to spiking neural networks, ASR systems with a broad vocabulary, and current ASR systems based on SNNs is provided in section 13.2.1. The neural coding strategy that transforms auditory data into spike-based representation is introduced in part 13.3 along with the spike neural method. They present a tandem learning framework to train deep SNN-based acoustic models. Section 13.4 compares the SNN-based acoustic model of the ANN-based implementations and presents experimental findings in the learning capabilities and energy efficacy of the former two kinds of model's gratitude tasks consisting of speech recognition on phone, low-resource ASR, and conventional large-vocabulary ASR. Section 13.5 concludes by presenting the experimental results.

13.2 Related Work

13.2.1 Spiking Neural Networks

Models for the data processing in biotic neural networks, where information is connected and transferred by stereotyped action abilities or spikes, were first examined in third-generation spiking neural networks. Research in the field of neuroscience has shown that the frequency and temporal structure associated with these spike trains play crucial roles in biochemical brain networks as information carriers. Spiking neurons integrate synaptic current from their spike trains asynchronously, as explained in section 3.1. When a neuron's membrane potential reaches the firing threshold, the neuron fires an output spike, which travels up the axon to other neurons in the network.

Traditional artificial neural networks (ANNs) employ analog neurons and SNNs follow the same connectionism concept, so they both use network designs that are either feedforward or recurrent. Based on the initial output spike, the SNN may make an early categorization judgement (Figure 13.1). Nevertheless, when additional data is gathered, the categorization conclusion is usually made better with time. In traditional ANNs, the output layer must wait for all previous layers to be completely updated before it can analyze any new data. This is a major difference. Thus, biological neural systems can execute complicated tasks at a high pace, even though the transmission and processing of information in neural substrates is much slower than in contemporary transistors.

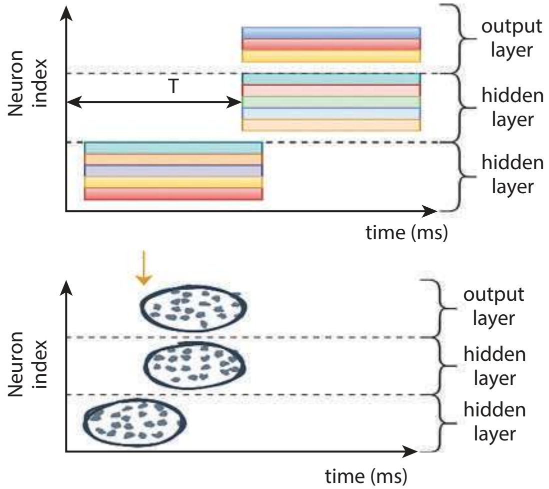


Figure 13.1 Early categorization judgement.

13.2.2 Automatic Speech Recognition with Large Vocabulary

Figure 13.2 shows that traditional ASR systems transform voice signals to their corresponding text using acoustic and linguistic data stored in three separate components: (1) an acoustic model that uses speech attributes to maintain statistical representations of various speech units, such as phones (2) A linguistic model for parabolizing co-occurring word sequences (3) a dictionary of pronunciations to enable the phonetic the transcriptions to be translated into orthography. In the decoding step, these resources are used together to ascertain the most probable hypothesis.

Gaussian Mixture Methods for frame-level phone fundaments and Hidden Markov Models for extent modeling may be used for acoustic modeling. There has been a recent shift toward acoustic models based on artificial neural networks, which provide high-tech performance for a change of automated speech gratitude applications. Numerous end-to-end ANN designs are suggested for straight plotting voice characteristics to word, by the additional language mechanisms being used optionally, in addition to the many ANN architectures investigated for acoustic modeling.

When the voice recognition job is formulated using Bayesian methods, the stochastic explanation for acoustic modeling becomes clearer. $O = (o_1, o_2, \dots . o_t)$ represents the resultant frame-wise characteristics from an intended speech signals that was segmented into T overlapping frames. The output that is recognized by an ASR system is the word sequence $\hat{W}$

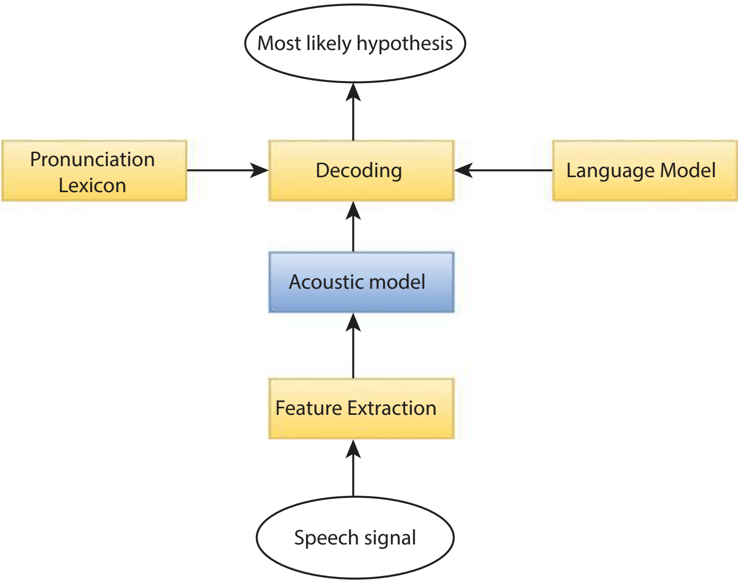


Figure 13.2 Traditional ASR systems.

with the greatest probability, after assigning the probability $P(W|O)$ to all conceivable word sequences $W = [w_1, w_2, \ldots.]$.

$$\dot{\hat{W}} = \arg maxP(W|O) \tag{13.1}$$

By using the Bayes' rule in the following way, split the probability P(W|O) into two portions:

$$\hat{W} = \arg max \frac{P(O|W)P(W)}{P(O)} \tag{13.2}$$

P(O) is independent of W and may be ignored. The outcome is

$$\hat{W} = \arg maxP(O|W)P(W) \tag{13.3}$$

where the theoretical basis of traditional ASR systems is defined in a formal sense. The linguistic model (P(W)) is trained on an extensive textual corpus of the goal language and provides the previous likelihood of the word order W. Assigned to acoustic model is the probability P(O|W), which is the chance that the observable feature classification O will be found given the sequence of words W. In order to distinguish between various acoustic units, the acoustic model records data on the acoustic aspect of speech signals. It is common exercise to use many three-state hidden Markov models for each phone in the phonic script, one for each possible triphone. To decrease model parameters, these HMM states' emission probabilities are connected across models. To assign these probabilities for the frame-level coupled triphone HMM state, the ANN-based acoustic models output layer is trained and constructed appropriately. In order to make the output more commonly used in probability distributions, the output layer applies the SoftMax process. To calculate likelihood, these numbers are scaled with each class's prior probabilities from the training data. The most probable hypothesis is found by combining these likelihood values with the language model's decoding probabilities.

The Spectro temporal motion of the speech signal and the ability to distinguish between distinct telephones in the language being studied are described by speech characteristics, which are used as input to the acoustic model. It is usual practice to combine a GMM-HMM acoustic model with MFCC feature extraction. Initial steps in extracting MFCC characteristics are (1) using a short-time Fourier transform, (2) FBANK calculates energy at each Mel frequency in the log domain using triangular Mel-scaled filter banks. (3) decorrelation of FBANK features using a discrete cosine transform. ANN-based acoustic models can manage feature correlation; therefore, the third stage is generally bypassed and FBANK data are employed. This study compares the ASR performance of deep SNNs with traditional ANNs in a variety of settings, such as phone gratitude, low-resource, and regular large-vocabulary ASR. The goal is to find the greatest ASR solution. The effects of the representation of feature space and its dimensions on the SNN-based acoustic model have been investigated by reporting ASR performance achieved using common speech features.

13.2.3 Spiking Neural Network Speech Recognition

Speech recognition tasks benefit from SNNs' ability to represent and interpret spatial-temporal data. For speech signal discrimination, an unverified spiking-timing-dependent plasticity rule-based SNN feature separator was proposed. On the isolated spoken digit identification test, competitive classification accuracies were established by linking these SNN-based extractors of features using Provision Hidden Markov system and HMM classifiers. SOM-SNN architecture for voice and ambient sound recognition. This architecture employs a feature representation based on the biologically inspired self-organizing map, which converts acoustic data from frames into a sparse and discriminative spike-based representation. A fully SNN-based recognition of speech framework was demonstrated, which uses an innovative membranes potential-driven aggregate-labeling algorithm for learning to train SNNs, and then uses threshold coding to encode the spectral information of successive frames. This allows the SNNs to handle the temporal dynamics of the speech signal. The memory capacity of RSNNs is greater than that of the feedforward frameworks discussed previously. When it comes to voice recognition, their ability to record lengthy periods is invaluable. An enticing architecture of low-power very-large-scale-integration (VLSI) voice recognition using spiking liquid-state machines (LSMs) was introduced. Combine the vanilla with a neural adaptation mechanism to show off top-notch phone detection precision on the TIMIT dataset. This is a first: RSNNs are now competing with LSTM networks in terms of performance on the voice recognition challenge. Phone categorization and isolated spoken digit identification are the extents of these early efforts on SNN-based ASR systems. Here we show that deep SNNs can compete with ANN-based ASR systems in terms of accuracy on LVCSR tasks.

13.3 Methodology

13.3.1 Model of Spiking Neurons

SNN-based acoustic models get frame-based characteristics first. Usually, these characteristics are thought to remain constant across the brief time of segmentation frames because of the short time-based length of these frames and the moderate fluctuation of the speech signal. To efficiently handle this stationary frame-based information with low computing charges, this study presents the integrate and fire neuron model that uses the rearrange by deduction technique. The neural model used in this study does not rely much on spike timings, therefore IF neurons are a good fit. However, they cannot match the complex temporal dynamics of real neurons.

The following process converts the incoming pulses to synaptic current at neuron j at layer l in a discrete-time imitation with a total of Ns period steps:

$$z_j^l(t) = \sum_i w_{ji}^{l-1}.\theta_i^{l-1}(t) + b_j^l \tag{13.4}$$

when a signal spike of afferent nerve cell i occurs at time step t, as shown by $\theta^{l-1}(t)$. Furthermore, the meaning of w^{l-1} is the synapse weightiness that links presynaptic neuron i with layer l – 1. A continuous injecting current might be seen of as b^{l} in this context.

The input current $z_j^l(t)$ is integrated into neuron j's membrane potential $V_j^l(t)$ according to Equation (13.5), as illustrated in Figure 13.3. With each new frame-based feature input, $V_j^l(0)$ the is reset to zero. This assumes unitary membrane resistance. Assuming normalized synaptic weights, an output spike occurs when $V_j^l(t)$ reaches the activation limit ϑ (Equation 13.6), where is set to 1 in all experiments.

$$V_j^l(t) = V_j^l(t-1) + z_j^l(t) - \vartheta.\theta_j^l(t-1) \tag{13.5}$$

$$\theta_j^l(t) = O\left(V_j^l(t) - \vartheta\right) \, with \; O(x) = \begin{Bmatrix} 1, & if \; x \geq 0 \\ 0, & otherwise \end{Bmatrix} \tag{13.6}$$

Equations (13.4) and (13.5) provide the allowed aggregate crust voltage of neuronal j (not at all fire) in layers l as

$$V_j^{l,f} = \sum_i w_{ji}^{l-1}.c_i^{l-1} + b_j^l.N_s \tag{13.7}$$

In Equation (13.8), c_i^{l-1} represents the effort spike frequency from pre-synaptic neuronal i at layer l – 1.

$$c_i^{l-1} = \sum_{t=1}^{N} \theta_i^{l-1}(t) \tag{13.8}$$

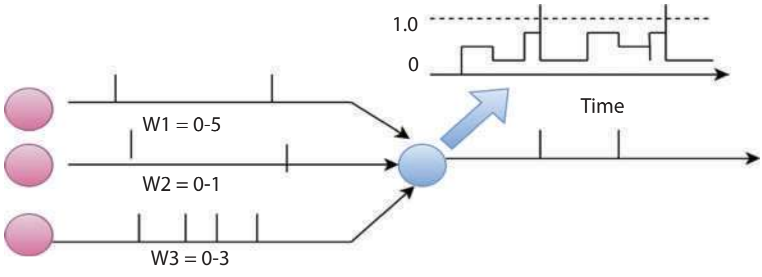


Figure 13.3 New frame-based feature input.

The $V_j^{l,f}$ summarizes pre-synaptic neuron spikes' membrane potential contributions without considering their temporal structures. This intermediate quantity bridges SNN and linked ANN layers for parameters optimization, as stated in the tandem training framework.

13.3.2 Neurocoding Arrangement

Spike trains are used by SNNs, therefore specific techniques are needed to encrypt continuous-valued feature courses and decode classification results from output neuron activity. Consequently, implement the suggested spiking neural encoding strategy. This encoding method begins by passing a weighted layer consisting of corrected linear units ReLU neurons through a frame-based feature input vector X^0, which may be described as $X^0 = \left\lfloor x_1^0, x_2^0 \ldots, x_n^0 \right\rfloor$T, in the following way:

$$V_j^{0,f}(0) = a_j^0 = p\left(\sum_i w_{ji}^0 . x_i^0 + b_j^0\right) \qquad (13.9)$$

when w_{ji}^0 is the synaptic strength measured from input x_i^0 to ReLU neuron j. A neuron's associated bias term is represented by b_i^0, while the activation function of ReLU is denoted by ρ(·).With each ReLU neuron having an activation value of a_i^0, we may find its free aggregate membrane potential, $V_j^{0,f}(0)$. the amount according to Equations (13.10) and (13.11) and then depict it using spike trains across the encoding time frame Ns.

$$\theta_j^0(t) = O\left(V_j^{0,f}(t-1) - \vartheta\right) \qquad (13.10)$$

$$V_j^{0,f}(t) = V_j^{0,f}(t-1) - \vartheta \cdot \theta_j^0(t) \qquad (13.11)$$

The following is a graphical representation of the neural encoding layer's output, which includes the spike trains s^0 and spike count c^0.

$$s^0 = \left\{\theta^0(1) \ldots \theta^0(N_s)\right\} \qquad (13.12)$$

$$c^0 = \sum_{t=1}^{N_s} \theta^0(t) \qquad (13.13)$$

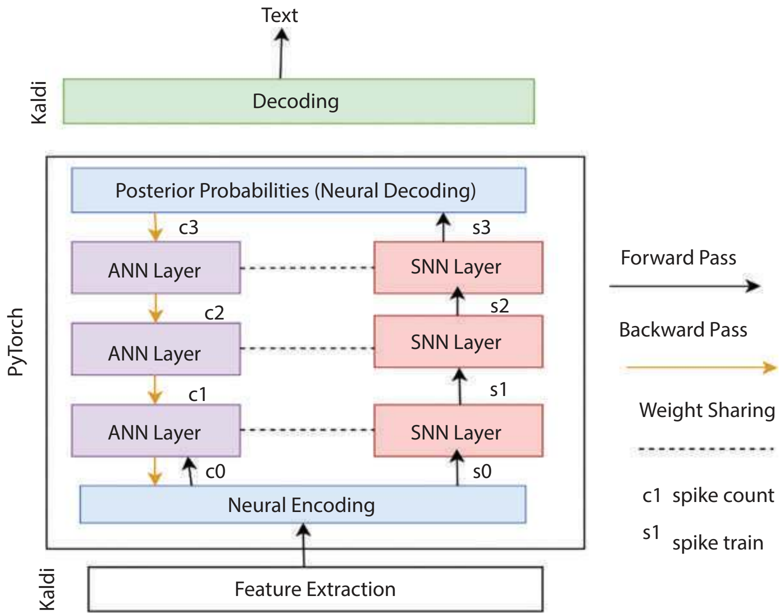


Figure 13.4 Tandem neural network SNN training flowchart, using SNN layers to calculate spike frequency.

Encoding layer conducts weighted transformation in an end-to-end learning system. The program adjusts the input representation to fit the encoding time frame Ns and displays the converted data using spike counts. Since input data may be programmed in a short time, this encoding strategy speeds up inference. Figure 13.4 shows how to apply spike count c^1 and s^1 to successive ANN and SNN layers for paired erudition from the brain encoding layer. For neural decoding, output spiking neurons' free aggregated membrane potential is employed to enable smooth learning with high-precision error gradients at the output layer. Since input vectors of features and output classes have a lower dimensionality than hidden layers, deploying them on edge devices will need less processing.

13.3.3 Deep SNN Training with Tandem Learning

ANN neuron activation value and IF neuron spike count are linked in the simultaneous neural network SNN learning rule. Since input features are represented as spike counts, spike train time-based structure is unimportant. Thus, SNN layer effective non-linear transformation is

$$c_j^l = f\left(s^{l-1}; w_j^{l-1}, b_j^l\right) \tag{13.14}$$

where spiking neurons transform at f(). Detecting an analytical expression from s^{l-j} to c_j^l is not feasible owing to the state-dependent nature of spike creation. Assuming equally distributed synaptic currents from s^{l-j} across the encoding time frame simplifies the spike generating procedure. Determine the interspike interval as follows.

$$ISI_j^l = p\left(\frac{\vartheta}{V_j^{l.f}/N_s}\right) = p\left(\frac{\vartheta}{\left(\sum_i w_{ji}^{l-1} c_i^{l-1} + b_j^l.N_s\right)/N_s}\right) \tag{13.15}$$

Accordingly, the estimated “spike count” al may be calculated using

$$a_j^l = \frac{N_s}{ISI_j^l} = \frac{1}{\vartheta}.p\left(\sum_i w_{ji}^{l-1} c_i^{l-1} + b_j^l.N_s\right) \tag{13.16}$$

An ANN layer of ReLU neurons may calculate a_j^l given a particular firing threshold ϑ by setting the spike sum as c_j^{l-1} input and the aggregate continuous injected current b_j^l·Ns as bias. This spike production interpretation enables the ANN layer to mimic spike-train level error gradients. In a high-dimensional environment, the cosine disparities between the estimated ‘spike count’ a l with the SNN output spike count $arec^l$ are very tiny, demonstrating the connected ANN layers may approximate high-quality error gradients.

Create a dual neural network as depicted in Figure 13.4 using this approach. The SNN layers define the precise spike representation and transmit the total peak count and spike trains to the succeeding ANN and SNN levels during activation forward propagation. This interlaced layer topology synchronizes input sent to the linked ANN and SNN layers. ANN is utilized to train SNN, however, solely SNN is used for inference.

13.3.4 The SNN-Based Acoustic Model

Section 2.2 describes how popular speech components were collected from training records to train deep SNN-based acoustic models, this work’s key contribution. To gain temporal context, these input voice characteristics are spliced over numerous frames before being supplied to SNNs.

A GMM-HMM-based ASR system aligns speech characteristics with target senone labels after learning the SNN-based acoustic model. Passing input voice frames through several spiking neurons during training teaches the deep SNN to chart speech characteristics to sense later probability.

The qualified SNN model acoustic scores are merged with the model of language and pronunciation lexicon data during inference. Weighted finite-state transducers are often used to create the search graph with plausible hypotheses by unifying ASR resources. The WFST-based decoding is motivated by (1) the simple synthesis of ASR resources to translate HMM values to word sequences. (2) WFST-based search methods that accelerate decoding.

A lattice of plausible hypotheses is identified throughout the search. The weighted total of lattice hypotheses' acoustic and linguistic model scores determines the ASR output value. WFST-based decoding was employed in this study. The SNN-based acoustic modeling's recognition performance in numerous recognition contexts is tested in the following ASR studies.

13.4 Results and Discussion

13.4.1 TIMIT Corpus Phone Recognition

Speaker-independent features' greatest performance on TIMIT corpus development and test sets is shown in Figure 13.5. To compare ASR performance, the top panel shows several high-tech schemes utilizing ANN and SNN constructions. SNN-based acoustic models are adaptable to various speech characteristics and perform similarly or slightly worse than ANN by the same system configuration, as demonstrated in Figure 13.5. Particularly, the ANN scheme taught with the conventional 13-dimensional FBANK feature has the best development set PER of 16.9%. The identical SNN system with the same feature has 17.3% (18.7%) PER on the testing set. The wider temporal context examined by the recurring Li-GRU model explains why state-of-the-art ASR schemes have 1% lower PER than the suggested SNN-based phone recognition scheme. Still, spiking neural networks struggle with phone recognition. The only recent study using recurring spiking neural networks on this corpus shows good test results with a PER of 26.4%. Compared to this early SNN-based acoustic modeling work, the system has a much lower PER. Since the suggested system decodes using an auditory and linguistic model, these findings are not comparable. Experimental findings on the TIMIT telephone detection test suggest SNN-based audio modeling have promising possibilities. Compare ANN

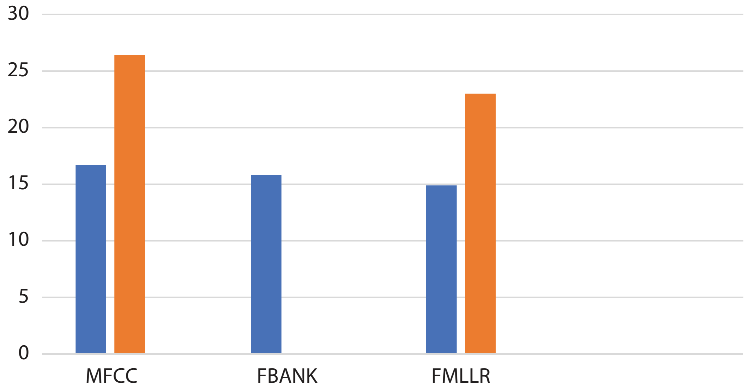


Figure 13.5 TIMIT development and test set PER (%).

and SNN performances in newer LVCSR corpora as the phone detection challenge in the TIMIT corpus is simpler than recent LVCSR workloads.

13.4.2 Librispeech Corpus Experiments Using LVCSR

Auditory models employing the authorized 100 and 360-h train subset of the Librispeech corpora were used to compare ANN and SNN models in the conventional LVCSR situation in the final ASR trials. ANN systems outperform the SNN system across all speech characteristics after 100 h of data for training, as seen in Figure 13.6 center panel. Because discrete spike counts have less representation power, SNN models may perform somewhat worse. This is encouraging even compared to high-tech ASR schemes utilizing more complicated ANN structures, as seen in Figure 13.6 center panel.

Increased training data benefits ANN and SNN systems. For development (test) sets, the greatest SNN models' WERs decreased from 10.0% (10.3%) to 9.2% (9.4%) as training data increased from 100 to 360 hours. ANN and SNN-based acoustic replicas perform similarly for LVCSR responsibilities. These findings indicate that SNNs may be suitable for acoustic modeling.

13.4.3 SNN-Based ASR Systems Energy Efficiency

SNN-based ASR systems on low-power neuromorphic circuits may offer exceptional performance gains and promising modeling capabilities. This section illuminates this potential by comparing ANN- and SNN-based

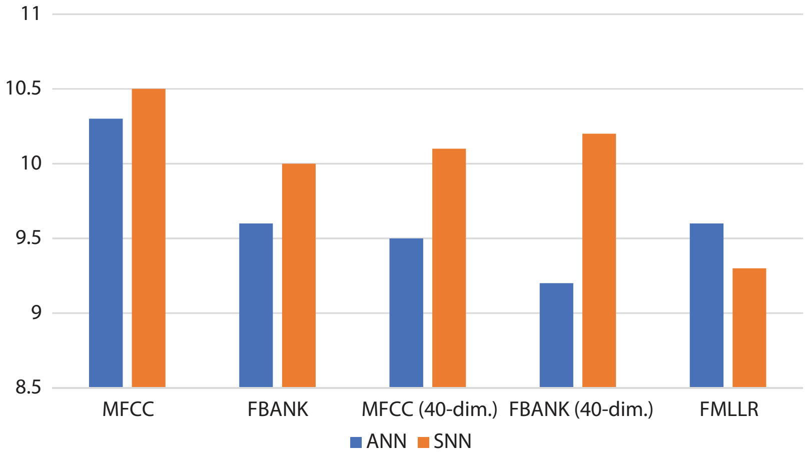


Figure 13.6 ANN systems outperform SNN system.

acoustics models energy efficiency. Information moves are the most energy-intensive process for data-driven AI. SynOps(SNN)/SynOps(ANN) of 5 randomly selected sentences through the TIMIT quantity and provide a relation of average synapse processes per feature categorization. The 40-dimensional MFCC, FBANK, and FMLLR characteristics in Figure 13.7 and Figure 13.8 were analyzed to see how feature representations affect them.

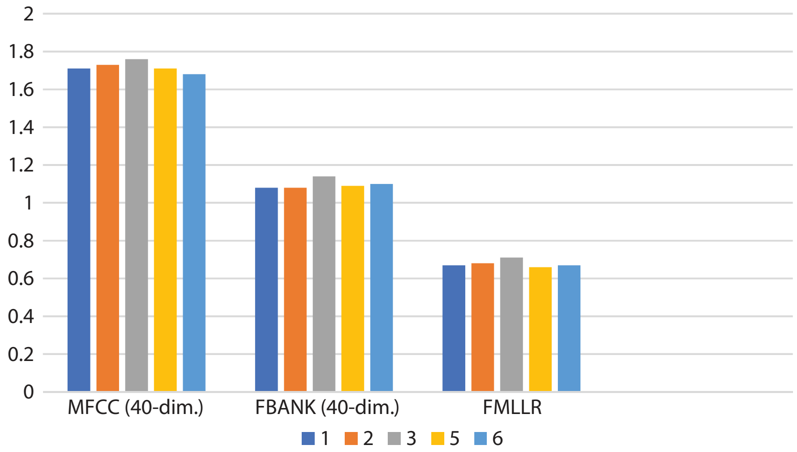


Figure 13.7 The 40-dimensional MFCC, FBANK, and FMLLR characteristics.

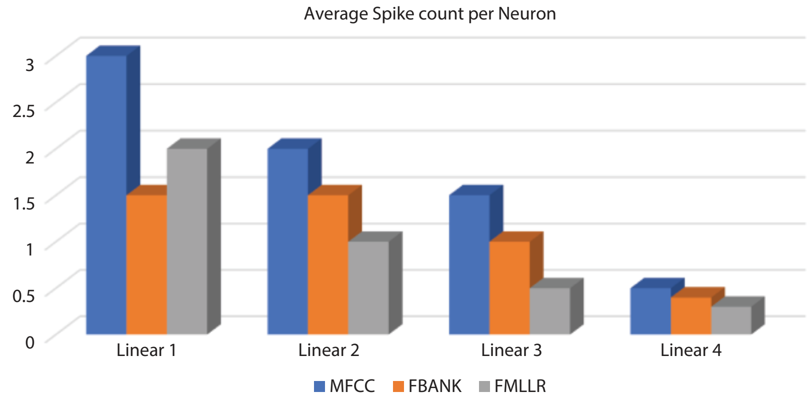


Figure 13.8 Feature representations of affection.

This experiment found that FMLLR had the smallest average spike rate among the three characteristics. Although the FMLLR feature is not always present in ASR circumstances, the speaker-dependent feature's greater discrimination may explain it. The average number of synaptic processes for SNNs with MFCC and FBANK features is somewhat larger than ANNs; however, AC operations are substantially cheaper than MAC operations. MAC operations cost 14 times more than AC processes and need 21 times more spot space, according to a Global Foundry 28 nm process evaluation. This work suggests that installing SNNs on neuromorphic chips that are used for inference might save energy and chip area. SNN-based acoustic models' energy savings rely on chip designs and materials, which are outside the scope of this study.

Autonomous voice recognition technologies have transformed the human–computer interaction. Computing effectiveness, real-time presentation, data security, and other issues have arisen as ASR services demand grow rapidly. Thus, it inspires new ideas to overcome such issues. Brain-inspired spike neuronal systems for big vocabulary ASR problems use event-driven control like biological neural systems. A unique SNN-based ASR framework that models frame-level properties and maps them to acoustic units. To discover the most probable term classification for the input voice indication, this frame-level output will combine word-level data collected by the language model.

13.5 Conclusion

Recurrent neural networks representing temporal signals use extended temporal contextual information in input signals. Speking neuron recurrent networks improve large-vocabulary ASR recognition. To minimize ANN inference computational costs and memory footprint, researchers have used network compression, quantization, and knowledge distillation. neuromorphic computer architectures offer great potential since biological neural networks employ a different computational paradigm than ANNs. Use connectionism to encode ANN and SNN information on network connectivity and strength. Initial network reduction and quantification initiatives may assist SNN reduce memory and computing costs. Synthetic semiconductor cochlea audio sensors convert audio inputs into spiking events. The environment delivers temporally sparse information, hence these sensors code more effectively than microphone sensors. Initial silicon cochlea sensor input spiking ASR studies are fascinating. The event-driven semiconductor cochlea audio sensor and SpiNNaker neuromorphic processor are connected via the Address-Event Representation protocol in an SNN-based spoken command recognition framework. To ensure real-time performance, buffering is included. These study' ASR workloads are small relative to modern standards due to the absence of event-based corpora. Mic-collected large-scale ASR corpora may be converted into spiking events via perceptually directed auditory neural encoding. Without altering audio quality, this encoding method may reduce 50% of spikes. After studying neuromorphic auditory front-end research, predict energy-efficient SNN-based ASR systems.

SNN-based big lexicon ASR systems opens up potential for integrating advanced ASR engines into power-limited mobile and embedded devices. The long-term performance of SNN-based ASR systems is predicted to increase as research on innovative neuromorphic auditory front-ends, SNN designs, neuromorphic computer architectures, and ultra-low-power stable memory devices grows.

Bibiliography

1. Hinton, G., *et al.*, Deep neural networks for acoustic modelling in speech recognition: The shared views of four research groups. *IEEE Signal Process Mag.*, 29, 6, 82–97, 2012.
2. Lippmann, R.P., Review of neural networks for speech recognition. *Neural Comput.*, 1, 1, 1–38, 1989.

3. Abdel-Hamid, O., *et al.*, Convolutional neural networks for speech recognition. *IEEE/ACM Trans. Audio Speech Lang. Process.*, 22, 10, 1533–1545, 2014.
4. Mohamed, A.-R., Dahl, G.E., Hinton, G., Acoustic modelling using deep belief networks. *IEEE Trans. Audio Speech Lang. Process.*, 20, 1, 14–22, 2011.
5. Sak, H., *et al.*, Fast and accurate recurrent neural network acoustic models for speech recognition, arXiv preprint arXiv:1507.06947, 2015.
6. Seltzer, M.L., Yu, D., Wang, Y., An investigation of deep neural networks for noise robust speech recognition. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, 2013.
7. Nassif, A.B., *et al.*, Speech recognition using deep neural networks: A systematic review. *IEEE Access*, 7, 19143–191655, 2019.
8. Zhang, Y., *et al.*, Towards end-to-end speech recognition with deep convolutional neural networks, arXiv preprint arXiv:1701.02720, 2017.
9. Abdel-Hamid, O., *et al.*, Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition. *2012 IEEE international conference on Acoustics, speech and signal processing (ICASSP)*, IEEE, 2012.
10. Graves, A., Mohamed, A.-R., Hinton, G., Speech recognition with deep recurrent neural networks. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, 2013.
11. Deng, L., Hinton, G., Kingsbury, B., New types of deep neural network learning for speech recognition and related applications: An overview. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, 2013.
12. Qian, Y., *et al.*, Very deep convolutional neural networks for noise robust speech recognition. *IEEE/ACM Trans. Audio Speech Lang. Process.*, 24, 12, 2263–2276, 2016.
13. Cui, X., Goel, V., Kingsbury, B., Data augmentation for deep neural network acoustic modelling. *IEEE/ACM Trans. Audio Speech Lang. Process.*, 23, 9, 1469–1477, 2015.
14. Sak, H., Senior, A.W., Beaufays, F., Long short-term memory recurrent neural network architectures for large scale acoustic modelling, 2014.
15. Bhargava, N., Kumar Sharma, A., Kumar, A., Rathoe, P.S., An adaptive method for edge preserving denoising. *2017 2nd International Conference on Communication and Electronics Systems (ICCES)*, Coimbatore, India, pp. 600–604, 2017, doi: 10.1109/CESYS.2017.8321149.
16. Kingsbury, B., Lattice-based optimization of sequence classification criteria for neural-network acoustic modelling. *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, 2009.
17. Choudhary, N., Rathore, P.S., Kumar, D., Comparative Evaluation of Marker-Controlled Method and Gradient Distance Transformation through Watershed Segmentation. *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, Pune, India, pp. 1–6, 2024, doi: 10.1109/I2CT61223.2024.10543576.

18. Kumar, A., Chatterjee, J. M., Díaz, V. G., A novel hybrid approach of SVM combined with NLP and probabilistic neural network for email phishing. *Int. J. Electr. Comput. Eng.*, 10, 1, 486, 2020.
19. Benkhaddra, I., Kumar, A., Setitra, M.A., *et al.*, Design and Development of Consensus Activation Function Enabled Neural Network-Based Smart Healthcare Using BIoT. *Wireless Pers. Commun.*, 130, 1549–1574, 2023, https://doi.org/10.1007/s11277-023-10344-0.

14

Brain–Computer Interface for Humanoid Robot Control Adaptation

B. Sai Chandana[1]*, K.S. Chakradhar[2], T. Rajasanthosh Kumar[3] and Makhan Kumbhkar[4]

[1]School of CSE, VIT-AP University, Amravati, India
[2]Department of ECE, Mohan Babu University, Tirupathi, India
[3]Department of Mechanical Engineering, Puducherry Technological University, Puducherry, India
[4]Department of Computer Applications, ICAR-Indian Institute of Soybean Research, Indore, India

Abstract

Modern developments in both robotics and neuroscience have made it possible to show off the first brain–computer interfaces (BCIs) for commanding robots with human-like characteristics. But earlier BCIs depended on command and control at a higher level, depending on behaviours that were hardwired in. Conversely, the BCI user may have a heavy cognitive strain due to the monotony of low-level control. To overcome these issues, an adaptive hierarchical method of brain–computer interaction was suggested. In this method, users train the BCI system to do new tasks automatically; thereafter, they may call upon these abilities directly as high-level instructions, which eliminates the need for repetitive manual control. This research delves into the use of hierarchical BCIs for training and controlling a PR2 humanoid robot. Consider using explicit command sequences to enable users to design complicated tasks with numerous state spaces. Three people successfully trained and controlled the PR2 robot with humanoids using a hierarchical EEG-based BCI to pour milk over cereal in a simulated domestic scenario. The first hierarchical BCI training for non-navigational tasks is presented. The example is the first to use one model for training a difficult problem with several state spaces.

***Keywords*:** Brain–computer interface, hierarchical method, humanoid robot, non-navigational task

**Corresponding author*: saichandana.bolem@vitap.ac.in

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (227–242) © 2025 Scrivener Publishing LLC

14.1 Introduction

The idea of humanoid robots working remotely under human supervision has piqued the imagination of many in the robotics field. In both safe and hazardous locations, people rely on humanoid robots to help carry out various duties [1]. Robots with human characteristics have also been considered for use in the field of communications. The usage of a brain–computer interface for direct brain-based regulators is a novel approach for operating a robot with human characteristics (refer to Figure 14.1) [2–4]. Achieving such control requires dealing with the large degrees of freedom that are inherent to robots designed to mimic humans [5]. This is the issue with other prevalent methods of managing humans, whether utilizing a joystick, voice recognition, or visual feedback systems [6]. The combination of a humanoid with a BCI, which often has poor throughput, compounds this issue [7]. The amount and precision of instructions the user may discern between in any one time period are constrained by the BCI system's low throughput [8–10].

The use of BCIs as controllers for robotic systems has drawbacks, such as poor throughput [10–13]. The bandwidth of control is limited with non-invasive BCIs because of the poor signal-to-noise ratio. This is especially true for BCIs that rely on scalp EEG inputs [14]. Although invasive BCIs that connect to brain neurons provide fine-grained control, the constant micromanagement required by such devices might wear people down [15–17].

Adaptive hierarchical BCI (HBCI) design, which enables the operator to continuously teach the structure fresh and beneficial activities, was previously offered as a solution to these issues [18]. To begin, the user shows the robot a lower-level ability by telling it to turn left, for example, rather than more complex instructions [19–21]. First, the user issues the general instruction, and the robot subsequently does the task to a certain extent on its own [22]. After learning a command, the user is no longer reliant on the arduous task of controlling it moment by moment due to this higher-level control [23–25].

Limiting their sphere of application, these prior HBCIs were exclusively applicable to navigational tasks. This work investigates the potential of HBCIs during a close-range setting, defined here as the collection of points within the manipulatable range of a humanoid robot [26]. They demonstrate the process of learning arm trajectories with the HBCI. Within a limited area, the operator may direct the robot's grasping mechanisms using this technique. Secondly, they provide a straightforward method for memorizing sequences, which permits the operator to construct new higher-level abilities by combining several lower-level capabilities with motor primitives, including opening and shutting the gripper and rotating it [27, 28]. This sequence of instructions will allow the user to accomplish the more complicated job in the future. The training of more engaging tasks is made possible by the fact that sequencing in this context includes individual instructions at all levels of hierarchy control, so it is not limited to someone's robot state space.

The method shown here is readily extensible to any degree of hierarchy; however, it only lets the operator sequence lower and middle-level abilities [29]. There is no limit to the number of levels of instructions that may be included in a sequence; in fact, sequences describe whole lists of jobs. This method may easily be extended to make breakfast, which involves a whole number of difficult operations; for instance, it shows how to pour milk over cereal [30].

Three participants were able to effectively operate and train a PR2 humanoid robot using a hierarchical EEG-based BCI, and present their findings. The PR2 has to be trained to mimic the motions of pouring milk over cereal. These results point to the possibility that HBCIs provide a flexible and effective means of training and controlling humanoid robots to solve complicated problems through brain signals [31].

14.2 The System Architecture

Three parts to the modular system make up the hierarchical architecture:

- A brain–computer interface that uses steady-state visually evoked potentials (SSVEPs) to amplify the brain's already-present noise. A user is exposed to visual stimuli that oscillate for the BCI to function. At the base of the skull, in the occipital lobe, is where the electrical activity that corresponds to the frequencies of this oscillation (and its multiples) may be detected. By selecting a stimulus (and, by extension, a frequency), the user gives instructions. Figure 14.1a shows how the BCI tries to deduce the user's selected command by measuring the appropriate EEG activity. However, it should be noted that other BCI paradigms that provide discrete categorization might also be used;
- A flexible and hierarchical menu;

- The Willow Garage PR2 semi-humanoid robot, shown in Figure 14.1b has two degrees of freedom (DOFs) in its head and seven or eight DOFs in each of its limbs; however, other humanoid robots might also be used.

To operate a higher degree of freedom robot that is semi-humanoid, they scaled up the interface and made various improvements to the proposed system. The inter-component communication was handled using ROS. Detailed descriptions of each part are provided below.

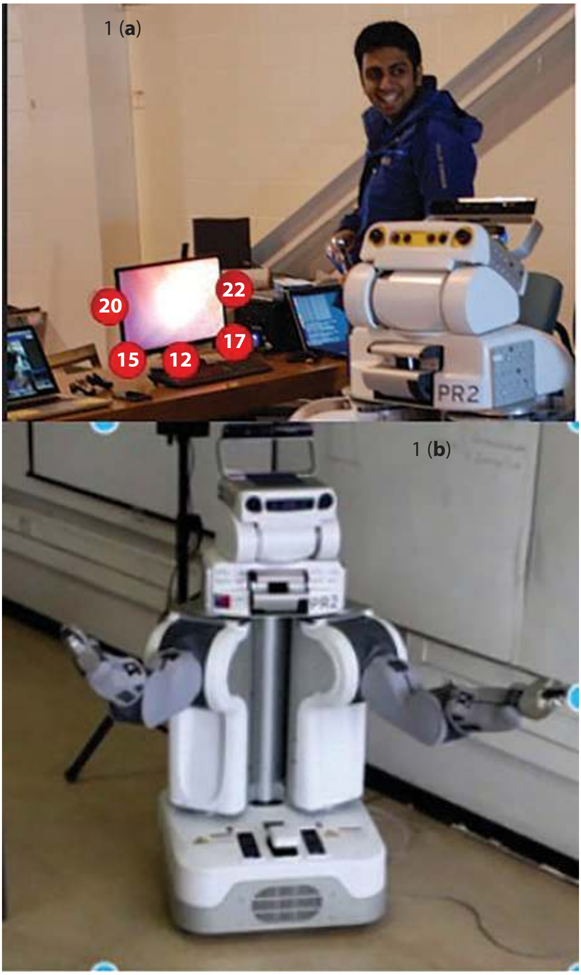


Figure 14.1 (a) BCI measures EEG activity to determine the user's instruction with LED lights. (b) Semi-humanoid PR2 robot.

14.2.1 BCI Based on SSVEP

SSVEPs were generated using light-emitting diode stimuli. For SSVEP-based BCIs, they discovered that LEDs provide more frequency flexibility. Five red LEDs were placed in spherical lightboxes. The frequency at which they oscillated were 12, 15, 17, 20, and 22 Hertz. To maximize the stimulus impact, the lights were attached to the monitor alongside their respective set menu items and draped behind a diffusive material, as seen in Figure 14.2.

A bipolar recording of continuous EEG signals was performed using electrodes that were gold positioned at two conventional sites on the brain: Fpz on the forehead and Oz in the middle of the back. The Fpz position is where they establish ground. A 256-Hz digitalization followed by a 60-Hz notch filtering of the data.

To categorize the user's input, they used the Fast Fourier Transform (FFT) to estimate the spectrum power of the signal as shown in Figure 14.3. Every half second, the Fourier transform (FFT) was performed to a 1.0-s slice of the electroencephalogram (EEG) data, and the squared value of the power at each frequency was then calculated. A 4-second power-value average was the data set used for the categorization. The user's selection for the time frame was determined by calculating the average frequency and sending the corresponding orders to the robotic and menu system.

The hierarchical architecture supports other BCI paradigms beyond SSVEP. A hierarchical command structure may be used in any BCI that allows discrete categorization as shown in Figure 14.4. The usage of a hierarchical command structure is not restricted to SSVEP.

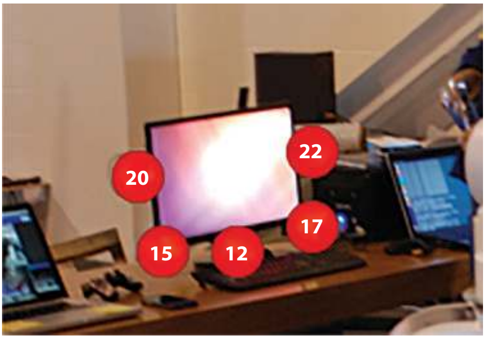


Figure 14.2 Control interface: SSVEP.

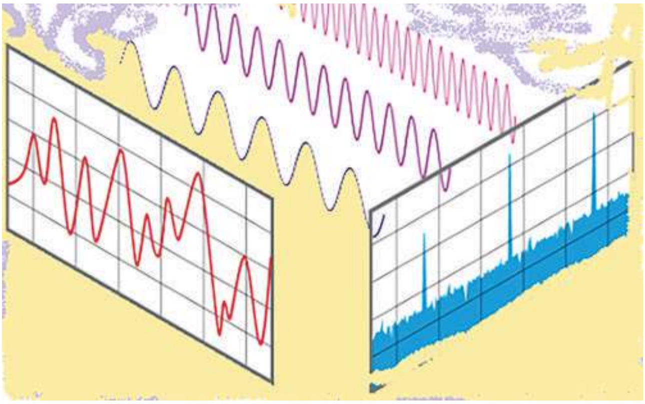


Figure 14.3 Fast Fourier Transform.

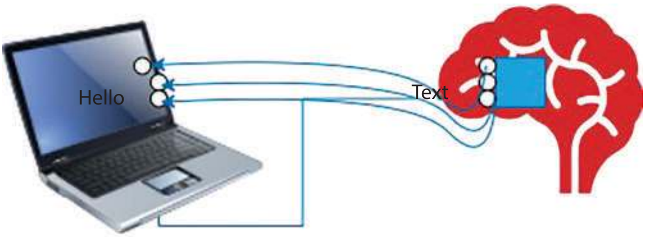


Figure 14.4 Brain–computer interface.

14.2.2 Adaptable Hierarchy

The user may access all parts of the system using the hierarchical menu as shown in Figure 14.5. From this menu, the user may guide the robot's movements and engage with its learning system to impart new abilities. There are five menu options shown down the screen's edge. The screen's central portion shows a live feed from the robot's head camera. Five options—Train, Training Series, Navigate, Playback, and Playback Series—are shown on the menu at the beginning of the experiment.

Users may teach the machine new arm trajectory skills by choosing the "train" command. The user may complete the new skill's training or move the arm backwards, forward, left, or right using this menu. The user is given the choice to either store the skill or dismiss the acquired data when they pick "stop" from the save menu. Several confirmation options are shown to the user to prevent misclassification, which strengthens the selection. A policy that maps robotic states to actions is generated by the learning system using recorded data when the user submits a sample trajectory of the ability to learn.

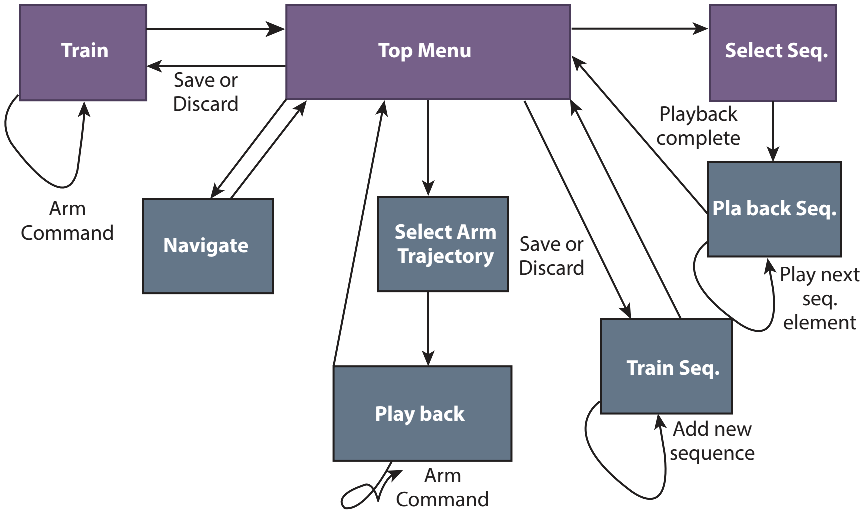


Figure 14.5 Hierarchical menu.

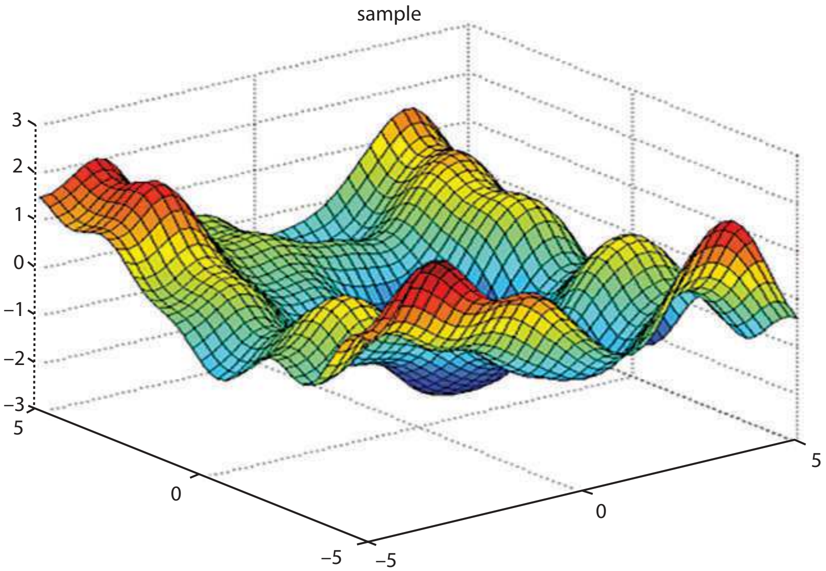


Figure 14.6 Gaussian process.

The policy is formulated through Gaussian Processes (see the Gaussian process example in Figure 14.6). With this strategy, adjust the beginning point of playback to some extent since it interpolates for locations that

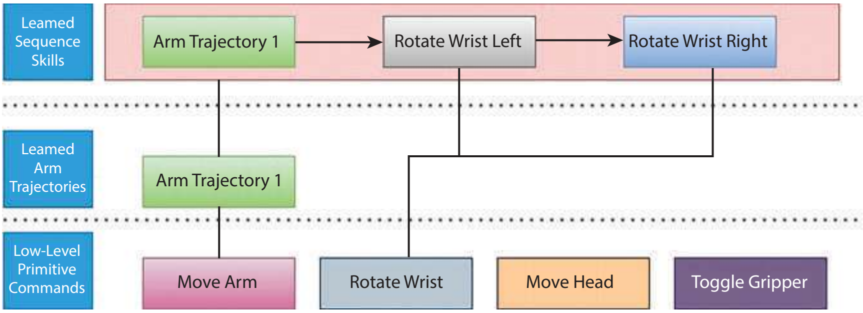


Figure 14.7 BCI architecture.

aren't on the original training route. Once the user has mastered a skill, they may either put it to the test via the Playback menus or incorporate it into a more advanced command sequence.

Be aware that there is no hard limit to the number of possible screen-to-playback sequences; in this instance, they just have four options. To accommodate all of the necessary abilities, they might permit an infinite number of menu panels. In addition, the HBCI system may filter present sequences according to how relevant they are to the robot's current state. For instance, if the gripper is empty, the system will not display sequences that involve dropping an object since they do not involve grasping. This way, the user doesn't have to go through as many screens.

One may create a more sophisticated set of instructions, including gripper opening and closing and arm trajectory training, using the "train sequence" option. The user's intent is validated when a confirmation box appears after they choose a talent to include in the new command sequence. The sequence may be enhanced with a variety of abilities. The user may access their newly saved sequence from the Sequence Playback menu once they've clicked "Exit" and saved it. Each item in a user-defined high-level command sequence will run in succession when the user presses the execute button. Figure 14.7 shows hierarchical BCI architecture, in which the user may control the robot's movement by angling its head to the left, right, up, and down while it carries out these orders.

14.2.3 Robot and Software for Robots

A Willow Garage PR2 mimic-humanoid robot served as the experimental subject for the present study. Applying ROS's pre-programmed library functions, they introduced the user to fundamental lower-level motor primitives. Some examples of these primitives are the right wrist link's rotation and the right gripper's open/closed toggle. The ability to rotate the head-mounted camera is also included in the library's head joint manipulation features. The last step is to rotate the gripper such that it moves laterally, forwards, and rightwards concerning the body.

The suggested position-based trajectories learning framework is the primary emphasis of the present system, which is learning how to move the arms. More complicated learning schemes, such as those involving the simultaneous or independent control of numerous joints, might be implemented using this method. Such strategies will soon be investigated by experts. The user was able to monitor the robot's movements in real-time appreciation to a camera mounted on its helmet.

14.3 Procedure for Experimentation

Three physically capable men took part in the research. Written informed permission was obtained from all individuals. The participants were told to teach the robot a series of commands that would resemble pouring milk. The experiment presupposes that the robot's right gripper is holding an opened container containing milk. Pouring the milk into the receiving container requires the user to twist their wrist and move the container across to the opposite side of the table (see the sequence execution summary in Table 14.1). They added difficulty to the assignment by placing a blue block between the beginning point of the right gripper and the objective point to demarcate the path.

Instructions were given to the subject in a single session to

- First, train an arm trajectory that goes around the difficulty and ends up where they want it to;

Table 14.1 Sequence execution summary (subjects 1 to 3).

Numbers	**1**	**2**	**3**
1	Y	Y	Y
2	Y	Y	Y
3	Nil	Y	Y

- Second, build a set of commands that represents the whole complicated skill;
- Third, play back the learnt series from three distinct beginning points.

The position-based approach to learning arm trajectories was shown resilient after three resets of the robot while the user was instructed to execute the learned skill. The original location of the robot's beginning state was changed with each reset. Almost every user started from the same place. From the starting position, they watched whether the gripper terminated above the bowl of cereal to determine success.

14.4 Results

A higher-level command sequence and an arm trajectory were both defined by the patients using the HBCI. There was never an instance when the user had to use more commands than necessary to accomplish a more complicated job because of the command sequence. Since this is the same as the number of instructions needed to do the job utilizing a mix of lower- and mid-level abilities, then compared directly with the number of commands required to define the sequence initially. The number of instructions needed to complete the assignment was considerably decreased in every instance when the sequence skill was used as displayed in Figure 14.8.

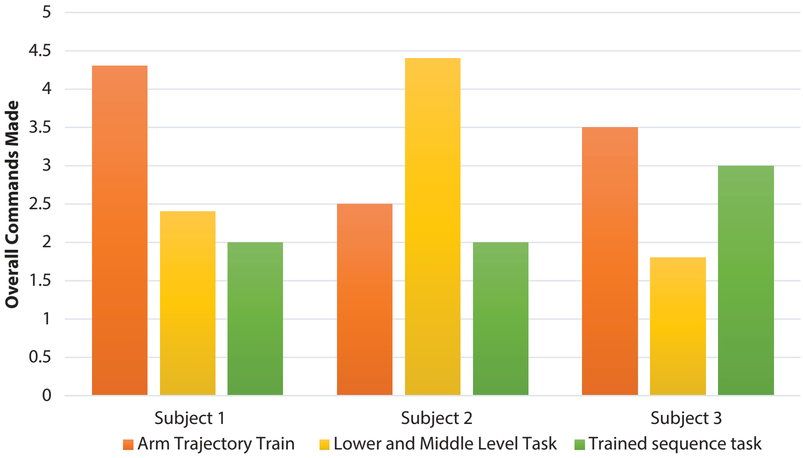
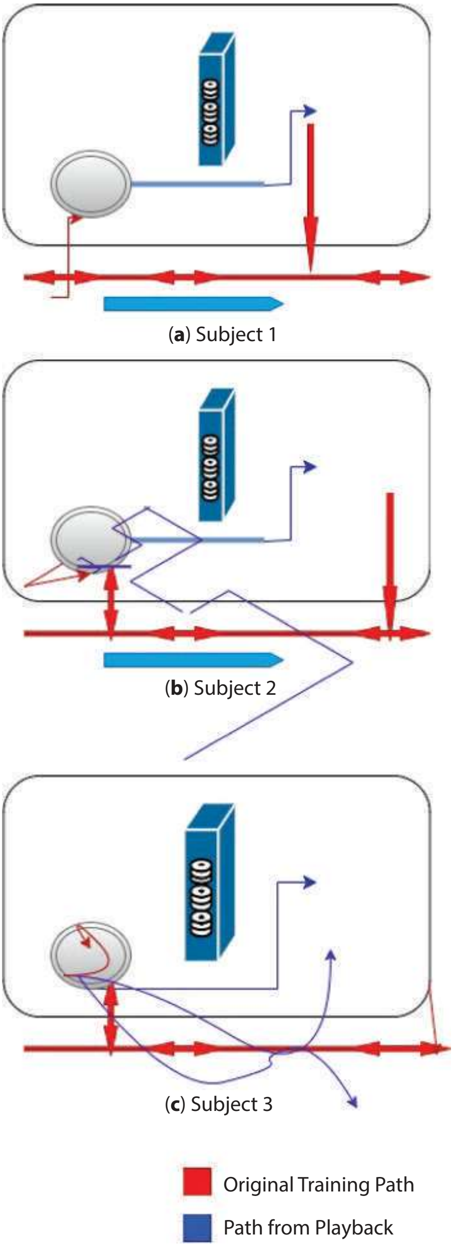


Figure 14.8 Task vs. users count.



(**a**) Subject 1

(**b**) Subject 2

(**c**) Subject 3

Figure 14.9 (a), (b), (c) Arm trajectories [red denotes the first training path. Blue paths represent the effect of implementing the taught policy. Beginning states are circles. Squares refer to successful outcomes].

Tracking the user's command input during each trial phase. The cognitive burden is directly proportional to the user's command volume. Figure 14.9 (a), (b), and (c) shows the arm trajectory graphs, which provide some fascinating observations. To begin, except for one user, the training sequence was successfully performed for all users regardless of the beginning condition. When compared to the initial state, the learnt policies for arm trajectories seem to be rather strong.

Second, it seems that the quantity of noise in the user's initial training data determines how robust a particular arm trajectory skill is. For instance, there were instances when Subject 1 struggled to distinguish between left and right arm movements. Starting from the lower starting state, the robot was unable to finish the job due to this, since it impacted the form of the learnt course of action in that region. Even though Subject 2's training information was noisy towards the conclusion of the trajectory, the learnt policy was strong enough to enable the robot to independently complete the job (although it slightly overshot the target on one occasion). This provides further evidence showing that the position-based trajectories learning technique mitigates the impact of data noise introduced by the users during training. When the recorded trajectories of Subject 3's arm travelled across space, it seemed as if Subject 3 had complete command of the robot. The findings suggest that the trajectory learning approach may automatically eliminate certain inefficiencies in the user's original training example.

The outcomes of observing user instructions in the BCI system are shown in Figure 14.10. Users may specify a skill once and then perform it

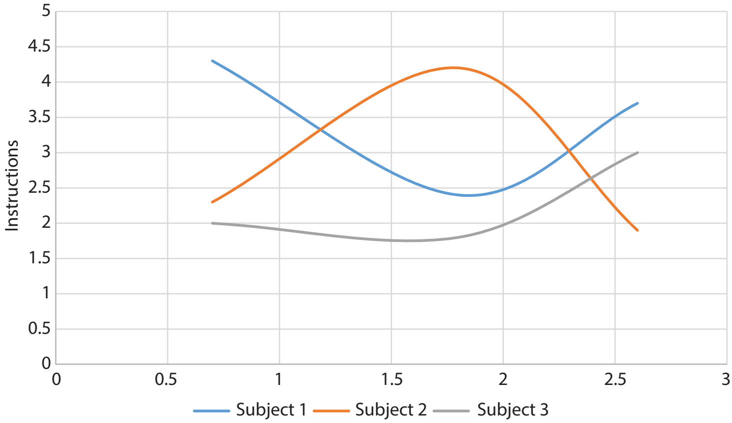


Figure 14.10 BCI system outcome.

as several times as they want when they train a sequence. Choosing a learnt command sequence from a hierarchical menu system involves very little cognitive work, as one would assume when compared to training or executing a sequence. Users utilized about 15 instructions when they individually described the job by choosing the arm trajectories and wrist rotation primitives. A significant decrease in the number of instructions needed to complete a job is implied by this reduction. As a result, the BCI's throughput is enhanced and the user's demand is reduced.

14.5 Conclusion

The idea of using BCIs to control robots that mimic humans is a relatively new one. To address the limitations of both low-level and high-level control, they provide hierarchical BCIs as a novel method for operating humanoid robots. This method enables the customization of humanoid control to suit the user's specific demands and surroundings on a schedule that suits them. As a proof-of-concept, this study's results from three participants show that:

- A hierarchical BCI allows for the multi-level teaching of new abilities to a humanoid robot;
- A substantial decrease in mental strain may be achieved by raising the command hierarchy by one level;
- An individual may complete complicated tasks by sending signals to their brain, provided that the user has acquired both basic motor instructions and more advanced abilities.

Currently, everyone concentrated on:

- Adding support for learning across several state spaces to the BCI design; for example, a user might learn both navigational instructions and hand/arm motions;
- Introducing pre-programmed robotic abilities like automatic grabbing or navigation to minimize the time required for user training and enable the user to concentrate on learning more advanced command sequences;
- Dealing with the noise and uncertainty that comes with brain-based robotic control using increasingly advanced algorithms for machine learning and probabilistic reasoning;

- Investigating the whole spectrum that constitutes human biological authority of humanoid robots by augmenting brain impulses with additional signals including eye movement, speech, and muscle-based instructions.

References

1. Bryan, M., *et al.*, An adaptive brain-computer interface for humanoid robot control. *2011 11th IEEE-RAS International Conference on Humanoid Robots*, IEEE, 2011.
2. Chamola, V., *et al.*, Brain-computer interface-based humanoid control: A review. *Sensors*, 20, 13, 3620, 2020.
3. Gandhi, V., *et al.*, EEG-based mobile robot control through an adaptive brain–robot interface. *IEEE Trans. Syst. Man Cybern.: Syst.*, 44, 9, 1278–1285, 2014.
4. Bell, C.J., *et al.*, Control of a humanoid robot by a noninvasive brain–computer interface in humans. *J. Neural Eng.*, 5, 2, 214, 2008.
5. Batula, A.M., *et al.*, Developing an optical brain-computer interface for humanoid robot control, in: *Foundations of Augmented Cognition: Neuroergonomics and Operational Neuroscience: 10th International Conference, AC 2016, Held as Part of HCI International 2016, Toronto, ON, Canada, July 17-22, 2016, Proceedings, Part I 10*, Springer International Publishing, Switzerland, 2016.
6. Stephygraph, L.R. and Arunkumar, N., Brain-actuated wireless mobile robot control through an adaptive human–machine interface, in: *Proceedings of the International Conference on Soft Computing Systems: ICSCS 2015, Volume 1*, Springer India, India, 2016.
7. Choi, B. and Jo, S., A low-cost EEG system-based hybrid brain-computer interface for humanoid robot navigation and recognition. *PLoS One*, 8, 9, e74583, 2013.
8. Rakshit, A., Konar, A., Nagar, A.K., A hybrid brain-computer interface for closed-loop position control of a robot arm. *IEEE/CAA J. Autom. Sin.*, 7, 5, 1344–1360, 2020.
9. Tang, J., *et al.*, Humanoid robot operation by a brain-computer interface. *2015 7th International Conference on Information Technology in Medicine and Education (ITME)*, IEEE, 2015.
10. Aznan, N.K.N., *et al.*, Using variable natural environment brain-computer interface stimuli for real-time humanoid robot navigation. *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, 2019.
11. Andreu-Perez, J., *et al.*, A self-adaptive online brain–machine interface of a humanoid robot through a general type-2 fuzzy inference system. *IEEE Trans. Fuzzy Syst.*, 26, 1, 101–116, 2016.

12. Escolano, C., Antelis, J.M., Minguez, J., A telepresence mobile robot controlled with a noninvasive brain–computer interface. *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, 42, 3, 793–804, 2011.
13. Hramov, A.E., Maksimenko, V.A., Pisarchik, A.N., Physical principles of brain–computer interfaces and their applications for rehabilitation, robotics and control of human brain states. *Phys. Rep.*, 918, 1–133, 2021.
14. Alimardani, M. and Hiraki, K., Passive brain-computer interfaces for enhanced human-robot interaction. *Front. Rob. AI*, 7, 125, 2020.
15. Chung, M., *et al.*, A hierarchical architecture for adaptive brain-computer interfacing. *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.
16. Cao, L., *et al.*, A brain-actuated robotic arm system using non-invasive hybrid brain–computer interface and shared control strategy. *J. Neural Eng.*, 18, 4, 046045, 2021.
17. Clanton, S.T., Brain-computer interface control of an anthropomorphic robotic arm, Carnegie Mellon University, 2011.
18. Prataksita, N., *et al.*, Brain-robot control interface: Development and application. *2014 IEEE International Symposium on Bioelectronics and Bioinformatics (IEEE ISBB 2014)*, IEEE, 2014.
19. Al Faiz, M.Z. and Al-Hamadani, A.A., Online brain-computer interface based five classes EEG to control the humanoid robotic hand. *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, IEEE, 2019.
20. Korovesis, N., *et al.*, Robot motion control via an EEG-based brain–computer interface by using neural networks and alpha brainwaves. *Electronics*, 8, 12, 1387, 2019.
21. Merel, J.S., A multi-agent control framework for co-adaptation in brain-computer interfaces. *Advances in Neural Information Processing Systems 26*, 2013.
22. Gergondet, P., *et al.*, Multitask humanoid control with a brain-computer interface: user experiment with HRP-2, in: *Experimental Robotics: The 13th International Symposium on Experimental Robotics*, Springer International Publishing, Heidelberg, 2013.
23. Lampe, T., *et al.*, A brain-computer interface for high-level remote control of an autonomous, reinforcement-learning-based robotic system for reaching and grasping. *Proceedings of the 19th International Conference on Intelligent User Interfaces*, 2014.
24. Vourvopoulos, A. and Liarokapis, F., Robot navigation using brain-computer interfaces. *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, IEEE, 2012.
25. Xu, Y., *et al.*, Shared control of a robotic arm using non-invasive brain–computer interface and computer vision guidance. *Rob. Auton. Syst.*, 115, 121–129, 2019.

26. Bhattacharyya, S., Shimoda, S., Hayashibe, M., A synergetic brain-machine interfacing paradigm for multi-DOF robot control. *IEEE Trans. Syst. Man Cybern.: Syst.*, 46, 7, 957–968, 2016.
27. Chae, Y., Jo, S., Jeong, J., Brain-actuated humanoid robot navigation control using asynchronous brain-computer interface. *2011 5th International IEEE/EMBS Conference on Neural Engineering*, IEEE, 2011.
28. Scherer, R., *et al.*, Non-invasive brain-computer interfaces: Enhanced gaming and robotic control, in: *Advances in Computational Intelligence: 11th International Work-Conference on Artificial Neural Networks, IWANN 2011, Torremolinos-Málaga, Spain, June 8-10, 2011, Proceedings, Part I 11*, Springer Berlin Heidelberg, Berlin, Germany, 2011.
29. Stankevich, L. and Sonkin, K., Human-robot interaction using brain-computer interface based on EEG signal decoding, in: *Interactive Collaborative Robotics: First International Conference, ICR 2016, Budapest, Hungary, August 24-26, 2016, Proceedings 1*, Springer International Publishing, Cham, Switzerland, 2016.
30. Galán, F., *et al.*, A brain-actuated wheelchair: asynchronous and non-invasive brain–computer interfaces for continuous control of robots. *Clin. Neurophysiol.*, 119, 9, 2159–2169, 2008.
31. Choudhary, N., Rathore, P.S., Kumar, D., Comparative Evaluation of Marker-Controlled Method and Gradient Distance Transformation through Watershed Segmentation. *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, pp. 1–6, Pune, India, 2024, doi: 10.1109/I2CT61223.2024.10543576.

15

Evaluation and Validation of Type 1 Diabetes Clinical Data by GAN

Robin Rohit Vincent[1]*, Senthilkumar Moorthy[2], F. Nisha[3] and Soumya[4]

[1]Department of Computer Science & Engineering, School of Computer Science & Engineering and Information Science, Presidency University, Itgalpur, Rajanakunte, Yelahanka, Bengaluru, India
[2]Department of Management Information Systems, A'Sharqiyah University, Ibra, Sultanate of Oman
[3]SOET-CSE, CMR University (Lakeside Campus), Bangalore, India
[4]SoCSE and ISE, Presidency University, Bangalore, India

Abstract

To supplement data collected from individual patients' continuous glucose monitors, this research presents a system that uses procreative inspiring network architecture to create simulated data groups. Enhance the overall efficacy of machine-learning-based prediction models by using these synthetic data sets. Two groups of people with type 1 diabetes mellitus were tested, and the results were found to be significantly different from one another. To begin, with the original data's statistical distributions, the selected approach may reproduce the inherent features of individual patients. The second contribution tests and compares several estimate models for the challenge of prediction night hypoglycemia occurrences in type 1 diabetic patient roles, highlighting the possibility of synthetic information to enhance the effectiveness of machine learning methods. Both the generative and analytical models' outcomes are promising, and they provide a standard for training future ML models using generative approaches.

**Corresponding author*: robinrohit@gmail.com; ORCID: https://orcid.org/0000-0003-1537-3902
Senthilkumar Moorthy: ORCID: 0000-0001-9680-406X
F. Nisha: ORCID: https://orcid.org/0000-0001-9538-3048
Soumya: ORCID: https://orcid.org/0009-0001-6374-1434

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (243–260) © 2025 Scrivener Publishing LLC

***Keywords*:** ML, GAN, type 1, glucose monitor, forecasting nocturnal hypoglycemia

15.1 Introduction

The creation of sophisticated systems in several application areas has been made possible by machine learning methods in the last few decades, making data collecting a major priority. One area that has seen the effects of this trend is the rapid expansion of data-based model use in apps designed to help with type 1 diabetic mellitus (T1DM) treatment. Modern technology has made it easier to collect data for type 1 diabetes clinical trials and commercial uses. Among the most noteworthy developments is the widespread use of continuous glucose monitors (CGMs), which can record blood glucose levels in real-time. However, owing to the wide range of causes and situations impacting type 1 diabetic patients, the majority of the data obtained is not openly accessible and is sometimes heterogeneous. This makes it difficult to get enough samples or information to satisfy the specified goals. Data on other primary disruptions, such as insulin dosages, meals, or physical activity, is usually recorded manually by patients, which makes this data susceptible to human mistakes and is thus not ideal for most applications produced in this sector. Data protection legislation presents yet another major obstacle; while some projects do release anonymized data for research purposes, collecting and processing such data is typically linked to costly and time-consuming clinical trials and bureaucratic red tape; the situation gets even more complicated when data exchanges between nations are involved [1–10].

This research hopes to make a difference by:

1. Suggest a strategy to increase the number of individuals who undergo continuous blood glucose monitoring.
2. Using a data-augmented technique to develop a classifier for detecting hypoglycemia throughout the night [11–15].

A deep learning model's assertion that it can produce accurate CGM systems has the first impact. The next influence is an analysis of the effectiveness of enhanced data-based and non-augmented nocturnal hypoglycemia prediction systems. Using clinical products and statistically based criteria, the produced time series will be assessed for their ability to mimic the inherent features of particular people with type 1 diabetes without directly replicating parameters. In response to the need to develop effective predictive models built on limited data resources, the synthetic data generator has two main uses: first, it can alleviate persistent data shortage, which occurs when larger samples are too expensive or too difficult to obtain; second, it can address time-related data shortage, which occurs when answers are needed early on in the data collection process. Although this idea may be used in any method that builds models using blood glucose, it has the potential to be beneficial in a wide variety of clinical circumstances. In the area of nocturnal hypoglycemia prediction, where training models are applied to tiny and highly skewed data sets, the nocturnal hypoglycemia classifier aims to show how synthetic data might be advantageous. When faced with difficult circumstances and limited CGM data, the updated algorithms strive to enhance prediction performance. The goal of developing these models is to better understand how to anticipate cases of hypoglycemia throughout the night. This system aims to prevent significant blood glucose drops in patients by providing forecasts, allowing them to act accordingly, such as by eating a small snack. The suggested method can also extract structures from the data, which can be used to anonymize samples and make the data sets available for sharing more freely since they wouldn't be subject to privacy regulations [16–24].

15.1.1 Modern Technology

Diabetes control has greatly improved with recent advancements in artificial intelligence (AI). The restrictions created by inadequate data sets in terms of both number and quality affect the majority of AI-based applications. This is mirrored in the modelling approaches that are more commonly used, which focus on the population level. As a result, there are limited examples of models that address the individual level, which limits the possibilities and slows down the growth in various fields where these methods are applied. To assess innovative therapies in silico and produce synthetic data, so-called virtual simulants were established to tackle the difficulty of data collecting and human trials. In the field of diabetes, in particular, there is a plethora of software that offers capabilities to create virtual patients, mostly to validate regulator algorithms. The FDA's approval of one T1D simulant for use in pre-clinical studies of some insulin therapies marked a significant improvement in these programs. To create a fresh blood glucose time series, these simulators adhere to certain mathematical functions that link carbohydrate ingestion and insulin delivery.

One major issue with these tools is that they do not adequately reflect actual samples since they react to mathematical functions, even if the formulae utilized are explicitly meant to mimic human metabolism. The reality gap issue is a well-known obstacle to deep learning's use of synthetic data. To address this data gap and sidestep the reality gap, a plethora of deep learning approaches have emerged. In image analysis, for instance, generative adversarial networks (GANs) are used to augment a limited dataset of photos with synthetic data, to use them in cataloguing issues down the road. The results show that this approach is successful. In a related scenario, the authors provide a GAN that can generate electrocardiograms (EKGs) that are realistic enough to use for training machine learning applications. Also, numerous studies employ modified GANs for picture creation to generate realistic models for data anonymization in the cardiology sector. With a focus on the area of artificial data for glucose forecast, this article presents a model that uses various data increase methods to achieve state-of-the-art results for affected role with type 2 diabetes mellitus over dissimilar time prospects by balancing data sets and refining the presentation of glucose forecast models.

15.1.2 Metabolic Syndrome

T1DM is a chronic disease that causes an insulin shortage by destroying pancreatic cells. A peptide hormone, insulin is essential for the control of blood glucose levels and the preservation of homeostasis by facilitating their delivery into cells and tissues. Patients with type 1 diabetes must take insulin injections orally to keep their blood glucose levels steady. Hypoglycemia (low blood sugar) and hyperglycemia (high blood sugar) are two potential outcomes of inadequate regulation of these levels. The most serious consequences of uncontrolled high blood sugar, or hyperglycemia, which is definite as glucose levels over 180 mg/dL, are diabetic ketoacidosis and hyperosmolar hyperglycemic condition, both of which may last for years. Neurological dysfunction in a hypoglycemic patient may range from relatively minor symptoms (dizziness, somnolence) to life-threatening coma and death. The severity of hypoglycemia determines the range of its consequences. Decreases in blood glucose levels below 70 mg/dL are referred to as level 1 hypoglycemia (L1 Hypo), whereas those below 54 mg/dL are denoted to as level 2 hypoglycemia (L2 Hypo). The time in range (TIR) of blood glucose levels is defined as the interval between 70 and 180 mg/dL.

15.2 Methodology

15.2.1 Gathering and Preparing Data

The two data sets that were intended to be used had distinct T1DM cohorts. The first group includes those who were part of the observational study. Ten persons with type 1 diabetes were followed for 12 weeks in a home-free environment, with blood glucose levels taken from continuous glucose monitors (CGMs). Patients having more than four hypoglycemia per week are included in the dataset as those prone to hypoglycemia. The other data set includes six adult T1DM patients who used a Medtronic Elite CGM sensor and an insulin pump for up to eight weeks to track their blood sugar levels. To get reliable data sets, several data preparation steps were used. To begin, there were missing CGM readings that were identified by an exploratory investigation. Both short (less than 1 hour) and extended periods contain missing data. In shorter durations, they are usually caused by a sudden signal loss, whereas in longer durations, they are associated with issues that need a longer reaction time to resolve, such as difficulties with the battery, the need to replace sensors, software issues, etc. Consequently, linear interpolation was used to fill in missing data between two samples that were separated by less than 1 hour. Next, to meet the requirements of this research, retrieve time series. This has been carried out for each patient separately, yielding a unique dataset for each patient. The data is organized into a 288-column matrix, with each column indicating glucose levels at 5-minute intervals.

15.2.2 Networks of Generative Adversaries

An artificial neural network approach called GAN with a convolutional-based construction was employed to produce the artificial information that was used to enhance the original data sets. When compared to recurrent neural networks, this GAN design has shown to be more effective in producing visual and temporal data. This neural network architecture merges two more compact networks that were developed systematically. Generators and discriminators are the names of these networks. The first one takes latent space, a distribution of random variables, and attempts to generate samples that look like actual data. It is possible to teach the discriminator to tell the difference between fake and genuine models. The GAN training approach incorporates a Minmax optimizer, which allows for the generation of more realistic samples by maximizing the discriminator's efforts to minimize error and the generator's efforts to increase it. Two separate neural networks, the discriminator and the generator, have different designs. This paper's methodology is a convolutional layer-based, one-dimensional adaption of a generalized artificial neural network (GAN) for picture synthesis and discriminator. Generator architecture makes use of normal distribution random noise as latent space. A thick input layer that may represent 50 different low-definition copies of the same sequence receives the latent space. An output sequence of 288 blood glucose readings representing a single day of a patient is generated from these 50 versions by sequential up-sampling and convolutional filtering. Figure 15.1 depicts this design.

An adaption of image classification models, the selected discriminator architecture has an output layer of a sole nerve cell by a sigmoidal instigation purpose, as well as a succession of convolution and leaky remedied linear unit's layers. Because of this, the discriminator can only return 1 for "real" and 0 for "synthetic" as output values. In Figure 15.2, the discriminator's structure. Due to the discriminator's training on the difference between real and synthetic samples within a given time frame, it can be labelled as "not real" sequences that do not fit within a given window; consequently, the generalized ANN is unable to generate original data rolling windows.

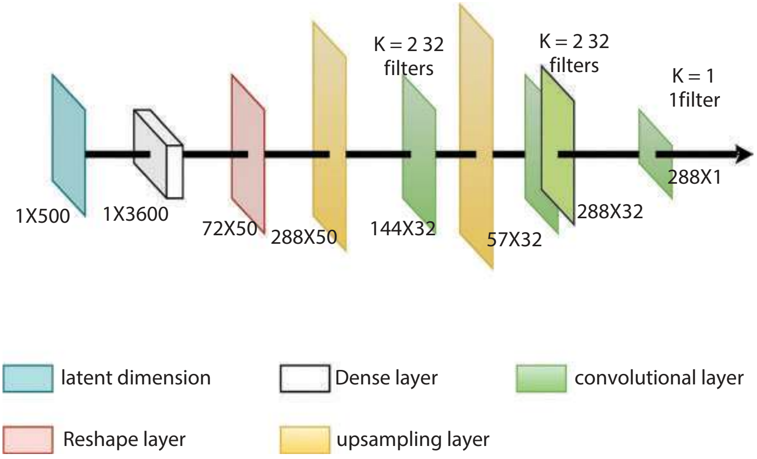


Figure 15.1 The construction of the generator is shown graphically.

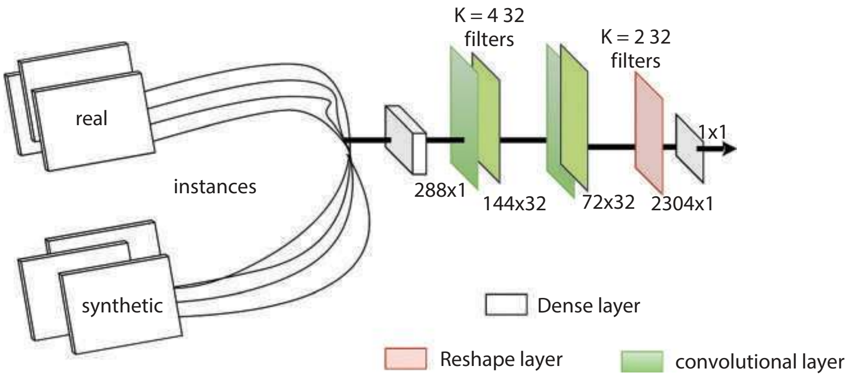


Figure 15.2 The discriminator architecture visualized.

15.2.3 Enhanced Nighttime Data-Based Sugar Lows Predictor

One practical use of augmented data is to fill in missing data points in incomplete or unbalanced datasets. This is accomplished by expanding the training set for a nocturnal hypoglycemia classifier using deep generative models. To evaluate the efficacy of estimate models trained on both the original and enhanced data sets the latter of which is an expanded version of the former that incorporates synthetically produced data a unci-dimensional convolutional classifier has been used. Like the discriminator in the GAN model, the architecture of the applied approach is shown in Figure 15.3. Classification issues and, more exactly, glucose prediction, have shown remarkable results for this sort of artificial neural network.

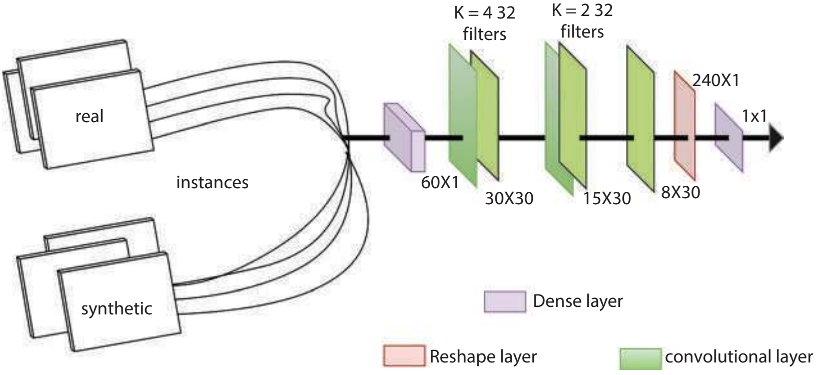


Figure 15.3 The nocturnal hypoglycaemia classifier is shown graphically.

Whether or not patients experience nocturnal hypoglycemia dictates the categorization of days into two distinct groups.

Classification issues, and glucose prediction, in particular, have shown exceptional performance from this kind of artificial neural network. Whether a patient has nocturnal hypoglycemia determines the sort of day they will have. Three repeated readings below 70 mg/dL from the continuous glucose monitor between 6:00 a.m. and 10:00 p.m. is considered the incidence of nocturnal hypoglycemia. Predictions are based on CGM readings taken during the last five hours, namely between five o'clock and ten o'clock in the evening, by the installed system. To train or assess the algorithm, the data acquired after 10:00 p.m. is utilized purely for labelling occurrences into nights with hypoglycemia and darks without hypoglycemia. Nocturnal hypoglycemia is usually the result of a buildup of decisions made throughout the day, and the hypoglycemia predictor takes into account the fact that a patient's blood glucose levels before bedtime significantly affect the likelihood of nocturnal hypoglycemia. Predicting whether a patient will have nocturnal hypoglycemia and giving them the chance to take precautions against it should be possible within the application's capabilities.

Figure 15.4 shows the Barcelona Data Set of Ten Patient's Total Incidences. These numbers reflect the total occurrences available for each patient. Both the frequency and duration of hypoglycemia episodes throughout the night are much lower in the Ohio dataset (Figure 15.5). The models have been evaluated using Stratified k-fold cross-validation. It should be mentioned that the testing data was only derived from non-synthetic sources.

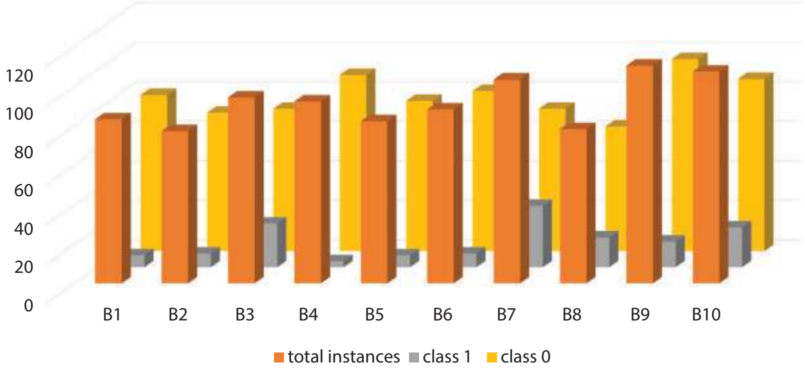


Figure 15.4 Barcelona data set's ten patients' total incidences. A sleep time without hypoglycemia is classified as class 0, whereas a sleep period with hypoglycemia is classified as class 1.

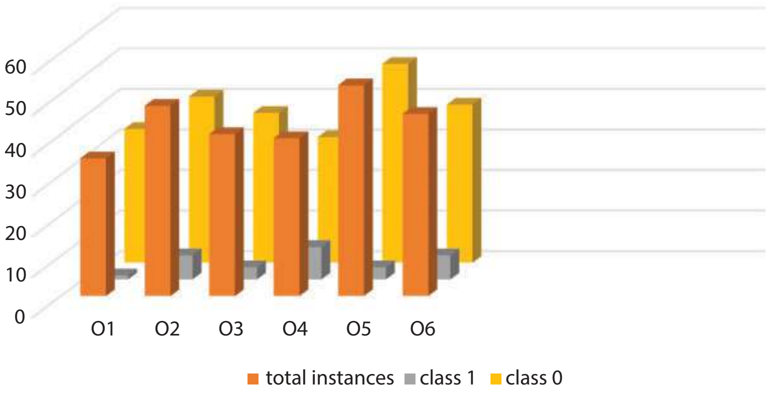


Figure 15.5 Total incidents for six patients from Ohio. Hypoglycemia is present in class 1 and absent in class 0.

15.3 Methods of Evaluation

Use several assessment criteria to evaluate the generative model and its consequences in glucose prediction cases. First, time-in-range measures were used to compare actual and synthetic data. The generated patients' clinical similarity to actual patients will be determined by these metrics, which are established clinical outcomes in diabetic patients. The generative models according to their output values are identical to those of the actual data sets used for training. If the p-value is more than 0.05, which indicates a statistical significance among the two samples, then the test may be considered a success in identifying differences between the matched groups. contrasted more significant statistical measures that provide useful information about distributions, such as mean blood glucose, Jensen-Shannon (JS) distance, variance, standard deviation (SD), and Z-test.

JS divergence determines how similar two probability distributions or time series are by using Kullback-Leibler divergence as an application. It differs from the Kullback-Leibler divergence in that its possible values are limited to the integers 0 and 1, with 0 denoting two identical distributions. Equation (15.1) was used to determine the square root of the Jensen-Shannon JS deviation, which is referred to as the JS distance.

$$\sqrt{\frac{D(P\,||\,M)+D(Q\,||\,M)}{2}} \tag{15.1}$$

In this case, M is defined as $M = \frac{1}{2}(P + Q)$

When comparison real-real, real-synthetic, and synthetic-synthetic samples, be sure to evaluate the means of the acquired values for numerical system of measurement that connect one distribution to another (Z-test, JS, variance). The projected values for each statistic may be shown globally in this way. When testing between synthetic and real-synthetic pairs, the expected result is 0.50, the same as it is when testing between real-real pairs when the JS distance is 0.50. The generative model also fails to reproduce the initial data, as seen by JS heat maps. For comparisons amongst themselves, they should have dark blue tones, but for all other comparisons, they should have green tones. In addition, compare the synthetic samples for each day with the real data set and utilize multiple ranges of values to test the GAN model. Several measures have been used to verify the nocturnal hypoglycemia classifier, including accuracy (ACC), sensitivity (SEN), specificity (SP), Matthew's correlation coefficient (MCC), and geometric mean (G mean), which is defined as the square of SEN times SP.

15.4 Results

The outcomes of the generative and classification models should be presented here. gather six data sets for each persistent in the Ohio cohort and ten groups for each enduring in the unique Barcelona cohort. Table 15.2 shows the results for the Ohio data set and Table 15.1 shows the results for the Barcelona data set when comparing artificial and actual patients separately. For every set of days, the metrics are shown, with the synthetic patients having many samples that correspond to their real-life counterparts' number of days. examine the changing ranges and average glucose values over time by comparison the actual and imitation data sets and scheming a p-value using the Wilcoxon test. Keep in mind that any distribution resulting in p-values less than 0.05 will be considered distinct by this test. The findings obtained for each patient are higher than this threshold, indicating that they are not different and are therefore legitimate.

Table 15.1 Results from Barcelona cohort patient actual and synthetic data comparison. Accept all p-values as they exceed 0.0

Patient		hyper	TIR	L1 hypo	L2 hypo	mean	JS	Variance	z-value	SD
B1	Real syn	48.80 37.36	45.21 54.02	3.74 5.67	2.26 2.95	187.20 164.73	0.23 0.21	-1.34 -1.37	0.40 0.59	74.67 58.59
B2	Real syn	28.79 43.57	61.94 50.10	5.01 3.59	4.26 2.74	147.95 180.08	0.19 0.18	-0.95 -1.56	0.54 072	47.80 56.69
B3	Real syn	35.08 38.67	56.69 54.74	5.24 4.05	2.99 2.54	158.78 171.32	0.21 0.18	-1.30 -1.74	0.51 0.77	56.76 54.35
B4	Real syn	44.72 38.02	47.80 54.13	4.07 4.65	3.40 3.20	175.15 170.66	0.22 020	-0.95 -1.25	0.38 0.54	69.61 66.39
B5	Real syn	39.52 45.13	54.94 51.54	4.29 2.35	1.24 0.98	163.54 184.42	0.19 0.18	-1.14 -1.42	0.46 0.60	55.67 59.73

(*Contin*

Table 15.1 Results from Barcelona cohort patient actual and synthetic data comparison. Accept all p-values as they exceed 0.0 (*Continued*)

Patient		hyper	TIR	L1 hypo	L2 hypo	mean	JS	Variance	z-value	SD
B6	Real	51.37	41.11	3.21	4.31	185.65	0.22	-1.29	0.66	59.43
	syn	37.81	49.83	5.19	7.18	163.71	0.22	-1.29	0.66	59.43
B7	Real	35.40	50.53	7.21	6.85	153.87	0.25	-1.30	0.46	63.71
	syn	27.70	59.22	6.55	6.53	146.27	0.22	-1.42	0.54	54.71
B8	Real	33.84	56.30	7.06	3.80	154.30	0.23	-1.26	0.46	61.44
	syn	36.86	56.75	4.16	2.23	168.57	0.20	-1.56	0.54	63.66
B9	Real	46.01	45.59	4.85	3.55	176.19	0.24	-1.27	0.42	69.30
	syn	33.94	54.01	5.47	6.58	157.40	0.22	-1.57	0.66	57.23
B10	Real	36.59	47.59	6.30	9.52	160.35	0.29	-1.08	0.29	81.94
	syn	31.48	50.62	6.40	11.50	151.59	0.26	-1.40	0.44	74.35

Table 15.2 Results from Ohio cohort patient actual and synthetic data comparison. Since all p-values above 0.05, they are acceptable.

Patient		Hyper	TIR	L1 hypo	L2 hypo	Mean	JS	Variance	Z-value	SD
O1	Real Syn.	39.27	56.92	2.74	1.07	167.12	0.21	−1.13	0.32	62.70
		32.91	61.88	4.57	0.64	163.45	0.20	−1.92	0.50	59.84
O2	Real Syn.	25.72	72.13	1.85	0.30	148.04	0.17	−1.14	0.33	43.13
		20.14	67.88	7.60	4.38	132.59	0.15	−1.90	0.85	34.86
O3	Real Syn.	60.30	38.60	1.03	0.07	195.18	0.14	−0.81	0.57	51.19
		50.99	47.41	1.29	0.31	196.22	0.13	−2.12	1.07	48.59
O4	Real Syn.	25.13	67.62	5.12	2.14	144.20	0.21	−1.32	0.33	53.73
		29.83	60.96	5.59	3.62	151.29	0.20	−1.49	0.54	56.82
O5	Real Syn.	38.61	60.66	0.56	0.17	167.99	0.15	−1.06	0.43	43.11
		54.98	44.10	0.78	0.13	195.53	0.14	−1.07	0.68	49.08
O6	Real Syn.	31.16	65.06	3.19	0.59	153.16	0.19	−1.16	0.38	50.15
		28.86	61.93	5.56	3.65	152.75	0.16	−1.43	0.88	41.71

Figure 15.6 includes the Barcelona heat maps with the JS assessment to show how the models perform while assessing the uniqueness of the synthetic data. Dark blue on the heat map specifies that the samples match when the JS value is 0, which is seen when comparing real-real samples. The second heat map in the picture to determine the degree of similarity between the produced samples. Since no dark blue values are seen outside the diagonal, deduce that there are no repeated samples. To demonstrate that the suggested approach does not faithfully reproduce the initial data, the final heat map displays a comparison of synthetic and real data. Regarding the question of whether all values are typically lesser than in the real-real contrast, no such comparisons with zero values are seen. Display these heat maps using models of patient number 1 of the Barcelona data as an example, even though they exhibit like patterns for all of the individual patients utilized in the study.

As a significant methodological restriction, the insufficiency of events renders the testing data unfit for a proper evaluation of the classifier, hence patient roles with less than five nocturnal hypoglycemia episodes (O1, B4, O5, and O3) were excluded from the predictive model training and evaluation processes. A stratified k-Fold cross authentication procedure was used to validate the replicas for the remaining patients. For each fold, separate data sets were used for exercise and testing. The goal was to find the sweet spot for the augmented data set's number of days by comparing the average metrics of models qualified with varying numbers of synthetic samples. the number of simulated days and data sets ranging from 500 to 5000 occurrences were used to train the prediction model 20 times for each patient. Figure 15.7 shows that the results recover as the quantity of samples increases, although the rise is sluggish for values more than 1500 synthetic models.

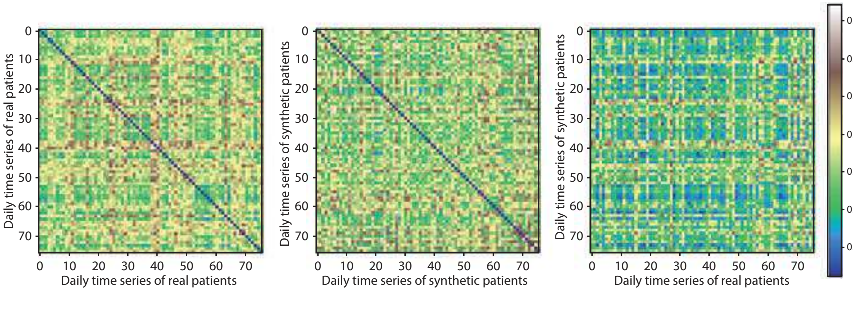


Figure 15.6 Barcelona patient distance heat maps.

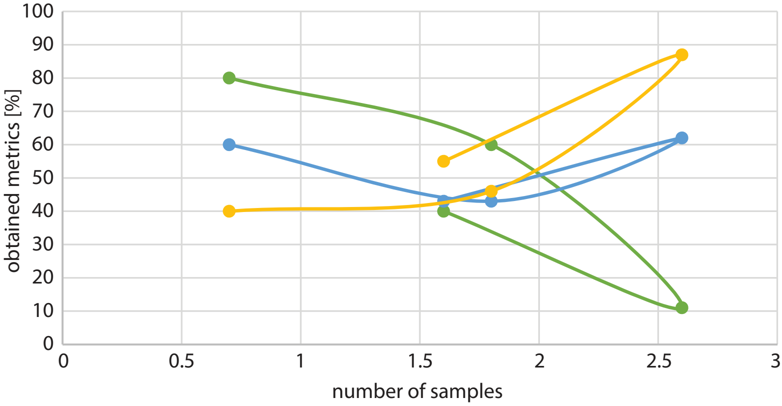


Figure 15.7 Displaying metrics linked to days utilized to enhance the actual data collection.

15.5 Discussion

Tables 15.1 and 15.2 show the findings, which show that the variables employed to quantitatively define each patient in each artificial data set correspond with their genuine counterparts. In addition to correlating with the other variables of mean, variance, JS, SD, and Z-Value, the acquired CGM values adhere to the similar temporal supplies of L1 Hypo, Hyper, TIR, and L2 Hypo as the unique data. These results demonstrate the viability of the suggested GAN design for producing a novel time series of blood glucose readings. The results for the various statistics demonstrate that the synthetic data sets are accurate depictions of the originals. According to the results of the many tests, which were measured in terms of range, mean, and other metrics, the GAN model successfully translated the inherent properties of the original into synthetic data. A step forward in closing the reality gap has been hinted at by this implementation, which is good news for the field of machine learning. Because the approach is based on a machine-designed mathematical function, the produced datasets may react more realistically to real-world actions. The use of GAN has the possible to become commonplace when designing machine learning applications due to this feature and the ability to indefinitely augment actual data sets with extremely modest computing costs. Machine learning systems may legitimately employ synthetic data to enhance limited and unbalanced datasets, as shown by the nocturnal hypoglycemia classifier that was qualified with together actual and increased data sets. This study's tests demonstrated that all patients who participated in the trial improved on many commonly used assessment metrics, including SP, SEN, ACC, MCC, and G mean, with a percentage improvement indicated for each patient and metric averaged out. The nocturnal hypoglycemia classifier, which was trained with together real and increased data sets, demonstrates that machine learning algorithms may legally use synthetic information to improve constrained and extreme data sets. Every single patient who took part in the experiment showed improvement on a battery of widely used assessment metrics, including SP, ACC, MCC, SEN, and G mean. The results were presented as an average percentage improvement for each patient and measure.

Since there is a significant data imbalance in the particular challenge of predicting nocturnal hypoglycemia, SP and SEN are the most important parameters to consider. High scores on the first one indicates a good prediction of nocturnal hypoglycemia and elevated results, while high scores on the second indicate good classification of nights when patients will not have hypoglycemia and won't encourage them to increase their blood glucose levels. An additional essential consideration is that, unlike the nocturnal categorization issue, this one use solely data on blood glucose levels and does not take into account insulin, carbohydrate consumption, or other significant disruptions like exercise. Regardless, the suggested classifier that makes use of enhanced data has done quite well for patients in both groups, despite this limitation.

15.6 Conclusion

When building machine learning models to prediction actual medical occurrences, the method outlined in this article establishes a standard for dealing with the issue of limited data. The ability of generative models to retain their essential properties while growing limited data sets has been shown. This study's findings suggest that data set augmentation might help investigators realistically increase machine learning models' prediction accuracy by allowing them to supplement tiny and uneven data sets. The suggested models may also be used to anonymize samples, which might lead to safer and more accessible data sets in light of data defense laws, as they can depict the unique glycemic response of each patient. Because mathematical models cannot detention the entire human physiology that impacts diabetic patients, this methodology, when applied with training signals of meals, insulin, or exercise, paves the way for more realistic simulators that could help reduce the reality gap.

References

1. Zhu, T., *et al.*, Blood glucose prediction for type 1 diabetes using generative adversarial networks. *CEUR Workshop Proceedings*, vol. 2675, 2020.
2. García-Domínguez, A., *et al.*, Optimizing Clinical Diabetes Diagnosis through Generative Adversarial Networks: Evaluation and Validation. *Diseases*, 11, 4, 134, 2023.
3. Yin, Y., *et al.*, Tetrathiomolybdate partially alleviates erectile dysfunction of type 1 diabetic rats through affecting ceruloplasmin/eNOS and inhibiting corporal fibrosis and systemic inflammation. *Sex. Med.*, 10, 1, 100455–100455, 2022.
4. Zhang, Y., Gan, P., Huo, Y., The efficacy of sodium glucose co-transport-2 inhibitors on glycemic control for patients with type 1 diabetes mellitus: A protocol for systematic review and meta-analysis. *Medicine*, 100, 27, 244, 2021.
5. Gan, C.Y.X., Transferring self-management responsibilities in type 1 diabetes families; initial assessment of individual and family self-management theory. *Diabetes Care Children Young People*, 8, 3, 1–6, 2019.
6. S. R. Swarna, A. Kumar, P. Dixit and T. V. M. Sairam, Parkinson's Disease Prediction using Adaptive Quantum Computing, *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, pp. 1396–1401, Tirunelveli, India, 2021, doi: 10.1109/ICICV50876.2021.9388628
7. Mujahid, O., *et al.*, Conditional synthesis of blood glucose profiles for T1D patients using deep generative models. *Mathematics*, 10, 20, 3741, 2022.
8. Yang, B., *et al.*, Identification of autophagy-related genes as potential biomarkers for type 1 diabetes mellitus. *Ann. Transl. Med.*, 10, 11, 2022.
9. Heise, T., *et al.*, A Glucose Clamp Trial Investigating the Biosimilarity of Gan & Lee Insulin Glargine Injection (Insulin Glargine 100 U/mL) with US and EU Lantus® Comparator Products in Patients with Type 1 Diabetes Mellitus.
10. Thomaidou, S., *et al.*, IFNɣ but not IFNα increases recognition of insulin defective ribosomal product-derived antigen to amplify islet autoimmunity. *Diabetologia*, 66, 11, 2075–2086, 2023.
11. Cichosz, S.L., Jensen, M.H., Hejlesen, O., Short-term prediction of future continuous glucose monitoring readings in type 1 diabetes: development and validation of a neural network regression model. *Int. J. Med. Inf.*, 151, 104472, 2021.
12. Gupta, P., *et al.*, Impact of incidence and progression of diabetic retinopathy on vision-specific functioning. *Ophthalmology*, 125, 9, 1401–1409, 2018.
13. Zhang, M., Flores, K.B., Tran, H.T., Deep learning and regression approaches to forecasting blood glucose levels for type 1 diabetes. *Biomed. Signal Process. Control*, 69, 102923, 2021.
14. Fan, W., *et al.*, Plasma-derived exosomal mRNA profiles associated with type 1 diabetes mellitus. *Front. Immunol.*, 13, 995610, 2022.

15. Gan, W.Z., *et al.*, Omics-based biomarkers in the diagnosis of diabetes. *J. Basic. Clin. Physiol. Pharmacol.*, 31, 2, 20190120, 2019.
16. Pratley, R.E., *et al.*, Effect of continuous glucose monitoring on hypoglycemia in older adults with type 1 diabetes: a randomized clinical trial. *Jama*, 323, 23, 2397–2406, 2020.
17. Yuan, H., *et al.*, BioBART: Pretraining and evaluation of a biomedical generative language model, arXiv preprint arXiv:2204.03905, 2022.
18. Mirasol, R., *et al.*, Self-Reported Hypoglycemia in Insulin-Treated Patients with Diabetes: Results from the Philippine Cohort of the International Operations Hypoglycemia Assessment Tool (IO HAT) Study. *J. ASEAN Fed. Endocr. Soc.*, 33, 1, 12, 2018.
19. Theilade, S., *et al.*, Arterial stiffness is associated with cardiovascular, renal, retinal, and autonomic disease in type 1 diabetes. *Diabetes Care*, 36, 3, 715–721, 2013.
20. Wang, Y., *et al.*, The m6A methylation profiles of immune cells in type 1 diabetes mellitus. *Front. Immunol.*, 13, 10307285, 2022.
21. S. R. Burri, A. Kumar, A. Baliyan and T. A. Kumar, Predictive Intelligence for Healthcare Outcomes: An AI Architecture Overview. *2023 2nd International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)*, pp. 1–6, Villupuram, India, 2023, doi: 10.1109/ICSTSN57873.2023.10151477
22. Bhargava, N., Rathore, P.S., Jha, M., Goswami, A., Adaptive Clustering Algorithm for Big Health Data Classification. *2023 3rd Asian Conference on Innovation in Technology (ASIANCON)*, Ravet IN, India, pp. 1–4, 2023, doi: 10.1109/ASIANCON58793.2023.10270544.
23. Datta, P., Kumar, A., Das, P., A Bibliometric Survey of Diabetic Retinopathy Research in the Last Decade. *2023 International Conference on Artificial Intelligence and Smart Communication (AISC)*, Greater Noida, India, pp. 146–150, 2023, doi: 10.1109/AISC56616.2023.10084973.
24. Datta, P., Das, P., Kumar, A., Techniques in Detecting Diabetic Retinopathy: A Review, in: *Contemporary Issues in Communication, Cloud and Big Data Analytics*, Lecture Notes in Networks and Systems, vol. 281, H.K.D. Sarma, V.E. Balas, B. Bhuyan, N. Dutta, (Eds.), Springer, Singapore, 2022, https://doi.org/10.1007/978-981-16-4244-9_34.

16

Exploring Neuromorphic Computing with Deep Learning: Unveiling Opportunities, Applications, and Overcoming Challenges

Yogesh Kumar Sharma[1]*, Smitha[2], Shaik Saddam Hussain[3] and Leena Arya[1]

[1]*Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram, Guntur District, A.P., India*
[2]*Muscat College, University of Stirling, Muscat, Oman*
[3]*VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, India*

Abstract

To create more intelligent systems that use less energy, researchers are looking at how deep learning and neuromorphic computing might work together. Although deep learning algorithms have shown to be very effective in several domains, the prohibitive computing costs linked to training them have limited their use. Neuromorphic computing, a new method that takes motivation from the biology and structure of the brain of an individual, offers promise by using practical artificial brain cells to do computations. By bringing together deep learning with intelligent devices that prioritize energy saving in their autonomous operations, this junction has the potential to pave the way for a genuinely ubiquitous AI. Neuromorphic hardware has several benefits over traditional digital computer designs, such as enormous data throughput, quicker processing speeds, reduced power consumption, more integration density, and analogue computing. This is why Neuromorphic hardware is a promising substitute for using deep learning models in practical settings. Reviewing neuromorphic computing using deep learning methodologies, this article discusses its potential, uses, and obstacles. We go over some of the potential benefits of neuromorphic computing technologies for the future of computing and the ways in which algorithms and apps built on these platforms might be improved.

**Corresponding author*: dr.sharmayogeshkumar@gmail.com

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (261–286) © 2025 Scrivener Publishing LLC

Keywords**:** Deep learning, neuromorphic algorithms, neuromorphic applications, energy, neuromorphic computing

16.1 Introduction

The computing world is actively seeking new technologies to provide ongoing performance advances as the conclusion of Dennard scaling and Moore's law draws near. Among these emerging forms of computing are neuromorphic computers. Initially used by Carver Mead in the late 1980s, the term "neuromorphic" now refers to a broader range of hardware implementations of brain-inspired computing systems, not limited to mixed analogue-digital deployments [1]. With advancements in the field and the emergence of large-scale financing possibilities for these technologies, such the "European Union's Human Brain Project" (EUHBP) and the DARPA Synapses endeavor, this has altered.

Computers that do not follow the von Neumann model but instead use synapses and neurons to simulate the way actual brains work are called neuromorphic computers. Programs and data are stored in memory sections of a Von Neumann computer, which are physically isolated from the CPU. However, synapses and neurons govern computation and memory in a neuromorphic computer [2]. Neuromorphic computers construct programs using the structure and characteristics of the neural network, as opposed to von Neumann computers, which depend on explicit instructions. Neuromorphic computers differ from von Neumann computers in that they utilize spikes as input rather than binary numbers. The length, magnitude, and shape of each spike may carry numerical information. The optimal method for transforming binary data into spikes and vice versa is still under investigation in Neuromorphic computing [3].

Cloud computing, which is primarily a result of the need to concentrate shared high-performance computer equipment at specific sites, is degrading the environment since it requires significant energy resources. Every internet search uses energy. Billions of worldwide searches waste energy and contribute to climate change [4]. Machine learning algorithm training is a common energy-hungry data center application. Data centers require 200 terawatt-hours of energy per year, which is anticipated to rise by orders of magnitude by 2030 if nothing is done to reduce their energy needs.

Researchers are trying to figure out ways to make computer system components consume less energy [5, 6]. The foundation of neuromorphic computing is the notion that the brain retains and analyses information simultaneously. This statement is in opposition to von Neumann's computer model, which distinguishes between data processing and storage. The primary factor behind the high energy consumption of most computer systems is this. The transfer of data among the CPU and memory consumes valuable time and energy, resulting in inefficiency. Neuromorphic computing focuses primarily on hardware advancements, but these developments are going to make a big impact on the foreseeable future of computing. This article provides a comprehensive analysis of the latest developments in Neuromorphic computing methods and how they can be used in practice.

Several compelling demonstrations have indicated potential neuromorphic computing platforms may overcome conventional von Neumann designs in certain computational tasks. Currently, there are several challenges that must be addressed in the development of neuromorphic systems. Proficiency in fields like as material biology, electronics, information technology, and neurology is crucial, given that the field is interdisciplinary. Given the significance of comprehending the brain's mechanisms for processing and storing information, as well as creating novel materials that can replicate both of these neural processes in computer and electronic components, this challenge offers promising prospects for individuals new to the field, particularly neuroscience researchers and supplies investigators. Neuromorphic computing has many important hurdles without a model hierarchy that may simplify and promote universality [7].

The achievement of classical computing can be linked to the Turing completion hypothesis of the von Neumann structure. The stacking hierarchy guarantees uniform program execution across different hardware platforms, irrespective of their individual characteristics. Currently, neuromorphic computing lacks a hierarchy of models. This allows neuromorphic hardware abstraction at a higher level. The literature extensively explores the advantages of neuromorphic computing and strategies for their implementation. The most attractive feature of neuromorphic computers for computation is their capacity to operate on significantly lower power compared to conventional computing systems. These systems operate with minimal power consumption because to their event-driven and substantially parallel nature, which ensures that only a small portion of the system is actively engaged at any given time. Energy efficiency is a compelling reason to consider neuromorphic computers as a possible solution to the increasing issue of rising energy expenses in computing and the prevalence of energy-limited activities including the network's edge [8].

Moreover, the neural network-based processing structure of several AI and ML programs makes them very compatible with neuromorphic systems. Once more, a wide range of computations may be performed on neuromorphic computers because of their inherent computing capabilities.

Recently, the focus in the creation of neuromorphic computers has been on these properties, which are all based on brain characteristics. However, it is still uncertain if these attributes are the only computationally important components of biological brains. While neurons as well as synapses have been selected as the primary computing components of neuromorphic computers, it is worth considering that glial cells, another kind of neurological component, may also play a role in computation. While there is ongoing discussion on the ideal level of abstraction for neuromorphic computers, there is data indicating the advantages of including neurons and synapses [9].

Researchers are now employing or proposing to employ several physical kinds of neuromorphic hardware, which differ from upcoming computer technology. Several extensive neuromorphic machines have been created to pursue different methodologies and objectives. SpiNNaker6 with BrainScale were funded by the EUHBP (European Union's Human Brain Project), which seeks to facilitate extensive neuroscience simulations. An improved digital neuromorphic processor enabling the use of rather more complicated neuron models is the "online learning digital spiking neuromorphic" (ODIN). A proposal was put forward [10].

The Tianjic chip is a neuromorphic platform that seeks to expand the range of calculations and applications. The software is compatible with both traditional artificial neural networks (ANNs) and neuromorphic spiking neural networks (NSNNs), allowing it to address a wide range of problems. Several academic programs, including DYNAPs, IFAT, BrainScales-2 and Neurogrid5,, are among the increasing number of commercial initiatives that are specifically focused on neuromorphic factors systems, including IBM's TrueNorth alongside Intel's Loihi. Using neuromorphic hardware like BrainScales-2 to optimise learning-to-learn scenarios for spiking neural networks at much faster timescales than biological timescales has been successful [11].

Large-scale neuromorphic computers use silicon and complementary metal oxide semiconductor technology, but the neuromorphic community is working hard to find new materials like ferroelectric, non-filamentary, topological insulators, and channel-doped biomembranes. Memristors, known for its resistive memory and capacity to integrate CPU and memory, are widely acknowledged in the literature as a significant approach for constructing neuromorphic computers. Nevertheless, alternative technologies such as optoelectronic devices have also been employed. Neuromorphic computers exhibit variations in their operating speeds, energy consumption, and physical appearance, which are contingent upon the specific device and material employed in their construction. The wide range of devices and materials used in constructing neuromorphic electronics nowadays allows for the customisation of specific properties required for different applications [12].

Most of the present research on neuromorphic computing primarily concentrates on the hardware, including systems, devices, and materials, as described previously. However, for the future optimisation of neuromorphic computers, leveraging their unique computational features, and shaping their hardware design, it is essential to combine these computers with neuromorphic algorithms and applications. From our perspective, we examine the current state of neuromorphic algorithms and applications and anticipate the future possibilities of neuromorphic computing in the fields of computer science and mathematical science. The term "neuromorphic computing" is now used to many types of developments. As previously discussed, it was determined that the original concept was limited to hybrid digital and analogue systems. In this study, we consider all hardware implementations, including digital, hybrid analogue-digital, and analogue, as neuromorphic, regardless of whether they employ spiking neural networks or not [13].

The literature extensively explores the advantages of neuromorphic computers and provides insights on their implementation. The most attractive feature of neuromorphic computers for computation is their capacity to operate on significantly lower power compared to conventional computing systems. These systems operate with minimal power consumption because to their event-driven and massively parallel nature, which ensures that only a small portion of the system is actively engaged at any given time. The energy efficiency of neuromorphic computers makes them a compelling option for addressing the increasing energy expenses of computing and the prevalence of energy-limited applications, particularly those located at the network's periphery. Furthermore, the present AI and ML applications are particularly compatible with neuromorphic computers because to their inherent utilisation of computation in the style of neural networks. Furthermore, neuromorphic computers have potential for a wide array of calculations due to their intrinsic computational characteristics [14].

Recently, the development of neuromorphic computers has primarily emphasized these qualities, which are derived from the brain. Nevertheless, it remains uncertain if they are the only crucial elements of biological brains for computing. Glial cells, a type of neurological component, offer potential value in computing, despite neurons and synapses being chosen as the primary processing units for neuromorphic computers. The question of whether neurons and synapses are the optimal level of abstraction for neuromorphic computers remains a subject of ongoing debate.

However, empirical evidence has demonstrated their use in this context. Unlike many upcoming computer technologies, researchers are now developing or using several physical forms of neuromorphic hardware. The creation of several large-scale neuromorphic computers has been driven by various approaches and objectives. The European Union's Human Brain Project financed the development of SpiNNaker6 and BrainScale to enable the execution of extensive neuroscience simulations. The online-learning digital spiking neuromorphic (ODIN) is an enhanced digital neuromorphic processor proposed for the use of somewhat more complex neuron models [15].

Several neuromorphic systems, like as the Tianjic chip, are striving to enhance the range of computations and their possible applications. Due to its compatibility with both classic artificial neural networks (ANNs) and newer neuromorphic spiking ANNs, it is capable of addressing a diverse set of challenges. In addition to commercial efforts like as TrueNorth by IBM and Loihi by Intel, several academic programs, including DYNAPs, Neurogrid5, IFAT, and BrainScales-2, have also started focusing on neuromorphic systems. Neuromorphic hardware such as BrainScales-2 has a useful use in optimising learning-to-learn situations for spiking neural networks. This involves using an optimisation method to define the learning process, but at a considerably quicker rate than what occurs in biological systems [16].

Although large-scale neuromorphic computers currently rely on silicon and conventional complementary metal oxide semiconductor technology, the neuromorphic community is actively researching alternative materials for neuromorphic implementations. These materials include ferroelectric, non-filamentary, topological insulators, and channel-doped biomembranes. While optoelectronic devices and other device types have been utilized, memristors, which encompass resistive memory and can integrate CPU and memory, have been widely acknowledged in the literature as a primary method for building neuromorphic computers. Neuromorphic computers can exhibit differences in their visual appearance, tactile sensation, and operational efficiency, depending on the specific device and materials utilised in its fabrication. Currently, the diverse assortment of devices and materials utilized in constructing neuromorphic circuits enables the tailoring of certain characteristics to suit different applications [17].

As previously said, the majority of current research in neuromorphic computing focuses on systems, devices, and materials. However, in order to fully use the unique computing capabilities and impact hardware design, neuromorphic algorithms and applications must be integrated with neuromorphic computers. From this perspective, we analyse the present condition of neuromorphic algorithms and their practical uses, while also making predictions about the future of neuromorphic computing in the fields of computer science and computational science. The phrase "neuromorphic computing" has been applied to several sorts of breakthroughs. As mentioned before, the previous debate concluded that the basic notion only involved combining digital and analogue systems to a certain extent [18].

In this study, we specifically examine spiking neuromorphic computers that utilise spiking neural networks. However, we consider all hardware implementations, including digital, hybrid analogue-digital, and analogue, as neuromorphic. The chapter is structured into distinct sections that encompass neuromorphic deep learning algorithms, applications, prospects, problems, and a conclusion.

16.2 Neuromorphic Deep Learning Algorithms

It is common practice to build spiking neural networks (SNNs) to be deployed on neuromorphic computers while programming them. In particular, SNNs' neurons and synapses incorporate ideas of time into computing, similar to how most Neuromorphic computers draw inspiration from biological brain networks. For instance, spiking neurons may have a gradual loss of energy over time, which is determined by a certain time constants. Additionally, SNN cells and synapses are possibly connected with a time delay, as indicated by reference [19].

The question of how to specify an SNN for a specific task is frequently addressed by neuromorphic implementation algorithms. The numerous algorithmic approaches to neuromorphic computing systems include both machine learning algorithms, which involve the deployment of preexisting SNNs to a neuromorphic computer, and non-machine learning algorithms, which involve the hand-construction of SNNs to solve specific tasks. In this context, "training" and "learning" techniques refer to the process of fine-tuning the parameters of an SNN, typically the synaptic weights, to address a specific issue [20]. This is of great importance.

16.2.1 Spiking Neural Networks

When contrasted with other artificial neural network variations, such as multilayer perceptrons, striking neural networks employ a more biological approach to modelling the behaviour of their neurons and synapses. The most distinguishing characteristic of SNNs is their capacity to incorporate

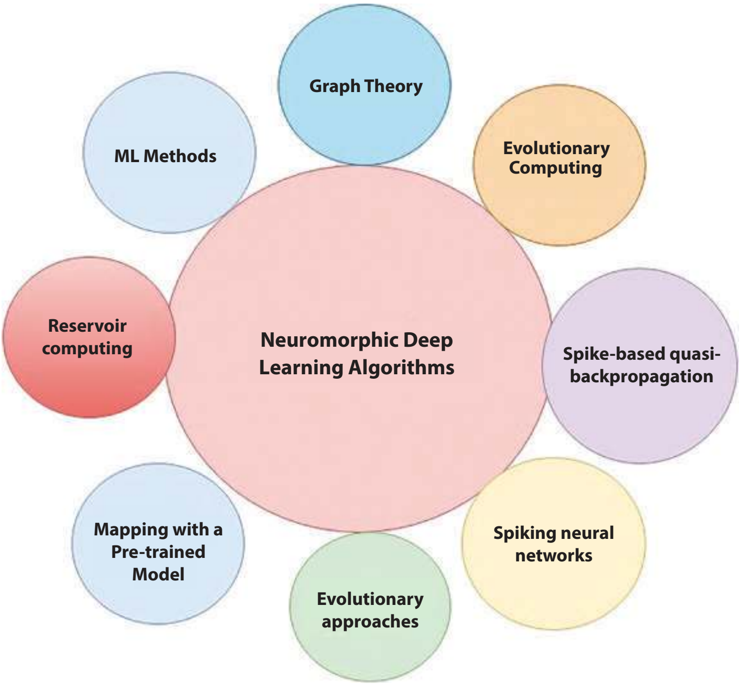


Figure 16.1 Neuromorphic deep learning algorithms.

time into their operations when contrasted with more conventional artificial neural networks. The neural network models employed in SNNs range from basic integrate and fire models, which involve the integration of the charge over time until a threshold value is achieved, to more sophisticated and realistic models, such as the Hodgkin-Huxley neuron model, which simulate the behaviour of specific components of real neurons, such as ion channels. The activity of the synapses and neurons in SNNs may be influenced by time components (Figure 16.1) [21].

As time progresses, neurons in spiking neural networks accumulate an increasing amount of charge from outside stimuli and internal conversations, with spikes delivered by adjacent neurons in the network being the most common source. Each neuron has a threshold value; when the neuron's charge level reaches this value, it activates and signals are transmitted throughout each of its synapses. In addition, neurons can impose a concept of leakage, in which the charge that has accumulated but is beneath the threshold progressively dissipates over time. Additionally, neurons may exhibit an axon postponement, which indicates that the information that is transmitted from the cell to its synapses requires a certain amount of time to reach them. Each synapse is composed of a pre-synaptic neuronal as well as a post-synaptic neuron, whose function is responsible for the connection of neurons [22].

Every summary carries some weight, which might be either encouraging or discouraging. Potentially, the time it takes for messages to go from pre-synaptic neurons to post-synaptic neurons is impacted by the associated synaptic delay value. One way synapses learn is by changing their weight value over time in reaction to what's happening in the network. The ability to activate and deactivate synapses in neuromorphic computers allows for the creation of a network architecture inside the fabric of connection. Furthermore, it is typically possible to program neuron and synaptic features, such as thresholds, synaptic weights, axonal delays, and synaptic delays, using neuromorphic designs [23].

In contrast to conventional artificial neural networks, SNNs may be executed on neuromorphic hardware that is event-driven or asynchronous, making them more suitable for the temporal dynamics of spiking neurons and synapses. Despite receiving input data simultaneously and having the network structured into layers, SNNs allow for information to travel asynchronously, meaning that it arrives at various times. Here we have a visual representation of an example SNN in action inside the domain of time. Synapses are displayed here with a delay. Spikes are the primary means of signal transmission across the network. By contrasting the network's operation at time t (on the left) and time t + 1 (on the right), the evolution of the state is demonstrated [24].

16.2.2 Spike-Based Quasi-Backpropagation

The effective deep learning methods of backpropagation and stochastic gradient descent don't work with spiking neurons since their activation functions remain unchanged. Spiking neurons often utilise a non-differentiable threshold function. Additionally, SNNs' temporal processing might complicate training and learning. SNNs may be less accurate than equivalent artificial neural networks due to the changes needed to apply deep learning methods with SNNs [25].

Adjusting layer weights and using a smoothed activation function to create error gradients and surrogate gradients affect deep learning training. Many presentations have showed near-state-of-the-art classification performance on MNIST's handwritten digit dataset using spike error gradient computing. Many approximations have been made to recurrent neural network training rules to leverage SNNs' intrinsic temporal component. The Spiking Speech Command (SSC) and Spiking Heidelberg Digits

(SHD) neuromorphic datasets employ real-time recurrent learning and time-based backpropagation [25].

16.2.3 Mapping with a Pretrained Model

Many attempts for implementing a neuromorphic approach to a problem begin by training a deep neural network (DNN) and then converting it to a spiking neural network (SNN) for inference using a mapping procedure. This is possible due to the well-established instruction process of DNNs. Applying these methods to well-known datasets such as MNIST, CIFAR-10, and ImageNet has led to impressive performance that rivals the current state-of-the-art. Additionally, these methods offer the potential for substantial energy savings compared to deep neural networks that rely on multiply and accumulate computations.

The majority of the first conversion methods either averaged the results instead of max-pooling or normalized the weights or activation levels. Alternatively, DNNs might be trained with certain restrictions so that their activation functions gradually begin to resemble those of spiking neurons [26].

The Few Spikes neuron model (FS-neuron), put out by Stockl and colleagues, allows SNNs to express complicated activation functions temporally with a maximum of two spikes. This leads to a novel mapping method. On benchmark picture classification datasets, they achieved accuracy comparable to that of deep neural networks with many fewer time steps per inference than previously revealed conversion algorithms. Some of the mapping methods stated above have been used in various applications that have been shown using neuromorphic hardware. Current technologies like IBM's TrueNorth and Intel's Loihi have been shown to be efficient for tasks including object identification, medical picture analysis, and keyword spotting [27].

It should be noted that when a traditional DNN is trained and subsequently mapped to neuromorphic hardware, particularly new hardware systems, the accuracy may suffer due to the transition from DNNs to SNNs and the neuromorphic hardware itself. Synaptic weight values realized by neuromorphic hardware systems using developing hardware devices like memristors are sometimes imprecise, and there may be cycle-to-cycle and device fluctuation as well. It is crucial to think about how these features could affect a mapped network's inference performance while developing a mapping approach. Another issue is that SNNs aren't often trained using methods that make use of deep learning techniques. This makes them unable to surpass the capabilities of more conventional artificial neural networks. When it comes to SNN computation, for instance, the temporal component is mainly ignored by the majority of gradient descent-style methods, including mapping techniques [28].

16.2.4 Reservoir Computing

Another well-liked technique for SNNs is reservoirs computing, commonly referred to as liquid state machines. In reservoir computing, a sparse recurrent SNN is defined. A liquid's definition is usually random if there is input reparability, which requires different inputs to yield separate outputs, and fading memory, which prevents signals from propagating through the reservoir indefinitely. The untrained liquid and a readout method, like a linear regression, which determines the reservoir's output, are both included in a reservoir computing solution. The fact that the SNN component of reservoir computing does not require training is a major advantage [29].

Utilizing the sparse and ongoing connections including synaptic delays in networking of spiking neurons, reservoir computing in SNNs projects the input into a higher-dimensional circumstances, both spatially and temporally. Several studies have demonstrated the benefits of using spike-based reservoir computing for processing signals with changing timestamps. This computing framework has seen a wide range of implementations, from basic reservoir networks used for bio-signal processing and prosthetic control applications to more intricate hierarchical layers of liquid state machines interconnected with supervised mode layers for applications related to video and audio signal processing [30].

16.2.5 Evolutionary Approaches

Another method that has been used is an evolutionary approach to SNN training and design. An initial population of possible solutions is generated at random in an evolutionary algorithm. When a population is assessed and given a score, it may be utilized for two purposes: selection, which involves choosing people with higher performance, and reproduction, which consists of generating new individuals via a mix of old ones and mutations. To find out how many neurons exist in the network and how they are linked with synapses, as well as other parameters like neuron thresholds and synaptic delays, evolutionary methods may be used for convolutional neural networks (SNNs) utilized for Neuromorphic computing.

These approaches are attractive because they are not specific to any network architecture (e.g., feed-forward or recurrent) and because differentiability in the activation functions is not required. Adjust the parameters and fine-tune the network design with these. In spite of their flexibility, evolutionary processes might not be as fast as other training approaches when it comes to convergence. The evolutionary methods have been especially effective in control-related applications, such as autonomous robot navigation and video games [31].

16.2.6 Non-Deep Learning Algorithms

Although Neuromorphic computers have traditionally been associated with machine learning, they have now been explored for application in non-ML techniques as well. The field of graph theory is one prominent source for algorithms that have been translated into Neuromorphic implementations. A directed graph is the building block of a Neuromorphic computer. Therefore, any graph of interest may be immediately inserted into one with suitable parameters, and the spike raster can show graph characteristics. These techniques are appealing because they are not limited to any particular network structure (e.g., feed-forward and recurrent), and differentiation in the process of activation functions doesn't seem necessary. Use these to fine-tune the network architecture settings. Despite their versatility, evolutionary processes may not be as rapid as other training methods when it comes to convergence. The evolutionary approaches have proven particularly useful in control-related programs, such as independent navigation for robots and game development [31].

Random walks have also been implemented in neuromorphic computers. A random walk starts with the random selection of a starting node and then moves an agent along an edge that branches out from that node. Over several iterations, the accidental agent's trips to numerous sites may provide critical information about the underlying network [32]. It is common practice to conduct many random walks and then combine the findings in order to do a random-walk analysis. While conventional hardware excels at parallel processing, aggregation and analysis often struggle and use much energy due to sequential operations, even when using GPUs or other parallel architectures [33].

Random walks in low-energy neuromorphic systems may be studied naturally in parallel, as Severa and colleagues demonstrated in certain circumstances. Smith and colleagues used energy-efficient time-scalable approaches to approximate solutions for particle transport and heat flow on complicated shapes using neuromorphic deployment of discrete-time Markov chains. Cook's key work on graphs, which are a subtype of relational structures, has found use in unsupervised learning tasks and cortical network learning, exhibiting compliance with neuromorphic hardware [34].

Several researches have shown that neuromorphic systems can perform similarly to other traditional techniques that employ CPUs and GPUs to tackle NP-complete problems in terms of time-to-solution and solution accuracy [35]. This shows that neuromorphic computing can also discover approximation solutions to NP-complete issues. Consider the quadratic unconstrained binary optimization (QUBO) issue [36], which Alom and colleagues sought to solve using the IBM TrueNorth Neurosynaptic Technology. Mniszewski used the IBM TrueNorth system to approximate a solution to the NP-complete graph partitioning problem, which he recast as the QUBO problem; in certain cases, neuromorphic solutions beat the answers provided by the D-Wave quantum computer. To approach a solution for the boolean satisfiability (SAT) issue [37]. Fonseca and Furber developed an application framework for addressing SpiNNaker-based NP-complete constraint SAT problems for broader graphical structures such as Bayes' nets, which are NP-complete for random variables with unconstrained probabilities, and used neuromorphic hardware to perform inference and sampling [38].

16.3 Neuromorphic vs. Deep Learning Algorithms

New algorithms, devices, and materials are being developed using Neuromorphic computing technology. On top of that, neuromorphic computing has started to show some promising outcomes in several emerging applications. The development and deployment of this technology still face several obstacles. There are several important distinctions between organic neurons and artificial neural networks based on Perceptrons. Accurately simulating the activity of real neurons is of utmost importance in neuromorphic computing [39]. Take biological neurons as an example; they are very dependent on time. Turning a neuron on or off does not last forever. In reality, what happens is an activation with a frequency attached to it. A neuron that has been triggered begins a brief synapse before entering a resting potential, where it awaits the arrival of fresh action potentials. Because of this property, a neuromorphic system's output will behave sinusoidally given a steady input [40].

Despite the complexity making them more challenging to manage, neuromorphic systems, when constructed on neuromorphic technology, have shown to be more energy efficient. Adaptive robotic arm control is one area where neuromorphic systems have shown great promise in terms of both performance and energy cost; nevertheless, these systems have not yet found widespread usage in the many domains where deep learning has demonstrated usefulness. As AI methods and software continue to advance, they are becoming more and more useful in scientific study, thereby expanding the AI revolution. A number of factors cause this quick uptake of AI technology. To begin, there has been significant development in the design and performance capabilities of neural models, particularly convolutional neural networks (CNNs). These models can do tasks on data that were before impossible and are now much beyond the capabilities of humans [41].

Second, these models can be easily retrained using application-specific data and are publicly available and shared. As an example, models that were initially trained on ImageNet [42] have been adapted for particular applications by being retrained on pictures that are relevant to those applications. This makes it easier for domain scientists to use these models without having extensive AI knowledge, which increases acceptance. Finally, CNNs have a robust software stack, which means that domain experts may run these models with little to no understanding of the hardware required to run them. Computer devices as diverse as laptops and supercomputers have all made use of the CNN software stack. The fourth point is that the necessary hardware to run these models has evolved fast and is now accessible to most people. The adoption of CNN-based approaches is further boosted by this hardware's availability [43].

Last but not least, there is a wealth of instructional resources available online, including tutorials, videos, blogs, and industry-sponsored training. The majority of this material is freely accessible at all times. Together, these reasons have accelerated the widespread use of artificial intelligence systems based on convolutional neural networks (CNNs) in the scientific community. Despite all these advantages, there are still some scientific fields that either don't have the resources to implement CNNs or would benefit significantly from a Neuromorphic computing approach, like Swap computing, which stands for weight, shallow size, width and power. Utilizing Spiking Neural Networks (SNNs) for event-driven computation, Neuromorphic computers are built on devices with non-von Neumann architecture [44].

16.4 Areas of High-Impact Studies

We present a variety of scientific disciplines that provide good potential for the research and application of Neuromorphic computing. This is not an exhaustive list, but rather an attempt to highlight some prospective sectors where Neuromorphic computing might have a significant societal

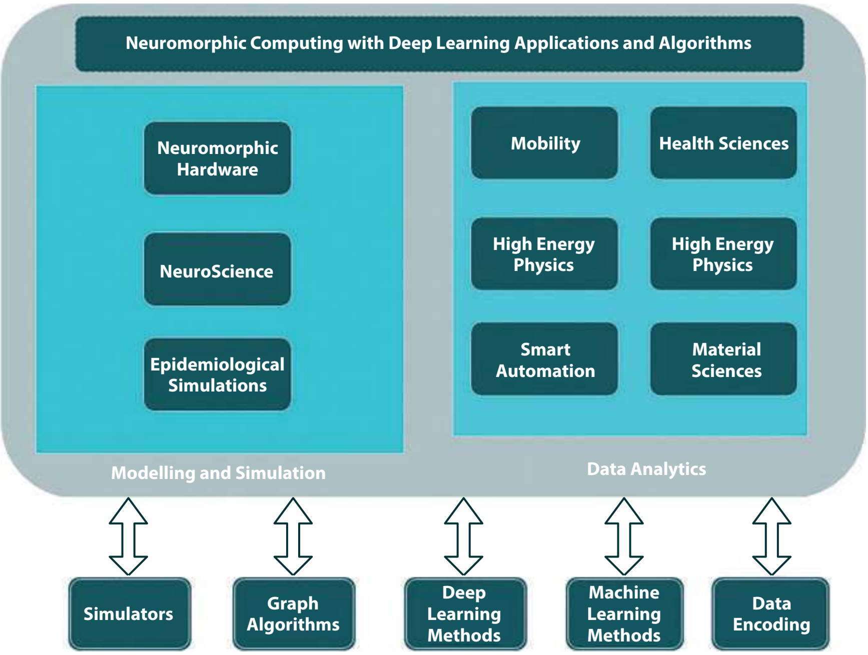


Figure 16.2 Neuromorphic applications and algorithms.

influence or an immediate effect (within the next year or two). There are various fields in which Neuromorphic computing presents little or no challenges. On the other hand, considerable challenges remain, but if they are conquered, society will profit enormously [45].

Figure 16.2 provides a visual summary of the whole project. "Modeling & Simulation" deals with physically reproducing a phenomenon for better understanding it, whereas "Data Analytics" processes data for categorizing or predicting physically occurring occurrences, primarily based on time information. In addition, we pinpoint a number of "Non-Neuromorphic" computer science domains (depicted at the figure's base) that can facilitate Neuromorphic computing advancement and reap its benefits (i.e., mutual co-development). The expansion of these technological subdivisions might create the path for neuroscience-based computing to spread to hitherto undiscovered scientific topics [46].

16.4.1 Neuromorphic Hardware

The performance and behavior of neuromorphic systems may be better understood via simulation, which in turn can drive architectural design choices for future iterations of these systems. Current attempts to mimic neuromorphic devices upon GPUs and central processing units suffer from a lack of scaling, size constraints on SNNs, and extremely long runtimes [47]. Accurately replicating neuromorphic systems requires a simulation engine that adheres to basic neuromorphic features such as co-located processing and memory, intrinsic scalability, and event-driven computing. Our working hypothesis is that the neuromorphic computers of today may serve as a perfect model for the neuromorphic computers of the future. These methods are already in use in high-performance computing, where they are used to predict how future HPC systems will perform [48].

16.4.2 Neuroscience

Neuromorphic computing was originally used in computational neuroscience modeling to understand brain regions [49]. Simulation of brain activity using abstract models of spiking neurons and synapses is now easy [50]. The GPU-enabled simulation framework GeNN and digital neuromorphic hardware SpiNNaker have simulated brain microcircuits with over 70,000 neurons [51]. SNNs may be used to explore neurogenesis in the dentate gyrus, episodic memories, and different neuroscience cognitive models [52]. This also opens more opportunities to construct new prediction models for other AI fields [53].

16.4.3 Epidemiological Simulations

With the use of adaptable, resilient, and efficient epidemiological models, we may successfully prepare for pandemics like COVID-19, Ebola, and swine flu. Local, state, and federal governments also use these epidemiological models to guide essential choices. Many holes in our epidemiological modelling methods have been revealed in the last few years by the COVID-19 pandemic. They have affected policy, revealing inadequacies in healthcare, health research, and logistics management. Neuromorphic computing can offer flexible, robust, the power, environment, and efficient in time the epidemiological simulations.

Upadhyay *et al.* [53] discussed that a spiking successful in the Susceptible-Infected-Recovered (SIR) model, a popular epidemiological model [54]. SNN spike-threshold-fire processes and advanced network contagion models' individual-scale dynamics are similar. Further research along this line will allow us to undertake epidemiological simulations with larger populations and more contagion properties. This may help governments make crucial decisions during worldwide pandemics.

16.4.3.1 Mobility

The electrification and autonomy of mobility have lately seen fresh development, and this growth is expected to continue for decades to come. The importance of 5G cellular connectivity to mobility is also expected to grow. At the moment, convolutional neural network (CNN) methods are all the rage, especially for in-car and fixed-location video sensors like traffic signals. However, at the moment, vehicle movement is the primary user of cars' limited electrical capacity.

Consequently, a power-hungry computer environment cannot be accommodated by this electrical capability. In addition, a plethora of sensors, including cameras, lidar, radar, and cellular communications, will likely be essential to autonomous navigation, with individual sensors necessitating processing, analysis, and fusion. Therefore, using a CNN-based technique would require gadgets that use much power. Neuromorphic computing's very low SWaP is well-suited to this application domain [55].

16.4.4 High-Energy Physics

Innovations in detector and sensing technology have been a boon to high-energy physics testing methods, which in turn have helped with the development of relevant theories. Additionally, machine learning methods are being used more and more to handle the massive amounts of data produced by these trials effectively [56]. The capacity to handle data with very low latency and power efficiency in a noisy environment is a significant limitation of these data processing units.

A newly exhibited "Application Specific Integrated Chip" or ASIC utilizes 95 mW of power, contains over 2,000 configurable settings, and compresses data on the HL-LHC front-end programming detector using an artificially intelligent automatic encoder. For neutrinos bouncing studies, deep-learning CNN techniques identify the component interaction point [57]. EONS-trained SNNs, with 90 neurons and 86 synapses, were utilized to classify neutrino particle segments utilizing memristive Neuromorphic prosthetic equipment with a limit of 2μJ per prediction.

16.4.5 Power Electronics

Artificial neural networks (ANN) with recurrent neural networks (RNNs) [58] were utilized to simulate the electrically active dynamics of control systems used for power electronics. SNNs have been demonstrated to be effective in predicting the current condition of solar thermal power plant controllers, enabling the detection of irregularities and anticipation of future states [24]. Further investigation is necessary to ascertain the most effective use of SNNs for tasks such as regression, however, as their efficiency in the present investigation was shown to be worse when compared to other AI models such as decision tree approaches and “non-linear auto-regressive exogenous” model also know as NARXs [59]. There is a great opportunity to use the inherent temporal characteristics of SNNs in the field of power electronics, where they are well-suited for modeling dynamic systems.

16.4.6 Health Sciences

Many convolutional neural networks (CNN)-based methods in the medical field are geared toward image processing in fields like radiology and pathology [60]. Both the lack of a need for real-time processing and a stationary computer environment make this an ideal scenario for the strategies mentioned above. On top of that, you may train and build CNN architectures using any number of publicly accessible data sets. Nevertheless, there are still openings in the medical field where neuromorphic devices’ portability, low power consumption, and real-time computing capabilities might lead to better patient care. One potential application of neuromorphic devices is to monitor a patient’s vital signs via a network of sensors in order to spot patterns or trends that may otherwise go unnoticed. A smartwatch, a wristband, or even an embedded gadget in a hospital bed may all house such a device. Unfortunately, there aren’t enough data sets or simulators to train and build SNNs, which is a significant obstacle to this use.

16.4.7 Smart Automation

The proliferation and development of the IoT devices offers significant prospects for the automation of smart homes and the advancement of intelligent manufacturing, resulting in improved energy and time utilization and consequent cost savings. The typical focus of most CNN-based systems is on cameras that may be mounted to monitor people or machines in operation. Time series prediction is achieved by the utilization of certain multilayer LSTM approach. The potential of neuromorphic computing to integrate sensors and detect or predict events is immense. However, there continues to stay much scope for enhancement in this domain. Placing devices in close proximity to the sensor sites can enhance the advantages of Neuromorphic computing minimal SWaP (size, weight, and power), hence increasing the potential for saving time and energy. Moreover, SNNs trained and improved by evolutionary optimization [61] are particularly well-suited for learning by reinforcement and control tasks.

16.5 Challenges and Opportunities

The computational fields of neuroscience and epidemiology are ideal for the modelling and simulation capabilities offered by Neuromorphic computers. As far as computers are concerned, Neuromorphic processors are the wave of the future when it comes to low-power machine-learning accelerators. The majority of scientific applications need HPC resources due to the massive volumes of data they analyze.

The widespread use and progress of IoT devices present substantial opportunities for automating smart homes and advancing intelligent manufacturing. This leads to enhanced efficiency in energy and time usage, resulting in cost savings. CNN-based systems often prioritize cameras that may be installed to monitor individuals or machinery throughout their operation. Time series prediction is accomplished by the deployment of a specific multilayer Long Short-Term Memory (LSTM) method.

The capacity of Neuromorphic computing to include sensors and identify or forecast occurrences is vast. Nevertheless, there is still much room for improvement in this field. By positioning devices near the sensor locations, the benefits of Neuromorphic computing's low SWaP (size, weight, and power) may be maximized, resulting in greater potential for time and energy savings. In addition, SNNs that have been trained and enhanced via evolutionary optimization [62] are especially suitable for acquiring knowledge through reinforcement and control tasks.

Numerous pieces of Neuromorphic hardware, such as digital and mixed-signal processors, have been shown so far [63]. The DYNAPs, BrainScales one, BrainScales two and all the bundled signal Neuromorphic devices all use analog circuits to simulate neurons and synapses [64]. Analog hardware systems are less power-hungry and can accommodate different biological time constants via device and circuit dynamics, but they have issues with device mismatches [4].

Inherently scalable and event-driven, Neuromorphic computers are the way of the future. In addition, they outperform CPUs and GPUs on machine learning tasks in terms of power consumption by an order of magnitude, all while maintaining computing speed [65]. These features make them ideal for real-time, low-latency signal processing, which is essential in event-based sensing applications [66].

Neuromorphic computing is projected to provide the most significant advantages to applications that need optimum size, weight parameters, and power consumed. Examples of such applications include autonomous systems like self-driving cars, drones, and automated guided vehicles, as well as embedded systems like control circuits, signal processing, and power electronics. The Internet of Things, which involves intelligent automation, and remote sensing, particularly in the field of high energy physics, are also notable examples. Neuromorphic computers can provide valuable modeling and simulation capabilities that may be advantageous for the computational domains of neurology and epidemiology.

16.6 Conclusion

When faced with problems in data analysis or using function approximates in simulation systems, many scientific areas swiftly embraced and used deep learning, also known as convolutional neural networks. However, applications requiring little space, weight, and power aren't a good fit for convolutional neural networks. Furthermore, compared to similar SNNs, CNNs have inference times that are at least ten times longer. Compared to CNNs, SNNs have a steeper learning curve and fewer developmental resources.

Regardless of these obstacles, Neuromorphic computing has great potential for growth in scientific domains where CNNs are either not applicable or perform inadequately compared to support vector machines. Neuromorphic computing has the potential to revolutionize several scientific fields, as we highlight in our work. Furthermore, we went over the software and hardware improvements that Neuromorphic computing needs to reach its full potential. In order to provide the groundwork for neuromorphic computing and increase its usage in scientific contexts, it is necessary to create a developmental ecosystem that is receptive to new ideas and geared toward neuromorphic computing. This will lower the entrance barrier and boost acceptance of the technology.

References

1. Li, R., Gong, Y., Huang, H., Zhou, Y., Mao, S., Wei, Z., Zhang, Z., Photonics for Neuromorphic Computing: Fundamentals, Devices, and Opportunities. *Adv. Mater.*, 37, 2, 2312825, 2025.

2. Ajani, S.N., Khobragade, P., Dhone, M., Ganguly, B., Shelke, N., Parati, N., Advancements in Computing: Emerging Trends in Computational Science with Next-Generation Computing. *Int. J. Intell. Syst. Appl. Eng.*, *12*, 7s, 546–559, 2024.
3. Sarkar, K., Shiuly, A., Dhal, K.G., Revolutionizing concrete analysis: An in-depth survey of AI-powered insights with image-centric approaches on comprehensive quality control, advanced crack detection and concrete property exploration. *Constr. Build. Mater.*, *411*, 134212, 2024.
4. Pradeep, S. and Sharma, Y.K., A pragmatic evaluation of stress and performance testing technologies for web based applications, in: *2019 Amity International Conference on Artificial Intelligence (AICAI)*, 2019, February, IEEE, pp. 399–403.
5. Daidone, M., Ferrantelli, S., Tuttolomondo, A., Machine learning applications in stroke medicine: Advancements, challenges, and future prospectives. *Neural Regener. Res.*, *19*, 4, 769–773, 2024.
6. Alqahtani, H. and Kumar, G., Machine learning for enhancing transportation security: A comprehensive analysis of electric and flying vehicle systems. *Eng. Appl. Artif. Intell.*, *129*, 107667, 2024.
7. Zhang, P., Wang, C., Lam, E.Y., Neuromorphic imaging and classification with graph learning. *Neurocomputing*, *565*, 127010, 2024.
8. Sun, H., Wang, H., Dong, S., Dai, S., Li, X., Zhang, X., Deng, L., Liu, K., Liu, F., Tan, H., Xue, K., Optoelectronic Synapses Based on Triple Cation Perovskite and Al/MoO3 interface for Neuromorphic Information Processing. *Nanoscale Adv.*, 6, 559–569, 2024.
9. Sharma, Y.K. and Rokade, M.D., Deep and machine learning approaches for anomaly-based intrusion detection of imbalanced network traffic. *IOSR J. Eng.*, 3, 1, 63–67, 2019.
10. Himmi, S., Parret, V., Chhatkuli, A., Van Gool, L., MS-EVS: Multispectral Event-Based Vision for Deep Learning Based Face Detection, in: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 616–625, 2024.
11. Wang, W., Kim, N.Y., Lee, D., Yin, F., Niu, H., Ganbold, E., Park, J.W., Shin, Y.K., Li, Y., Kim, E.S., Operant conditioning reflex implementation in a transparent Ta2O5–3x/Ta2O5– x homo-structured optoelectronic memristor for neuromorphic computing application. *Nano Energy*, *119*, 109102, 2024.
12. Bihl, T., Farr, P., Di Caterina, G., Vicente-Sola, A., Manna, D., Kirkland, P., Liu, J., Combs, K., Exploring spiking neural networks (SNN) for low Size, Weight, and Power (SWaP) benefits, 2024.
13. Luo, T., Wong, W.F., Goh, R.S.M., Do, A.T., Chen, Z., Li, H., Jiang, W., Yau, W., Achieving Green AI with Energy-Efficient Deep Learning Using Neuromorphic Computing. *Commun. ACM*, *66*, 7, 52–57, 2023.
14. Rathi, N., Chakraborty, I., Kosta, A., Sengupta, A., Ankit, A., Panda, P., Roy, K., Exploring neuromorphic computing based on spiking neural networks: Algorithms to hardware. *ACM Comput. Surv.*, *55*, 12, 1–49, 2023.

15. Mehonic, A. and Eshraghian, J., Brains and bytes: Trends in neuromorphic technology. *APL Mach. Learn.*, *1*, 2, 265, 2023.
16. Ahmed, L.J., Dhanasekar, S., Sagayam, K.M., Vijh, S., Tyagi, V., Singh, M., Norta, A., Introduction to Neuromorphic Computing Systems, in: *Neuromorphic Computing Systems for Industry 4.0*, pp. 1–29, IGI Global, Norta, 2023.
17. Ottati, F., Gao, C., Chen, Q., Brignone, G., Casu, M.R., Eshraghian, J.K., Lavagno, L., To spike or not to spike: A digital hardware perspective on deep learning acceleration. *IEEE J. Emerging Sel. Top. Circuits Syst.*, 13, 1015–102, 2023.
18. Singh, K., Kumar, A., Sharma, Y.K., Rai, A.K., AIoT-based e-commerce, in: *AIoT Technologies and Applications for Smart Environments*, p. 215, 2023.
19. Eshraghian, J.K., Ward, M., Neftci, E.O., Wang, X., Lenz, G., Dwivedi, G., Bennamoun, M., Jeong, D.S., Lu, W.D., Training spiking neural networks using lessons from deep learning. *Proc. IEEE*, 111, 1016–1054, 2023.
20. Sharma, Y.K., Athithan, S., Sachi, S., Singh, A.K., Jain, A., Devi, S., Copy and Move Forged Image Detection by Deep Learning, in: *2023 World Conference on Communication & Computing (WCONF)*, 2023, July, IEEE, pp. 1–6.
21. Yildirim, B., Razmi, P., Fathollahi, A., Gheisarnejad, M., Khooban, M.H., Neuromorphic deep learning frequency regulation in stand-alone microgrids. *Appl. Soft Comput.*, *144*, 110418, 2023.
22. Zins, N., Zhang, Y., Yu, C., An, H., Neuromorphic computing: A path to artificial intelligence through emulating human brains, in: *Frontiers of Quality Electronic Design (QED) AI, IoT and Hardware Security*, pp. 259–296, Springer International Publishing, Cham, 2023.
23. Mehonic, A., Sebastian, A., Rajendran, B., Simeone, O., Vasilaki, E., Kenyon, A.J., Memristors—From in-memory computing, deep learning acceleration, and spiking neural networks to the future of neuromorphic and bio-inspired computing. *Adv. Intell. Syst.*, *2*, 11, 2000085, 2020.
24. Marković, D., Mizrahi, A., Querlioz, D., Grollier, J., Physics for neuromorphic computing. *Nat. Rev. Phys.*, *2*, 9, 499–510, 2020.
25. Dabos, G., Mourgias-Alexandris, G., Totovic, A., Kirtas, M., Passalis, N., Tefas, A., Pleros, N., End-to-end deep learning with neuromorphic photonics, in: *Integrated Optics: Devices, Materials, and Technologies XXV*, vol. 11689, pp. 56–66, SPIE, 2021, March.
26. Woźniak, S., Pantazi, A., Bohnstingl, T., Eleftheriou, E., Deep learning incorporating biologically inspired neural dynamics and in-memory computing. *Nat. Mach. Intell.*, *2*, 6, 325–336, 2020.
27. Dahiya, N., Sharma, Y.K., Rani, U., Hussain, S., Nabilal, K.V., Mohan, A., Nuristani, N., Hyper-parameter tuned deep learning approach for effective human monkeypox disease detection. *Sci. Rep.*, 13, 1, 15930, 2023.
28. Shastri, B.J., Tait, A.N., Ferreira de Lima, T., Pernice, W.H., Bhaskaran, H., Wright, C.D., Prucnal, P.R., Photonics for artificial intelligence and neuromorphic computing. *Nat. Photonics*, *15*, 2, 102–114, 2021.

29. Pradeep, S., Sharma, Y.K., Lilhore, U.K., Simaiya, S., Kumar, A., Ahuja, S., Chakrabarti, T., Developing an SDN security model (EnsureS) based on lightweight service path validation with batch hashing and tag verification. *Sci. Rep.*, 13, 1, 17381, 2023.
30. Deng, L., Tang, H., Roy, K., Understanding and bridging the gap between neuromorphic computing and machine learning. *Front. Comput. Neurosci.*, *15*, 665662, 2021.
31. Nayak, S. and Sharma, Y.K., A modified Bayesian boosting algorithm with weight-guided optimal feature selection for sentiment analysis. *Decis. Anal. J.*, 8, 100289, 2023.
32. Kumar, N., Verma, H., Sharma, Y.K., Adversarial Attacks on Graph Neural Network: Techniques and Countermeasures, in: *Concepts and Techniques of Graph Neural Networks*, pp. 58–73, IGI Global, India, 2023.
33. Choi, S., Yang, J., Wang, G., Emerging memristive artificial synapses and neurons for energy-efficient neuromorphic computing. *Adv. Mater.*, *32*, 51, 2004659, 2020.
34. Venkataramani, S., Ranjan, A., Roy, K., Raghunathan, A., AxNN: Energy-efficient neuromorphic systems using approximate computing, in: *Proceedings of the 2014 International Symposium on Low Power Electronics and Design*, 2014, August, pp. 27–32.
35. Kumar, N., Verma, H., Sharma, Y.K., Graph Convolutional Neural Networks for Link Prediction in Social Networks, in: *Concepts and Techniques of Graph Neural Networks*, pp. 86–107, IGI Global, Pilani, India, 2023.
36. Sachi, S., Singh, A.K., Jain, A., Devi, S., Sharma, Y.K., Athithan, S., Hate Speech Detection Using the GPT-2 and Natural Language Processing, in: *2023 Intelligent Methods, Systems, and Applications (IMSA)*, 2023, July, IEEE, pp. 276–280.
37. Sharma, Y.K. and Khatal Sunil, S., Health Care Patient Monitoring using IoT and Machine Learning, in: *IOSR Journal of Engineering (IOSR JEN) National Conference on "Recent Innovations in Engineering and Technology" MOMENTUM-19*, 2019.
38. Balyan, A.K., Ahuja, S., Sharma, S.K., Lilhore, U.K., Machine learning-based intrusion detection system for healthcare data, in: *2022 IEEE VLSI Device Circuit and System (VLSI DCS)*, 2022, February, IEEE, pp. 290–294.
39. Swapna, M., Sharma, Y.K., Prasad, B.M.G., A survey on face recognition using convolutional neural network, in: *Data Engineering and Communication Technology: Proceedings of 3rd ICDECT-2K19*, pp. 649–661, Springer Singapore, Singapore, 2020.
40. Rokade, M.D. and Sharma, Y.K., MLIDS: A machine learning approach for intrusion detection for real-time network dataset, in: *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)*, 2021, March, IEEE, pp. 533–536.

41. Jadhav, D., Sharma, Y.K., Arora, D.P., Profound Learning Approach for Shot Boundary Location, in: *2nd International Conference on Communication & Information Processing (ICCIP)*, 2020, April.
42. Jadhav, M., Kumar Sharma, Y., Bhandari, G.M., Currency identification and forged banknote detection using deep learning, in: *2019 International Conference on Innovative Trends and Advances in Engineering and Technology (ICITAET)*, 2019, December, IEEE, pp. 178–183.
43. Guleria, K., Sharma, A., Lilhore, U.K., Prasad, D., Breast cancer prediction and classification using supervised learning techniques. *J. Comput. Theor. Nanosci.*, 17, 6, 2519–2522, 2020.
44. Lilhore, U.K., Simaiya, S., Algarni, A.D., Elmannai, H., Hamdi, M., Hybrid model for detection of cervical cancer using causal analysis and machine learning techniques. *Comput. Math. Methods Med.*, 2022, 275, 2022.
45. Lilhore, U.K., Simaiya, S., Pandey, H., Gautam, V., Garg, A., Ghosh, P., Breast cancer detection in the IoT cloud-based healthcare environment using fuzzy cluster segmentation and SVM classifier, in: *Ambient Communications and Computer Systems: Proceedings of RACCCS 2021*, pp. 165–179, Springer Nature Singapore, Singapore, 2022.
46. Lilhore, U.K., Simaiya, S., Prasad, D., Guleria, K., A hybrid tumour detection and classification based on machine learning. *J. Comput. Theor. Nanosci.*, 17, 6, 2539–2544, 2020.
47. Lilhore, U.K., Dalal, S., Faujdar, N., Margala, M., Chakrabarti, P., Chakrabarti, T., Velmurugan, H., Hybrid CNN-LSTM model with efficient hyperparameter tuning for prediction of Parkinson's disease. *Sci. Rep.*, 13, 1, 14605, 2023.
48. Lilhore, U.K., Manoharan, P., Sandhu, J.K., Simaiya, S., Dalal, S., Baqasah, A.M., Raahemifar, K., Hybrid model for precise hepatitis-C classification using improved random forest and SVM method. *Sci. Rep.*, 13, 1, 12473, 2023.
49. Ramesh, T.R., Lilhore, U.K., Poongodi, M., Simaiya, S., Kaur, A., Hamdi, M., Predictive analysis of heart diseases with machine learning approaches. *Malays. J. Comput. Sci.*, 1, 1, 132–148, 2022. https://doi.org/10.22452/mjcs.sp2022no1.10.
50. Sharma, S.K., Lilhore, U.K., Simaiya, S., Trivedi, N.K., An improved random forest algorithm for predicting the COVID-19 pandemic patient health. *Ann. Rom. Soc. Cell Biol.*, 25, 1, 67–75, 2021.
51. S. R. Swarna, A. Kumar, P. Dixit and T. V. M. Sairam, Parkinson's Disease Prediction using Adaptive Quantum Computing, *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, Tirunelveli, India, 2021, pp. 1396–1401, doi: 10.1109/ICICV50876.2021.9388628
52. Trivedi, N.K., Simaiya, S., Lilhore, U.K., Sharma, S.K., COVID-19 Pandemic: Role of Machine Learning & Deep Learning Methods in Diagnosis. *Int. J. Curr. Res. Rev.*, 13, 06, 150–156, 2021.

53. Upadhyay, N.K., Jiang, H., Wang, Z., Asapu, S., Xia, Q., Joshua Yang, J., Emerging memory devices for neuromorphic computing. *Adv. Mater. Technol.*, *4*, 4, 1800589, 2019.
54. Hassan, A., Prasad, D., Khurana, M., Lilhore, U.K., Simaiya, S., Integration of internet of things (IoT) in health care industry: An overview of benefits, challenges, and applications, in: *Data Science and Innovations for Intelligent Systems*, pp. 165–180, 2021.
55. Woo, J., Kim, J.H., Im, J.P., Moon, S.E., Recent advancements in emerging neuromorphic device technologies. *Adv. Intell. Syst.*, *2*, 10, 2000111, 2020.
56. Chakraborty, I., Jaiswal, A., Saha, A.K., Gupta, S.K., Roy, K., Pathways to efficient neuromorphic computing with non-volatile memory technologies. *Appl. Phys. Rev.*, *7*, 2, 278, 2020.
57. Rahimi Azghadi, M., Chen, Y.C., Eshraghian, J.K., Chen, J., Lin, C.Y., Amirsoleimani, A., Mehonic, A., Kenyon, A.J., Fowler, B., Lee, J.C., Chang, Y.F., Complementary metal-oxide semiconductor and memristive hardware for neuromorphic computing. *Adv. Intell. Syst.*, *2*, 5, 1900189, 2020.
58. Iwagi, E., Tsuno, T., Imai, T., Nakashima, Y., Kimura, M., Multilayer Crossbar Array of Amorphous Metal-Oxide Semiconductor Thin Films for Neuromorphic Systems. *IEEE J. Electron Devices Soc.*, *10*, 784–790, 2022.
59. Bhargava, P.N., Rathore, P.S., Vaishnav, P., Rai, M., Utilizing Artificial Neural Networks and Multivariate Patient Data for Anemia Detection using WEKA based Approach for Diagnosis. *2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS)*, Pudukkottai, India, pp. 643–648, 2023, doi: 10.1109/ICACRS58579.2023.10404355.
60. S. R. Burri, A. Kumar, A. Baliyan and T. A. Kumar, Predictive Intelligence for Healthcare Outcomes: An AI Architecture Overview, *2023 2nd International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)*, pp. 1–6, Villupuram, India, 2023, doi: 10.1109/ICSTSN57873.2023.10151477
61. Schuman, C.D., Potok, T.E., Patton, R.M., Birdwell, J.D., Dean, M.E., Rose, G.S., Plank, J.S., A survey of neuromorphic computing and neural networks in hardware, arXiv preprint arXiv:1705.06963, 2017.
62. Poon, C.S. and Zhou, K., Neuromorphic silicon neurons and large-scale neural networks: challenges and opportunities. *Front. Neurosci.*, *5*, 108, 2011.
63. Tang, J., Yuan, F., Shen, X., Wang, Z., Rao, M., He, Y., Sun, Y., Li, X., Zhang, W., Li, Y., Gao, B., Bridging biological and artificial neural networks with emerging neuromorphic devices: fundamentals, progress, and challenges. *Adv. Mater.*, *31*, 49, 1902761, 2019.
64. Sekhar, U.S., Vyas, N., Dutt, V., Kumar, A., Multimodal Neuroimaging Data in Early Detection of Alzheimer's Disease: Exploring the Role of Ensemble Models and GAN Algorithm, in: *2023 International Conference on Circuit Power and Computing Technologies (ICCPCT)*, 2023, August, IEEE, pp. 1664–1669.

65. Ghosh-Dastidar, S. and Adeli, H., Spiking neural networks. *Int. J. Neural Syst.*, *19*, 04, 295–308, 2009.
66. Tavanaei, A., Ghodrati, M., Kheradpisheh, S.R., Masquelier, T., Maida, A., Deep learning in spiking neural networks. *Neural Netw.*, *111*, 47–63, 2019.

17

Quantum Neurocomputing: Bridging the Frontiers of Quantum Computing and Neural Networks

Smitha[1]*, Yogesh Kumar Sharma[2], Muniraju Naidu Vadlamudi[3] and Leena Arya[2]

[1]Muscat College, University of Stirling, Muscat, Oman
[2]Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur District, A.P., India
[3]Koneru Lakshmaiah Education Foundation, Hyderabad, Telangana, India

Abstract

This chapter explores the notion of 'quantum neural computing' (QCN) under the context of many developing research fields in neural networks. We prioritize the development of innovative neuron and network models that facilitate rapid training. Additionally, we explore models of consciousness and attention, data processing through the cytoskeleton, the microtubule and quantum modelling. We examine current discoveries in neuroscience that challenge reductionist theories of the brain's data-processing mechanisms. Neural networks have achieved significant advancements in both the corporate and academic realms. An unresolved issue is the efficient construction of neural networks utilizing quantum computing hardware. A summary of quantum neurocomputing, the connection underlying quantum computing key developments and neural network advancements, and an examination of the key components of a quantum neural network have also been covered.

***Keywords*:** Quantum computing, neural networks, quantum neural networks, data processing, quantum neurocomputing

***Corresponding author*: smitha@muscatcollege.edu.om

Abhishek Kumar, Pramod Singh Rathore, Sachin Ahuja and Umesh Kumar Lilhore (eds.) Integrating Neurocomputing with Artificial Intelligence, (287–306) © 2025 Scrivener Publishing LLC

17.1 Introduction

Quantum computing, a novel technology, is rapidly gaining popularity. It employs the fundamental concepts of quantum mechanics to address problems that conventional computers cannot solve. The limitations imposed by classical computer technology fuel the substantial need for brain and quantum data processing. Traditional computers can effectively process symbols and numbers when the bit registers are relatively tiny, with a size of less than 128 bits [1]. Two significant obstacles prohibit it from handling multichannel signals alongside a length greater than 100 bits. The primary problem with the software implementation of sequence processing in traditional computers is that it requires many gates (d4.8) per Rent's Law to handle d-bit registers.

Contrarily, 2-D operators are often required by computer programs that can do universal computations on patterns. Because of this, using an algorithmic approach to pattern analysis is not an option. This is an issue that artificial neural networks (ANNs) can handle because of their innovative design, which allows them to learn from examples rather than instructions and control very lengthy bit strings. Even with little subject expertise, ANN can handle complicated issues [2].

Also, ANN is resilient and can perform distributed computations simultaneously. As stated by quantum rules, the fundamental objective of quantum computing is to minimize the physical dimensions of system elements. A quantum analogue of conventional computer architecture, using quantum bits and gates, is the focus of quantum computing research. Many characteristics of classical computers are preserved in quantum computers. Wideband signals are not within their capabilities, and they are not amenable to example-based training. The efficacy of these systems is contingent upon the robust quantum algorithms [3].

There are many issues with classical neural networks, such as a lack of guidelines for ideal topologies, lengthy training processes, and insufficient memory. Quantum computing, grounded in the principles of quantum physics, possesses inherent parallel distributed processing capabilities and exponential memory capacity. However, its potential is greatly hindered by its hardware limitations. While ANN utilizes a non-linear approach, quantum computing is a theoretical framework that operates on the principle of linearity [4].

Several fundamental ideas are included in ANNs or artificial neural networks. Consider the following: the concept of a processing element (neuron), the operation that this element carries out (usually, adding up all the inputs and then non-linearly mapping the result to an output value), the architecture of the network's connections between neurons, the dynamics of the network, and the learning rule that regulates the adjustment of interaction strengths. To overcome these obstacles, a hybrid method integrating quantum and neural processing called a quantum neural network, is necessary [5, 6]. This method overcomes the limits of classical computers, neurocomputers, and quantum computers.

Several studies have shown the link between quantum physics and neural networks via personal analogies. Neural networks are so effective because their nodes (neurons) analyze data in a distributed, massively parallel fashion, and their transformations are non-linear. Quantum mechanics, on the other hand, introduces the idea of superposition, which might lead to an even more substantial quantum parallelism. This allows us to handle massive data sets using neural networks trained in quantum computing [7].

The full chapter is structured into many parts. The first section provides an in-depth analysis of quantum computation, and the second section delves into quantum machine learning, the fourth section examines QNN, and the fifth section concludes the study.

17.2 Quantum Computation

A quantum computer presents one type of computer that utilizes the principles of quantum mechanics. Using specialized hardware, quantum computing creates and manipulates quantum states, utilizing the wave-particle duality observed at very small scales. A quantum computer is one type of computer that utilizes the principles of quantum mechanics. Using specialized hardware, quantum computing creates and manipulates quantum states, utilizing the wave-particle duality observed at very small scales. This includes quantum superposition and entanglement. A scalable quantum computer might do certain computations at an exponentially faster rate (concerning the scaling of input sizes) than any contemporary "classical" computer, and the workings of these quantum devices defy explanation by traditional physics [8].

The present level of knowledge is primarily experimental and unrealistic, with several barriers to practical implementations; yet, a large-scale quantum computer may decrypt commonly used encryption techniques and assist scientists in conducting physical simulations. On top of that, many real-world problems are unsolvable with quantum speedups, and scalable quantum computers show little promise for many more [9].

The qubit is the basic building block in quantum computing, similar to the bit in normal digital electronics. Unlike ordinary bits, qubits can exist in two "basis" states simultaneously, making them unique. Quantum computers are often non-deterministic since a standard bit's output is probabilistic when a qubit is measured. Wave interference effects may amplify the intended measurement findings if a quantum computer confidently manipulates the qubit. The goal of developing quantum algorithms is to facilitate the rapid and accurate execution of computations on a quantum computer [10].

Creating high-quality qubits via physical engineering has been difficult. Quantum decoherence causes noise to enter computations when a physical qubit is not physically separated from its surroundings. Since quantum operations often require initializing qubits, performing regulated qubit interactions, and measuring the resultant quantum states, completely isolating qubits is paradoxically undesirable. A buildup of inaccuracy occurs because each action is prone to noise and adds mistakes [11]. Developing scalable qubits with extended coherence periods and reduced error rates is a goal of the experimental research in which national governments have invested extensively. Superconductors and ion traps are the most exciting new technologies because they can isolate electrical currents by removing electrical resistance and confining a single ion using electromagnetic fields [12].

Given sufficient time, classical computers, which do not use quantum computing, can theoretically solve computational tasks on par with quantum computers. Instead of being computably efficient, quantum algorithms are superior in terms of temporal complexity. According to quantum complexity theory, some quantum algorithms may do specific tasks with exponentially smaller computing steps than the top non-quantum methods [13]. Theoretically, a large-scale quantum computer might do such calculations far more quickly than a conventional computer. Basic computing activities like sorting are shown not to enable any asymptotic quantum speedup; hence, quantum speedup is neither universal nor even standard across computational jobs. Although the field has received much interest due to claims of quantum supremacy, these claims have only been proven on artificial workloads, and there will be few real-world applications shortly [14].

Although there is reason to be optimistic about quantum computing due to the many new theoretical hardware possibilities made possible by quantum physics, this optimism is somewhat offset by our growing knowledge of its limits. Specifically, low-polynomial speedups may be undermined by noise and the application of quantum error correction, even though noiseless quantum computers have historically had their quantum speedups approximated.

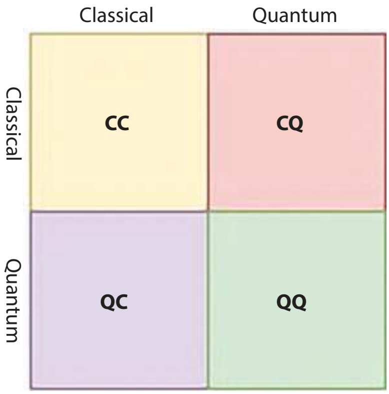
17.3 Quantum Machine Learning Technique

Neural networks have advanced in industry and academia. How to develop neural networks employing quantum computing devices is a difficult question. In this part, we discuss a quantum neural network model that computes practical problems of quantum machines with real-time triggered environment-induced decoding [15]. The difficulties of physical implementations are greatly reduced by this paradigm, which employs (classically controlled) single-qubit computations and measures. The method [15] overcomes the problem where the current state space grows rapidly with the growing number of neurons to decrease memory demands and permit speedy optimization, leveraging conventional techniques for optimization [16].

This type of machine learning is called quantum machine learning (QML), and it occurs when machine learning systems use quantum methods. “quantum-enhanced machine learning” commonly refers to quantum computational techniques that analyze conventional data. QML uses qubits, quantum processes, and quantum networks to speed up and store program algorithms [17].

Machine learning techniques handle massive data sets. Some methods use conventional and quantum computing to deliver sophisticated subroutines to a quantum device. Quantum computers can perform these tasks faster and more complexly. Instead of data, quantum algorithms can investigate quantum states. The quantum machine learning structure is shown in Figure 17.1 [18].

The term “quantum machine learning” encompasses two distinct meanings. Firstly, it refers to applying traditional machine learning methods to



	Classical	Quantum
Classical	CC	CQ
Quantum	QC	QQ

Figure 17.1 Standard structure of QML.

Table 17.1 A comprehensive examination of different QML.

References	Method used	Outcome	Limitation
[5]	Quantum Circuit/ Feedforward NN	Improved convergence, reduced training time	Limited scalability to large datasets
[6]	Quantum Boltzmann Machine/RNN	Higher accuracy in time-series prediction	Sensitivity to hyperparameter tuning
[7]	Quantum Hopfield Network	Effective pattern retrieval	Limited storage capacity
[8]	Quantum Convolutional Neural Network (QCNN)	Enhanced feature extraction in image data	Resource-intensive computations
[9]	Quantum Long Short-Term Memory Network (QLSTM)	Improved memory retention	Challenges in implementation on real hardware
[10]	Quantum Autoencoder	Efficient dimensionality reduction	Dependency on quantum hardware advancements
[11]	Neural Quantum State Tomography	Accurate reconstruction of quantum states	Limited to small-scale quantum systems
[12]	Quantum Restricted Boltzmann Machine	Effective in unsupervised learning	Training complexity increases with system size
[13]	Quantum Recurrent Spiking Neural Network	Improved temporal processing in spiking neurons	Highly dependent on input encoding schemes
[14]	Quantum Generative Adversarial Network	Enhanced generation of realistic samples	Sensitivity to hyperparameter choices

analyze data generated by quantum studies, specifically the mathematical modelling of quantum systems. Secondly, it pertains to advancements in quantum computing, including the creation of novel quantum experiments or the instruction of a quantum system's transition between phases [19].

Another area of QML is studying the similarities in methodology and structure among certain biological and computerized learning systems, particularly neural networks. For instance, traditional deep learning and quantum physics have potential advantages, such as employing confident mathematical and statistical approaches. Furthermore, the term "quantum learning theory" refers to a study exploring broader concepts of learning theory concerning quantum technology [20]. Table 17.1 displays a comprehensive examination of different.

17.3.1 Applying Machine Learning Techniques in Quantum Computers

Quantum-enhanced machine learning (QEML) is a specialized area within machine learning models which leverages quantum computing to augment and potentially accelerate conventional ML techniques. The typical approach for constructing such algorithms is converting the given conventional data set inside a quantum computing device to enable information that uses quantum processing. Subsequently, quantum computing is assessed using measurements of the quantum structure, and quantum data extraction protocols are executed [21].

Imagine a binary classification issue where the outcome is determined by measuring a qubit. Several QML algorithms have already been tested on devices with restricted capabilities or that are specifically designed for a particular purpose. However, certain algorithms are still theoretical and need a functional quantum computer globally [22].

17.3.2 Quantum-Enhanced Reinforcement Learning

Quantum-Enhanced Reinforcement Learning (QERL), a distinct subject of machine learning, might potentially reap the advantages of quantum advancements. It stands apart from unsupervised and supervised learning. A quantum entity can modify its behavior through QERL by engaging with conventional and quantum surroundings and frequently obtaining rewards for its actions [23]. In some cases, a quantum speedup may be achievable if the agent possesses quantum processing skills or if composite contextual probing is feasible.

Two possible ways to implement such protocols are using entrapped and superconductivity circuits. Investigations with entrapped have demonstrated a quantum acceleration in the inbuilt decision-making process of the agent. Similarly, investigations using an optical configuration demonstrate quantum acceleration in the learning time through fully coherent communication between the agent that makes decisions and the external environment [24].

17.3.3 Quantum Annealing

Finding a function's local minimum and maximum across a collection of candidate functions is the goal of quantum annealing, an optimization method. This is a way to find the observables of a function by discretizing it with several local minima or maxima [25].

Quantum tunneling differentiates itself from the simulation of annealing by including the passage of electrons from a higher energy state to a lower one by tunneling via kinetic and potential obstacles. Quantum annealing relies on a superposition of all conceivable system states, each having an equal weight. Subsequently, the time-dependent spatiotemporal equation governs the system's temporal progression, which influences the magnitude of every situation as time elapses. The system's immediate Hamiltonian may be obtained by attaining the fundamental form [26].

17.4 Quantum Neural Networks

Models of computational neural networks grounded on quantum mechanics are known as quantum neural networks. Using the notion of the quantum mind, which proposes that quantum effects contribute to cognitive

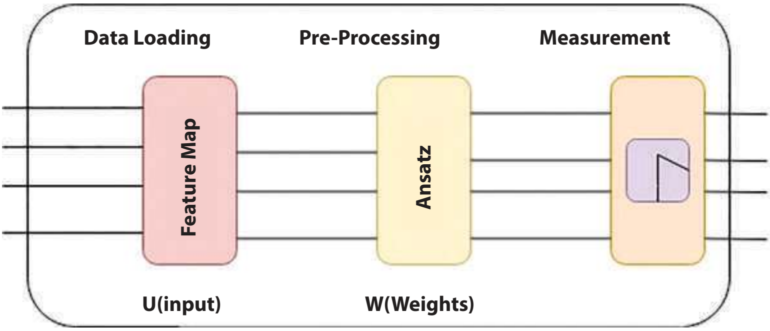


Figure 17.2 Basic model for QNN.

Table 17.2 Comparative analysis of QNNS vs traditional methods.

Aspect	QNNs	Traditional methods
Basic Principle	Utilizes quantum mechanics principles like superposition and entanglement	Based on classical computing principles
Data Processing	Can process data in quantum states, enabling parallel computations	Processes data in binary states (0s and 1s)
Computational Power	Potential for exponential speedup in certain tasks	Linear or polynomial speedup
Memory Requirements	It may require fewer resources due to quantum superposition	Typically requires large memory for large datasets
Training Algorithms	Quantum algorithms like Quantum Approximate Optimization Algorithm (QAOA)	Classical algorithms like backpropagation
Error Correction	More susceptible to noise and requires quantum error correction techniques	Less susceptible to noise, traditional error correction used
Scalability	Currently limited by quantum hardware constraints	Highly scalable with current classical hardware
Current Maturity	In early experimental stages with limited real-world applications	Well-established with numerous practical applications
Potential Applications	Cryptography, optimization, drug discovery, complex simulations	Image and speech recognition, natural language processing, predictive analytics
Challenges	Quantum decoherence, error rates, and hardware limitations	Overfitting, vanishing gradients, and large resource requirements
Learning Capability	Promises superior performance for specific complex tasks	Proven effective across a broad range of applications
Hardware Requirements	Requires quantum computers (e.g., superconducting qubits, ion traps)	Runs on classical computers (e.g., CPUs, GPUs)
Energy Efficiency	Potentially more energy-efficient for specific quantum tasks	Energy consumption is higher due to classical hardware requirements

function separately, the first concepts on quantum brain computing were released in 1995. However, the majority of research on Quantum Neural Networks (QNNs) integrates the advantages of quantum computation with those of conventional artificial neural network models, which are widely employed in Machine Learning (ML) for the important task of pattern recognition, to develop more sophisticated algorithms [27].

The challenge of training conventional neural networks, particularly for large data applications, is a driving factor in these studies. It is believed that quantum computing properties, such as quantum parallelism, interference, and entanglement, may be harnessed for functional purposes. Such models of quantum neural networks are primarily theoretical suggestions that will have to wait for complete application in practical trials since the technical implementation of quantum computers is still in its infancy. The model of a quantum neural network (QNN) is shown in Figure 17.2 [28].

The majority of QNNs are built using the feed-forward method. Like its classical analogues, this structure accepts input from a single layer of qubits and transfers it to another. The information is evaluated by this layer of qubits, which then sends the result on to the next layer. The last layer of qubits is reached at the end of the journey. The layers do not need to have the same amount of qubits; in other words, their breadth is not fixed. Like traditional artificial neural networks, this structure learns its way around obstacles. This is discussed in the following section. One way to categorize a QNN is as a traditional computer that uses quantum data, another as a quantum computer that uses classical data, and another as a hybrid [29]. Table 17.2 presents a comparative analysis of QNNS Vs traditional methods.

17.4.1 Quantum Perceptrons

Numerous suggestions seek a quantum analogue of the perceptron unit, the building block of neural networks. Since linear operations characterize a quantum development and lead to probabilistic observation, the mathematical structure of quantum theory does not instantly correlate to non-linear activation functions.

There's a problem here. From the controversial postulate of non-linear quantum drivers to the more recent proposal by Schuld, Sinayskiy and Petruccione of immediate implementation of the activation function within the circuit-based quantum computing paradigm utilizing the quantum phase estimation technique, there is a spectrum of possible quantum mechanical mathematical representations of the perceptron activation function [30]. Table 17.3 presents Quantum Perceptrons, key elements.

Table 17.3 Quantum perceptrons, key elements.

Element	**Quantum perceptrons**
Basic unit	**Qubit**
State Representation	Superposition of states
Information Encoding	Quantum states (amplitude and phase)
Activation Function	Quantum gates (e.g., Pauli-X, Hadamard, Controlled-NOT)
Weight Representation	Quantum gates adjusting qubit states
Learning Mechanism	Quantum algorithms (e.g., Quantum Approximate Optimization Algorithm - QAOA)
Input Data	Quantum states or classical data encoded into qubits
Output Data	Quantum states measured to provide classical output
Noise and Error Handling	Quantum error correction techniques
Parallelism	Inherent due to quantum superposition
Computational Basis	Quantum linear algebra operations
Training Algorithm	Variational Quantum Eigensolver (VQE), Quantum Gradient Descent
Hardware Requirements	Quantum processors (e.g., superconducting qubits, ion traps)
Entanglement Utilization	Exploits entanglement for complex correlations
Measurement	Quantum measurement collapses the state to classical bit values
Scalability	Limited by qubit count and coherence time
Optimization	Quantum optimization algorithms
Implementation	Quantum circuits and gates designed for specific tasks

17.4.2 Quantum Networks

Investigators have attempted to implement larger-scale neural networks in a quantum setting. One way to construct quantum neurons is to begin with conventional neurons and then generalize them to produce unitary gates. Quantum control via unitary gates or classical control by measuring network states are viable options for controlling neuronal interactions. Various kinds of networks and quantum neuron implementations, including photonic ally-implemented neurons, may be used to apply this high-level theoretical method extensively [31].

Table 17.4 Quantum networks comparative analysis.

Aspect	**Quantum networks**	**Classical networks**
Basic Unit	Qubits	Bits
State Representation	Superposition of quantum states	0, 1
Information Encoding	Quantum states (amplitude and phase)	Binary encoding (0 or 1)
Data Processing	Quantum gates and circuits	Logical gates and circuits
Activation Function	Quantum gates (e.g., Pauli-X, Hadamard, CNOT)	Functions like sigmoid, ReLU, softmax
Network Structure	Quantum circuits composed of qubits and quantum gates	Layers of neurons interconnected with weights
Learning Mechanism	Quantum algorithms (e.g., QAOA, VQE, Quantum Gradient Descent)	Gradient descent, backpropagation
Parallelism	Inherent due to quantum superposition and entanglement	Achieved through multi-core processors and GPUs
Error Handling	Quantum error correction techniques	Classical error correction methods

(*Continued*)

Table 17.4 Quantum networks comparative analysis. (*Continued*)

Aspect	**Quantum networks**	**Classical networks**
Scalability	Limited by current quantum hardware capabilities	Highly scalable with existing hardware
Hardware Requirements	Quantum processors (e.g., superconducting qubits, ion traps)	Classical processors (CPUs, GPUs, TPUs)
Training Algorithms	Quantum-specific algorithms, variational circuits	Classical training algorithms (SGD, Adam)
Data Input	Quantum states or classical data encoded into qubits	Binary or continuous data
Output Data	Quantum states measured to provide classical output	Binary or continuous data
Noise and Error Rates	Susceptible to quantum decoherence and noise	Less susceptible to noise
Optimization	Quantum optimization algorithms	Classical optimization algorithms
Current Maturity	Experimental and early research stages	Mature and widely used in various applications
Potential Applications	Quantum machine learning, optimization, complex simulations	Broad applications including AI, data analysis
Entanglement Utilization	Exploits entanglement for complex correlations	No entanglement, relies on classical correlations
Energy Efficiency	Potentially more energy-efficient for specific tasks	Typically higher energy consumption
Community and Research	Growing research community with increasing interest	Established community with extensive literature

To acquire knowledge of the connection between inputs and outputs for specific training sets, most learning computational methods use traditional feedback processes that modify the variables of the quantum framework until they come together to an optimal configuration. This is done in the quantum reservoir processor, a data processing system. This paradigm is based on training an artificial neural network. Analytical models of quantum computing have also taken a go at learning as an attribute optimization issue [32].

Algorithmic design may benefit from quantum neural networks. With adjustable mutual interactions in qubits, one can train the network to learn the intended output algorithm's behavior from a set of desirable input-output relations using the classical backpropagation method. In this way, an algorithm is "learned" by the quantum network [33]. Table 17.4 presents Quantum networks comparative analysis.

17.4.3 Quantum Associative Memory

In 1999, Antonio Martinez and Daniel Ventura introduced the initial quantum memory with association's method. The authors [34, 35] offer a way a circuit-based quantum computing system can simulate memory with associations rather than attempting to adapt the conceptual framework of ANN simulations into quantum mechanics. The memories that hold phases in Hopfield artificial neural networks are encoded in the weights comprising the connections between neurons via the superposition process.

Next, the memory configuration nearest to an input is retrieved using a quantum retrieval method comparable to Grover's. As mentioned earlier, memory cannot be fully said to be content-addressable due to the limits of recovering only specific patterns [36]. The concept of a "content-addressable quantum memory" (CAQM) that might recuperate patterns from corrupted inputs was first proposed by Marshall *et al.* [24]. These memories can hold an exponential (in terms of n qubits) number of patterns. However, they can only be used once due to the no-cloning theorem and are destroyed upon measurement. With Marshall *et al.* [24], probabilistic framework for quantum memory with associations, unlike traditional associative memories, it is possible to construct and reuse the model many times for any quadratic array of recorded sequences [37].

17.4.4 Quantum Convolution Neural Network

Quantum convolution neural network (QCNN) is a novel MV architecture that uses circuits as convolution filters. The main inspirations were CNNs and QML. It uses a DNN and a "variational quantum circuit" (VQC) to superpose a quantum state with restricted qubits, taking advantage of extremely parallel processing. To optimize NISQ devices repeatedly without noise in the circuit parameter or quantum error correction, we want to prevent noise's negative impacts [38].

The quantum circuit must handle spatial information well for QCNN to be a CNN. Convolution filters are the easiest way to use geographical data. QCNNs [39] use one or more quantum convolutional filters with organized or randomly generated quantum circuits to modify input data. Measurement, encoder, and parameterized quantum circuits make up a quantum convolutional filter. The quantum convolutional filter extends the CNN filter since it has trainable parameters [40].

Quantum neural networks use hierarchical topologies to lower qubits by two per layer. These topologies provide shallow circuit depth for n input qubits with O (log (n)) layers. They also avoid the "barren plateau," a major issue with PQC-based algorithms, making them trainable. Despite not including the matching quantum operation, the QCNN model is viable because of the pooling layer [41].

A QCNN pooling layer is commonly between two convolutional layers following each other. Its objective is to decrease representation spatial size while maintaining crucial properties for parameter reduction, network computation performance, and over-fitting control. Complete Tomography on the state reduces it to a qubit, which may be processed in the subway. Max pooling is the most popular pooling layer unit type. In feed-forward neural networks, the final module is a fully connected layer that links to all activations in the previous layer. Translational invariance requires equal blocks of parameterized quantum gates inside a layer, differentiating the QCNN design [42].

17.4.5 Dissipative Quantum Neural Network

Dissipative QNNs (DQNNs) are layers of qubits coupled by Perceptrons that can have any unitary architecture. The network layer of a DQNN assigns each node a unique collection of qubits and a unique quantum perceptron unitary. As the name implies, the network sends input state information through a layer-to-layer transition mapping on the qubits of the two neighbouring layers [43]. As they trace the last layer, the auxiliary qubits create the output layer and remove the input layers, another definition of dissipative. Deep Convolutional Neural Networks (DQNNs) perform supervised learning to train a unitary matrix connecting input and output quantum states. This task trains on quantum states and classical labels [44].

Dissipative quantum generative adversarial networks (DQGANs) are inspired by powerful classical GANs and are created for the unsupervised learning of unlabeled training data. DQGANs have two DQNN generators and a discriminator. The generator generates training states that the discriminator cannot differentiate from the real ones, whereas the discriminator must separate natural training states from phoney ones. Adversarial and alternative network training helps the generator understand the fundamental characteristics of the training set and build sets that expand it. DQGAN trains using quantum data and quantum architecture [45].

17.5 Conclusion and Future Directions

The revolutionary combination of neural networks and quantum computing is poised to transform the computing industry, known as quantum neurocomputing. This revolutionary convergence is transforming the computing field with its incredible potential. With its incredible computational power and lightning-fast data analysis capabilities, quantum neurocomputing is truly a marvel. This is achieved by employing principles from quantum mechanics, such as superposition and entanglement. Understanding the principles of quantum mechanics is crucial in making this possible. Traditional computers encounter some obstacles that are particularly challenging to surmount. These challenges involve tasks that require a deep understanding of complex systems, the ability to process and analyze large volumes of data, and the skill to identify subtle patterns. Combining these two factors will probably result in notable progress in addressing these challenges. Using quantum computing and neural networks, exciting opportunities for progress in cryptography, materials science, and artificial intelligence are beginning to unfold.

To progress in quantum neurocomputing, it is essential to address the existing constraints in theory and technology. For progress to be made, this condition is crucial. Scientists are currently delving into different research fields, such as enhancing the durability and error-resistance of quantum hardware, exploring hybrid classical-quantum frameworks, and creating efficient quantum algorithms tailored for neural network operations. Here are a few of the most noteworthy areas of research that warrant attention. Adopting a multidisciplinary approach and promoting collaboration will be instrumental in discovering solutions to these challenges. This will happen because it will facilitate the progress of breakthroughs that can expedite the integration of quantum neurocomputing into practical uses. Given the progress of these technologies, we can anticipate the rise of intricate and intelligent systems that will address the most formidable problems of our era. It is evident that these technologies will significantly impact businesses that heavily depend on intricate computer operations.

References

1. Hamid, M., Alam, B., Pal, O., Qamar, S., Investigating Classification with Quantum Computing, in: *Intelligent Data Analytics, IoT, and Blockchain*, pp. 302–314, Auerbach Publications, Boca Raton, Florida, 2024.
2. Conti, C., *Quantum Machine Learning: Thinking and Exploration in Neural Network Models for Quantum Science and Quantum Computing*, Springer Nature, Heidelberg, Germany, 2024.
3. Dahiya, N., Sharma, Y.K., Rani, U., Hussain, S., Nabilal, K.V., Mohan, A., Nuristani, N., Hyper-parameter tuned deep learning approach for effective human monkeypox disease detection. *Sci. Rep.*, 13, 1, 15930, 2023.
4. Gong, L.H., Pei, J.J., Zhang, T.F., Zhou, N.R., Quantum convolutional neural network based on variational quantum circuits. *Opt. Commun.*, *550*, 129993, 2024.
5. Pradeep, S., Sharma, Y.K., Lilhore, U.K., Simaiya, S., Kumar, A., Ahuja, S., Chakrabarti, T., Developing an SDN security model (EnsureS) based on lightweight service path validation with batch hashing and tag verification. *Sci. Rep.*, 13, 1, 17381, 2023.
6. Das, M., Naskar, A., Mitra, P., Basu, B., Shallow quantum neural networks (SQNNs) with application to crack identification. *Appl. Intell.*, 54, 2 1–16, January 2, 2024.
7. Nayak, S. and Sharma, Y.K., A modified Bayesian boosting algorithm with weight-guided optimal feature selection for sentiment analysis. *Decis. Anal. J.*, 8, 100289, 2023.
8. Vijay, A., Bhargava, H., Pareek, A., Suravajhala, P., Sharma, A., Quantum Machine Learning for Biological Applications, in: *Bioinformatics and Computational Biology*, pp. 75–86, Chapman and Hall/CRC, Boca Raton, Florida, USA, 2024.
9. Xu, S., Wilhelm-Mauch, F., Maass, W., Quantum Feature Embeddings for Graph Neural Networks, in: *Hawaii International Conference on System Sciences*, 2024, January.
10. Hong, Y.Y., Rioflorido, C.L.P.P., Zhang, W., Hybrid deep learning and quantum-inspired neural network for day-ahead spatiotemporal wind speed forecasting. *Expert Syst. Appl.*, *241*, 122645, 2024.
11. Kumar, N., Verma, H., Sharma, Y.K., Adversarial Attacks on Graph Neural Network: Techniques and Countermeasures, in: *Concepts and Techniques of*

Graph Neural Networks, pp. 58–73, IGI Global, Pennsylvania, United States, 2023.

12. Arrazola, J.M., Bromley, T.R., Izaac, J., Myers, C.R., Brádler, K., Killoran, N., Machine learning method for state preparation and gate synthesis on photonic quantum computers. *Quantum Sci. Technol.*, *4*, 2, 024004, 2019.
13. Ajani, S.N., Khobragade, P., Dhone, M., Ganguly, B., Shelke, N., Parati, N., Advancements in Computing: Emerging Trends in Computational Science with Next-Generation Computing. *Int. J. Intell. Syst. Appl. Eng.*, *12*, 7s, 546–559, 2024.
14. Kumar, N., Verma, H., Sharma, Y.K., Graph Convolutional Neural Networks for Link Prediction in Social Networks, in: *Concepts and Techniques of Graph Neural Networks*, pp. 86–107, IGI Global, Pennsylvania, USA, 2023.
15. Otterbach, J.S., Manenti, R., Alidoust, N., Bestwick, A., Block, M., Bloom, B., Caldwell, S., Didier, N., Fried, E.S., Hong, S., Karalekas, P., Unsupervised machine learning on a hybrid quantum computer, arXiv preprint arXiv:1712.05771, 2017.
16. Data, B., Check for Big Data, Artificial Intelligence, and Quantum Computing in Sports Benno Torgler, in: *21st Century Sports: How Technologies Will Change Sports in the Digital Age*, p. 169, 2024.
17. Zhou, M.G., Liu, Z.P., Yin, H.L., Li, C.L., Xu, T.K., Chen, Z.B., Quantum Neural Network for Quantum Neural Computing. *Research*, *6*, 0134, 2023.
18. Singh, K., Kumar, A., Sharma, Y.K., Rai, A.K., AIoT-based e-commerce, in: *AIoT Technologies and Applications for Smart Environments*, p. 215, 2023.
19. Huang, S., Chang, Y., Lin, Y., Zhang, S., Hybrid quantum-classical convolutional neural networks with privacy quantum computing. *Quantum Sci. Technol.*, *8*, 2, 025015, 2023.
20. Sharma, Y.K., Athithan, S., Sachi, S., Singh, A.K., Jain, A., Devi, S., Copy and Move Forged Image Detection by Deep Learning, in: *2023 World Conference on Communication & Computing (WCONF)*, 2023, July, IEEE, pp. 1–6.
21. Dubey, A.K., Kumar, A., García-Díaz, V., Sharma, A.K. Kanhaiya, K., Study and analysis of SARIMA and LSTM in forecasting time series data. *Sustain. Energy Technol. Assessments*, 47, 101474, 2021.
22. Kim, J., Huh, J., Park, D.K., Classical-to-quantum convolutional neural network transfer learning. *Neurocomputing*, *555*, 126643, 2023.
23. Sachi, S., Singh, A.K., Jain, A., Devi, S., Sharma, Y.K., Athithan, S., Hate Speech Detection Using the GPT-2 and Natural Language Processing, in: *2023 Intelligent Methods, Systems, and Applications (IMSA)*, 2023, July, IEEE, pp. 276–280.
24. Marshall, S.C., Gyurik, C., Dunjko, V., High-dimensional quantum machine learning with small quantum computers. *Quantum*, *7*, 1078, 2023.
25. Sharma, Y.K. and Khatal Sunil, S., Health Care Patient Monitoring using IoT and Machine Learning, in: *IOSR Journal of Engineering (IOSR JEN) National Conference on "Recent Innovations in Engineering and Technology" MOMENTUM-19*, 2019.

26. Perdomo-Ortiz, A., Benedetti, M., Realpe-Gómez, J., Biswas, R., Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers. *Quantum Sci. Technol.*, *3*, 3, 030502, 2018.
27. Pradeep, S. and Sharma, Y.K., A pragmatic evaluation of stress and performance testing technologies for web-based applications, in: *2019 Amity International Conference on Artificial Intelligence (AICAI)*, 2019, February, IEEE, pp. 399–403.
28. Schuld, M. and Petruccione, F., *Supervised learning with quantum computers*, vol. 17, Springer, Berlin, 2018.
29. Sharma, Y.K. and Rokade, M.D., Deep and machine learning approaches for anomaly-based intrusion detection of imbalanced network traffic. *IOSR J. Eng.*, 12, 15, 63–67, 2019.
30. Mahajan, R.P., A quantum neural network approach for portfolio selection. *Int. J. Comput. Appl.*, *29*, 4, 47–54, 2011.
31. Balyan, A.K., Ahuja, S., Sharma, S.K., Lilhore, U.K., Machine learning-based intrusion detection system for healthcare data, in: *2022 IEEE VLSI Device Circuit and System (VLSI DCS)*, 2022, February, IEEE, pp. 290–294.
32. Cao, Y., Guerreschi, G.G., Aspuru-Guzik, A., Quantum neuron: an elementary building block for machine learning on quantum computers, arXiv preprint arXiv:1711.11240, 2017.
33. Swapna, M., Sharma, Y.K., Prasad, B.M.G., A survey on face recognition using convolutional neural network, in: *Data Engineering and Communication Technology: Proceedings of 3rd ICDECT-2K19*, pp. 649–661, Springer Singapore, Singapore, 2020.
34. Cai, X.D., Wu, D., Su, Z.E., Chen, M.C., Wang, X.L., Li, L., Liu, N.L., Lu, C.Y., Pan, J.W., Entanglement-based machine learning on a quantum computer. *Phys. Rev. Lett.*, *114*, 11, 110504, 2015.
35. Rokade, M.D. and Sharma, Y.K., MLIDS: A machine learning approach for intrusion detection for real-time network dataset, in: *2021 International Conference on Emerging Smart Computing and Informatics (ESCI)*, 2021, March, IEEE, pp. 533–536.
36. Schuld, M. and Petruccione, F., *Machine learning with quantum computers*, Springer, Berlin, 2021.
37. Jadhav, D., Sharma, Y.K., Arora, D.P., Profound Learning Approach for Shot Boundary Location, in: *2nd International Conference on Communication & Information Processing (ICCIP)*, 2020, April.
38. Schuld, M., Sinayskiy, I., Petruccione, F., An introduction to quantum machine learning. *Contemp. Phys.*, *56*, 2, 172–185, 2015.
39. Jadhav, M., Kumar Sharma, Y., Bhandari, G.M., Currency identification and forged banknote detection using deep learning, in: *2019 International Conference on Innovative Trends and Advances in Engineering and Technology (ICITAET)*, 2019, December, IEEE, pp. 178–183.

40. Kumar, A., Chatterjee, J. M., Díaz, V. G., A novel hybrid approach of SVM combined with NLP and probabilistic neural network for email phishing. *Int. J. Electr. Comput. Eng.*, 10, 1, 486, 2020.
41. Bhargava, P.N., Rathore, P.S., Vaishnav, P., Rai, M., Utilizing Artificial Neural Networks and Multivariate Patient Data for Anemia Detection using WEKA based Approach for Diagnosis. *2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS)*, Pudukkottai, India, pp. 643–648, 2023, doi: 10.1109/ICACRS58579.2023.10404355.
42. Dubey, A.K., Kumar, A. & Agrawal, R. An efficient ACO-PSO-based framework for data classification and preprocessing in big data. *Evol. Intel.,* 14, 909–922, 2021.
43. Wani, S., Ahuja, S., Kumar, A., Application of Deep Neural Networks and Machine Learning algorithms for diagnosis of Brain tumour. *2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, Greater Noida, India, pp. 106–111, 2023, doi: 10.1109/CISES58720.2023.10183528.
44. Trivedi, N.K., Simaiya, S., Lilhore, U.K., Sharma, S.K., COVID-19 Pandemic: Role of Machine Learning & Deep Learning Methods in Diagnosis. *Int. J. Curr. Res. Rev.*, 13, 06, 150–156, 2021.
45. Benkhaddra, I., Kumar, A., Setitra, M.A., *et al.*, Design and Development of Consensus Activation Function Enabled Neural Network-Based Smart Healthcare Using BIoT. *Wireless Pers. Commun.*, 130, 1549–1574, 2023, https://doi.org/10.1007/s11277-023-10344-0.

Index

Also of Interest

Check out these related titles from Scrivener Publishing

Networked Sensing Systems, Edited by Rajesh Kumar Dhanaraj, Malathy Sathyamoorthy, Balasubramaniam S., and Seifedine Kadry, ISBN: 9781394310869. *Networked Sensing Systems* is essential for anyone seeking innovative and sustainable solutions across diverse sectors, as it explores the integration of cutting-edge IoT technologies and digital transformation aimed at enhancing resource efficiency and addressing climate change challenges.

Multimodal Data Fusion for Bioinformatics Artificial Intelligence, Edited by Umesh Kumar Lilhore, Abhishek Kumar, Narayan Vyas, Sarita Simaiya, and Vishal Dutt, ISBN: 9781394269938. *Multimodal Data Fusion for Bioinformatics Artificial Intelligence* is a must-have for anyone interested in the intersection of AI and bioinformatics, as it not only delves into innovative data fusion methods and their applications in omics research but also addresses the ethical implications and future developments shaping the field today.

Artificial Intelligence-Based System Models in Healthcare, Edited by A. Jose Anand, K. Kalaiselvi and Jyotir Moy Chatterjee, ISBN: 9781394242498. Artificial Intelligence-Based System Models in Healthcare provides a comprehensive and insightful guide to the transformative applications of AI in the healthcare system.

EXPLAINABLE MACHINE LEARNING MODELS AND ARCHITECTURES: Real-Time System Implementation, Edited by Suman Lata Tripathi and Mufti Mahmud, ISBN: 9781394185849. This cutting-edge new volume covers the hardware architecture implementation, the software implementation approach, and the efficient hardware of machine learning applications.

Natural Language Processing for Software Engineering, Edited by Rajesh Kumar Chakrawarti, Ranjana Sikarwar, Sanjaya Kumar Sarangi, Samson Arun Raj Albert Raj, Shweta Gupta, Krishnan Sakthidasan Sankaran, and Romil Rawat, ISBN: 9781394272433. Discover how Natural Language Processing for Software Engineering can transform your understanding of agile development, equipping you with essential tools and insights to enhance software quality and responsiveness in todays rapidly changing technological landscape.

MACHINE LEARNING TECHNIQUES FOR VLSI CHIP DESIGN, Edited by Abhishek Kumar, Suman Lata Tripathi, and K. Srinivasa Rao, ISBN: 9781119910398. This cutting-edge new volume covers the hardware architecture implementation, the software implementation approach, and the efficient hardware of machine learning applications with FPGA or CMOS circuits, and many other aspects and applications of machine learning techniques for VLSI chip design.

Intelligent Green Technologies for Smart Cities, Edited by Suman Lata Tripathi, Souvik Ganguli, Abhishek Kumar, and Tengiz Magradze, ISBN: 9781119816065. Presenting the concepts and fundamentals of smart cities and developing "green" technologies, this volume, written and edited by a global team of experts, also goes into the practical applications that can be utilized across multiple disciplines and industries, for both the engineer and the student.

Hybrid Intelligent Optimization Approaches for Smart Energy: A Practical Approach, Edited by Senthilkumar Mohan, A. John, Sanjeevikumar Padmanaban, and Yasir Hamid, ISBN: 9781119821243. Written and edited by a group of experts in the field, this is a comprehensive and up-to-date description of current energy optimization techniques, such as artificial intelligence techniques, machine learning, deep learning, and IoT techniques and their future trends.

NANODEVICES FOR INTEGRATED CIRCUIT DESIGN, Edited by Suman Lata Tripathi, Abhishek Kumar, K. Srinivasa Rao, and Prasantha R. Mudimela, ISBN: 9781394185788. Written and edited by a team of experts in the field, this important new volume broadly covers the design of nano-devices and their integrated applications in digital and analog integrated circuits (IC) design.

DESIGN AND DEVELOPMENT OF EFFICIENT ENERGY SYSTEMS, edited by Suman Lata Tripathi, Dushyant Kumar Singh, Sanjeevikumar Padmanaban, and P. Raja, ISBN 9781119761631. Covering the concepts and fundamentals of efficient energy systems, this volume, written and edited by a global team of experts, also goes into the practical applications that can be utilized across multiple industries, for both the engineer and the student.

Electrical and Electronic Devices, Circuits, and Materials: Technical Challenges and Solutions, edited by Suman Lata Tripathi, Parvej Ahmad Alvi, and Umashankar Subramaniam, ISBN 9781119750369. Covering every aspect of the design and improvement needed for solid-state electronic devices and circuit and their reliability issues, this new volume also includes overall system design for all kinds of analog and digital applications and developments in power systems.

MODELING AND OPTIMIZATION OF OPTICAL COMMUNICATION NETWORKS, Edited by Chandra Singh, Rathishchandra R Gatti, K.V.S.S.S.S. Sairam, and Ashish Singh, ISBN: 9781119839200. *Modeling and Optimization of Optical Communication Networks* is a comprehensive and authoritative book that delves into the optical networks, principles, technologies, and practical applications of optical networks, equipping readers with the knowledge needed to design, implement, and optimize optical networks for various applications, from telecommunications to scientific research. With its comprehensive coverage and up-to-date insights, this book serves as an essential reference in the field of optical networks.

WIRELESS COMMUNICATION SECURITY: Mobile and Network Security Protocols, Edited by Manju Khari, Manisha Bharti, and M. Niranjanamurthy, ISBN: 9781119777144. Presenting the concepts and advances of wireless communication security, this volume, written and edited by a global team of experts, also goes into the practical applications for the engineer, student, and other industry professionals.

ADVANCES IN DATA SCIENCE AND ANALYTICS, edited by M. Niranjanamurthy, Hemant Kumar Gianey, and Amir H. Gandomi, ISBN: 9781119791881. Presenting the concepts and advances of data science and analytics, this volume, written and edited by a global team of experts, also goes into the practical applications that can be utilized across multiple disciplines and industries, for both the engineer and the student, focusing on machining learning, big data, business intelligence, and analytics.

ARTIFICIAL INTELLIGENCE AND DATA MINING IN SECURITY FRAMEWORKS, Edited by Neeraj Bhargava, Ritu Bhargava, Pramod Singh Rathore, and Rashmi Agrawal, ISBN 9781119760405. Written and edited by a team of experts in the field, this outstanding new volume offers solutions to the problems of security, outlining the concepts behind allowing computers to learn from experience and understand the world in terms of a hierarchy of concepts.

MACHINE LEARNING AND DATA SCIENCE: Fundamentals and Applications, Edited by Prateek Agrawal, Charu Gupta, Anand Sharma, Vishu Madaan, and Nisheeth Joshi, ISBN: 9781119775614. Written and edited by a team of experts in the field, this collection of papers reflects the most up-to-date and comprehensive current state of machine learning and data science for industry, government, and academia.

MEDICAL IMAGING, Edited by H. S. Sanjay, and M. Niranjanamurthy, ISBN: 9781119785392. Written and edited by a team of experts in the field, this is the most comprehensive and up-to-date study of and reference for the practical applications of medical imaging for engineers, scientists, students, and medical professionals.

SECURITY ISSUES AND PRIVACY CONCERNS IN INDUSTRY 4.0 APPLICATIONS, Edited by Shibin David, R. S. Anand, V. Jeyakrishnan, and M. Niranjanamurthy, ISBN: 9781119775621. Written and edited by a team of international experts, this is the most comprehensive and up-to-date coverage of the security and privacy issues surrounding Industry 4.0 applications, a must-have for any library.

CYBER SECURITY AND DIGITAL FORENSICS: Challenges and Future Trends, Edited by Mangesh M. Ghonge, Sabyasachi Pramanik, Ramchandra Mangrulkar, and Dac-Nhuong Le, ISBN: 9781119795636. Written and edited by a team of world renowned experts in the field, this groundbreaking new volume covers key technical topics and gives readers a comprehensive understanding of the latest research findings in cyber security and digital forensics.

DEEP LEARNING APPROACHES TO CLOUD SECURITY, edited by Pramod Singh Rathore, Vishal Dutt, Rashmi Agrawal, Satya Murthy Sasubilli, and Srinivasa Rao Swarna, ISBN 9781119760528. Covering one of the most important subjects to our society today, this editorial team delves into solutions taken from evolving deep learning approaches, solutions allow computers to learn from experience and understand the world in terms of a hierarchy of concepts.

MACHINE LEARNING TECHNIQUES AND ANALYTICS FOR CLOUD SECURITY, Edited by Rajdeep Chakraborty, Anupam Ghosh and Jyotsna Kumar Mandal, ISBN: 9781119762256. This book covers new methods, surveys, case studies, and policy with almost all machine learning techniques and analytics for cloud security solutions.

SECURITY DESIGNS FOR THE CLOUD, IOT AND SOCIAL NETWORKING, Edited by Dac-Nhuong Le, Chintin Bhatt and Mani Madhukar, ISBN: 9781119592266. The book provides cutting-edge research that delivers insights into the tools, opportunities, novel strategies, techniques, and challenges for handling security issues in cloud computing, Internet of Things and social networking.

DESIGN AND ANALYSIS OF SECURITY PROTOCOLS FOR COMMUNICATION, Edited by Dinesh Goyal, S. Balamurugan, Sheng-Lung Peng and O.P. Verma, ISBN: 9781119555643. The book combines analysis and comparison of various security protocols such as HTTP, SMTP, RTP, RTCP, FTP, UDP for mobile or multimedia streaming security protocol.

SMART GRIDS FOR SMART CITIES VOLUME 2: Real-Time Applications in Smart Cities, Edited by O.V. Gnana Swathika, K. Karthikeyan, and Sanjeevikumar Padmanaban, ISBN: 9781394215874. Written and edited by a team of experts in the field, this second volume in a two-volume set focuses on an interdisciplinary perspective on the financial, environmental, and other benefits of smart grid technologies and solutions for smart cities.

SMART GRIDS AND INTERNET OF THINGS, Edited by Sanjeevikumar Padmanaban, Jens Bo Holm-Nielsen, Rajesh Kumar Dhanaraj, Malathy Sathyamoorthy, and Balamurugan Balusamy, ISBN: 9781119812449. Written and edited by a team of international professionals, this groundbreaking new volume covers the latest technologies in automation, tracking, energy distribution and consumption of Internet of Things (IoT) devices with smart grids.