

Computational intelligence for signal and image processing, volume II

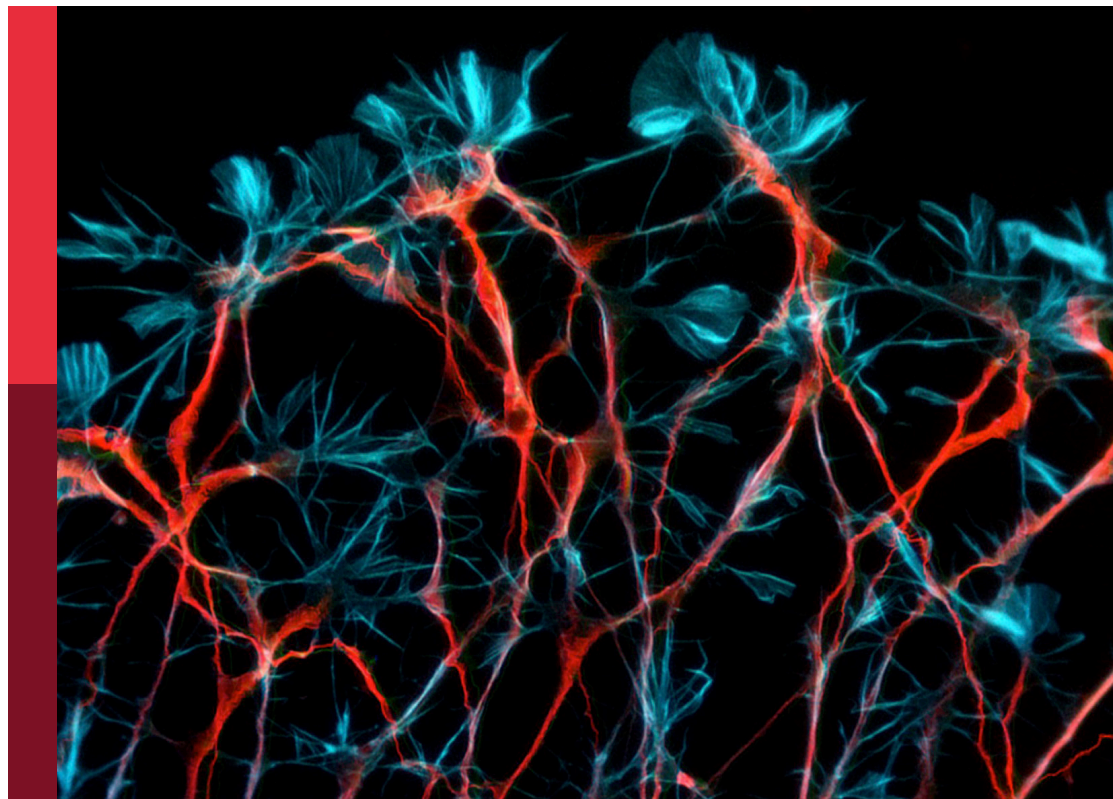
Edited by

Deepika Koundal and Jussi Tohka

Published in

Frontiers in Computational Neuroscience

Frontiers in Artificial Intelligence



FRONTIERS EBOOK COPYRIGHT STATEMENT

The copyright in the text of individual articles in this ebook is the property of their respective authors or their respective institutions or funders. The copyright in graphics and images within each article may be subject to copyright of other parties. In both cases this is subject to a license granted to Frontiers.

The compilation of articles constituting this ebook is the property of Frontiers.

Each article within this ebook, and the ebook itself, are published under the most recent version of the Creative Commons CC-BY licence. The version current at the date of publication of this ebook is CC-BY 4.0. If the CC-BY licence is updated, the licence granted by Frontiers is automatically updated to the new version.

When exercising any right under the CC-BY licence, Frontiers must be attributed as the original publisher of the article or ebook, as applicable.

Authors have the responsibility of ensuring that any graphics or other materials which are the property of others may be included in the CC-BY licence, but this should be checked before relying on the CC-BY licence to reproduce those materials. Any copyright notices relating to those materials must be complied with.

Copyright and source acknowledgement notices may not be removed and must be displayed in any copy, derivative work or partial copy which includes the elements in question.

All copyright, and all rights therein, are protected by national and international copyright laws. The above represents a summary only. For further information please read Frontiers' Conditions for Website Use and Copyright Statement, and the applicable CC-BY licence.

ISSN 1664-8714
ISBN 978-2-8325-6157-7
DOI 10.3389/978-2-8325-6157-7

About Frontiers

Frontiers is more than just an open access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers journal series

The Frontiers journal series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the *Frontiers journal series* operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews. Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the *Frontiers journals series*: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area.

Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers editorial office: frontiersin.org/about/contact

Computational intelligence for signal and image processing, volume II

Topic editors

Deepika Koundal — University of Eastern Finland, Finland

Jussi Tohka — University of Eastern Finland, Finland

Citation

Koundal, D., Tohka, J., eds. (2025). *Computational intelligence for signal and image processing, volume II*. Lausanne: Frontiers Media SA.
doi: 10.3389/978-2-8325-6157-7

Table of contents

- 04 **Editorial: Computational intelligence for signal and image processing, volume II**
Deepika Koundal and Jussi Tohka
- 07 **Prediction of emotion distribution of images based on weighted K -nearest neighbor-attention mechanism**
Kai Cheng
- 21 **Prediction of Alzheimer's disease stages based on ResNet-Self-attention architecture with Bayesian optimization and best features selection**
Nabeela Yaqoob, Muhammad Attique Khan, Saleha Masood, Hussain Mobarak Albarakati, Ameer Hamza, Fatimah Alhayan, Leila Jamel and Anum Masood
- 37 **Semi-supervised active learning using convolutional auto- encoder and contrastive learning**
Hezi Roda and Amir B. Geva
- 51 **An improved Dijkstra cross-plane image encryption algorithm based on a chaotic system**
Pijun Hou, Yuepeng Wang, Ziming Shi and Pan Zheng
- 69 **A novel multi-feature fusion attention neural network for the recognition of epileptic EEG signals**
Congshan Sun, Cong Xu, Hongwei Li, Hongjian Bo, Lin Ma and Haifeng Li
- 81 **Wearable sensors based on artificial intelligence models for human activity recognition**
Mohammed Alarfaj, Azzam Al Madini, Ahmed Alsafran, Mohammed Farag, Slim Chtourou, Ahmed Afifi, Ayaz Ahmad, Osama Al Rubayyi, Ali Al Harbi and Mustafa Al Thunaian
- 98 **Facial emotion recognition using deep quantum and advanced transfer learning mechanism**
Shtwai Alsubai, Abdullah Alqahtani, Abed Alanazi, Mohemmed Sha and Abdu Gumaei
- 117 **FacialNet: facial emotion recognition for mental health analysis using UNet segmentation with transfer learning model**
In-seop Na, Asma Aldrees, Abeer Hakeem, Linda Mohaisen, Muhammad Umer, Dina Abdulaziz AlHammadi, Shtwai Alsubai, Nisreen Innab and Imran Ashraf
- 130 **Motion feature extraction using magnocellular-inspired spiking neural networks for drone detection**
Jiayi Zheng, Yaping Wan, Xin Yang, Hua Zhong, Minghua Du and Gang Wang



OPEN ACCESS

EDITED AND REVIEWED BY
Si Wu,
Peking University, China

*CORRESPONDENCE

Jussi Tohka
✉ jussi.tohka@uef.fi

RECEIVED 21 February 2025
ACCEPTED 26 February 2025
PUBLISHED 11 March 2025

CITATION

Koundal D and Tohka J (2025) Editorial:
Computational intelligence for signal and
image processing, volume II.
Front. Comput. Neurosci. 19:1581047.
doi: 10.3389/fncom.2025.1581047

COPYRIGHT

© 2025 Koundal and Tohka. This is an
open-access article distributed under the
terms of the [Creative Commons Attribution
License \(CC BY\)](#). The use, distribution or
reproduction in other forums is permitted,
provided the original author(s) and the
copyright owner(s) are credited and that the
original publication in this journal is cited, in
accordance with accepted academic practice.
No use, distribution or reproduction is
permitted which does not comply with these
terms.

Editorial: Computational intelligence for signal and image processing, volume II

Deepika Koundal^{1,2} and Jussi Tohka^{2*}

¹School of Computer Science, UPES, Dehradun, India, ²A.I. Virtanen Institute for Molecular Sciences, University of Eastern Finland, Kuopio, Finland

KEYWORDS

computational intelligence, signal processing, image processing, artificial intelligence, machine learning

Editorial on the Research Topic

Computational intelligence for signal and image processing, volume II

1 Introduction

The second volume of this Research Topic features eight research articles that explore the use of Computational Intelligence in Signal and Image Processing II applications (Koundal and Ding, 2023; Wang et al., 2023). This edition explored brain-inspired algorithms and examined how they have driven the development of new methodologies in image/video and signal processing (Pan et al., 2024; Saikumar et al., 2022). It emphasized the significant potential of brain-inspired algorithms to transform multiple fields to drive innovation and enhance efficiency. Advancements in Artificial Intelligence (AI) and machine learning have significantly impacted on a variety of fields, ranging from healthcare and emotion recognition (Bing et al., 2024; Xia et al., 2023; Zhu, 2023) to image encryption (Chu et al., 2024) and human activity classification. By highlighting the interconnectedness of deep learning, neuro-fuzzy systems, neural networks, and other AI methods, it underscored their essential role in understanding and modeling the complexities of brain functions (Ye et al., 2024; Wen et al., 2024; Hao et al., 2023). This work connected neuroscience and technology by examining how brain insights can inspire the creation of novel algorithms and applications across diverse fields (Zhu, 2024; Hu et al., 2022). These innovations represent a significant leap forward in their respective domains, offering practical solutions with potential for real-world applications (Gezawa et al., 2023; Song et al., 2025).

2 Contributions

Cheng introduced a weighted closest neighbor algorithm to predict emotional distribution in abstract paintings. Emotional features have been extracted and assigned K-values that are followed by an encoder-decoder model that utilized a pre-trained network to enhance classification. Incorporating a blank attention mechanism, the model accurately identified emotional semantics- outperforming existing methods. This approach addressed the challenge of emotion recognition in abstract art. However, limitations are positional link detection and dataset constraints which suggest future expansion for more comprehensive classification. Sun et al. introduced a classification system for

epileptic electroencephalography (EEG) signals using an attention network that has integrated nonlinear dynamic and time-frequency features. The system consisted of three modules: a parallel convolutional network for high-resolution Hilbert spectrum extraction, a residual-connected convolution module for nonlinear dynamic feature learning via grayscale recurrence plots, and a self-attention fusion module. The given system significantly improved the classification accuracy on multiple EEG databases that offered a promising approach to aid epilepsy diagnosis and treatment with broad clinical applications. [Yaqoob et al.](#) introduced an automated framework for Alzheimer's disease (AD) stage prediction using a Fuzzy Entropy-controlled Path-Finding Algorithm (FEcPFA) and ResNet-Self architecture. This method addressed dataset imbalance through data augmentation, incorporated a self-attention module to extract key information and Bayesian optimization (BO) to optimize hyperparameters. This framework improved the diagnosis accuracy, reduced computational time, and offered potential for early AD detection, though challenges like overfitting remain. Future improvements include using more diverse MRI datasets. [Hou et al.](#) developed an improved Dijkstra-based image encryption algorithm for color images that addressed the inefficiencies of traditional methods to treat color planes separately. Their approach integrated a new 1D chaotic system with enhanced randomness and an adaptive diffusion algorithm. The Dijkstra algorithm is used for cross-plane pixel scrambling to ensure better security and encryption efficiency. This method provided robust encryption for both medical images and standard RGB images by outperforming existing techniques in terms of security, quality, and robust to attacks especially in telemedicine applications. [Roda and Geva](#) introduced a pool-based semi-supervised active learning method for image classification using both labeled and unlabeled data. The approach involved clustering the latent space of a pre-trained convolutional autoencoder and applied a novel contrastive clustering loss (CCL) to enhance clustering even with limited labeled data. The system queries the most uncertain samples for annotation by iterating until the budget is exhausted. Empirical results show high accuracy with fewer labeled samples by offering an effective solution for image classification tasks with reduced annotation costs. [Alarfaj et al.](#) proposed a novel human activity recognition (HAR) approach using sensor-specific convolutional neural networks (CNNs) for accelerometers, gyroscopes, and barometers. Each CNN model is tailored to capture the unique features of its sensor type by addressing challenges with diverse data shapes. A late-fusion technique combined predictions from multiple models to significantly improving accuracy. [Alsubai et al.](#) introduced a facial emotion recognition system using a Modified ResNet model enhanced with quantum computing and advanced transfer learning. By integrating quantum convolutional layers with parameterized filters and employing residual connections, the system reduced the computation time and improved performance. The Modified up Sampled Bottle Neck Process (MuS-BNP) ensured computational efficiency. The model achieved superior accuracy, recall, precision and F1-score by overcoming challenges in distinguishing similar facial expressions. The results highlighted the system's potential for faster, more accurate facial emotion detection, using quantum computing and deep learning. [Na et al.](#) introduced FacialNet, a framework for facial emotion recognition

(FER) using UNet image segmentation and transfer learning with EfficientNetB4. The approach is validated through cross-validation by offering high reliability and promised real-world applications in emotion-aware systems to enhance mental health assessments through more accurate emotion recognition. [Zheng et al.](#) introduced the Visual-Magnocellular Dynamics Dataset (VMD) with a multi-frame spike temporal encoding strategy to enhance dynamic visual information processing. They proposed the DT-MSTS backpropagation method for improved motion feature extraction in SNNs. Additionally, they integrated MG-SNN with YOLO to develop a retinal-inspired neural network for drone motion extraction and object detection. The study highlights the benefits of combining retinal mechanisms with SNNs, explores software-based deployment of neuromorphic chips, and suggests future directions for handling complex spatiotemporal data in real-world detection tasks.

3 Conclusion

This editorial presented nine research articles focused on the applications of Computational Intelligence for Signal and Image Processing. The studies highlighted significant advancements across various fields, from emotion recognition in abstract art to medical applications. Techniques such as deep learning, transfer learning, and quantum computing have shown great potential in improving accuracy, efficiency, and security. Despite their successes, challenges like dataset limitations, overfitting, and computational time remain. Future work includes expanding datasets, and refining models to enhance applicability in real-world settings across domains such as healthcare and mental health.

Author contributions

DK: Formal analysis, Methodology, Writing – original draft. JT: Writing – review & editing, Supervision.

Acknowledgments

We sincerely thank the authors for their significant contributions to this Research Topic. We also express our gratitude to the reviewers for their thorough and timely evaluations, which greatly enhanced the quality of this publication. Finally, we acknowledge the continuous support from the editorial team of Frontiers in Computational Neuroscience, whose dedication was crucial in the successful completion of this Research Topic.

Conflict of interest

The authors declare that the research was conducted without any commercial or financial interests that could be perceived as a potential conflict of interest.

The author(s) declared that they were an editorial board member of Frontiers, at the time of submission. This had no impact on the peer review process and the final decision.

Generative AI statement

The author(s) declare that Gen AI was used in the creation of this manuscript. The author(s) have used ChatGPT (<https://chat.openai.com/>) to rephrase individual sentences for increasing clarity. After using this tool/service, the author(s) reviewed and edited the content as needed and take full responsibility for the content of the publication.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Bing, P., Liu, W., Zhai, Z., Li, J., Guo, Z., Xiang, Y., et al. (2024). A novel approach for denoising electrocardiogram signals to detect cardiovascular diseases using an efficient hybrid scheme. *Front. Cardiovasc. Med.* 11, 1277123. doi: 10.3389/fcvm.2024.1277123
- Chu, L., Su, Y., Zan, X., Lin, W., Yao, X., Xu, P., et al. (2024). A deniable encryption method for modulation-based DNA storage. *Interdiscipl. Sci.* 16, 872–881. doi: 10.1007/s12539-024-00648-5
- Gezawa, A. S., Liu, C., Jia, H., Nanekaran, Y. A., Almutairi, M. S., and Chiroma, H. (2023). An improved fused feature residual network for 3D point cloud data. *Front. Comput. Neurosci.* 17, 1204445. doi: 10.3389/fncom.2023.1204445
- Hao, S., Jiali, P., Xiaomin, Z., Xiaoqin, W., Lina, L., Xin, Q., et al. (2023). Group identity modulates bidding behavior in repeated lottery contest: neural signatures from event-related potentials and electroencephalography oscillations. *Front. Neurosci.* 17:1184601. doi: 10.3389/fnins.2023.1184601
- Hu, F., Qiu, L., and Zhou, H. (2022). Medical device product innovation choices in Asia: an empirical analysis based on product space. *Front. Public Health* 10:871575. doi: 10.3389/fpubh.2022.871575
- Koundal, D., and Ding, B. (2023). Computational intelligence for signal and image processing. *Front. Comput. Neurosci.* 17:1284600. doi: 10.3389/fncom.2023.1284600
- Pan, H., Wang, Y., Li, Z., Chu, X., Teng, B., and Gao, H. (2024). A complete scheme for multi-character classification using EEG signals from speech imagery. *IEEE Trans. Biomed. Eng.* 71, 2454–2462. doi: 10.1109/TBME.2024.3376603
- Saikumar, K., Rajesh, V., Srivastava, G., and Lin, J. C. W. (2022). Heart disease detection based on internet of things data using linear quadratic discriminant analysis and a deep graph convolutional neural network. *Front. Comput. Neurosci.* 16:964686. doi: 10.3389/fncom.2022.964686
- Song, W., Ye, Z., Sun, M., Hou, X., Li, S., and Hao, A. (2025). AttrDiffuser: adversarially enhanced diffusion model for text-to-facial attribute image synthesis. *Pattern Recognit.* 2025:111447. doi: 10.1016/j.patcog.2025.111447
- Wang, W., Yan, D., Wu, X., He, W., Chen, Z., Yuan, X., et al. (2023). Low-light image enhancement based on virtual exposure. *Signal Proc. Image Commun.* 118:117016. doi: 10.1016/j.image.2023.117016
- Wen, H., Zhong, Y., Yao, L., and Wang, Y. (2024). Neural correlates of motor/tactile imagery and tactile sensation in a BCI paradigm: a high-density EEG source imaging study. *Cyborg Bionic Syst.* 5:0118. doi: 10.34133/cbsystems.0118
- Xia, J., Cai, Z., Heidari, A. A., Ye, Y., Chen, H., and Pan, Z. (2023). Enhanced moth-flame optimizer with quasi-reflection and refraction learning with application to image segmentation and medical diagnosis. *Curr. Bioinform.* 18, 109–142. doi: 10.2174/1574893617666220920102401
- Ye, W., Wang, J., Chen, L., Dai, L., Sun, Z., and Liang, Z. (2024). Adaptive spatial-temporal aware graph learning for EEG-based emotion recognition. *Cyborg Bionic Syst.* 5:88. doi: 10.34133/cbsystems.0088
- Zhu, C. (2023). Research on emotion recognition-based smart assistant system: emotional intelligence and personalized services. *J. Syst. Managem. Sci.* 13, 227–242. doi: 10.33168/JSMS.2023.0515
- Zhu, C. (2024). Computational intelligence-based classification system for the diagnosis of memory impairment in psychoactive substance users. *J. Cloud Comp.* 13:119. doi: 10.1186/s13677-024-00675-z



OPEN ACCESS

EDITED BY

Deepika Koundal,
University of Petroleum and Energy
Studies, India

REVIEWED BY

Arvind Dhaka,
Manipal University Jaipur, India
Mohit Mittal,
Institut National de Recherche en
Informatique et en Automatique
(INRIA), France
Tariq Ahmad,
Guilin University of Electronic
Technology, China

*CORRESPONDENCE

Kai Cheng
✉ chengkai7300@163.com

RECEIVED 06 December 2023

ACCEPTED 28 March 2024

PUBLISHED 17 April 2024

CITATION

Cheng K (2024) Prediction of emotion
distribution of images based on weighted
 K -nearest neighbor-attention mechanism.
Front. Comput. Neurosci. 18:1350916.
doi: 10.3389/fncom.2024.1350916

COPYRIGHT

© 2024 Cheng. This is an open-access article
distributed under the terms of the [Creative
Commons Attribution License \(CC BY\)](#). The
use, distribution or reproduction in other
forums is permitted, provided the original
author(s) and the copyright owner(s) are
credited and that the original publication in
this journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

Prediction of emotion distribution of images based on weighted K -nearest neighbor-attention mechanism

Kai Cheng*

School of Artificial Intelligence, Xidian University, Xi'an, China

Existing methods for classifying image emotions often overlook the subjective impact emotions evoke in observers, focusing primarily on emotion categories. However, this approach falls short in meeting practical needs as it neglects the nuanced emotional responses captured within an image. This study proposes a novel approach employing the weighted closest neighbor algorithm to predict the discrete distribution of emotion in abstract paintings. Initially, emotional features are extracted from the images and assigned varying K -values. Subsequently, an encoder-decoder architecture is utilized to derive sentiment features from abstract paintings, augmented by a pre-trained model to enhance classification model generalization and convergence speed. By incorporating a blank attention mechanism into the decoder and integrating it with the encoder's output sequence, the semantics of abstract painting images are learned, facilitating precise and sensible emotional understanding. Experimental results demonstrate that the classification algorithm, utilizing the attention mechanism, achieves a higher accuracy of 80.7% compared to current methods. This innovative approach successfully addresses the intricate challenge of discerning emotions in abstract paintings, underscoring the significance of considering subjective emotional responses in image classification. The integration of advanced techniques such as weighted closest neighbor algorithm and attention mechanisms holds promise for enhancing the comprehension and classification of emotional content in visual art.

KEYWORDS

image emotions, classification, weighted closest neighbor algorithm, emotional features, abstract paintings

1 Introduction

Image data are essentially used for transferring information. The amount of picture data is even increasing at an exponential speed owing to the advent of the Internet (Cetinic and She, 2022; Zou et al., 2023). Because of the fast-paced nature of modern society, people's ability to extract information from photos is also accelerating, necessitating more accuracy and efficiency in identifying image data on the network. Based on this necessity, an effective image processing technique that makes use of computer vision is required for humans to manage and use picture data more effectively.

Sentiment analysis, often called opinion mining, is the process of using natural language processing, text analysis, computational linguistics, and biometrics to systematically unpack subjective information and emotional states. The notion was initially introduced by Yang et al. (2023). Sentiment analysis has gained significant economic and societal significance in the last several years and has been applied extensively in the domains of opinion monitoring (Chen et al., 2023), topic inference (Ngai et al., 2022), and comment analysis and decision-making (Bharadiya, 2023). For monitoring public opinion, the government can make timely policy interventions and accurately determine the direction of public opinion. When it comes to product recommendations, merchants can better understand user needs and suggestions by gauging user satisfaction with product evaluations and enhancing product quality. In the finance domain, trending financial topics can even be used to predict stock direction. Furthermore, sentiment analysis is frequently used for various tasks involving natural language processing. To increase the accuracy of the system, more exact terms for sentiment expression are chosen for machine translation (Chan et al., 2023) by evaluating the sentiment tendency of the input text. The pixel density extraction of the image information is shown in Figure 1.

Various classification techniques will be broken down into different levels for the sentiment analysis task: output results will categorize the methods into sentiment intensity classification and sentiment polarity classification; granularity of the processed text will divide them into three research levels: word level, sentence level, and chapter level; research methodology will separate them into unsupervised learning, semi-supervised learning, and supervised learning, and so on. The majority of the conventional sentiment classification algorithms employ manually created feature selection techniques for feature extraction, such as the maximum entropy model (Chandrasekaran et al., 2022), plain Bayes (Wang et al., 2022), support vector machines (Zhao et al., 2021a), and so on. However, these techniques have limitations, such as being labor-intensive, time-consuming, and hard to train. As a result, they are not well-suited for use in the current large-scale application scenarios.

With advancements in machine learning, research efforts (Milani and Fraternali, 2021) led to the development of deep learning methods that give neural networks a hierarchical structure. This development subsequently resulted in an explosion of deep learning research. Feature learning, at the heart of deep learning, uses hierarchical networks to convert unprocessed input into more abstract and higher-level feature information. With its superior learning capacity to optimize automated feature extraction, deep learning has produced remarkable research achievements in recent years in the domains of speech recognition, picture processing, and natural language processing. The application of deep learning techniques to text sentiment analysis has gained popularity as a natural language processing study area. Among these techniques, Song et al. (2021) used a convolutional neural network to classify text emotion for the first time, and the results were superior to those of conventional machine learning techniques.

The study of human eyesight is where attention mechanism first emerged. According to cognitive science, humans have a tendency to ignore other observable information in favor of focusing on a certain portion of the information based on the

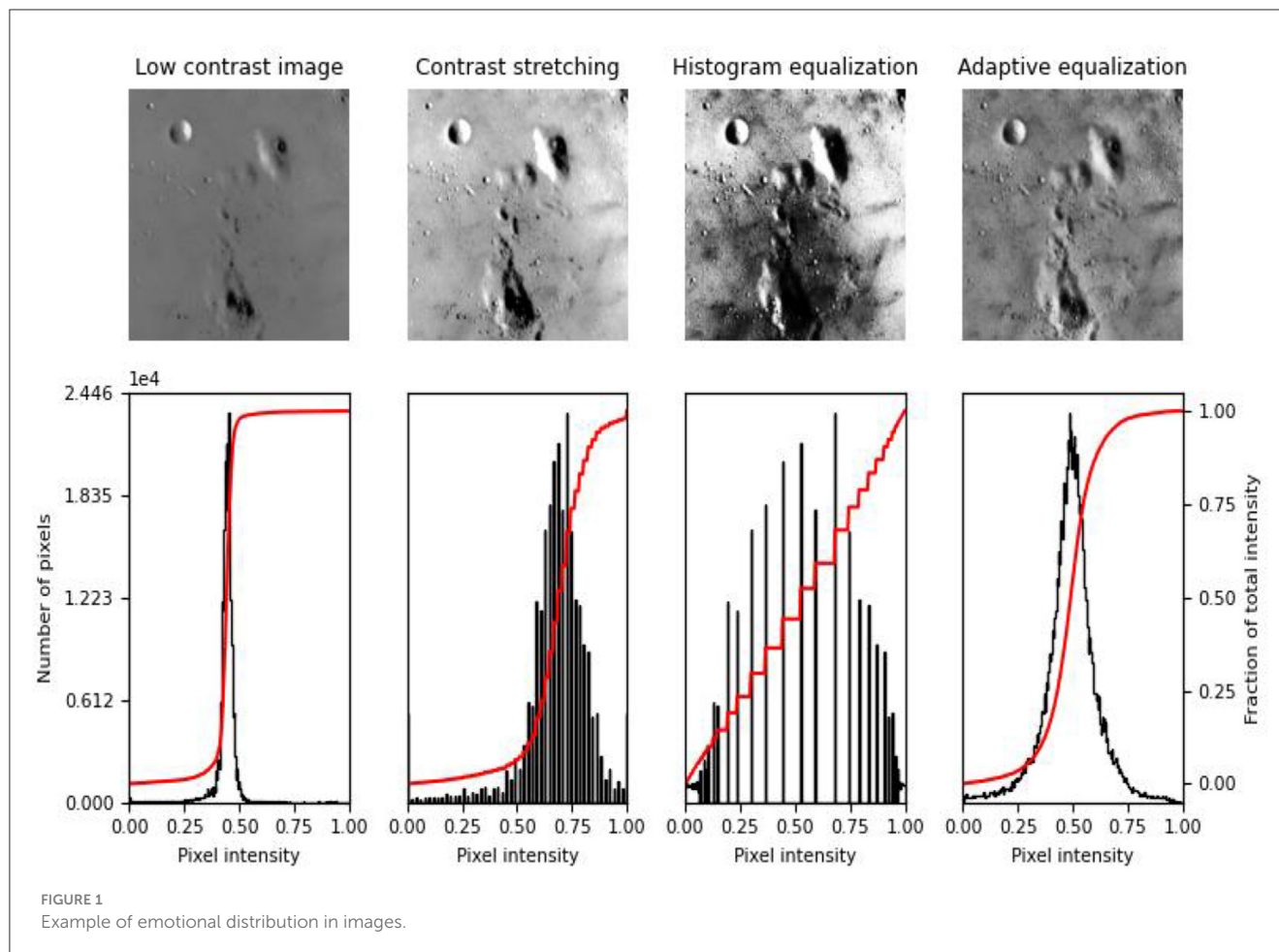
demand imposed by the information processing bottleneck. The primary objective of attention mechanism is to efficiently separate valuable information from a vast quantity of data. To understand the word dependencies inside the phrase and grasp the internal structure of the sentence, the self-attention mechanism—a unique form of attention mechanism—is incorporated into the sentiment classification job. To establish an accurate and efficient technique for sentiment analysis based on deep learning technology and self-attention mechanism, this study examines the present technical issues in the field of sentiment analysis from the standpoint of the real demands of sentiment analysis.

2 Related studies

Natural language processing has attracted extensive research attention (McCormack and Lomas, 2021) because it introduced the idea of sentiment analysis. There are three prominent methods for conducting sentiment analysis at present: the sentiment dictionary approach, the classical machine learning approach, and the deep learning approach.

Experts must annotate the sentiment polarity of the text's terms in order for researchers to perform sentiment analysis based on sentiment dictionary. Based on semantic rules and sentiment dictionary, researchers compute the text's sentiment score and determine the sentiment tendency. Among these researchers, Toisoul et al. (2021) demonstrated positive findings on a multi-domain dataset by expanding the domain-specific vocabulary by extracting subject terms from the corpus using latent Dirichlet allocation (LDA) modeling based on the pre-existing sentiment lexicon. Peng et al. (2022) used the point mutual information (PMI) technique to assess the similarity of adjectives in WordNet. The polar semantics (ISA) approach was then used to generate numerous fixed sentence constructions in order to examine the target text sentiment tendency. To create a Chinese microblogging sentiment dictionary, Liu et al. (2021a) first identified microblogging sentences using information entropy and then filtered network sentiment terms using the sentiment-oriented pointwise mutual information (SO-PMI) method.

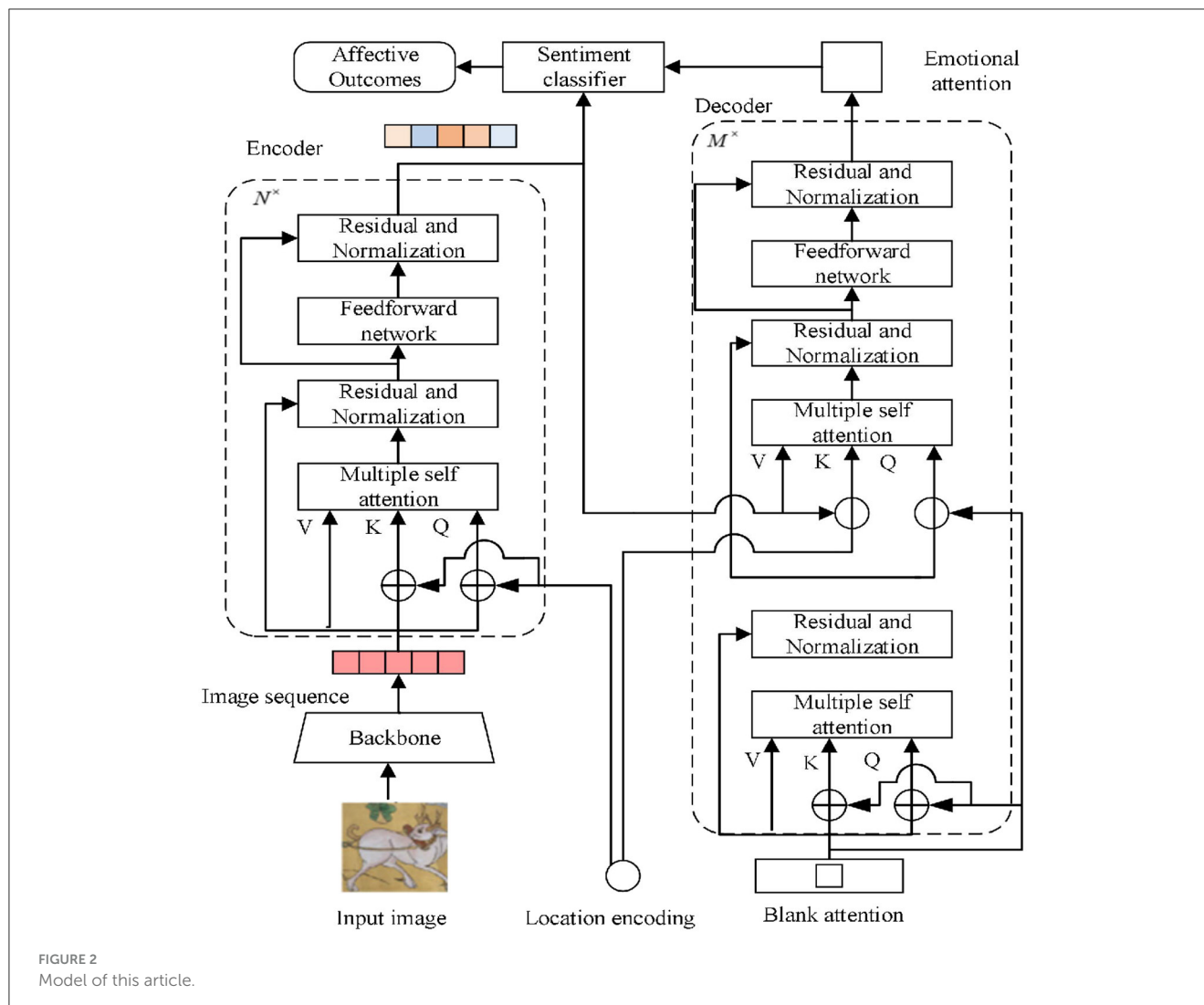
Ding et al. (2021) introduced the idea of the primary word and used weight priority calculations to determine the text's semantic inclination degree. These developments paved the way for accomplishing more difficult sentiment analysis tasks. The approach based on sentiment dictionary has the benefit of being more accurate in classifying text at the word or phrase level. However, the system migration is not good, and the sentiment dictionaries are often geared to certain domains. These days, one of the most popular techniques for sentiment analysis is classical machine learning-based techniques. Using simple bag-of-words features from a movie review dataset, Yang et al. (2021) was the first to use machine learning techniques to the sentiment binary classification issue and produced superior experimental outcomes. Utilizing Twitter comments as test data, Roy et al. (2023) classified emotions into six categories—happiness, sadness, disgust, fear, surprise, and anger—and employed plain Bayes for text sentiment analysis. The data were processed with consideration for lexical and expression features, leading to a high classification accuracy. To



address the sentiment classification problem, [Sahoo et al. \(2021\)](#) merged a genetic algorithm with simple Bayes, and the results of the experiments indicated that the combined model outperformed the individual models. To extract rich sentiment data and include them in the basic feature model, [Liu et al. \(2021b\)](#) used machine learning techniques with numerous rules, which increased the classification result in microblog sentiment classification trials. In order to complete the study of sentiment analysis, [Sampath et al. \(2021\)](#) included semantic rules into the support vector machine model. The experiment confirmed that the support vector machine model with the inclusion of semantic rules performed better in the sentiment classification task. Deeper text semantic information is hard to learn, even while machine learning-based techniques enhance the sentiment classification performance and lower the reliance on sentiment lexicon.

Text sentiment analysis based on deep learning has garnered a much interest from academics at both national and international levels due to its superior performance in the fields of picture processing and natural language processing. [Zhang et al. \(2023\)](#) used deep neural network training to create the Collobert and Weston (C&W) model, which was then used to perform well on natural language processing tasks including sentiment classification and lexical annotation. To demonstrate the efficacy of single-layer convolutional neural networks (CNNs) in sentiment classification tasks, [Zhao et al. \(2021b\)](#) combined different sizes of convolutional

kernels with maximum pooling and performed comparison tests on seven datasets. The study employed convolutional neural networks for sentiment analysis tasks. A number of recurrent neural networks, including recurrent neural network (RNN), multiplicative RNN (MRNN), recursive neural tensor network (RNTN), and others, were progressively suggested by [Szubielska et al. \(2021\)](#). The RNTN model, for example, uses a syntactic analysis tree to determine word sentiment and then outputs the sentence's sentiment classification result in the form of word sentiment summation. To tackle the sentiment analysis problem utilizing a long short-term memory (LSTM) network with an expanded gate structure, which increases the model's flexibility, [Li et al. \(2022\)](#) employed Twitter comments as the experimental data. RNNs were utilized by [Zhou et al. \(2023\)](#) to model texts by taking into account their temporal information. [Li et al. \(2023\)](#) achieved outstanding results in a sentiment classification test by modeling utterances using a tree LSTM model to approximate the sentence structure. By segmenting a text according to sentences, obtaining vectors through convolutional pooling operation, and then inputting them into LSTM according to temporal relations to construct a CNN-LSTM model and apply it to the task of sentiment analysis, [Alirezazadeh et al. \(2023\)](#) primarily addressed the issue of temporal and long-range dependencies in a chapter-level text. [Teodoro et al. \(2023\)](#) constructed an experimental minimal convolutional neural



network (EMCNN) model using microblog comments as the experimental data, combining lexical and emoji characteristics. The model produced experimental findings that outperformed the benchmark model's performance.

3 Attention given

We propose an emotion classification method based on the attention mechanism that sets blank attention in the decoder and fuses the output sequence of the encoder to learn the image semantics to guide the model to learn the image emotion more accurately and reasonably via the learning mechanism of the decoder. This method is intended to address the characteristics of small numbers of abstract painting samples and rich image semantics. Figure 2 depicts the general flowchart of the procedure used in this article, along with the encoder-decoder architecture, the emotion classification module, and the backbone network for extracting picture feature sequences.

3.1 Image sequence generation

Since the encoder anticipates a sequence as input, the abstract painting dataset in this study has been uniformly normalized, meaning that its length and width are 224 and its number of channels is 3. To extract the image's features, the image is supplied into the backbone network. The residual network has a strong feature learning ability and adapts to the characteristics of the backbone convolutional network architecture. In this study, ResNet-50 is adopted as the backbone network to solve the network degradation problem brought by fewer samples of abstract paintings to simplify the model training parameters of this article to a certain extent, improve the training efficiency, and carry out comparative experiments with the residual network variant in the ablation experiments, and to assess the influence of the backbone network on the model accuracy rate (Ahmad et al., 2023). The abstract painting dataset is generated by the backbone network to generate canonical image features with a length and width of 7 and a channel count of 256 and is spread into a one-dimensional sequence, resulting in an image sequence of length 49 and a channel count of 256 to be fed to the encoder.

3.2 Encoders

By adjusting the number of encoder layers, the model demonstrates the significance of global image-level self-attention, guarantees that there is no appreciable loss of accuracy when $N = 6$, and prevents an increase in training difficulty brought on by the addition of too many parameters. This article adopts the position coding method of detection transformer (DETR), which uses the sine and cosine functions to encode the positions of rows and columns of the parity channel of the abstract painting feature map, adapting to the sequence input of the encoder–decoder architecture (Ahmad and Wu, 2023). The encoder–decoder architecture is not sensitive to the order of the image sequence and does not have the ability to learn the sequence position information. The calculation for the position coding as shown in Equation (1):

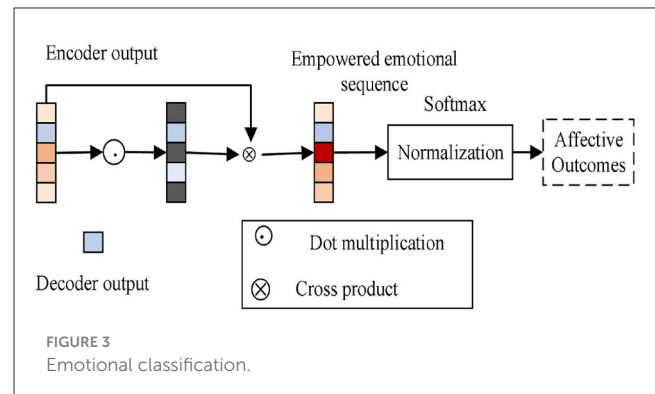
$$f(x)^i = \begin{cases} \sin\left(\frac{x}{10000^{i/128}}\right), i = 2k (k \in [0, 127]) \\ \cos\left(\frac{x}{10000^{i/28}}\right), i = 2k + 1 (k \in [0, 127]) \end{cases} \quad (1)$$

where x is the row and column spread value of point (p, q) and i is the channel of the feature map. For a feature map with a length and width of 7 and a channel count of 256, respectively, the row and column position encoding on the point with a channel of 10 and a coordinate value of (1,2) is $\sin [((1 \times 7) + 2)/(1,000,010/128)]$ and $\sin [((2 \times 7) + 1)/(1,000,010/128)]$, respectively, and the position encoding of the remaining image sequences of the channels is computed by this rule. Encoding finally generates a one-dimensional feature sequence with a length of 49 and a channel count of 256 with position information.

The Q, K, V in the encoder is a one-dimensional sequence of a fixed length of 49 and a channel count of 256, which is used as sentiment weights in translating the image sequence and ordering its position in each encoding session. As the model learns the feature dependencies between image sequences, the multi-head self-attention module supports the model by reinforcing the original features with sequence global information. This support enables the model to learn discriminative features for sentiment classification. The original image sequence serves as the input for the first coding layer, and the input for each succeeding layer is the image sequence encoded in the preceding layer. The picture feature sequences are given to the decoder after being encoded and learned by many coding layers of the encoder, avoiding the issue of delayed network convergence and poorer accuracy brought on by the increased depth of the model.

3.3 Decoders

The blank attention in this study has the same format as the feature sequence of the model input, that is, a sequence with a fixed length of 49 and a channel count of 256. Similar to the encoding phase, the blank attention is weighted as a query statement with Q, K, V of the first self-attention layer in the decoder, but at this point, the blank attention does not need to focus on the location information. At each decoding stage, the multi-head attention module transforms the blank attention sequences and generates



the output of the attention sequences with weights by avoiding the problem of slower model convergence through the residuals and normalization module.

The attention sequence with weights from the upper layer and the output sequence from the encoder are fed into the second self-attention layer. This study uses the same sine and cosine functions in the decoder as in the encoder to encode the position of the weighted attention sequences from the upper layers since the output sequence of the encoder contains positional information and needs to accommodate its positional connection. The positional encoding of the picture sequence for each channel is calculated for the weighted attention sequence of length 49 and a channel count of 256. This positional encoding is applied to the rows and columns of the parity channels. Ultimately, a weighted attention sequence of length 49 and a channel count of 256 with position information are obtained. It is combined with the output sequence of the encoder as a query statement and weighted with Q, K, V from the second layer in the decoder. In each decoding stage, the output sequence of the encoder is translated, and the sequence positions are sorted.

3.4 Classification of emotions

Figure 3 depicts the emotion classification module. The sentiment classification module combines the output sequences of the encoder and decoder to produce weighted sentiment sequences, which suppress redundant sentiment information in the model, direct the model to concentrate on deep and shallow sentiment information, and improve the model's ability to classify sentiment. The fully connected layer is used to map the weighted sentiment sequences, and the cross-entropy loss is minimized to produce stable sentiment classification results (Ahmad et al., 2021). The normalized exponential function is used to calculate the probability value of each type of sentiment; the abstract painting sentiment predicted by the model has the highest probability value.

The normalized exponential function is as shown in Equation (2):

$$S_i = \frac{e^i}{\sum_{j=1}^{12} e^j} \quad (2)$$

where S_i is the normalized value of a particular sentiment and computation $\max(S_i)$ is the abstract painting sentiment label predicted by the model.

One popular loss function for handling classification difficulties is the cross-entropy function, which is primarily used to quantify the difference between two probability distributions. The cross-entropy loss function is as shown in Equation (3):

$$Loss = \frac{1}{batch_size} \sum_i \sum_{c=1}^{12} y_{ic} \log_2(p_{ic}) \quad (3)$$

Each term in the cross-entropy function is p and q and p indicates the true probability distribution and q represents the predicted probability distribution. The cross-entropy function describes the difference between the two probability distributions. For the special case, the cross-entropy function of the binary classification problem, there are a total of two terms, i.e., probability distributions of classes 0 and 1, and there is $p(0) = 1 - p(1)$, so we can get the expression for the binary classification cross entropy loss function, where y_{ic} is the true value and p_{ic} is the probability of the predicted value.

4 Image preprocessing

4.1 Datasets

The abstract dataset, which includes 280 abstract paintings, was created by Machajdik. These paintings are better suited for challenges requiring the prediction of emotion distribution because they simply feature colors and textures and not any clearly discernible objects. The 230 participants in the dataset expressed their emotions by identifying these 280 photographs, with an average of 14 people doing so. The final sentiment category is determined by which of these sentiment markers received the most votes. Due to the ambiguity of emotions, several categories may have extremely similar or identical numbers of votes, making the classification process unclear. Therefore, the ratio of votes for each emotion category is used as a probability distribution to form a probability distribution of emotions corresponding to the image, as shown in Figure 3.

4.2 Feature extraction

Since abstract paintings contain only colors and textures and do not generate emotions through specific objects, the features extracted are emotional features based on the theory of artistry.

4.2.1 Color histogram

Artists use colors to express or trigger different emotions in observers, and extracting color histograms from color features is a common and effective method. The color histogram space H is defined as Equation (4):

$$H = [h(0), h(1), \dots, h(L_k)], \sum_{k=1}^K h(L_k) = 1 \quad (4)$$

where $h(L_k)$ denotes the frequency of the k th color. The similarity of the color histograms of the two images are measured using the Euclidean distance as shown in Equation (5):

$$D(H_s, H_d) = [(H_s - H_d)^T (H_s - H_d)]^{1/2} \quad (5)$$

4.2.2 Itten comparison

Itten successfully used the strategy of color combination by defining seven contrast attributes. Machajdik used seven contrast attributes such as light and dark contrast, saturation contrast, extension contrast, complementary contrast, hue contrast, warm and cool contrast, and simultaneous contrast of images as the emotional characteristics of artistry theory.

As in the case of light and dark contrast, the image is segmented into R_1, R_2, \dots, R_N , small chunks using the watershed segmentation algorithm, and the average h_n (Chroma) b_n (Brightness) s_n (Saturation) is calculated for each chunk. Calculation b_n belongs to five fuzzy luminance: $\left\{ \begin{array}{l} \text{Very Dark(VD), Dark(D), middle (M),} \\ \text{Light(L), Very Light(VL)} \end{array} \right\}$ affiliation function as shown in Equations (6–10).

$$VD = \begin{cases} 1 & b_n \leq 21 \\ \frac{39-b_n}{18} & 21 < b_n \leq 39 \\ 0 & \end{cases} \quad (6)$$

$$D = \begin{cases} \frac{b_n-21}{18} & 21 < b_n \leq 39 \\ \frac{55-b_n}{16} & 39 < b_n \leq 55 \\ 0 & \end{cases} \quad (7)$$

$$M = \begin{cases} \frac{55-b_n}{16} & 39 < b_n \leq 55 \\ \frac{b_n-55}{13} & 55 < b_n \leq 68 \\ 0 & \end{cases} \quad (8)$$

$$L = \begin{cases} \frac{b_n-55}{13} & 55 < b_n \leq 68 \\ \frac{84-b_n}{16} & 68 < b_n \leq 84 \\ 0 & \end{cases} \quad (9)$$

$$VL = \begin{cases} \frac{84-b_n}{16} & 68 < b_n \leq 84 \\ 1 & b_n > 84 \\ 0 & \end{cases} \quad (10)$$

Thus, a 1×5 dimensional vector for each small block of image R_1, R_2, \dots, R_N is obtained, and for the whole image, the light/dark contrast is defined as Equation (11):

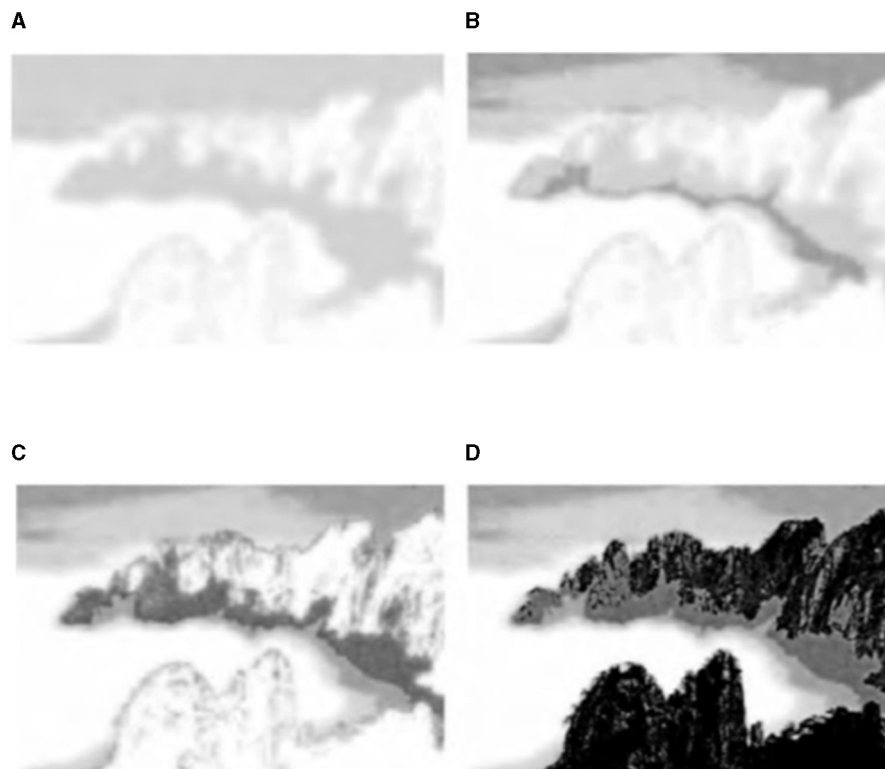


FIGURE 4
Layer stacking process effect and ink simulation results. **(A)** One grayscale layer overlay effect. **(B)** Two grayscale layer overlay effect. **(C)** Three grayscale layer overlay effect. **(D)** Seven grayscale layer overlay effect.

$$B(i) = \left[\frac{1}{\sum_{n=1}^N R_n} \sum_{n=1}^N R_n (B_n(i) - \bar{B}(i))^2 \right]^{1/2} \quad (11)$$

where $i = 1, \dots, 5$, R_n is the number of pixels in the split block.

In this way, the vector expression of the contrasting attributes of the images is obtained as features, and the similarity of the different images is calculated by the Euclidean distance.

The Itten model is also used to determine whether or not an image is harmonic, and it can also be used to identify an image's emotional expression. Select three to four of the image's prominent colors, connect them to the colors on the Itten hue wheel, and if they form a positive polygon, the image is harmonic. To determine the dominant chromaticity of an image, make a histogram of its N colors. Ignore the colors with a proportion of $<5\%$. The harmony of a polygon can be assessed by comparing its internal angles to those of a square polygon built from the same number of vertices.

4.2.3 Texture

The main idea behind the statistical approach to texture analysis is to symbolize textures by the randomness of the distribution of gray levels in a graph. We define z as a random variable representing the gray levels, L as the maximum gray level of the image, Z_i as the number of pixels with gray level i , 01 denotes the

gray level histogram, and with respect to z , the n th order moments are calculated as shown in Equation (12):

$$u_n(z) = \sum_{i=0}^L (Z_i - m)^n p(z_i) \quad (12)$$

$$m = \sum_{i=0}^L z_i p(z_i) \text{ is the mean value of } z.$$

The second-order moments are more important in texture description; it is a measure of grayscale contrast, where $R = \frac{1}{1+u_2(z)}$ indicates the smoothness of the image, and a smaller value of $u_n(z)$ corresponds to a smaller R value, indicating that the smaller the value of R , the smoother the image.

4.3 Weighted K -nearest neighbor sentiment distribution prediction algorithm

Assuming that there are M sentiment categories C_1, \dots, C_M and N training images, x_1, \dots, x_N (which also denote the corresponding features of the images) use $p = \{P_{n1}, \dots, P_{nm}, \dots, P_{nM}\}^T$ to denote the sentiment distribution of X_n , where P_{nm} denotes the probability that x_n expresses a sentiment of c_m , and for each image, there is $\sum_{m=1}^M P_{nm} = 1$.

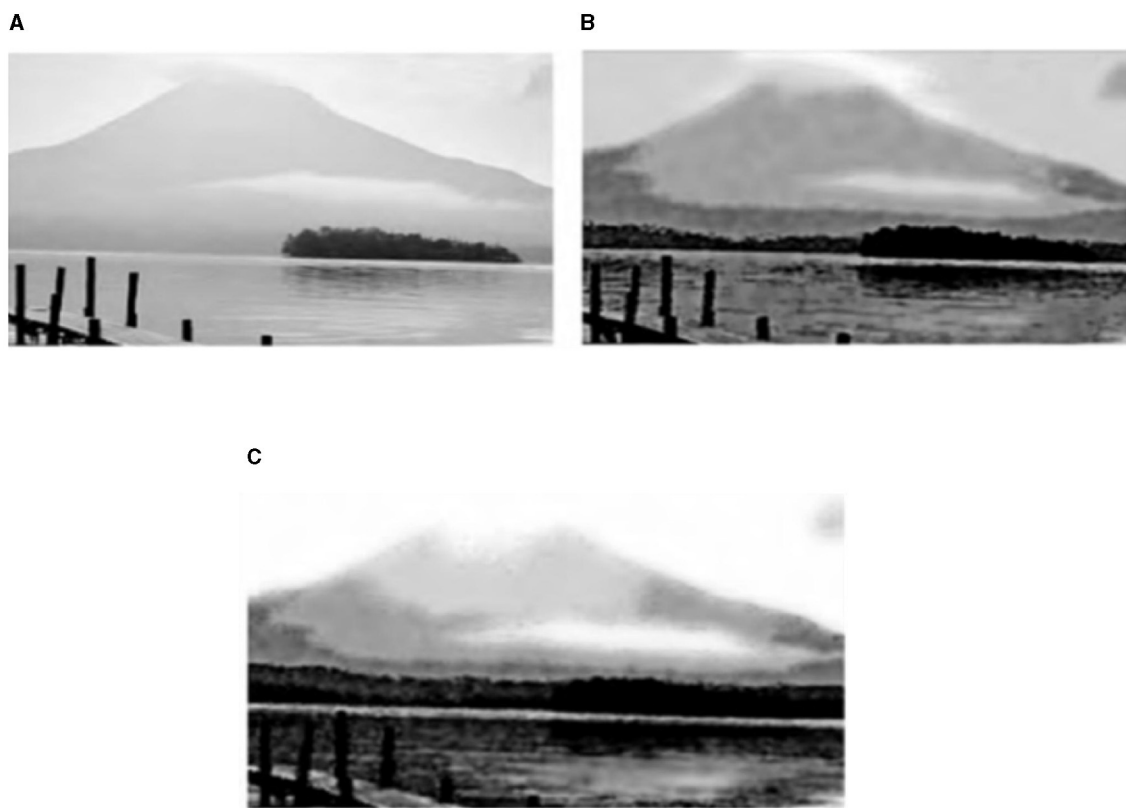


FIGURE 5

Comparison of direct eighth order color reduction and histogram prescribed ink simulation 1. (A) Landscape original 2. (B) Direct eighth order color reduction link effect. (C) Histogram normalization link effect.

Assuming that y is a test image, the goal of this study is to find the sentiment distribution $p = \{p_1, \dots, p_M\}^T$ of y , i.e., as shown in Equation (13).

$$f(\{x_n, p_n\}_{n=1}^N, y) \rightarrow p \quad (13)$$

Training sets that are very far away have little effect on y . Considering that including all training sets can slow down the run and irrelevant training samples can also mislead the algorithm's classification, the effect of isolated noise samples can be eliminated by taking a weighted average of the K -nearest neighbors.

Weighted K -nearest neighbor option denotes only the drizzle functions corresponding to the K training images that assign the larger weights to the closer nearest neighbors. denotes the sentiment distribution of the K training images nearest to the test image, which is considered as a basis function, and the sentiment distribution P of the test image y is computed by performing a distance-weighted summation of the basis function, i.e.,

$$P = \frac{\sum_{k=1}^K s_k p_k}{\sum_{k=1}^K s_k} \quad (14)$$

where s is the similarity between the test sample and the training sample, as shown in Equation (15).

$$s = e \left(-\frac{d(x_k, y)}{\beta} \right) \quad (15)$$

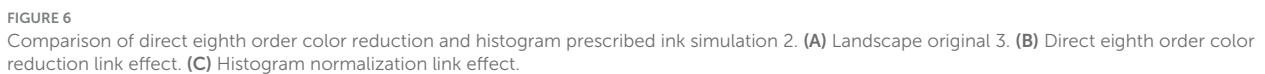
where d is the Euclidean distance and β is the average distance of y from the training images.

Algorithm: Weighted K -nearest neighbor sentiment distribution prediction algorithm.

Input: Training set (x_n, p_n) , test set y .

Output: Sentiment distribution p for the test set.

1. Calculate the distance d between the test set image y and each image in the training set.
2. Select the first k images $x_1 \dots x_k$ that are closest to y in the increasing order of distance.
3. $\beta = \frac{1}{k} \sqrt{(x_1 - y)^2 + \dots + (x_k - y)^2}$ is brought into Equation (14) in order to compute the similarity s .
4. Calculate the sentiment distribution of the test image y $P = \frac{\sum_{k=1}^K s_k p_k}{\sum_{k=1}^K s_k}$.



5.1 Landscape image

In the algorithm, the histogram specification serves to preset the weight of each ink color and enhance the recognition of the inked area. **Figures 5, 6** show a comparison of the ink simulation experiments for two other sets of landscape images and reveal the role of histogram specification in the simulation effect. **Figures 5B, 6B** show the effect of the algorithm based on direct eighth order grayscale color reduction, and **Figures 5C, 6C** show the effect of the algorithm based on histogram specification. The values of p_u (u_j) were [0.2, 0.15, 0.15, 0.15, 0.15, 0.15, 0.11, 0.06, 0.03] and [0.3, 0.125, 0.15, 0.125, 0.125, 0.1, 0.05, 0.025]. The direct eight-order color reduction approach is governed by the color values of the original

5.2 Abstract paintings

The existing sentiment classification networks ResNet and Swin Transformer and their variants are compared under the sentiment classification accuracy metrics in order to assess the effectiveness of the model in this article. The encoder-decoder structure with various numbers of layers is set up for this article's method; the one-layer encoder-decoder structure is defined as Tiny and the six-layer encoder-decoder structure is defined as Base. By training five batches of experimental findings and averaging them as the final results of the experimental data, five rounds of cross-validation were used to test the models. To accelerate the convergence of abstract painting sentiment classification, each

TABLE 1 Example of part of the abstract painting dataset.






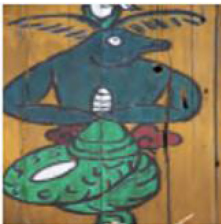



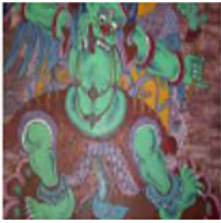

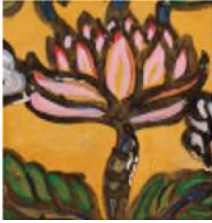
Emotion	Painting theme			
	Character	Ghosts	Animal	Plant
Negative				
	Be cautious	Aversion and resistance	Anxiety and tension	Faint and wilting
Neutral				
	Harmony and friendliness	Neutral and Pure	Be pragmatic and responsive	Impartial and impartial
Positive				
	Diligent and simple	Enthusiastic and proactive	Elegant and gentle	Beautiful and graceful

TABLE 2 Abstract painting emotion classification experiment.

Model	Classification accuracy (%)
ResNet-18	64.6
ResNet-34	68.4
ResNet-50	70.3
ResNet-101	71.4
Swin-T	70.1
Swin-S	72.7
Swin-B	73.2
Vit-T	72.7
Vit-B	76.8
Method of this article—Tiny	74.3
Method of this article—Base	80.8

model is fine-tuned based on the ImageNet pre-trained model, using the Adam W optimizer with a weight decay of 0.1/30 epoch and an initial learning rate of 0.0001, and trained based on the NVIDIA RTX 2080Ti.

TABLE 3 Backbone network experiment.

Method	Backbone	Parameter quantity (M)	Accuracy (%)
1	ResNet-18	29	72.5
2	ResNet-34	39	76.7
3	ResNet-50	42	80.8
4	ResNet-101	61	81.9

The actual Naxi Dongba abstract paintings were gathered from the literature on Na xi abstract paintings, and the abstract paintings were divided into four categories based on the subject matter of the painting's creation. For instance, in the abstract painting data set shown in [Table 1](#), the figures, ghosts and monsters, animals, and plants are shown from left to right, and the abstract paintings were divided into 12 different emotion categories based on the emotions they conveyed.

ResNet50 was used as the backbone network in order to extract image features and tested on the test set for sentiment classification of abstract paintings.

The experimental findings in [Table 2](#) demonstrate that the algorithm presented in this article is superior to ResNet,

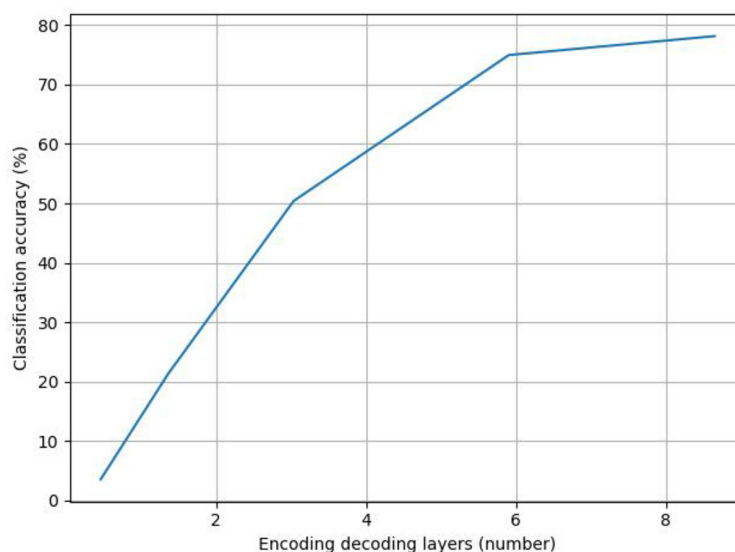


FIGURE 7
Analysis of the number of encoding and decoding layers.

TABLE 4 Ablation experiment.

Method	Decoder output	Encoder output	Accuracy (%)
1	X	✓	74.3
2	✓	X	77.9
3	✓	✓	80.8

Swin, and their variation network topologies for the job of sentiment recognition for abstract paintings. Established sentiment classification techniques like ResNet-101 and Swin-B achieved classification accuracies of 71.4 and 73.2%, respectively, whereas this article's method-Tiny and method-Base produced the best classification outcomes with classification accuracies of 74.3 and 80.8%, respectively.

The existing sentiment classification techniques do not account for the deeper sentiment elements that are buried in abstract paintings; instead, they focus on predicting the sentiment labels of abstract paintings while neglecting their linguistically complex and emotionally varied properties. The method in this article, in contrast, uses blank attention in the decoder and fuses the encoder's output sequence while learning the semantics of the abstract painting image as the emotion attention through the decoder's decoding learning mechanism. As a result, the method employed in this study is able to achieve a higher classification accuracy rate.

This study first conducts ablation experiments on the backbone network, compares a variety of Res Net variants to replace the backbone network, and keeps the structure of this article's model unchanged for the experiments in order to assess the impact of the number of parameters of the backbone network on the accuracy of sentiment classification. The results of the experiments are shown in Table 3.

The model parameter amount was 42 M and the classification accuracy was 80.8% when ResNet-50 was used as the backbone network. The number of model parameters was cut to 29 M with the use of ResNet-18, however the model's classification accuracy dropped by 8.3%. ResNet-34, on the other hand, reduced the number of model parameters by 3 M while increasing the classification accuracy of the model by 4.1% when utilized as the backbone network. The number of model parameters rises by 19 M when ResNet-101 is used as the backbone network, yet the classification accuracy increases by 1.1%. In this article, choosing ResNet-50 as the backbone network ensures that there is no significant decrease in the accuracy rate and avoids the increase in training difficulty due to the introduction of too many parameters.

Figure 7 displays the line graph of the experimental analysis of the number of coding-decoding layers; as the number of coding-decoding layers increases, the model's accuracy gradually increases, suggesting that adding more coding-decoding layers can, to a certain extent, increase the accuracy of the classification of the emotions in abstract paintings. The model uses six coding-decoding layers to achieve 80.8% classification accuracy, avoiding the overfitting issue that results from the stacking of coding-decoding layers. However, as the number of coding-decoding layers increases, the improvement in accuracy eventually slows down and becomes flat.

To prove the effectiveness of this article's attention mechanism for classifying the emotions of abstract paintings, two types of ablation models are set up to eliminate the decoder and encoder outputs, based on keeping the backbone network of the model as ResNet-50: ① The attention mechanism setup is not used in the ablation model, which eliminates the output of the decoder. Instead, the model uses the coded sequence output from the encoder as the basis for emotion classification. The classifier then normalizes the coded sequence to determine the likelihood of outputting emotion labels through the full connectivity layer

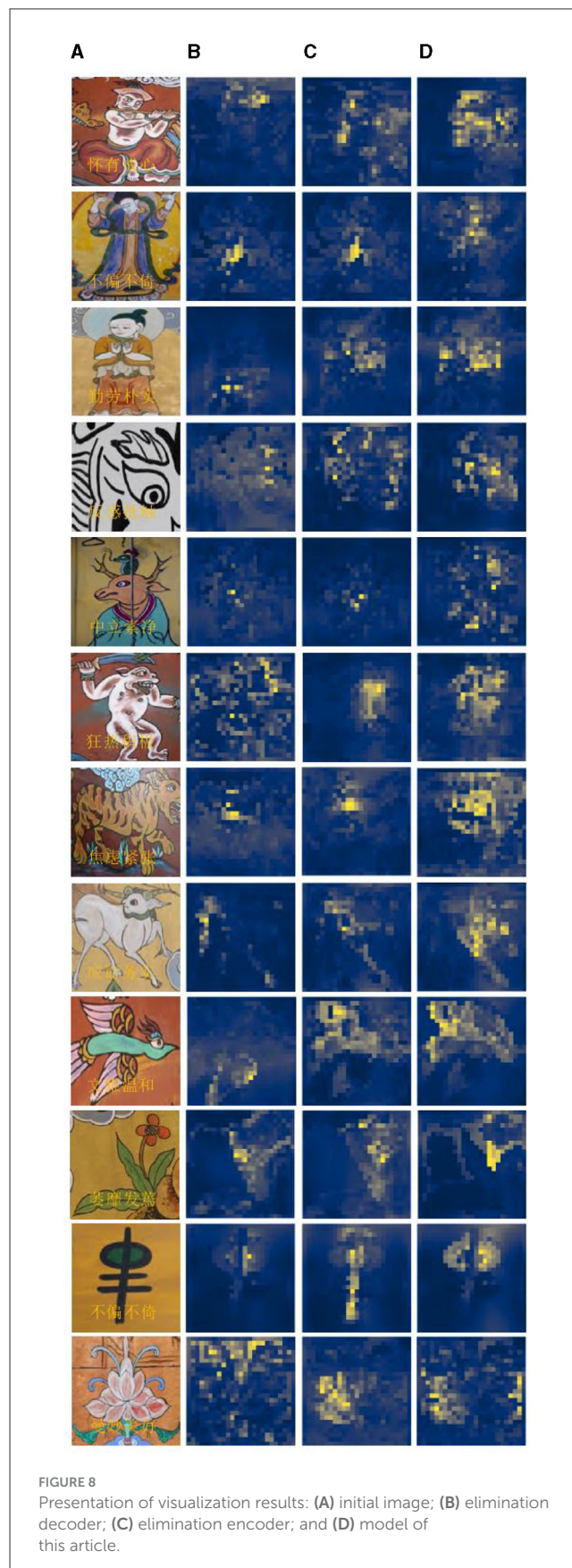
spreading. The fully linked layer disperses the coded sequence, and the normalization of the classifier determines the likelihood of producing emotion labels; ② The attention mechanism established in this research is kept in the ablation model that removes the encoder output, and the model continues to mine the picture semantics of the abstract paintings without fusing the encoder output with the attention mechanism. In Table 4, the experimental findings are displayed.

When the encoder is used to help classify the abstraction of drawing sentiments, the accuracy of sentiment classification decreased after eliminating the decoder output by 6.5%, showing higher classification accuracy than that of the ResNet-50 classification model; however, after eliminating the encoder output, the accuracy of sentiment classification of the ablation model decreased by 2.9%, which is higher than that of the ResNet-101 classification model and close to that of the ResNet-50 classification model. This finding shows that the attention mechanism in this study can help the model recognize abstract paintings' emotions more accurately by acting as a facilitator.

In this study, we used a full convolutional network to calculate the emotional weights of the model, visualize the weight heat map of the model, and simultaneously highlight and locate the regions in the heat map that significantly influence the expression of emotion.

Figure 8A provides an illustration of an abstract painting's original image, which is tagged with the predicted emotions derived from the image by the model test and contains 12 emotions as determined by the experimental data, respectively. The ablation model produced by the elimination decoder is depicted in Figure 8B, with loose regions of attention and unfocused regions of interest in the model's heat map; the regions of interest for abstract paintings of various subjects also differ significantly from one another. Figure 8C demonstrates that, despite being more compact, the model heat map's zone of interest suffers from ambiguous regions of interest and incorrect localization. It is also unresponsive to a smaller percentage of the neutral emotion image. The focus in the figure paintings is on the behavior and movements of the Dong ba figures, and the areas highlighted by the model labeled colors in the different image emotions correspond to the areas of the abstract paintings where the figures are holding arms, dancing, and making gestures, respectively. Figure 8D shows the model heat map of this article, which has a more concentrated region of interest and more stable localization. For emotionally complex animal paintings, the model expands the emotional expression to the animal's body area; for the plant paintings, the color highlighting points out the plant petal area, which corresponds to the plant's budding or blossoming gesture. In the ghost paintings, the model heat map focuses on the ghost behavior and action area.

The visualization experiments demonstrate the comparison experiments of the ablation model and the region of interest of the model described in this article. They also show how the relationship between the abstract painting emotion attention and the image emotion learned by this article model is more intimate and how this has a more immediate effect on the results of the emotion classification. It demonstrates how well the model in this study extracts the emotion from images of abstract paintings, making it more appropriate for classifying the emotions of abstract paintings.



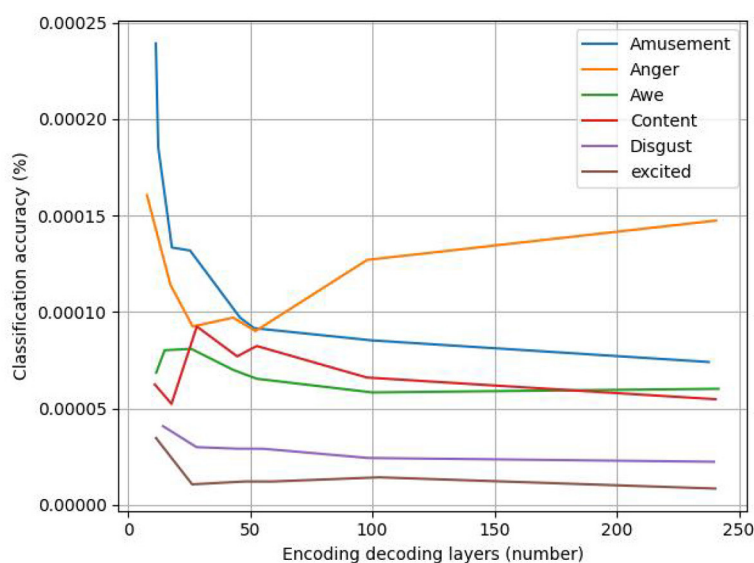


FIGURE 9
Effect of different K -values on the prediction of sentiment distribution.

5.3 Predictive distribution

The effect of different values of K ($K = 5, 10, 20, 40, 50, 100, 252$ using 10-fold cross-validation, where $K = 252$ is the global weighting of the training set) on the prediction of the sentiment distribution in a weighted KKN is shown in Figure 9.

In this example, the optimal K -value is affected by the sentiment category, and considering the average performance, it is considered that the best prediction is achieved at $K = 40, 50$, which outperforms the global weighting, and when $K = 252$, all the training images are used for distribution prediction.

better classification accuracy of 80.7% when compared to state-of-the-art techniques, thereby resolving the issues of rich material and difficulties in identifying the emotions shown in abstract paintings. Nevertheless, there are several drawbacks to the attention mechanism in this article, such as its incapacity to create the positional link between objects and scenes in abstract paintings. Furthermore, it is restricted by the dataset on abstract paintings and is unable to sufficiently address the issues of imprecise sentiment categorization and imprecise attention learnt from datasets that are made publicly available. Future research methods might thus expand the sentiment dataset to a broader picture data domain and further expand the abstract painting sentiment classification system to a multimodal level in order to overcome these problems.

6 Conclusion

The majority of early algorithms employed for sentiment classification were based on shallow machine learning and extract features using manually constructed feature selection techniques that have weak generalization ability, require extensive training times, and entail high labor costs. Because of its superior learning capacity to optimize feature extraction and prevent the flaws of manual feature selection, deep learning has produced positive research outcomes in the field of text sentiment categorization. The attention mechanism's primary objective is to swiftly separate valuable information from a vast amount of data. When applied to the sentiment classification task, it is capable of identifying word dependencies within sentences and identifying the internal organization of the sentence. Using a weighted closest neighbor technique, we provide a novel approach in this study to predict the discrete sentiment distribution of each picture in an abstract painting. Testing shows that the attention mechanism-based classification algorithm achieves a

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

KC: Writing – original draft.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

Acknowledgments

The author would like to show sincere appreciation to the development of those techniques that have contributed to this research.

Conflict of interest

The author declares that the research was conducted in the absence of any commercial or financial relationships

References

- Ahmad, T., and Wu, J. (2023). SDIGRU: spatial and deep features integration using multilayer gated recurrent unit for human activity recognition. *IEEE Trans. Comput. Soc. Syst.* 2023:3249152. doi: 10.1109/TCSS.2023.3249152
- Ahmad, T., Wu, J., Alwageed, H. S., Khan, F., Khan, J., and Lee, Y. (2023). Human activity recognition based on deep-temporal learning using convolution neural networks features and bidirectional gated recurrent unit with features selection. *IEEE Access* 11, 33148–33159. doi: 10.1109/ACCESS.2023.3263155
- Ahmad, T., Wu, J., Khan, I., Rahim, A., and Khan, A. (2021). Human action recognition in video sequence using logistic regression by features fusion approach based on CNN features. *Int. J. Adv. Comput. Sci. Appl.* 11:121103. doi: 10.14569/IJACSA.2021.0121103
- Alirezazadeh, P., Schirrmann, M., and Stolzenburg, F. (2023). Improving deep learning-based plant disease classification with attention mechanism. *Gesunde Pflanzen* 75, 49–59. doi: 10.1007/s10343-022-00796-y
- Bharadiya, J. (2023). Convolutional neural networks for image classification. *Int. J. Innov. Sci. Res. Technol.* 8, 673–677. doi: 10.5281/zenodo.8020781
- Cetinic, E., and She, J. (2022). Understanding and creating art with AI: review and outlook. *ACM Trans. Multimed. Comput. Commun. Appl.* 18, 1–22. doi: 10.1145/3475799
- Chan, J. Y. L., Bea, K. T., Leow, S. M. H., Phoong, S. W., and Cheng, W. K. (2023). State of the art: a review of sentiment analysis based on sequential transfer learning. *Artif. Intell. Rev.* 56, 749–780. doi: 10.1007/s10462-022-10183-8
- Chandrasekaran, G., Antonaela, N., Andrei, G., Monica, C., and Hemanth, J. (2022). Visual sentiment analysis using deep learning models with social media data. *Appl. Sci.* 12:1030. doi: 10.3390/app12031030
- Chen, X., Li, J., and Hua, Z. (2023). Retinex low-light image enhancement network based on attention mechanism. *Multimed. Tools Appl.* 82, 4235–4255. doi: 10.1007/s11042-022-13411-z
- Ding, F., Yu, K., Gu, Z., Li, X., and Shi, Y. (2021). Perceptual enhancement for autonomous vehicles: restoring visually degraded images for context prediction via adversarial training. *IEEE Trans. Intell. Transport. Syst.* 23, 9430–9441. doi: 10.1109/TITS.2021.3120075
- Li, W., Shao, W., Ji, S., and Cambria, E. (2022). BiERU: bidirectional emotional recurrent unit for conversational sentiment analysis. *Neurocomputing* 467, 73–82. doi: 10.1016/j.neucom.2021.09.057
- Li, X., Li, M., Yan, P., Li, G., Jiang, Y., Luo, H., et al. (2023). Deep learning attention mechanism in medical image analysis: basics and beyonds. *Int. J. Netw. Dyn. Intell.* 2023, 93–116. doi: 10.53941/ijndi0201006
- Liu, H., Nie, H., Zhang, Z., and Li, Y. F. (2021a). Anisotropic angle distribution learning for head pose estimation and attention understanding in human-computer interaction. *Neurocomputing* 433, 310–322. doi: 10.1016/j.neucom.2020.09.068
- Liu, T., Wang, J., Yang, B., and Wang, X. (2021b). NGDNet: nonuniform Gaussian-label distribution learning for infrared head pose estimation and on-task behavior understanding in the classroom. *Neurocomputing* 436, 210–220. doi: 10.1016/j.neucom.2020.12.090
- McCormack, J., and Lomas, A. (2021). Deep learning of individual aesthetics. *Neural Comput. Appl.* 33, 3–17. doi: 10.1007/s00521-020-05376-7
- Milani, F., and Fraternali, P. (2021). A dataset and a convolutional model for iconography classification in paintings. *J. Comput. Cult. Herit.* 14, 1–18. doi: 10.1145/3458885
- Ngai, W. K., Xie, H., Zou, D., and Chou, K. L. (2022). Emotion recognition based on convolutional neural networks and heterogeneous bio-signal data sources. *Inform. Fusion* 77, 107–117. doi: 10.1016/j.inffus.2021.07.007
- Peng, S., Cao, L., Zhou, Y., Ouyang, Z., Yang, A., Li, X., et al. (2022). A survey on deep learning for textual emotion analysis in social networks. *Digit. Commun. Netw.* 8, 745–762. doi: 10.1016/j.dcan.2021.10.003
- Roy, S. K., Deria, A., Shah, C., Haut, J. M., Du, Q., and Plaza, A. (2023). Spectral-spatial morphological attention transformer for hyperspectral image classification. *IEEE Trans. Geosci. Rem. Sens.* 61, 1–15. doi: 10.1109/TGRS.2023.3242346
- Sahoo, K. K., Dutta, I., Ijaz, M. F., Wozniak, M., and Singh, P. K. (2021). TLEFuzzyNet: fuzzy rank-based ensemble of transfer learning models for emotion recognition from human speeches. *IEEE Access* 9, 166518–166530. doi: 10.1109/ACCESS.2021.3135658
- Sampath, V., Murtua, I., Aguilar Martin, J. J., and Gutierrez, A. (2021). A survey on generative adversarial networks for imbalance problems in computer vision tasks. *J. Big Data* 8, 1–59. doi: 10.1186/s40537-021-00414-0
- Song, T., Zheng, W., Liu, S., Zong, Y., Cui, Z., and Li, Y. (2021). Graph-embedded convolutional neural network for image-based EEG emotion recognition. *IEEE Trans. Emerg. Top. Comput.* 10, 1399–1413. doi: 10.1109/TETC.2021.3087174
- Szubielska, M., Imbir, K., and Szymańska, A. (2021). The influence of the physical context and knowledge of artworks on the aesthetic experience of interactive installations. *Curr. Psychol.* 40, 3702–3715. doi: 10.1007/s12144-019-00322-w
- Teodoro, A. A., Silva, D. H., Rosa, R. L., Saadi, M., Wuttisittikulkij, L., Mumtaz, R. A., et al. (2023). A skin cancer classification approach using gan and roi-based attention mechanism. *J. Sign. Process. Syst.* 95, 211–224. doi: 10.1007/s11265-022-01757-4
- Toisoul, A., Kossai, J., Bulat, A., Tzimiropoulos, G., and Pantic, M. (2021). Estimation of continuous valence and arousal levels from faces in naturalistic conditions. *Nat. Machine Intell.* 3, 42–50. doi: 10.1038/s42256-020-00280-0
- Wang, Y., Song, W., Tao, W., Liotta, A., Yang, D., Li, X., et al. (2022). A systematic review on affective computing: emotion models, databases, and recent advances. *Inform. Fusion* 83, 19–52. doi: 10.1016/j.inffus.2022.03.009
- Yang, H., Wang, L., Xu, Y., and Liu, X. (2023). CovidViT: a novel neural network with self-attention mechanism to detect COVID-19 through X-ray images. *Int. J. Machine Learn. Cybernet.* 14, 973–987. doi: 10.1007/s13042-022-01676-7
- Yang, Z., Baraldi, P., and Zio, E. (2021). A multi-branch deep neural network model for failure prognostics based on multimodal data. *J. Manufact. Syst.* 59, 42–50. doi: 10.1016/j.jmjsy.2021.01.007
- Zhang, C., Li, M., and Wu, D. (2023). Federated multidomain learning with graph ensemble autoencoder GMM for emotion recognition. *IEEE Trans. Intell. Transp. Syst.* 24, 7631–7641. doi: 10.1109/TITS.2022.3203800
- Zhao, S., Jia, G., Yang, J., Ding, G., and Keutzer, K. (2021a). Emotion recognition from multiple modalities: fundamentals and methodologies. *IEEE Sign. Process. Mag.* 38, 59–73. doi: 10.1109/MSP.2021.3106895
- Zhao, W., Zhou, D., Qiu, X., and Jiang, W. (2021b). Compare the performance of the models in art classification. *PLoS ONE* 16:e0248414. doi: 10.1371/journal.pone.0248414
- Zhou, J., Pang, L., and Zhang, W. (2023). Underwater image enhancement method by multi-interval histogram equalization. *IEEE J. Ocean. Eng.* 48, 474–488. doi: 10.1109/OJEE.2022.3223733
- Zou, Q., Wang, C., Yang, S., and Chen, B. (2023). A compact periocular recognition system based on deep learning framework AttenMidNet with the attention mechanism. *Multimed. Tools Appl.* 82, 15837–15857. doi: 10.1007/s11042-022-14017-1

that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.



OPEN ACCESS

EDITED BY

Deepika Koundal,
University of Petroleum and Energy
Studies, India

REVIEWED BY

Vatsala Anand,
Chitkara University, India
Arvind Dhaka,
Manipal University Jaipur, India

*CORRESPONDENCE

Anum Masood
✉ anum.masood@ntnu.no
Muhammad Attique Khan
✉ attique.khan@ieee.org

RECEIVED 29 February 2024

ACCEPTED 28 March 2024

PUBLISHED 25 April 2024

CITATION

Yaqoob N, Khan MA, Masood S, Albarakati HM,
Hamza A, Alhayan F, Jamel L and Masood A
(2024) Prediction of Alzheimer's disease
stages based on ResNet-Self-attention
architecture with Bayesian optimization and
best features selection.
Front. Comput. Neurosci. 18:1393849.
doi: 10.3389/fncom.2024.1393849

COPYRIGHT

© 2024 Yaqoob, Khan, Masood, Albarakati,
Hamza, Alhayan, Jamel and Masood. This is
an open-access article distributed under the
terms of the [Creative Commons Attribution
License \(CC BY\)](#). The use, distribution or
reproduction in other forums is permitted,
provided the original author(s) and the
copyright owner(s) are credited and that the
original publication in this journal is cited, in
accordance with accepted academic practice.
No use, distribution or reproduction is
permitted which does not comply with these
terms.

Prediction of Alzheimer's disease stages based on ResNet-Self-attention architecture with Bayesian optimization and best features selection

Nabeela Yaqoob¹, Muhammad Attique Khan^{1*}, Saleha Masood²,
Hussain Mobarak Albarakati³, Ameer Hamza¹, Fatimah Alhayan⁴,
Leila Jamel⁴ and Anum Masood^{5*}

¹Department of Computer Science and Mathematics, Lebanese American University, Beirut, Lebanon, ²IRC for Finance and Digital Economy, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia, ³Department of Computer and Network Engineering, College of Computer and Information Systems, Umm Al-Qura University, Makkah, Saudi Arabia, ⁴Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia, ⁵Department of Physics, Norwegian University of Science and Technology, Trondheim, Norway

Alzheimer's disease (AD) is a neurodegenerative illness that impairs cognition, function, and behavior by causing irreversible damage to multiple brain areas, including the hippocampus. The suffering of the patients and their family members will be lessened with an early diagnosis of AD. The automatic diagnosis technique is widely required due to the shortage of medical experts and eases the burden of medical staff. The automatic artificial intelligence (AI)-based computerized method can help experts achieve better diagnosis accuracy and precision rates. This study proposes a new automated framework for AD stage prediction based on the ResNet-Self architecture and Fuzzy Entropy-controlled Path-Finding Algorithm (FECPFA). A data augmentation technique has been utilized to resolve the dataset imbalance issue. In the next step, we proposed a new deep-learning model based on the self-attention module. A ResNet-50 architecture is modified and connected with a self-attention block for important information extraction. The hyperparameters were optimized using Bayesian optimization (BO) and then utilized to train the model, which was subsequently employed for feature extraction. The self-attention extracted features were optimized using the proposed FECPFA. The best features were selected using FECPFA and passed to the machine learning classifiers for the final classification. The experimental process utilized a publicly available MRI dataset and achieved an improved accuracy of 99.9%. The results were compared with state-of-the-art (SOTA) techniques, demonstrating the improvement of the proposed framework in terms of accuracy and time efficiency.

KEYWORDS

Alzheimer's disease, MRI, deep learning, self-attention, convolutional neural network, optimization, fuzzy entropy

1 Introduction

Dementia is the seventh-greatest root cause of mortality and the main reason for impairment and vulnerability in elderly individuals (Koul et al., 2023). It is a rapidly spreading disorder among the elderly population, becoming increasingly common over the last decade (Sisodia et al., 2023). Dementia greatly impairs intellectual performance, interfering with daily tasks and interpersonal interactions (Nagdee, 2011). Alzheimer's disease (AD) is an inseparable subclass of dementia that can cause memory loss in a person (Mahmud et al., 2024). An individual affected by AD may struggle to recognize family members and experience difficulties in remembering daily activities. Moreover, it can cause ultimately lead to the death of the patient (Mohammad and Al Ahmadi, 2023). Due to these worse health conditions, it is also referred to as a progressive neurodegenerative disease. It affects behavioral functions, thinking abilities, decision-making, and language skills, often leading to memory loss in older people (Kellar and Craft, 2020).

Brain cell alteration may occur a decade or more before clinical signs appear. In the beginning, patients with AD experience unnoticed changes in their brains (Jansi et al., 2023). Throughout the early AD stage, the brain undergoes destructive transformations, including ectopic protein deposition that produces amyloid plaques and tau tangles. Neurons that were once fully functional cease to function properly, losing connections to other neurons and eventually undergoing cell death (Hoozemans et al., 2006). Several additional intricate alterations in the brain can also lead to Alzheimer's (Kasula, 2023). The hippocampus and entorhinal cortex, critical for cognitive control, seem to be the initial regions of impairment (Shrager et al., 2008). Furthermore, the signs of AD begin to manifest when nerve cells (neurons) in certain areas of the brain gradually shrink and eventually become destroyed or damaged (Khalid et al., 2023). In the final phase of AD, damage becomes widespread, and a large amount of brain tissue is destroyed (Blonicki Kallio, 2002; Carle, 2022).

AD is such a serious brain disease that it can result in a patient's death if not effectively treated (Gómez-Isla and Frosch, 2022). To overcome this disease, patients need good care, regular exercise, and some memory-sharpening activities as there is currently no specific medication for AD (Shamrat et al., 2023). In recent years, a significant increase has been observed in AD (Mirzaei and Adeli, 2022; Stevenson-Hoare et al., 2023). The number of deaths from Alzheimer's disease in 2020 increased by 15,925 compared to the 5 years before 2023, and 44,729 more deaths were recorded for all dementias, including Alzheimer's disease (Chua, 2023). Traditional machine learning (ML) techniques such as pre-processing (Wen et al., 2020), feature extraction (Rathore et al., 2017), feature selection (Balaji et al., 2023), feature fusion (Jia and Lao, 2022), and classification (Tanveer et al., 2020) have been employed by researchers as a four-step channel in the past few years. Classification is the bottommost step in which each object accredits a label, in either a supervised or unsupervised ML technique (Bondi et al., 2017). Deep learning (DL) (Shaukat et al., 2022) is a subtype of machine learning that falls under the umbrella of artificial intelligence, but DL is way more vigorous and flexible in comparison with ML (Fabrizio et al., 2021). Techniques such as shallow CNN (Marwa et al., 2023), DNN (Hazarika et al.,

2023), MultiAz-Net (Ismail et al., 2023), hybridized DL method (Hashmi, 2024), and RVFL (Goel et al., 2023) have been used in recent years, but these techniques yield low accuracy as compared to our proposed model (Shamrat et al., 2023).

1.1 Major challenges and gaps

Recent advances in ML and DL have opened up new avenues for assessing AD, but researchers are still grappling with the diagnosis of the disease (Shamrat et al., 2023). Few of them are related to insufficient and unbalanced datasets. Furthermore, major problems with AD patients are the complexity, diversity, and complicated neurobiological underlying AD (Dhakhnamoorthy et al., 2023). Architectural variation in scans is another main challenge to diagnosing and detecting AD. However, the influence of these challenges may vary from patient to patient. This research will focus on AD stages for classification using deep learning and feature optimization techniques.

1.2 Major contributions

The main contributions of this study are as follows:

- A fine-tuned ResNet-50 architecture has been modified by adding a self-attention layer and trained from scratch for feature extraction.
- Hyperparameters of the trained model are initialized using an optimization technique named Bayesian optimization.
- Improved the extracted self-attention features using an improved pathfinder optimization named the Fuzzy entropy-controlled path-finding algorithm (FECPPFA). The optimization algorithm selects the best features and improves the efficiency.
- The optimized selected features are finally classified using machine learning to classify the stages of AD.

This article is organized as follows: Section 2 reviews ML and DL techniques that have been applied to Alzheimer's disease, and Section 3 provides a comprehensive description of the datasets. The testing outcomes are shown in detail in Section 4. Section 5 summarizes our findings, and Section 6 discusses future work.

2 Related work

Due to the brain's intricacy, classifying AD is difficult (Dhakhnamoorthy et al., 2023). Thus, researchers are improving medical image processing to identify AD correctly. This section presents relevant literature in the domain of AD detection and diagnosis, which focuses primarily on classification techniques based on deep learning for MRI tissue structure analysis (Mohi et al., 2023). The deep belief network (DBN) was utilized by AI-Atroshi et al. (2022) to extract feature vectors from detected speech samples, which has an output accuracy of 90.2%. Shankar et al. (2022) used HAAR-based object identification techniques because they are more suitable with discriminant attributes and generated

37 spatial pieces of information from seven characteristics that produced 94.1% accuracy on the dataset taken from ADNI. To aid in the initial diagnosis of AD (FDN-ADNet), [Sharma et al. \(2022\)](#) used a DL network for all-level feature extraction from extracted sagittal plane slices of 3D MRI scans and a fuzzy hyperplane-oriented FLS-TWSVM for the classification of the retrieved features, which generated 97.29% accuracy on the publicly available ADNI dataset.

[Albright \(2019\)](#) presented the all-pairs pre-processing algorithm to train the model. For this experiment, setting data were taken from ADNI and divided into three datasets, i.e., LB1, LB2, and LB3, with an mAUC of 0.866. The 3D-CNN networks by [Soliman et al. \(2022\)](#) predicted AD. It learned basic traits that catch AD indicators to identify brains with Alzheimer's disease from healthy and normal brains using MRI scans. ADNI provided 3,013 photographs with 96.5% training accuracy and 80.6% tested accuracy. [Samhan et al. \(2022\)](#) adopted CNNs, VGG16, Adam, activation, and softmax optimizers. The Kaggle dataset of 10,432 images yielded 100% training accuracy, 0.0012 training loss, 97% validation accuracy, and 0.0832 verifying loss. [Jo et al. \(2022\)](#) proposed a unique deep learning-based genome-wide approach called SWAT-CNN that found SNPs associated with AD and a classification model for AD. It may be useful for a variety of biomedical applications and was tested on the GWAS dataset by the AD Neuroimaging Initiative (ADNI).

[Zhang et al. \(2022\)](#) adopted CNN models of various designs and capacities and assessed them thoroughly. The most appropriate model was then applied for AD diagnosis. To increase the transparency of the model, an explanation heatmap was produced for AD vs. cognitive normal (CN) classification tasks and pMCI vs. sMCI using two publicly available datasets. Interestingly, the study found that a moderately sized model could outperform one with the largest capacity. [Ghazal et al. \(2022\)](#) proposed the system named ADDLTA, in which the transfer learning (TL) approach was used in conjunction with brain medical resonance imaging (MRI) to classify the image into four categories: mildly demented (MD), moderately demented (MD), non-demented (ND), and very mildly demented (VMD), which gave 91.70% accuracy on simulation results based on the publicly assessable dataset by the Kaggle repository.

[Shanmugam et al. \(2022\)](#) focused on detecting different phases of cognitive impairment and AD in the early stages by utilizing TL in neuroimaging. GoogLeNet, AlexNet, and ResNet-18 were three pre-trained models adopted for classification, giving an accuracy of 96.39, 94.08, and 97.51%, respectively, on the ADNI dataset. [Prasath and Sumathi \(2024\)](#) suggested a compact architecture by merging two models, LeNet and AlexNet, that outperform DenseNet. Three parallel tiny filters (1×1 , 3×3 , and 5×5) replaced the convolution levels to recover key features that achieved 93.58% accuracy on the dataset taken from ADNI. [Sorour et al. \(2024\)](#) proposed a system for the automated diagnosis of Alzheimer's disease that integrates multiple customized deep-learning models to provide an objective evaluation. The very first methodology addresses AD diseases using SVM and KNN. The second approach combines rs-fMRI datasets from the ADNI repository with modified AlexNet and Inception blocks. This architecture gave 96.61% accuracy. A new optimized ensemble-based DNN learning model called MultiAz-Net is used by [Ismail](#)

[et al. \(2023\)](#) with diverse PET and MRI data to identify AD. The Multi-Objective Grasshopper Optimization Algorithm (MOGOA) optimizes MultiAz-Net layers, which produced 92.3% accuracy on the ADNI dataset. [Balaji et al. \(2023\)](#) suggested a DL approach to detect AD in its initial stages using multimodal imaging and the LSTM algorithm, combining MRI, PET, and traditional neuropsychological examination results. The suggested technique adjusted the learning weights to improve accuracy and employed Adam's optimization. The proposed architecture achieved 98.5% accuracy on 512 MRI and 112 PET scans.

3 Materials and methods

This section provides a comprehensive exposition of the experimental dataset and methodologies employed within. It elucidates the specifics of the experiments, including the nature of the dataset utilized and the methodologies adopted.

3.1 Dataset

A well-characterized repository has a significant role in the performance evaluation of a diagnosis system. In this experiment, a dataset was obtained from Kaggle. This dataset, known as Alzheimer's disease, consists of specimens of anonymously affected individuals with MRI scans and their appropriate class label details. This multiclass dataset contains four distinct classes and offers many different views, comprising over 5,000 MRI images. The four classes are shown in [Figure 1](#): mildly demented ([Shanmugam et al., 2022](#)), moderately demented ([Prasath and Sumathi, 2024](#)), non-demented ([Sorour et al., 2024](#)), and very mildly demented ([Ismail et al., 2023](#)). A brief explanation of the four classes of AD is given in [Table 1](#) for testing and training purposes. The data were imbalanced in each class. Each class consisted of a different number of images.

These datasets are the most prominent and effective for this publicly available domain. The major aim of this study is to yield high accuracy. Original MRI scans and augmented image distribution were utilized in the training and testing of the experiment. Mild demented contained 896 images; moderate demented contained 64 images; non-demented comprised 3,200 images; and very mild had 2,240 images. After the augmentation, we took 2,000 images from each class for further proceedings. [Figure 2](#) illustrates the AD stages with a brief description. Moreover, an image description that lists the number of classes and augmented images utilized in this study is found in [Table 2](#).

3.2 Proposed methodology

Our proposed study presents a deep learning-based methodology for classifying AD grades. First, the dataset was taken from Kaggle, a public repository. The data were unbalanced in each class, so different augmentation techniques were applied. The data have been enhanced by applying different enhancement methods. After the enhancement, we fine-tuned the ResNet-50

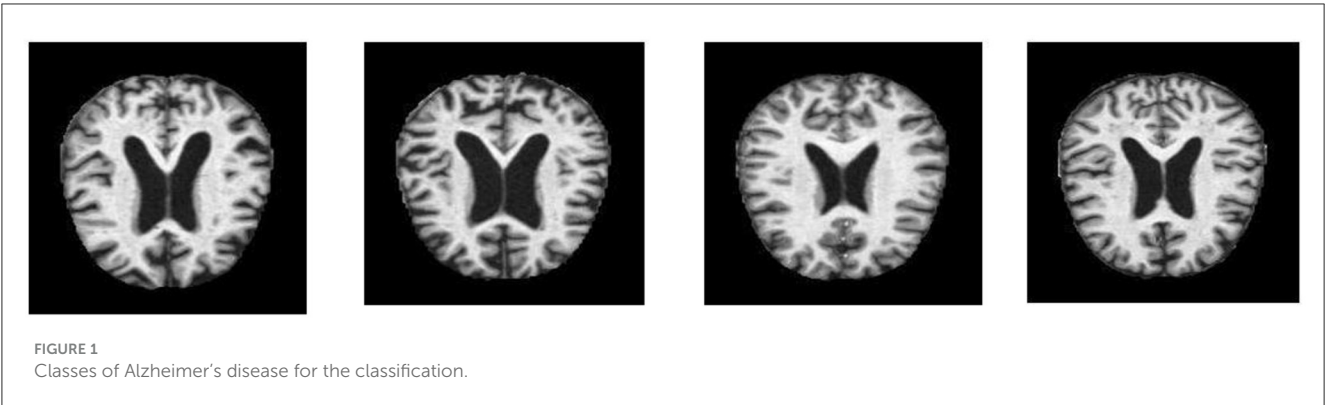
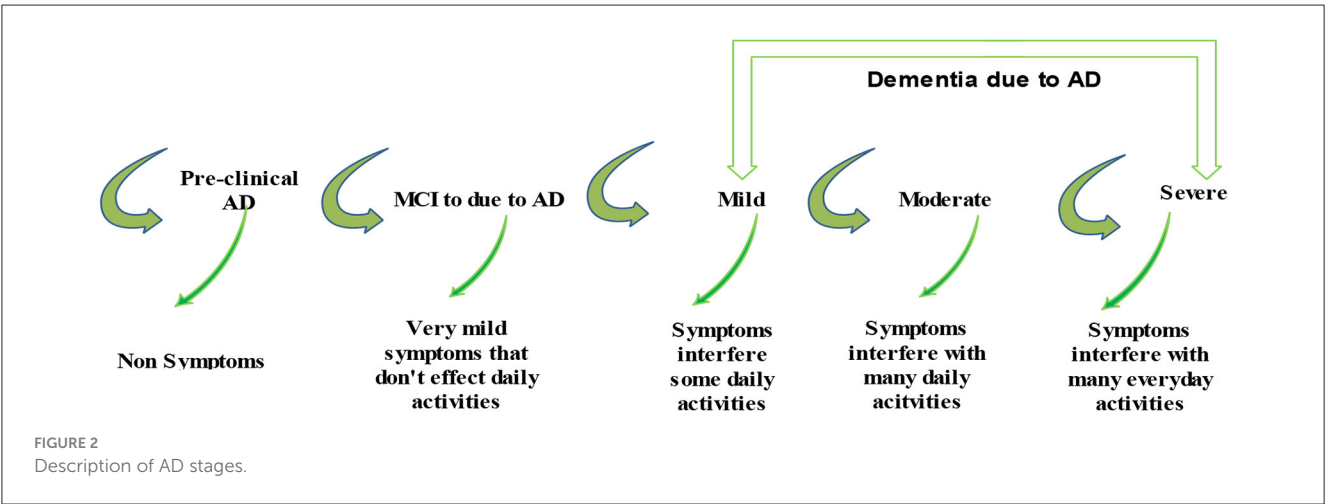


TABLE 1 Description of AD classes dataset.

References	Classes	Description	No of images
Shanmugam et al. (2022)	Mild demented	People may become socially withdrawn, and noticeable changes occur in their moods and personality. People may find it hard to remember the faces, people they met a long time ago, and recent events. Individuals do not recall what they are saying, cannot find their way to their desired location, and have lost focus and work abilities.	896
Prasath and Sumathi (2024)	Moderate demented	In this phase, the affected person requires help to do their routine work. Inability to recall important information, such as name of close relatives, home location, time, and date; however, the person knows their name and family member's names. The person lacks sensibility, forgets previous work, and struggles to keep track of finances and daily expenses while living alone.	3,200
Sorour et al. (2024)	Non-demented	It usually occurs in elderly persons. People may face difficulty in conversation and gradually memory loss.	64
Ismail et al. (2023)	Very mild demented	The person may find it hard to adjust to a new environment and experience apathy and repetition. Affected persons cannot complete the task. There seems to be low memory loss in this stage. Individuals may forget the names of people who lived with them.	2,240



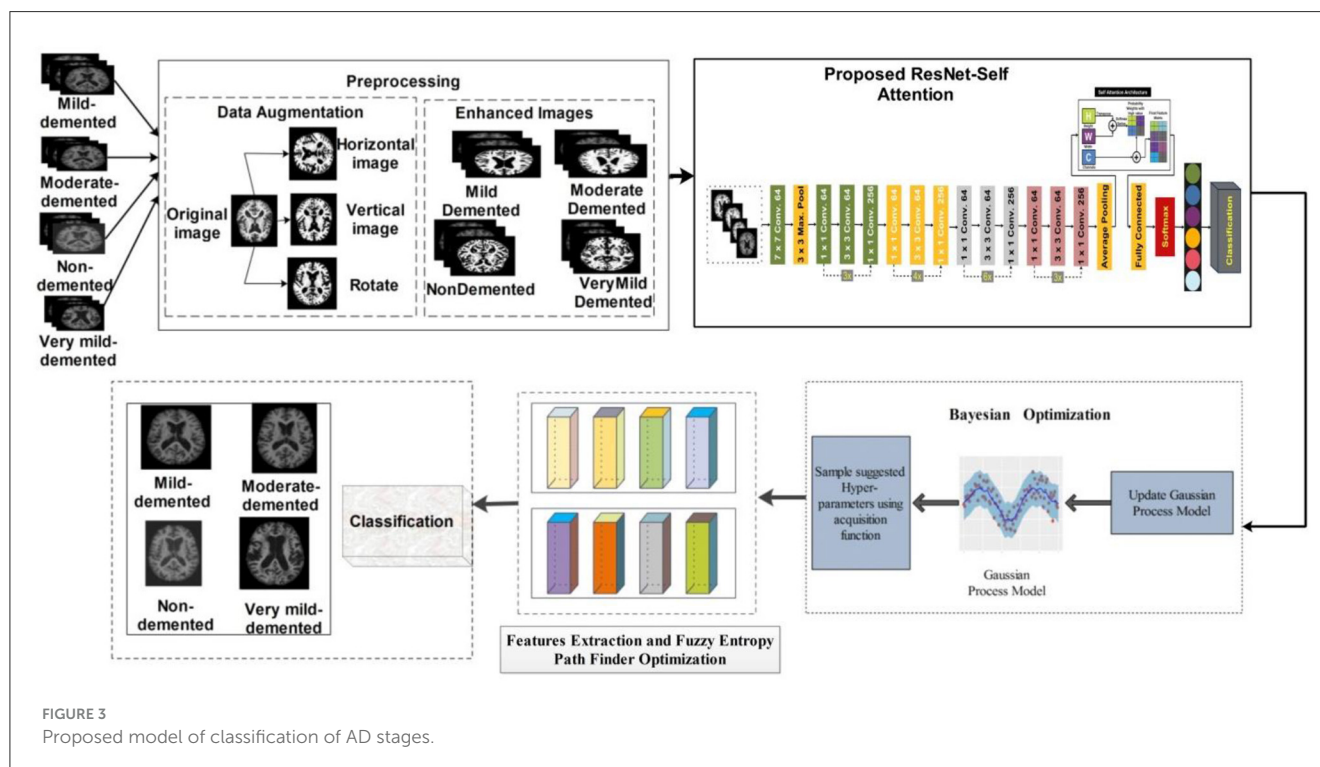
model and added Self-Attention layers. The modified model is trained on the augmented dataset and extracted deep features from the self-attention layer. The features are extracted from the self-attention layer. Bayesian optimization is employed for the selection of hyperparameters, instead of manual initialization. Moreover, PFA is utilized to select the optimal features. In the final stage, KNN, NN, and SVM classifiers are used to classify AD stages. The proposed model is represented in Figure 3.

3.2.1 Data augmentation

Augmentation is creating modified image variants from an existing image dataset to improve its variety artificially. Images are nothing more than a 2D collection of numbers for a computer. These numbers indicate intensity values, which may be modified to produce new, enhanced images. The primary goal of augmentation is to maintain parity among each group. It improved the outcomes and made them more precise and effective. In most cases, it was only useful for very small data sets. Images may be flipped

TABLE 2 Dataset image description.

S#	Dataset	Number of classes	Total images	Augmented images	Training/testing
1	Alzheimer's disease	Mild demented	896	3,200	3,200/2 = 1,600
2		Moderate demented	64	3,200	3,200/2 = 1,600
3		Non-demented	3,200	3,200	3,200/2 = 1,600
4		Very mild demented	2,240	3,200	3,200/2 = 1,600



horizontally, vertically, or rotated using this method. Both of these techniques expand the quantity of the dataset by producing images that have been flipped at various angles.

3.2.1.1 Horizontal flip

Complete rows and columns of image pixels are set aside horizontally. If the image on the right is flipped, the outcome will be on the left. The mathematical representation for the horizontal flip is shown by Equation 1.

$$H_F(-x, y) = H_O(x, y) \quad (1)$$

The given formula illustrates the horizontal flip of an image scan. HF shows the flipping function, while H_O represents the real image. The first half (x, y) displays the actual image, while quadrant two $(-x, y)$ displays the replica image. Therefore, the unedited version of the image resides within the first quarter, which is the right side, and after horizontal flipping, the image has been flipped to the second phase, which is the left side.

3.2.1.2 Vertical flip

Complete rows and columns of image pixels are set aside vertically. When an image is now displayed in the upward position

and flipped, the resulting image will be displayed in the downward motion. The mathematical representation for the vertical Flip is shown below:

$$H_v(x, -y) = H_O(x, y) \quad (2)$$

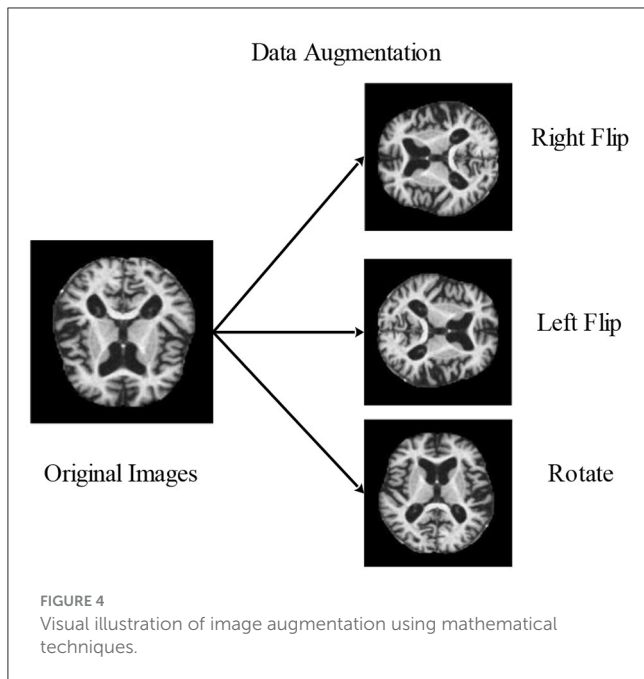
The given formula illustrates the vertical flip of an image scan. HV shows the flipping function, while H_O represents the real image.

The first half lies in (x, y) which displays the actual image, while the third quadrant third $(x, -y)$ displays the replica image. Therefore, Equation (2) demonstrates that the initial image resides in the first half on the right side. When the vertical flip is enforced, the image goes to the third half, which is in a downward direction. In short, it flipped the image along with the X-axis.

3.2.1.3 Rotate flip

A 3D graphic item is flipped by rotating it. The following is a mathematical representation by Equation 3.

$$g_{(i,j)}^{90^\circ} = \begin{bmatrix} \cos 90^\circ & -\sin 90^\circ \\ \sin 90^\circ & \cos 90^\circ \end{bmatrix} \begin{bmatrix} g_i \\ g_j \end{bmatrix} \quad (3)$$



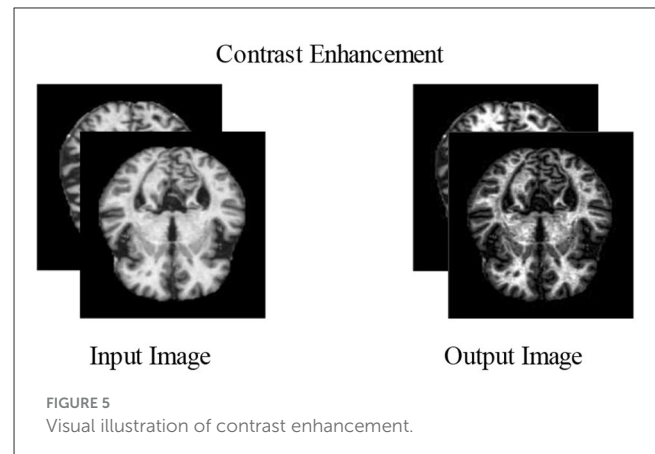
Consequently, image (I) is rotated by angle degrees counterclockwise around its center. To rotate the image counterclockwise, input a negative angle value, then (imrotate) will extend the resultant image (J) to encompass the entire rotated image. The methods for data augmentation used in the experiment are shown in [Figure 4](#).

3.2.2 Contrast enhancement

Increasing contrast is one of the most important and useful techniques for improving the essential elements of an image. Normally, raw images contain noise, distortion, and low contrast that lower the image quality, which sometimes causes the loss of useful information ([Perumal and Velmurugan, 2018](#)). Contrast enhancement improves the image qualities for further processing. More relevant characteristics may be extracted from the improved photographs for the classification stage than from the input image. The datasets chosen in this study have poor-quality images with low contrast levels. Due to this issue, we could end up incorrectly categorizing things. Contrast enhancement is further divided into two main groups, i.e., the spatial and frequency domains, such as morphological enhancement, histogram equalization, contrast stretching, contrast slicing, and some contrast enhancement. In our proposed experiment, two types of contrast enhancement have been adopted, one by one. First, a fast local Laplacian filter is applied to the augmented dataset. After this, a top-bottom hat filter is applied to the enhanced dataset to get better-quality MRI scans.

3.2.2.1 Fast local Laplacian filter

There are two main functions of FLLF. The first is applied to the raw images to boost the boundary detail and reduce the noise artifacts. The second is that the images are transformed from the RGB color system to the YUV color space to isolate the Y factor. Multiscale adjustments are crucial to photo editing but are



especially vulnerable to halos. Advanced edge-aware algorithms and careful parametric adjustments are needed to get outcomes without artifacts. These deficiencies were subsequently remedied through local Laplacian filters. These filters use typical Laplacian pyramids to generate a wide variety of effects. However, these filters are time-consuming, and their link to other methods is obscure.

3.2.2.2 Top-bottom hat filter

In this filter, the top-hat part is employed for objects with a light color on a darker backdrop, whereas the bottom-hat part is utilized for images with a dark color on a light background. The correction of the effects of non-lighting is a key purpose of the top-hat modification. When the shade is evident in an image, this filtering technique can effectively highlight the information in the image. The methods for contrast enhancement are visually presented in [Figure 5](#).

3.2.3 Bayesian optimization

Bayesian optimization incorporates Bayes' theorem to guide the search as a method for minimizing or maximizing an optimization technique. This method can be very helpful for optimization algorithms that are difficult to evaluate due to their complexity, noise, or cost. BO differs from other methods in that it considers previous parameter data by changing the baseline using Gaussian progress (GP). Additionally, BO has minimal iterations and a rapid convergence time. The BO approach may also eliminate local optimum in non-convex optimization circumstances. BO is a perfect pick for optimizing HPs due to its high convergence and resilience. All hyperparameters must be tuned to gain classification precision while utilizing DL architectures. The choice of hyperparameters substantially affects the accuracy and precision of the prediction. When optimizing hyperparameters, the objective is to choose the values that provide the highest quality validation findings. Hyperparameter optimization is written mathematically by [Equation 4](#).

$$x^* = \operatorname{argmin}_f(x) \quad (4)$$

where $f(x)$ shows the cost-minimizing objective score for evaluating hyperparameter optimization relative to the validation set, and x is

TABLE 3 Hyperparameter range of BO.

Hyperparameters	Ranges
L2Regularization	$(1e^{-10}, 1e^{-2})$
Section depth	(1, 3)
Momentum	(0.7, 0.98)
Learning rate	(0.0001, 1)

the set of hyperparameters whose values lie in that range. Training takes longer and is extremely difficult to do by hand with DNN models with numerous hyperparameters. ML and simulations employ BO. FFNN designs alter hyperparameters in CV-based techniques to enhance the performance of the mode network. Optimizing several parameters is faster by using it.

In contrast to other methods, BO updates the prior with Gaussian progress to adjust for past parameter values (GP). Additionally, BO converges quickly and with a small number of iterations. When addressing non-convex optimization problems, the BO approach may be able to sidestep localized optimality. BO is a great option for optimizing HPs due to its high convergence and resilience. The stopping condition of the BO algorithm is based on MaxTime. The BO algorithm stops when it reaches the MaxTime, which is 54,000 s. This time is approximately equivalent to 15 h. In this study, we utilized BO along with DCNN, which fine-tuned the hyperparameters to generate the lowest error rate with optimal results in an architecture. Optimizing parameters such as L2Regularization, Section Depth, Momentum, and Learning Rate have been used in this study, shown along with their ranges in Table 3, which represents the Bayesian optimization workflow. Figure 6 illustrates the BO.

3.2.4 Deep transfer learning

Transfer learning is applying a learned model to a different situation. The fact that it has the potential to train deep neural networks on very little training data has recently made it more famous in deep learning. Deep transfer learning is becoming more prominent in handling image classification issues as it is feasible to use built-in CNNs on publicly available datasets such as ImageNet to achieve top classification accuracy in several application domains. After transfer learning (TL), the framework is fine-tuned (FT) to relearn all FE and C. FT is performed by initializing feature extraction parameters and ImageNet weights, and classification parameters are updated along with TL weights. Figure 7 illustrates the deep transfer learning workflow.

3.2.5 Proposed ResNet-Self architecture

In this study, we proposed a modified ResNet-50 architecture based on the self-attention module named ResNet-Self. Initially, we consider the ResNet-50 architecture based on the residual blocks. In this network, 48 convolutional layers have been originally added, along with one max-pooling layer, one average pooling layer, and one fully connected layer. The residual blocks added in this network contain skip connections. In this network, bottleneck filters are applied, such as 1×1 , which reduces the number of parameters.

The depth size of this model is originally between 64- and 2,048, and filter sizes of 3×3 . Moreover, the stride is used 2 out of the residual blocks, and in the residual blocks, 1 stride is employed. The average pooling layer has been added at the end of this model for the features extraction that followed the fully connected and softmax layers. The initial performance of this model for AD stage classification was insufficient; therefore, we modified it with the latest concept named Self-Attention.

The proposed ResNet-Self architecture is illustrated in Figure 8. This figure shows that the self-attention layer was added after the global average pooling layer. A flattening layer has been added before the self-attention layer that converts the input into 1D. The first channel is passed to the Softmax function that combines with the second channel for the attention map creation. After that, the generated attention map is combined with a third channel for final attention features that are further utilized to classify AD stages.

3.2.5.1 Self-attention

The internal attention approach, sometimes called the self-attention (SA) strategy, uses internal information to automatically identify and highlight relevant information without needing external information. SA has low computational complexity and allows parallel computing. It consists of three characteristics matrices such as X , Y , and V , where these are defined by Equations 5–9.

$$\{Y, X\} \in R^{T \times T} \quad (5)$$

$$V \in R^{T \times J} \quad (6)$$

Initially, the correlation score has been computed among all rows of Y and X as follows:

$$P = XY^T \quad (7)$$

where Y^T denotes the transpose of Y and $P \in R^{T \times T}$. The softmax function is applied in the next step, which converts the correlation score into probability values. Mathematically, it is formulated as follows:

$$SM(P)(i, j) = \frac{e^{P(i, j)}}{\sum_{j=0}^{T-1} e^{P(i, j)}} \quad (8)$$

Hence, the final attention map has been obtained as follows:

$$AMp = SM(P)V \quad (9)$$

3.2.5.2 Proposed network training

After the design of the proposed model, the next step is training a model using the deep transfer learning concept. The entire model is trained from scratch, instead of any frozen layer. The hyperparameters of this network are presented in Table 3. Based on the selected hyperparameters using BO, the proposed model is trained on the augmented dataset. The best-returned value of the learning rate using BO is 0.00032, and the momentum value is 0.773. After the training process, the test data are employed for the extraction of the features.

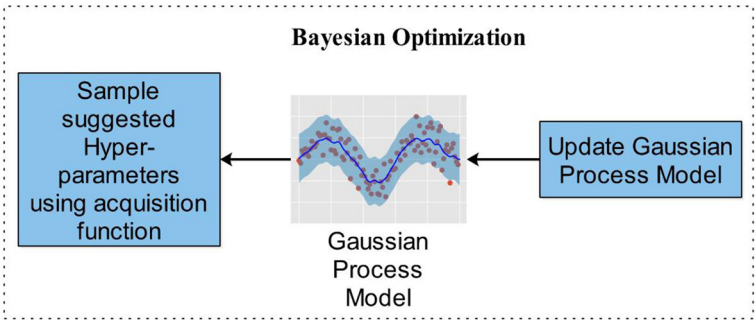


FIGURE 6
Bayesian optimization workflow for hyperparameters selection.

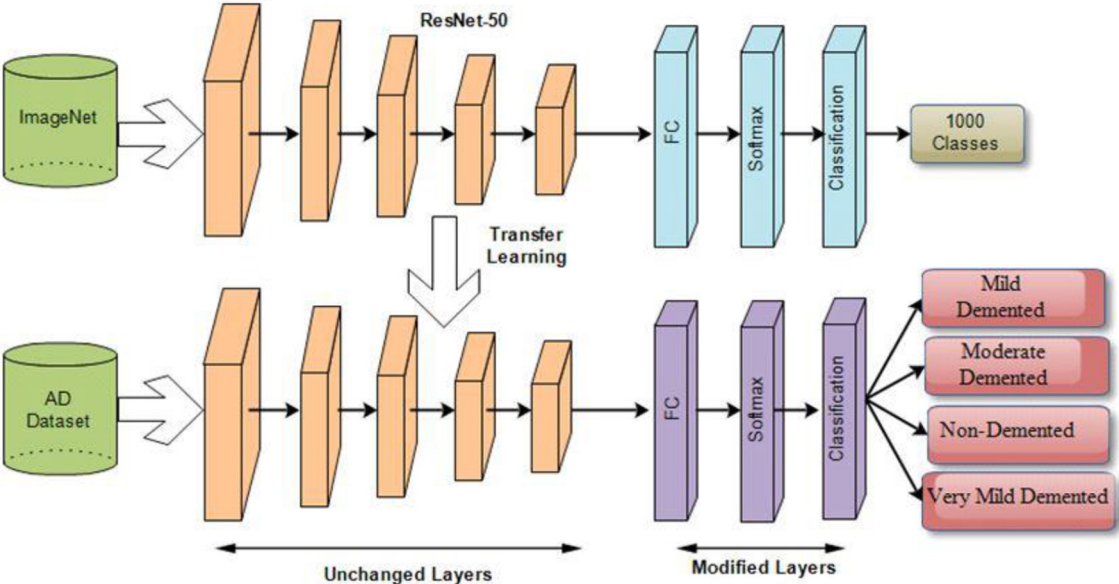


FIGURE 7
Deep transfer learning architecture for classification of AD stages.

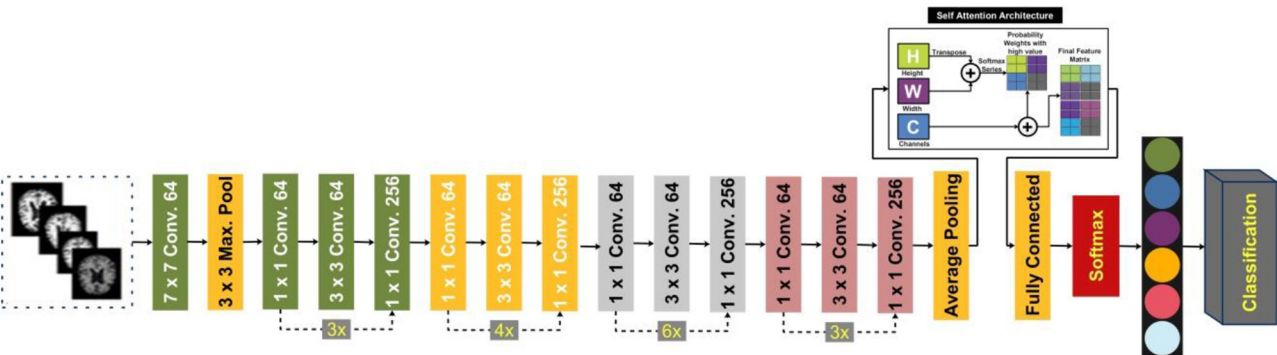


FIGURE 8
Proposed ResNet-Self architecture for classification of AD stages.

3.2.5.3 Deep features extraction

Typically, CNNs return the three levels of feature maps: low-level feature maps, mid-level feature maps, and high-level feature maps. All of these levels contain different information. Low-level feature maps contain simple patterns such as edges, corners, and textures. These maps have high dimensions. Mid-level feature maps contain abstract and more structured patterns like specific regions of the objects or textures, but high-level features contain discriminative and semantically significant information. The features (high-level) are extracted from the last stage of CNNs due to their lower dimensions. The low dimensionality reduces the memory requirement and computational complexity.

This study extracts deep features using the self-attention layer, instead of the global average pooling layer. The self-attention layer returns the prominent and relevant global information within the images. The testing images are utilized, and a trained model is opted for. The batch size was 128 during the deep features extraction. The self-attention layer features contain deeper information about the AD stages. The size of the extracted feature vector is $N \times 2,048$. The extracted features are analyzed and optimized using an improved pathfinder optimization algorithm.

3.2.6 Best features selection

In this study, we utilized an evolutionary optimization algorithm named Entropy Path Finder Optimization (EPFO) for the best feature selection. Features are selected at the initial step through original pathfinder optimization and later refined using an entropy approach that handles the uncertainty.

3.2.6.1 Path finder optimization

In contrast to previously suggested swarm intelligence, the Pathfinder approach does not specify which species group it belongs to. For instance, the seagull optimization algorithm restricts the number of seagulls, whereas the gray wolf optimization technique restricts the number of gray wolves, etc. The Pathfinder algorithm is based on many surviving traits and regulations of animals. Based on the best fitness of the organism, the Pathfinder algorithm divides cluster animals among two sorts of tasks: the leader (only with the lowest fitness value) and the follower. The leader must find the greatest food and label it for the followers. The markings left behind by the Pathfinder are used as a reference point by the followers, who then proceed to follow the Pathfinder. Hence, both the Pathfinder and the follower are skeptical. That is why the two distinct sorts of responsibilities may switch places with one another depending on the individual's level of search capability as the number of iterative steps of the method rises; that is, those who lead the way sometimes get followers. Similarly, followers may also play the role of a pathfinder. To optimize a task, the PFA is split into two segments. The initial stage is a period of exploration. The PFA changes the location using the following Equation (10):

$$x_p^{k+1} = x_p^k + 2r3 \cdot (x_p^k - x_p^{k-1}) + A \quad (10)$$

where x_p^{k+1} demonstrates the modified position vector of PFA. x_p^k indicates the present location of vectors in PFA, while x_p^{k-1} shows the former position of the vector of PFA. The ongoing iteration

count is denoted by the variable k . $R3$ is a random vector that is created in a uniform manner in the range $[0,1]$, whereas A is produced within every iteration by applying Equation (11). Step 2 is really the exploitation step, which is immediately preceded by the location change. The following update formula applies Equation (11):

$$x_i^{k+1} = x_i^k + R1 \cdot (x_j^k - x_i^k) + R2 \cdot (x_p^k - x_i^k) + E, i \geq 2 \quad (11)$$

where x_i^{k+1} indicates the updated location vector of the i -th integer just after location modification. x_i^k is the location vector of the i -th individual, x_j^k is the neighboring individual, and x_p^k is the Pathfinder. The variable k denotes the ongoing iteration count. Each of the vectors $R1$ and $R2$ is completely unpredictable. In this situation, $R1 = (\alpha r1)$ and $R2 = (\beta r2)$, and here, $R1$ and $R2$ are random vectors that are created uniformly in the range $[0,1]$. α determines the degree to which each component travels about its neighbors and is hence called the coefficient of iteration. β establishes a randomized spacing to make the herd fairly constant along with the leader and hence called the coefficient of attraction k_{max} . Mathematically, it is formulated by Equation (12).

$$h = \left(1 - \frac{k}{k_{max}}\right) \cdot \mu_1 \cdot D_{ij}, D_{ij} = \|x_i - x_j\| \quad (12)$$

Therefore, here μ_1 and μ_2 are randomly generated two vectors in the interval of $[1,1]$, D_{ij} is the gap between both individuals, (k) denotes the present iteration range, and k_{max} is the maximal quantity of repetitions. (A) and (h) may give random walk strides for all persons when the second part of Equations (10) and (11) and the third part of Equation (12) are equal to zero. As a result, in order to ensure that the motion will be in several directions and completely random, the values of (A) and (h) should be within the proper span.

After every update in the position, the KNN classifier is employed to measure the fitness value. The cost function of KNN is mathematically formulated as:

$$\tau_{cost} = \varphi_\alpha \times \epsilon_{err} + \varphi_\beta \times \left(\frac{\text{count of sel_feat}}{\text{Max(features)}}\right) \quad (13)$$

where α and β are denoted, the coefficient having values are 0.94 and 0.014, respectively. The ϵ_{err} presented the error value that is calculated by employing an Equation (14):

$$\epsilon_{err} = 1 - \partial_{accuracy} \quad (14)$$

3.2.6.2 Entropy selection

Assume U is a discrete random variable, and it is represented as $u = \{u_1, u_2, \dots, u_n\}$, then if an element u_i occurs with $p(u_i)$, the entropy $H(U)$ of U is formulated by Equation (15):

$$H(U) = - \sum_{i=1}^n p(u_i) \log p(u_i) \quad (15)$$

```

Load self-attention feature vector
Load parameters of PFA
Initialize the population of size 20
Calculate the fitness of initial
population
Find the Pathfinder
While k<maximum number of iterations
    α and β = random number in [1,2]
    update the position of Pathfinder
    using Equation (10) and check the
    bound
    if the new Pathfinder is better than old
        update the Pathfinder
    end
    for i=2 to maximum number of populations
        update the position of members using
        the Equation (11) and check the bound
    end
    calculate the new fitness of members
    find the best fitness using
    Equations (13) and (14)
    if best fitness < fitness of Pathfinder
        pathfinder = best member
        fitness = best fitness
    end
    for i=2 to maximum number of populations
        if new fitness of member (i) <
        fitness of member (i)
            update member
            - Find entropy of updated member using
            Eq. 15.
            - Compute the fuzziness using Eqs. 16-18
            - Find the fuzzy entropy value using
            Eq. 19.
        end
    end
    generate new A and ε
end

```

Algorithm 1. The pathfinder algorithm.

where n denotes the total number of features. The fuzzy C-Means clustering is utilized to construct the membership function of all features. The fuzzy membership method is defined in the following five steps.

In the first step, we assumed the number of clusters (C), where $2 \leq C \leq N$. In the next step, the j th center clusters are computed by the following Equation (16).

$$C_j = \frac{\sum_{i=1}^N \mu_{ij}^e u_{ij}}{\sum_{i=1}^N \mu_{ij}^g} \quad (16)$$

where $e \geq 1$ is a fuzziness coefficient and μ_{ij} is the degree of membership (DOM) for the i th data point u_i in j th cluster. Euclidean distance is computed in the third step using Equation (17).

$$D_{ij} = |C_j - \mu_i| \quad (17)$$

In the fourth step, the value of the fuzzy membership function is updated by Equation (18):

$$\mu = \frac{1}{\sum_{m=1}^e \left(\frac{D_{ij}}{D_{im}} \right)^{\frac{2}{e-1}}} \quad (18)$$

In the final step, we repeated steps 2–4 until the change in μ was less as per the previous values. Hence, the fuzzy entropy function is formulated as follows in Equation (19):

$$Fe(\check{H}) = -\lambda_c(\check{H}) \log \lambda_c(\check{H}) \quad (19)$$

where $\lambda_c(\check{H})$ is a class degree of the membership function (Khushaba et al., 2007). The fuzzy entropy process is applied to the selected features of Equation (12). The dimensions of the selected features are $N \times 1,467$. The final features are employed for the classification. The proposed fuzzy entropy-controlled pathfinder algorithm's pseudo-code is given in Algorithm 1.

4 Results and analysis

The proposed AD stage classification model undergoes evaluation using a Kaggle dataset, providing a robust framework for assessing its performance. The forthcoming section will comprehensively showcase all the experiments conducted and the corresponding results obtained, offering insights into the efficacy and potential of the proposed model in accurately diagnosing AD.

4.1 Experimental setup and evaluation measures

The experimental process of this study is discussed here. The proposed framework of AD is evaluated on a publically available dataset that includes four classes as mentioned in Section 3.1. The dataset is divided into 50:50 approaches, and training data augmentation is performed. The training data extracts and optimizes features for the best feature selection. The selected features are classified using machine learning classifiers, and the following measures are computed: recall rate, precision rate, F1-Score, MCC, and KAPPA. The entire experimental process has been conducted on MATLAB2023a using a personal computer with 128GB RAM, 512GB SSD, and a 12GB Graphics Card of NVIDIA3060 RTX.

4.2 Proposed ResNet-Self results (random values)

The proposed ResNet-Self CNN architecture is tested on 1,600 images in this experiment. The hyperparameters of this experiment are randomly initialized (related work knowledge such as learning rate 0.0001 and momentum 0.70) and performed training. Features are extracted from the testing data, and the maximum accuracy

TABLE 4 Proposed prediction results of AD stages using initialization of random hyperparameters.

S#	Classifiers	Precision	Recall	F1-score	Kappa	MCC	Accuracy	Time (s)
1	Fine KNN	98.73	98.73	98.72	96.60	98.30	98.7	66.002
2	NNN	99.93	99.93	99.92	99.80	99.90	99.93	33.532
3	MNN	99.90	99.90	99.90	99.73	99.87	99.90	50.399
4	Trilayered NN	99.65	99.65	99.65	99.07	99.53	99.65	41.09
5	Medium KNN	98.99	98.97	98.97	97.27	98.64	98.97	62.531
6	Coarse KNN	83.00	97.60	87.50	30.31	84.03	73.86	63.041
7	Cosine KNN	98.78	98.75	98.75	96.67	98.35	98.75	66.134
8	Bilayered NN	99.88	99.88	99.87	99.67	99.83	99.88	28.071
9	Medium Gaussian SVM	99.73	99.72	99.73	99.27	99.63	99.72	73.511

Bold values shows the best results.

TABLE 5 Proposed classification results after employing Bayesian optimization-based selection of hyperparameters.

S#	Classifiers	Precision	Recall	F1-score	Kappa	MCC	Accuracy	Time
1	Fine KNN	98.48	98.48	98.47	95.93	97.97	98.48	35.573
2	NNN	99.90	99.90	99.90	99.73	99.87	99.90	15.874
3	MNN	99.95	99.95	99.95	99.87	99.93	99.95	17.78
4	Trilayered NN	99.75	99.75	99.75	99.33	99.67	99.75	21.891
5	Medium KNN	98.75	98.72	98.72	96.60	98.31	98.72	34.139
6	Coarse KNN	97.11	97.08	97.07	92.20	96.11	97.08	34.571
7	Cosine KNN	98.45	98.40	98.40	95.73	97.89	98.40	37.183
8	Bilayered KNN	99.58	99.58	99.58	98.87	99.43	99.58	34.292
9	Medium Gaussian SVM	99.73	99.72	99.73	99.27	99.63	99.72	34.235

Bold values shows the best results.

of 99.93% for the NNN classifier was obtained (results seen in Table 4). The values of precision measure are 99.93%, and the Kappa value is 99.80%, respectively. The computational time taken by the NNN classifier is 33.532 (s), whereas the minimum noted time is 28.071 (s) for bilayered NN. The rest of the classifiers obtained accuracies of 98.7, 99.90, 99.65, 98.97, 73.86, 98.75, 99.88, and 99.72%, respectively.

4.3 Bayesian optimization results

This section presents the results obtained from Bayesian optimization (BO). We executed our BO algorithm 100 times and got the value for a learning rate of 0.00010195, momentum value of 0.81079, L2Regularization of $2.8724e^{-10}$, and section depth value of 3. These are the best feasible points. Based on these points, the classification was performed, and the results are noted in Table 5. The MNN classifier achieved a maximum accuracy of 99.95% in this table. The precision rate of this classifier is 99.95, the Kappa value of 99.87, and the MCC value of 99.95%, respectively. In addition, the computation time of this classifier is 17.78 (s). Compared to the results in Table 4, this experiment shows improved accuracy, precision, Kappa, and MCC values. Moreover, the computation

time of this experiment was less than that of the results in Table 4. The results show that selecting hyperparameters using BO can improve the accuracy and reduce the computational cost.

4.4 Proposed feature selection

Table 6 presents the AD stage classification results using the proposed selection of BO extracted features. In the first stage of this table, results are presented for the original pathfinder algorithm. The PFA was applied to the BO-based deep features extraction and performed classification. The maximum obtained accuracy for this experiment is 99.82%. The precision and recall values are 99.83 and 99.83%. In addition, Kappa and MCC measure values of 99.80 and 99.80%, respectively. Compared to Tables 4, 5, the selection results show better. Moreover, the computation time of each classifier is also noted, and the minimum noted time for this experiment is 12.338 (s), which is less than Tables 4, 5. Overall, the time is decreased after employing the optimization method.

To further improve (minimize) the computational time, we improved the PFA using Fuzzy Entropy formulation in this study. The proposed Fuzzy Entropy PFA (FEPFA) results are given in the second half of Table 6. The maximum obtained accuracy for

TABLE 6 Proposed classification results after employing Bayesian optimization and proposed feature selection algorithm.

S#	Classifiers	Precision	Recall	F1-score	Kappa	MCC	Accuracy	Time
Features selection using original PFA								
1	Fine KNN	94.34	94.32	94.32	84.82	92.43	94.31	23.406
2	NNN	99.83	99.83	99.82	99.80	99.80	99.82	12.338
3	MNN	99.82	99.80	99.82	99.47	99.80	99.80	14.71
4	Trilayered NN	99.80	99.80	99.80	99.47	99.73	99.80	16.088
5	Medium KNN	98.94	98.92	98.92	97.13	98.57	98.92	14.9
6	Coarse KNN	96.64	96.57	96.57	90.87	95.46	96.57	14.67
7	Cosine KNN	98.73	98.70	98.70	96.53	98.28	98.70	19.92
8	Bilayered KNN	99.88	99.88	99.87	99.67	99.83	99.80	15.04
9	Medium Gaussian SVM	99.73	99.73	99.73	99.27	99.63	99.73	17.014
Features selection using proposed fuzzy entropy PFA								
1	Fine KNN	99.90	99.90	99.90	99.73	99.87	99.90	18.987
2	NNN	99.93	99.92	99.92	99.80	99.90	99.90	10.231
3	MNN	99.90	99.90	99.90	99.73	99.87	99.90	13.395
4	Trilayered NN	99.73	99.73	99.72	99.27	99.63	99.73	11.411
5	Medium KNN	99.04	99.03	99.03	97.41	98.71	99.03	7.167
6	Coarse KNN	97.05	97.00	96.99	92.00	96.02	97.00	6.77
7	Cosine KNN	98.74	98.70	98.70	96.53	98.28	98.70	8.683
8	Bilayered KNN	99.85	99.82	99.85	99.60	99.80	99.85	10.336
9	Medium Gaussian SVM	99.70	99.70	99.70	99.20	99.60	99.70	9.118

Bold values shows the best results.

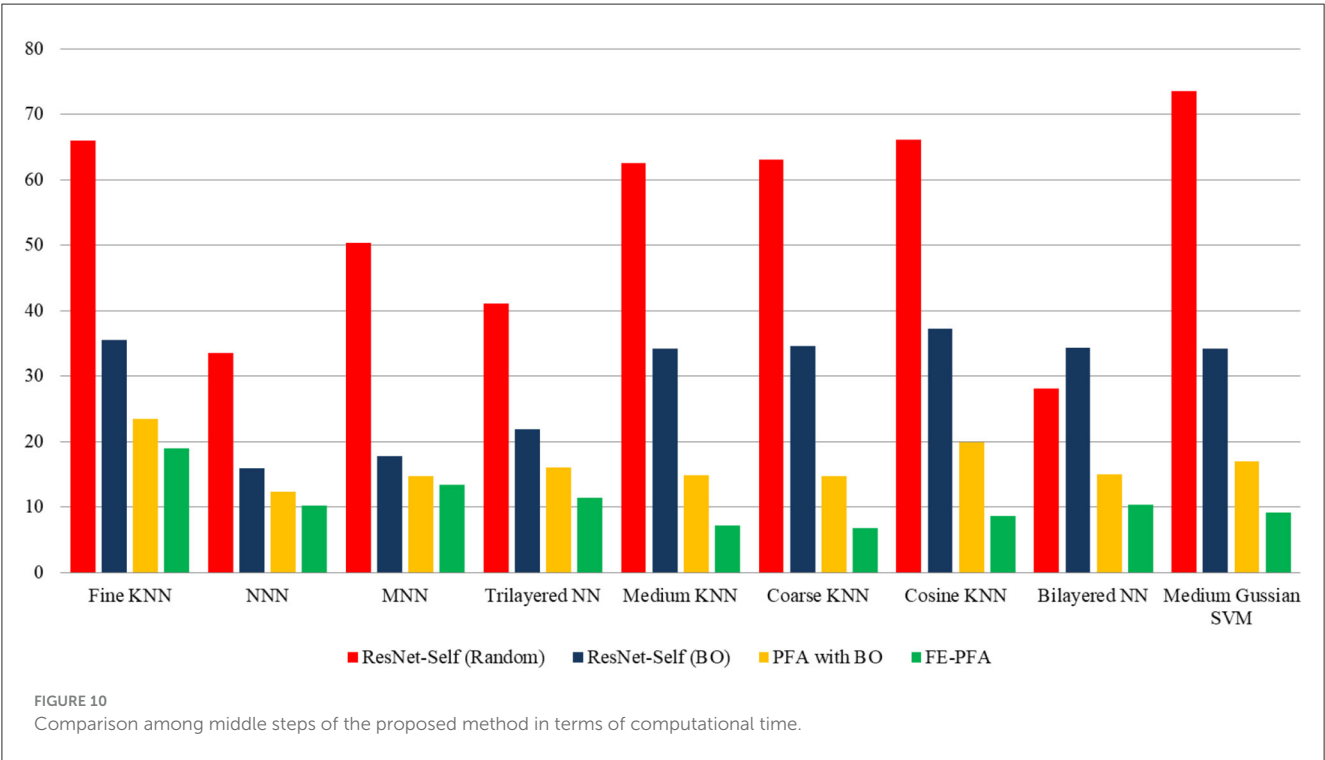
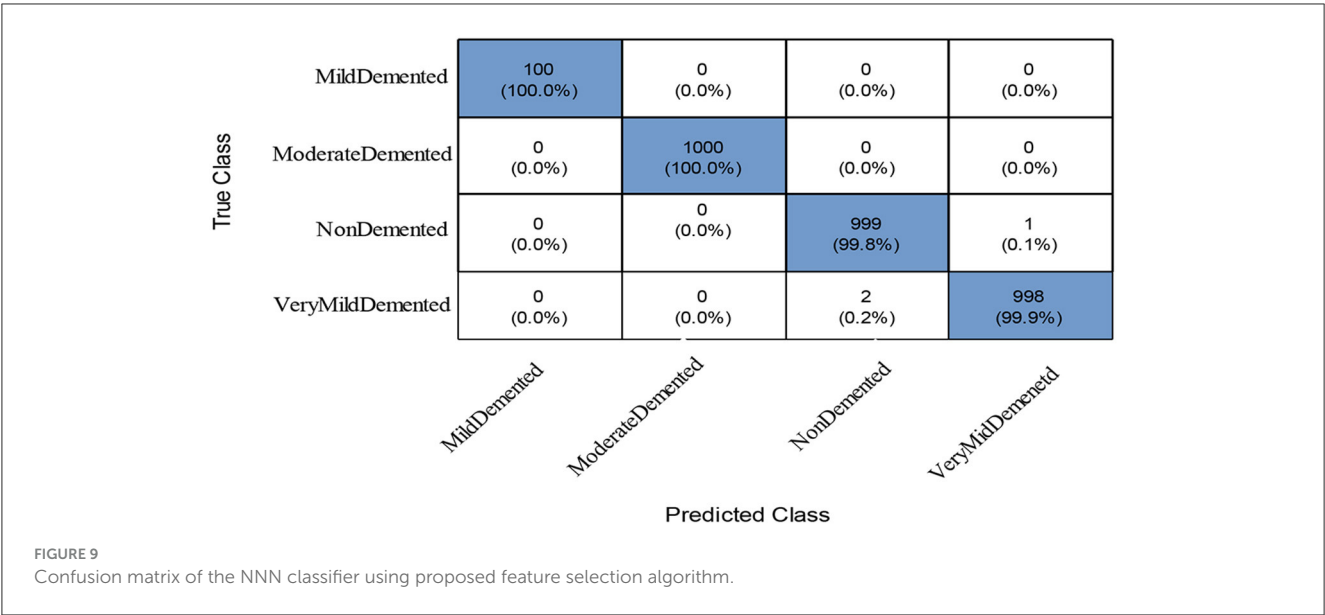
this technique is 99.90%, whereas the precision rate value of 99.93%. The Kappa and MCC values of this experiment are 99.80 and 99.90%, respectively. In addition, the computation time of this classifier is 10.231 (s), less than the original PFA (12.338). Overall, the performance of this technique is improved and time is minimized. The performance of the NNN classifier can be further verified using a confusion matrix illustrated in Figure 9. In this figure, the diagonal values represent the true predicted rates of each class.

4.5 Discussion and comparison

In this section, a detailed analysis of the proposed study has been conducted in the form of visual graphs and comparison with recent state-of-the-art (SOTA) techniques. The proposed framework of AD stage classification has been discussed in Section 3.1, and the visual illustration is shown in Figure 1. The MRI dataset has been used for the experimental process (a few sample images are shown in Figures 2, 3). The augmentation process has been performed to increase the number of images for a better training process. After that, a new model is proposed named ResNet-Self as shown in Figure 8 for the accurate classification of AD stages. The performance of AD stage classification is improved

by proposing new FEPFA techniques that select the best features. The results are presented in Tables 4–6. Table 4 presents results for the proposed ResNet-Self architecture using random initialization of hyperparameters. Table 5 presents the results of the proposed ResNet-Self after employing BO for hyperparameters selection. Table 5 shows better accuracy, precision rate, MCC, and Kappa performance than Table 4. The computational time and precision rate are further improved using the proposed FEPFA feature selection algorithm, and the results are presented in Table 6. In this table, accuracy is also improved and time is significantly decreased. In addition, a comparison is also conducted of the proposed FEPFA with the original PFA, showing the improvement in accuracy, precision, MCC, and computational time. Overall, the time comparison is illustrated in Figure 10. This figure clearly shows that the proposed selection method consumed less time than the other steps.

Table 7 compares the methods currently utilized for predicting AD. To enhance the categorization of early AD phases while reducing parameters and computational costs, a novel detection network named DAD-Net was introduced by Mohi et al. (2023). This network appropriately classified initial AD processes and depicted class activation characteristics as a heat map of the brain, achieving 99.2% accuracy using a Kaggle dataset. Additionally, AI-Atroshi et al. (2022) utilized convolutional layers with freeze



elements from ImageNet, achieving 99.27% accuracy on ADNI's MRI data collection for both binary and ternary classification. Authors in [Shankar et al. \(2022\)](#) employed a ResNet-18 architecture using a transfer learning concept and obtained an accuracy of 83.3% on Kaggle datasets. Authors in [Sharma et al. \(2022\)](#) utilized a CNN-based pre-trained network named ResNet-50 and achieved 91.78% accuracy. Authors in [Albright \(2019\)](#) proposed a ResNet-15 model and fused it with DenseNet-169 for the classification of AD prediction. They achieved an improved accuracy of 88.70% on Kaggle's AD dataset. Furthermore, [Soliman et al. \(2022\)](#) suggested a novel approach employing three pre-trained

CNN frameworks such as DenseNet196, VGG16, and ResNet-50, achieving 89% accuracy on MRI brain data from Kaggle. [Hashmi \(2024\)](#) proposed a compact architecture by merging LeNet and AlexNet models, achieving 93.58% accuracy on the ADNI dataset. [Goel et al. \(2023\)](#) proposed a system for automated AD diagnosis, integrating multiple customized deep-learning models. This architecture achieved 96.61% accuracy using rs-fMRI datasets and modified AlexNet and Inception blocks. [Ismail et al. \(2023\)](#) utilized a new optimized ensemble-based DNN learning model named MultiAz-Net and obtained 92.3% accuracy on the ADNI dataset.

TABLE 7 Comparison of proposed method results with existing techniques.

References	Years	Models	Datasets	Results
Ahmed et al. (2022)	2022	CNN based DAD-Net	Kaggle	99.22%
Naz et al. (2022)	2022	CNN using freeze features	ADNI	99.27%
Oktavian et al. (2002)	2022	CNN with ResNet-18	Kaggle	83.3%
Ebrahimi et al. (2021)	2021	CNN, ResNet-18, temporalCN, RNN	ImageNet	91.78%
Al Shehri (2022)	2022	ResNet-15	Kaggle	88.70%
Techa et al. (2022)	2022	ResNet-15	Kaggle	89%
Abunadi (2022)	2022	ResNet-18, AlexNet	Kaggle	99.94%
Prasath and Sumathi (2024)	2023	LeNet, AlexNet	ADNI	93.58%
Sorour et al. (2024)	2023	AlexNet, Inception blocks	ADNI	96.61%
Ismail et al. (2023)	2023	MOGOA	ADNI	92.3%
Proposed model		ResNet-50	Kaggle	99.99%

Bold values shows the best results.

5 Conclusion and future study

It is challenging to diagnose and predict Alzheimer’s disease using multiclass datasets promptly. A computerized technique is widely required for early AD prediction from MRI images. This study proposes a computerized framework based on deep-learning and optimization algorithms. A dataset balancing issue has been resolved at the initial stage using mathematical formulations that improved the training capability of the proposed ResNet-Self deep model. The proposed ResNet-Self model is a combination of ResNet-50 architecture modified by adding the self-attention module. The self-attention module shows improved accuracy; however, the random initialization of hyperparameters impacts the accuracy and computational time. Therefore, we implemented a BO technique that automatically initialized the hyperparameters for the training process. Moreover, we proposed a feature selection algorithm named FEcPFA that selects the best features and shows improved accuracy (99.90), precision rate, and Kappa value. In addition, the computational time is significantly reduced, which is the strength of FEcPFA. The optimized hyperparameters that make the proposed model less generalized and lead to overfitting are the limitations of the proposed framework. In the future, a new custom model will be proposed based on the fire module, and the output of that module will be employed with self-attention and cross-validation to overcome overfitting. In addition, more MRI datasets will be utilized for the experimental process.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding authors.

Author contributions

NY: Methodology, Project administration, Resources, Software, Writing—original draft. MK: Methodology, Project administration,

Resources, Software, Validation, Writing—original draft. SM: Formal analysis, Investigation, Methodology, Resources, Software, Writing—original draft. HA: Conceptualization, Data curation, Methodology, Project administration, Software, Writing—review & editing. AH: Data curation, Methodology, Software, Validation, Writing—original draft. FA: Conceptualization, Data curation, Funding acquisition, Investigation, Methodology, Visualization, Writing—review & editing. LJ: Conceptualization, Funding acquisition, Methodology, Software, Supervision, Validation, Visualization, Writing—review & editing. AM: Funding acquisition, Methodology, Software, Validation, Writing—original draft.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This research was funded by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R719), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Acknowledgments

The authors extend their appreciation to Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R719), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Authors are also thankful to NTNU for support in this work.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated

organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Abunadi, I. (2022). Deep and hybrid learning of MRI diagnosis for early detection of the progression stages in Alzheimer's disease. *Conn. Sci.* 34, 2395–2430. doi: 10.1080/09540091.2022.2123450
- Ahmed, G., Er, M. J., Fareed, M. M. S., Zikria, S., Mahmood, S., He, J., et al. (2022). Dad-net: classification of alzheimer's disease using adasyn oversampling technique and optimized neural network. *Molecules* 27:7085. doi: 10.3390/molecules2707085
- AI-Atroshi, C., Rene Beulah, J., Singamaneni, K. K., Pretty Diana Cyril, C., Neelakandan, S., and Velmurugan, S. (2022). Automated speech based evaluation of mild cognitive impairment and Alzheimer's disease detection using with deep belief network model. *Int. J. Healthc. Manag.* 21, 1–11. doi: 10.1080/20479700.2022.2097764
- Al Shehri, W. (2022). Alzheimer's disease diagnosis and classification using deep learning techniques. *PeerJ Comp. Sci.* 8:e1177. doi: 10.7717/peerj-cs.1177
- Albright, J. (2019). Forecasting the progression of Alzheimer's disease using neural networks and a novel pre-processing algorithm. *Alzheimers Dement* 5, 483–491. doi: 10.1016/j.trci.2019.07.001
- Balaji, P., Chaurasia, M. A., Bilfaqih, S. M., Muniasamy, A., and Alsaid, L. E. G. (2023). Hybridized deep learning approach for detecting Alzheimer's disease. *Biomedicine* 11:149. doi: 10.3390/biomedicine11010149
- Bloniecki Kallio, V. (2002). Using CSF Biomarkers to Understand Mechanisms of Behavioral Changes and Effects of Drug Treatment in Dementia.
- Bondi, M. W., Edmonds, E. C., and Salmon, D. P. (2017). Alzheimer's disease: past, present, and future. *J. Int. Neuropsychol. Soc.* 23, 818–831. doi: 10.1017/S135561771700100X
- Carle, P. (2022). *Senescent Human Astrocytes Produce Amyloid-Beta in a G3BP1-Dependent Manner*. McGill University.
- Chua, J. J. E. (2023). HEBP1-An early trigger for neuronal cell death and circuit dysfunction in Alzheimer's disease. *Semin. Cell Dev. Biol.* 139, 102–110. doi: 10.1016/j.semcdb.2022.07.005
- Dhakhnamoorthy, C., Mani, S. K., Mathivanan, S. K., Mohan, S., Jayagopal, P., Mallik, S., et al. (2023). Hybrid whale and gray wolf deep learning optimization algorithm for prediction of Alzheimer's disease. *Mathematics* 11:1136. doi: 10.3390/math11051136
- Ebrahimi, A., Luo, S., and Chiong, R. (2021). Deep sequence modelling for Alzheimer's disease detection using MRI. *Comput. Biol. Med.* 134:104537. doi: 10.1016/j.combiomed.2021.104537
- Fabrizio, C., Termine, A., Caltagirone, C., and Sancesario, G. (2021). Artificial intelligence for Alzheimer's disease: promise or challenge?. *Diagnostics* 11:1473. doi: 10.3390/diagnostics11081473
- Ghazal, T. M., Abbas, S., Munir, S., Khan, M. A., Ahmad, M., Issa, G. F., et al. (2022). Alzheimer disease detection empowered with transfer learning. *Comp. Mater. Continua* 70:020866. doi: 10.32604/cmc.2022.020866
- Goel, T., Sharma, R., Tanveer, M., Suganthan, K., and Pilli, R. (2023). Multimodal neuroimaging based Alzheimer's disease diagnosis using evolutionary RVFL classifier. *IEEE J. Biomed. Health Inf.* 3, 1–21. doi: 10.1109/JBHI.2023.3242354
- Gómez-Isla, T., and Frosch, M. P. (2022). Lesions without symptoms: understanding resilience to Alzheimer disease neuropathological changes. *Nat. Rev. Neurol.* 18, 323–332. doi: 10.1038/s41582-022-00642-9
- Hashmi, S. A. (2024). Malware detection and classification on different dataset by hybridization of CNN and machine learning. *Int. J. Intell. Syst. Appl. Eng.* 12, 650–667. Available online at: <https://ijisae.org/index.php/IJISAE/article/view/4004>
- Hazarika, R. A., Maji, A., Kandar, D., Jasińska, E., Krejci, P., Leonowicz, Z., et al. (2023). An approach for classification of Alzheimer's disease using deep neural network and brain magnetic resonance imaging (MRI). *Electronics* 12:676. doi: 10.3390/electronics12030676
- Hoozemans, J., Veerhuis, R., Rozemuller, J., and Eikelenboom, P. (2006). Neuroinflammation and regeneration in the early stages of Alzheimer's disease pathology. *Int. J. Dev. Neurosci.* 24, 157–165. doi: 10.1016/j.ijdevneu.2005.11.001
- Ismail, W. N., Rajeeva, P. P. F., and Ali, M. A. (2023). A meta-heuristic multi-objective optimization method for Alzheimer's disease detection based on multimodal Data. *Mathematics* 11:957. doi: 10.3390/math11040957
- Jansi, R., Gowtham, N., Ramachandran, S., and Praneeth, V. S. (2023). "Revolutionizing Alzheimer's disease prediction using InceptionV3 in deep learnin," in *2023 7th International Conference on Electronics, Communication and Aerospace Technology (ICECA) (IEEE)*, 1155–1160.
- Jia, H., and Lao, H. (2022). Deep learning and multimodal feature fusion for the aided diagnosis of Alzheimer's disease. *Neur. Comp. Appl.* 34, 19585–19598. doi: 10.1007/s00521-022-07501-0
- Jo, T., Nho, K., Bice, P., Saykin, A. J., and Alzheimer's Disease Neuroimaging Initiative (2022). Deep learning-based identification of genetic variants: application to Alzheimer's disease classification. *Brief. Bioinform.* 23:bbac022. doi: 10.1093/bib/bbac022
- Kasula, B. Y. (2023). A machine learning approach for differential diagnosis and prognostic prediction in Alzheimer's disease. *Int. J. Sustain. Dev. Comp. Sci.* 5, 1–8. Available online at: <https://www.ijscs.com/index.php/ijscs/article/view/397>
- Kellar, D., and Craft, S. (2020). Brain insulin resistance in Alzheimer's disease and related disorders: mechanisms and therapeutic approaches. *Lancet Neurol.* 19, 758–766. doi: 10.1016/S1474-4422(20)30231-3
- Khalid, A., Senan, E. M., Al-Wagih, K., Ali Al-Azzam, M. M., and Alkhraisha, Z. M. (2023). Automatic analysis of MRI images for early prediction of Alzheimer's disease stages based on hybrid features of CNN and handcrafted features. *Diagnostics* 13:1654. doi: 10.3390/diagnostics13091654
- Khushaba, R. N., Al-Jumaily, A., and Al-Ani, A. (2007). "Novel feature extraction method based on fuzzy entropy and wavelet packet transform for myoelectric control," in *2007 International Symposium on Communications and Information Technologies (IEEE)*, 352–357.
- Koul, A., Bawa, R. K., and Kumar, Y. (2023). An analysis of deep transfer learning-based approaches for prediction and prognosis of multiple respiratory diseases using pulmonary images. *Arch. Comp. Methods Eng.* 6, 1–27. doi: 10.1007/s11831-023-10006-1
- Mahmud, T., Barua, K., Habiba, S. U., Sharmen, N., Hossain, M. S., Andersson, K., et al. (2024). An explainable AI paradigm for Alzheimer's diagnosis using deep transfer learning. *Diagnostics* 14:345. doi: 10.3390/diagnostics14030345
- Marwa, E.-G., Moustafa, H., E.-D., Khalifa, F., and Khater, H. (2023). An MRI-based deep learning approach for accurate detection of Alzheimer's disease. *Alexandria Eng. J.* 63, 211–221. doi: 10.1016/j.aej.2022.07.062
- Mirzaei, G., and Adeli, H. (2022). Machine learning techniques for diagnosis of alzheimer disease, mild cognitive disorder, and other types of dementia. *Biomed. Signal Process. Control* 72:103293. doi: 10.1016/j.bspc.2021.103293
- Mohammad, F., and Al Ahmadi, S. (2023). Alzheimer's disease prediction using deep feature extraction and optimisation. *Mathematics* 11:3712. doi: 10.3390/math11173712
- Mohi, G., Bhagat, A., Ansarullah, S. I., Othman, M. T. B., Hamid, Y., Alkahtani, H. K., et al. (2023). A novel framework for classification of different Alzheimer's disease stages using CNN model. *Electronics* 12:469. doi: 10.3390/electronics12020469
- Nagdee, M. (2011). Dementia in intellectual disability: a review of diagnostic challenges. *Afr. J. Psychiatry* 14, 194–199. doi: 10.4314/ajpsy.v14i3.1
- Naz, S., Ashraf, A., and Zaib, A. (2022). Transfer learning using freeze features for Alzheimer neurological disorder detection using ADNI dataset. *Multim. Syst.* 28, 85–94. doi: 10.1007/s00530-021-00797-3
- Oktavian, M. W., Yudistira, N., and Ridok, A. (2002). Classification of Alzheimer's disease using the convolutional neural network (CNN) with transfer learning and weighted loss. *arXiv [preprint]*. doi: 10.48550/arXiv.2207.01584
- Perumal, S., and Velmurugan, T. (2018). Pre-processing by contrast enhancement techniques for medical images. *Int. J. Pure Appl. Math.* 118, 3681–3688. Available online at: https://www.researchgate.net/profile/Velmurugan-Thambusamy/publication/325361080_Preprocessing_by_contrast_enhancement_techniques_for_medical_images/links/5fc1ee66a6fcc6c6774288/Preprocessing-by-contrast-enhancement-techniques-for-medical-images.pdf
- Prasath, T., and Sumathi, V. (2024). Pipelined deep learning architecture for the detection of Alzheimer's disease. *Biomed. Signal Process. Control* 87:105442. doi: 10.1016/j.bspc.2023.105442
- Rathore, S., Habes, M., Iftikhar, M. A., Shacklett, A., and Davatzikos, C. (2017). A review on neuroimaging-based classification studies and associated feature extraction

methods for Alzheimer's disease and its prodromal stages. *Neuroimage* 155, 530–548. doi: 10.1016/j.neuroimage.2017.03.057

Samhan, L. F., Alfarrar, A. H., and Abu-Naser, S. S. (2022). *Classification of Alzheimer's Disease Using Convolutional Neural Networks*.

Shamrat, F. M. J. M., Akter, S., Azam, S., Karim, A., Ghosh, P., Tasnim, Z., et al. (2023). AlzheimerNet: An effective deep learning based proposition for alzheimer's disease stages classification from functional brain changes in magnetic resonance images. *IEEE Access* 11, 16376–16395. doi: 10.1109/ACCESS.2023.3244952

Shankar, V. G., Sisodia, D. S., and Chandrakar, P. (2022). A novel discriminant feature selection-based mutual information extraction from MR brain images for Alzheimer's stages detection and prediction. *Int. J. Imaging Syst. Technol.* 32, 1172–1191. doi: 10.1002/ima.22685

Shanmugam, J. V., Duraisamy, B., Simon, B. C., and Bhaskaran, P. (2022). Alzheimer's disease classification using pre-trained deep networks. *Biomed. Signal Process. Control* 71:103217. doi: 10.1016/j.bspc.2021.103217

Sharma, R., Goel, T., Tanveer, M., and Murugan, R. (2022). FDN-ADNet: Fuzzy LS-TWSVM based deep learning network for prognosis of the Alzheimer's disease using the sagittal plane of MRI scans. *Appl. Soft Comput.* 115:108099. doi: 10.1016/j.asoc.2021.108099

Shaukat, N., Amin, J., Sharif, M., Azam, F., Kadry, S., Krishnamoorthy, S., et al. (2022). Three-dimensional semantic segmentation of diabetic retinopathy lesions and grading using transfer learning. *J. Pers. Med.* 12:1454. doi: 10.3390/jpm12091454

Shrager, Y., Kirwan, C. B., and Squire, L. R. (2008). Neural basis of the cognitive map: Path integration does not require hippocampus or entorhinal cortex. *Proc. Nat. Acad. Sci. U. S. A.* 105, 12034–12038. doi: 10.1073/pnas.0805414105

Sisodia, P. S., Ameta, G. K., Kumar, Y., and Chaplot, N. (2023). A review of deep transfer learning approaches for class-wise prediction of Alzheimer's disease using MRI images. *Arch. Comp. Methods Eng.* 30, 2409–2429. doi: 10.1007/s11831-022-09870-0

Soliman, S. A., El-Dahshan, E.-S. A., and Salem, A.-B. M. (2022). "Deep learning 3D convolutional neural networks for predicting Alzheimer's disease (ALD)" in *New Approaches for Multidimensional Signal Processing: Proceedings of International Workshop, NAMSP 2021* (Springer), 151–162.

Sorour, S. E., Abd El-Mageed, A. A., Albarrak, K. M., Alnaim, A. K., Wafa, A. A., El-Shafeiy, E., et al. (2024). Classification of Alzheimer's disease using MRI data based on deep learning techniques. *J. King Saud Univ.* 36:101940. doi: 10.1016/j.jksuci.2024.101940

Stevenson-Hoare, J., Heslegrave, A., Leonenko, G., Fathalla, D., Bellou, E., Luckcuck, L., et al. (2023). Plasma biomarkers and genetics in the diagnosis and prediction of Alzheimer's disease. *Brain* 146, 690–699. doi: 10.1093/brain/awac128

Tanveer, M., Richhariya, B., Khan, R. U., Rashid, A. H., Khanna, P., Prasad, M., et al. (2020). Machine learning techniques for the diagnosis of Alzheimer's disease: a review. *ACM Transact. Multim. Comp. Commun. Appl.* 16, 1–35. doi: 10.1145/3344998

Techa, C., Ridouani, M., Hassouni, L., and Anoun, H. (2022). "Alzheimer's disease multiclass classification model based on CNN and StackNet using brain MRI data," in *International Conference on Advanced Intelligent Systems and Informatics* (Springer), 248–259.

Wen, J., Thibeau-Sutre, E., Diaz-Melo, M., Samper-González, J., Routier, A., Bottani, S., et al. (2020). Convolutional neural networks for classification of Alzheimer's disease: overview and reproducible evaluation. *Med. Image Anal.* 63:101694. doi: 10.1016/j.media.2020.101694

Zhang, F., Pan, B., Shao, P., Liu, P., Alzheimer's Disease Neuroimaging Initiative; Australian Imaging Biomarkers Lifestyle flagship study of ageing, Shen, S., et al. (2022). A single model deep learning approach for Alzheimer's disease diagnosis. *Neuroscience* 491, 200–214. doi: 10.1016/j.neuroscience.2022.03.026



OPEN ACCESS

EDITED BY

Jussi Tohka,
University of Eastern Finland, Finland

REVIEWED BY

Meiyu Huang,
China Academy of Space Technology (CAST),
China
Mohammad Shabaz,
Model Institute of Engineering and
Technology, India

*CORRESPONDENCE

Hezi Roda
✉ rodahe@post.bgu.ac.il

RECEIVED 10 March 2024

ACCEPTED 10 May 2024

PUBLISHED 30 May 2024

CITATION

Roda H and Geva AB (2024) Semi-supervised
active learning using convolutional
auto-encoder and contrastive learning.
Front. Artif. Intell. 7:1398844.
doi: 10.3389/frai.2024.1398844

COPYRIGHT

© 2024 Roda and Geva. This is an
open-access article distributed under the
terms of the [Creative Commons Attribution
License \(CC BY\)](#). The use, distribution or
reproduction in other forums is permitted,
provided the original author(s) and the
copyright owner(s) are credited and that the
original publication in this journal is cited, in
accordance with accepted academic practice.
No use, distribution or reproduction is
permitted which does not comply with these
terms.

Semi-supervised active learning using convolutional auto-encoder and contrastive learning

Hezi Roda^{1*} and Amir B. Geva^{1,2}

¹Electrical and Computer Engineering, Ben-Gurion University, Be'er Sheva, Israel, ²InnerEye Ltd CTO, Herzliya, Israel

Active learning is a field of machine learning that seeks to find the most efficient labels to annotate with a given budget, particularly in cases where obtaining labeled data is expensive or infeasible. This is becoming increasingly important with the growing success of learning-based methods, which often require large amounts of labeled data. Computer vision is one area where active learning has shown promise in tasks such as image classification, semantic segmentation, and object detection. In this research, we propose a pool-based semi-supervised active learning method for image classification that takes advantage of both labeled and unlabeled data. Many active learning approaches do not utilize unlabeled data, but we believe that incorporating these data can improve performance. To address this issue, our method involves several steps. First, we cluster the latent space of a pre-trained convolutional autoencoder. Then, we use a proposed clustering contrastive loss to strengthen the latent space's clustering while using a small amount of labeled data. Finally, we query the samples with the highest uncertainty to annotate with an oracle. We repeat this process until the end of the given budget. Our method is effective when the number of annotated samples is small, and we have validated its effectiveness through experiments on benchmark datasets. Our empirical results demonstrate the power of our method for image classification tasks in accuracy terms.

KEYWORDS

active learning, contrastive learning, clustering, semi-supervised learning, human-in-the-loop

1 Introduction

In recent years, computer vision has made significant advancements, primarily driven by machine learning and, more specifically, deep learning. However, these methodologies are highly dependent on having a substantial number of labeled samples. Acquiring such a large volume of data poses a significant challenge for several reasons. Initially, the process of annotating images is time-intensive, ranging from a few seconds for simple image classification to several hours for more complex image segmentation tasks. This makes it impractical to annotate a large data set in a short time frame. Additionally, image annotation often requires specialized expertise, adding another layer of complexity. In some cases, annotations require professionals, which increases the cost and complexity of the annotation process.

An effective strategy to address these issues involves employing an active learning methodology. Active Learning, often abbreviated as AL, entails the process of selecting and prioritizing data that require labeling to have the most significant impact on the training of

a machine learning task. Through the utilization of AL, machine learning algorithms can enhance their accuracy using a reduced number of training labels, thereby economizing time and resources during model training. Settles (2009) provides a comprehensive overview of various AL techniques in machine learning. In essence, there are three primary scenarios where active learning can be beneficial for those seeking to maximize accuracy while minimizing the number of labeled instances, typically involving the submission of queries in the form of unlabeled data instances to be labeled by an oracle, such as a human annotator. These scenarios include membership query synthesis (Angluin, 1988), stream-based selective (Atlas et al., 1989) sampling, and pool-based sampling. In this research, we will be focused on the third scenario, pool-based sampling (Lewis, 1995).

In numerous practical scenarios, it is often straightforward to gather a substantial amount of unlabeled data, which serves as a driving force behind the adoption of the pool-based sampling method. Let us consider a pool of unlabeled data P_u alongside a limited quantity of labeled data P_l . In pool-based sampling in each query, we will sample a small amount of data from P_u and annotate it with human oracle, then add it to P_l . Assuming we have a good query that selects the most relevant samples from P_u , P_l will be a good representative group of P_u .

Employing a pool-based sampling active learning approach, where the model selects samples for annotation, can decrease the quantity of labeled data required to achieve a similar model accuracy. This represents a significant benefit of active learning for deep learning tasks, which has only recently started to be investigated (Gal et al., 2017; Sener and Savarese, 2017; Sinha et al., 2019).

As previously mentioned in numerous practical scenarios, there is a significant volume of unlabeled data, which motivate our study. In this research, we present a novel approach that utilizes pool-based active learning to fully exploit all unlabeled data. The method we suggest begins by clustering the unlabeled data in the latent space. Then, it proceeds to choose the samples with the highest entropy based on their representation in the latent space and the clustering within that space. Our central concept involves clustering the unlabeled data from P_u , querying samples with the highest entropy for human annotation, and employing labeled data from P_l to refine the clustering via our suggested clustering contrastive learning. The above process iterates until either a satisfactory level of accuracy is achieved, the model converges, or the annotation budget is exhausted.

In addition to addressing the challenges posed by limited labeled data, our research holds promise for real-world applications where unlabeled data is abundant. By leveraging a pool-based active learning approach, our method enables the effective utilization of unlabeled data in scenarios where acquiring labeled samples is impractical or costly, such as medical imaging diagnosis, satellite image analysis, and industrial inspection. This capability maximizes the efficiency and effectiveness of machine learning models in practical settings, facilitating improved accuracy and insights from limited labeled samples. Furthermore, our approach can identify and prioritize hard examples for labeling, ensuring that the annotated data provide the most informative training signal for the model.

The contributions of the research are:

- A new approach is proposed to integrate Deep Clustering and Deep Active Learning (DAL) in order to maximize the extraction of information from both labeled and unlabeled data.
- Propose a novel *contrastive clustering loss* (CCL) that has the potential to enhance the transition from unsupervised clustering to a semi-supervised framework.
- Achieving a high level of accuracy in image classification with a reduced number of labeled samples.

2 Previous work

2.1 Deep clustering

There has been significant research on deep clustering in recent years. Most deep clustering algorithms can be categorized into two groups. The first group includes two-stage clustering algorithms that first generate a data representation before applying clustering. These algorithms leverage existing unsupervised deep learning frameworks and techniques. For instance, Tian et al. (2014) and Peng et al. (2016) utilize autoencoders to learn low-dimensional features of original data samples and subsequently apply conventional clustering algorithms like k-means to the learned representations. Mukherjee et al. (2019) introduces ClusterGAN a generative adversarial network that clusters the latent space by sampling latent variables from a combination of one-hot encoded variables and continuous latent variables. The second group comprises approaches that simultaneously optimize feature learning and clustering. These algorithms aim to explicitly define a clustering loss, resembling the classification error in supervised deep learning. Yang et al. (2016) propose a recurrent framework that integrates feature learning and clustering into a unified model with a weighted triplet loss, optimizing it end-to-end. Xie et al. (2016) suggests a clustering loss that operates on the latent space of an autoencoder, enabling the simultaneous acquisition of feature representations and cluster assignments. Building upon this, Guo et al. (2017) DCEC (Deep Clustering with Convolutional Autoencoders) enhances the method by proposing Convolutional Autoencoders (CAE), which surpasses DEC while ensuring the preservation of local structure. This study directly adopts the clustering loss and clustering layer from DCEC.

We briefly review their definitions:

The trainable parameters of the clustering layer are μ_{j1}^k which represent the cluster center. The intuition behind the math operation of that layer is it maps each embedded point in the latent space z_i into a soft label q_i by the student's t-distribution (Van der Maaten and Hinton, 2008).

$$q_{ij} = \frac{(1 + \|z_i - \mu_j\|^2)^{-1}}{\sum_j (1 + \|z_i - \mu_j\|^2)^{-1}} \quad (1)$$

Where q_{ij} is the j th entry of q_i , representing the probability of z_i belonging to cluster j .

The clustering loss is defined as:

$$L_{clu} = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \quad (2)$$

where P is the target distribution, defined as:

$$p_{ij} = \frac{q_{ij}^2 / \sum_i q_{ij}}{\sum_j (q_{ij}^2 / \sum_i q_{ij})} \quad (3)$$

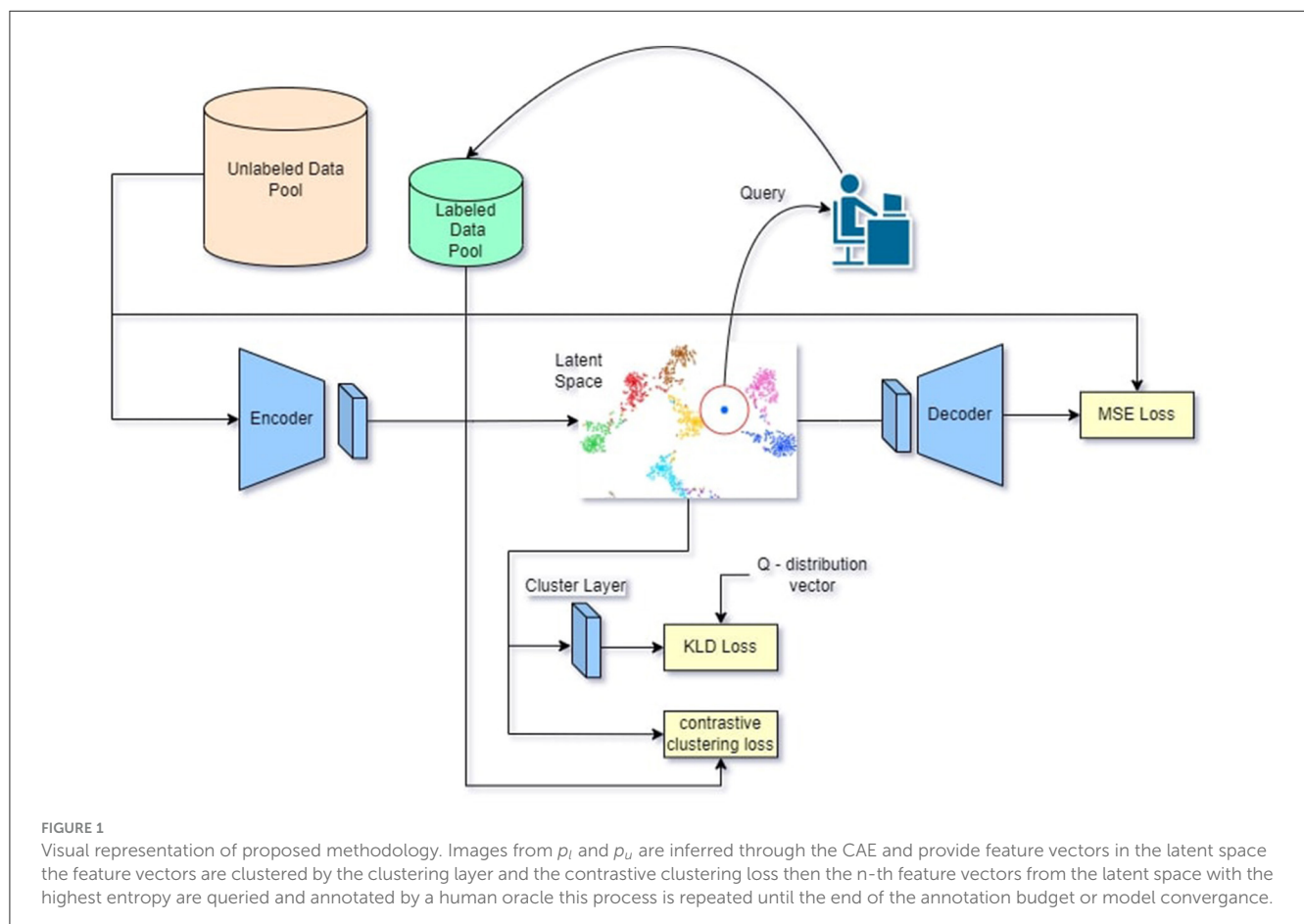
2.2 Active learning

Active learning is a subfield of machine learning empowering algorithms to select and prioritize the most informative data points for labeling, aiming to enhance model performance using less training data. Active learning scenarios commonly occur in three main contexts:

1. Membership Query Synthesis: In this scenario (Angluin, 1988), the learner synthesizes new instances to be labeled by an oracle, aiming to generate maximally informative instances, particularly beneficial when labeled data is scarce or expensive to obtain. 2. Stream-Based Selective Sampling: This scenario (Atlas et al., 1989) involves a continuous stream of unlabeled

instances, with the learner making real-time decisions on which instances to label based on the current model state and incoming data. Such scenarios are common in sequential data streams like online learning or sensor data. 3. Pool-Based Sampling: Here (Lewis and Gale, 1994), the learner is presented with a fixed pool of unlabeled instances and selects a subset for labeling, aiming to identify the most informative instances. This approach involves evaluating the informativeness of unlabeled samples, often utilizing query strategies like uncertainty sampling (Lewis and Gale, 1994), recently Liu and Li (2023) had an extensive work to explain this strategy even further, or query-by-committee (Seung et al., 1992). Active learning plays a crucial role in determining which data should be labeled to maximize the effectiveness of training supervised models. Traditional active learning methods are comprehensively reviewed by Settles (2009), while Ren et al. (2021) offer insights into the more contemporary Deep Active Learning (DAL) approach, integrating active learning with deep learning methodologies.

Notable active learning methodologies are Uncertainty Sampling (Lewis and Gale, 1994) and Variational Adversarial Active Learning (VAAL) (Sinha et al., 2019). VAAL integrates variational inference and adversarial training, leveraging a generator network to produce informative data points and a discriminator network to differentiate between real and generated instances, aiding in sample selection. Additionally, LADA (Kim



et al., 2021) introduces data augmentation techniques to improve the efficiency of data acquisition in deep active learning, while SRAAL (Zhang et al., 2020) integrates adversarial training techniques with active learning principles to address sample selection challenges.

Moreover, approaches like the Core-Set Approach (Sener and Savarese, 2017) and Bayesian Active Learning (BALD) (Houlsby et al., 2011) offer strategies for selecting informative instances, with Core-Set identifying a compact, diverse subset of unlabeled data, and BALD leveraging Bayesian inference for strategic instance selection. These methodologies collectively contribute to enhancing model training efficiency and performance in active learning settings.

2.3 Semi-supervised learning

Semi-supervised learning (SSL) is a specialized form of supervised learning that involves training on a small set of labeled data along with a large set of unlabeled data. Positioned between supervised and unsupervised learning, SSL is commonly used in scenarios where the availability of labeled data is limited due to constraints such as budgetary restrictions or data ambiguity, where the class of a sample is uncertain. Semi-supervised algorithms are

```

1 input : Labeled pool ( $P_l$ ), Unlabeled Pool ( $P_u$ ),
        Model parameters:  $\theta_E, \theta_D, \theta_c$ ,
        Hyperparameters:  $epochs, \alpha_1, \alpha_2, \alpha_3, \gamma$ 
2 output: Labeled pool ( $P_l$ ),  $Y_p$ 
3  $\theta_E, \theta_D \leftarrow preTraining(\theta_E, \theta_D, P_l, P_u)$ 
4  $\theta_c \leftarrow initCentroids(\theta_c, P_l)$ 
5 while  $budget \neq 0$  do
6   // Active Learning Loop
7    $Z_u = \theta_E(P_u)$ 
8    $X_s \leftarrow querySamples(Z_u, \theta_c, P_u)$ 
9    $P_l \leftarrow Annotate(X_s)$ 
10  for  $e$  in  $epochs$  do
11    for  $b$  in  $batches$  do
12       $Z_{ul} \leftarrow \theta_E(P_u, P_l)$ 
13       $X_r \leftarrow \theta_D(Z_{ul})$ 
14       $\mathcal{L}_{rec}$  compute using Eq. 5
15       $\mathcal{L}_{clu}$  compute using Eq. 2
16       $\mathcal{L}_{ccl}$  compute using Eq. 7
17       $\mathcal{L}_{total} \leftarrow \alpha_1 \cdot \mathcal{L}_{rec} + \alpha_2 \cdot \mathcal{L}_{clu} + \alpha_3 \mathcal{L}_{ccl}$ 
18       $\theta'_E, \theta'_c \leftarrow \theta_E, \theta_c - \gamma \nabla \mathcal{L}_{total}$ 
19       $\theta_c \leftarrow updateCentroids(Z_{ul})$ 
20      if  $updateCentroids$  is  $True$  then
21         $P \leftarrow updateP(z_{ul})$  compute using Eq. 1
```

Algorithm 1. Contrastive active learning.

designed to address such challenges. In this study, we propose an SSL approach for the classification of image data, aiming to leverage the benefits of both active learning (AL) and SSL. To achieve this, we suggested clustering contrastive loss (CCL) in conjunction with unsupervised training.

2.4 Entropy

Entropy Shannon (1948) is an information-theoretic measure of uncertainty. It quantifies the amount of information needed to encode a distribution. In active learning, entropy is widely used to select the most uncertain or ambiguous samples for annotation. The entropy can be shown as:

$$H(x) := - \sum_{x \in X} p(x) \log p(x)$$

(4)

3 Method

This study proposes a novel active learning approach based on pool-based sampling. It involves training a convolutional autoencoder (CAE) (Masci et al., 2011) to learn a low-dimensional latent space for both labeled and unlabeled samples. The latent space is then clustered using a clustering layer. After each iteration of the active learning process, a subset of data points associated with the latent space vectors is selected for annotation. To leverage information from the labeled data, the study introduces the contrastive clustering loss (CCL), which is a modified version of the contrastive loss (Chopra et al., 2005). The CCL operates on the latent space vectors, pulling samples of the same class toward their

TABLE 1 Algorithm symbols and their explanations.

Notation	Explanation
P_l	Labeled pool
P_u	Unlabeled pool
θ_E	Encoder model parameters
θ_D	Decoder model parameters
θ_c	Centroid parameters
$\alpha_1, \alpha_2, \alpha_3$	Losses weights
γ	Learning rate
Z_u	Encoded representations of unlabeled pool
X_s	Samples selected for annotation
P_l	Updated labeled pool
Z_{ul}	Encoded representations of both labeled and unlabeled pool
X_r	Reconstructed samples
θ'_E, θ'_c	Updated encoder and centroid parameters
∇	Gradient operator
P	P distribution

respective cluster centers and pushing samples of different classes apart.

3.1 Problem definition and notation

The main focus of this study is a semi-supervised active learning approach designed for image classification. Assuming there is a large set of unlabeled images P_u and a small set of labeled images P_l , along with a predetermined annotation budget, the goal is to select the most informative samples from the unlabeled set P_u to enhance the classification accuracy. These selected samples will be labeled by a human annotator and incorporated into the labeled set P_l . The initial step involves training a Convolutional Autoencoder (CAE) to learn a condensed representation of the images, referred to as latent space features. Each image i is transformed by the CAE into a feature vector z_i in the latent space. Subsequently, all latent space features $z_i, \forall i \in P_l \cup P_u$ are clustered into clusters, denoted as μ_j where j represents the centroid of the j -th cluster. Finally, the proposed cluster contrastive loss L_{ccl} (see Eq. 7) is applied to the labeled samples $z_l, \forall l \in P_l$. This loss function aims to attract the feature vectors z_j toward μ_j while pushing them away from $\mu_n, \forall n \neq j$, for all $n \neq j$.

3.2 Suggested method

The primary objective of this study is image classification, aiming to categorize images into their respective classes with optimal accuracy by leveraging labeled images from the restricted labeled data pool P_l . To achieve this, we introduce a pool-based active learning strategy that integrates contrastive learning and clustering, mutually enhancing their performance in every training cycle. Our approach follows a human-in-the-loop methodology, in which an active learning loop comprises model training, image querying, and annotation by an oracle. This iterative process continues until the budget is fully utilized.

The model consists of a CAE (Masci et al., 2011) and a clustering layer (Xie et al., 2016). Samples from P_l and P_u are fed into the model based on the active learning training stage. During each iteration of the active learning process, samples from P_u are chosen for labeling. The proposed module is depicted in Figure 1.

Prior to commencing the active learning iteration, certain initial steps are carried out. Initially, our CAE is pre-trained by reconstructing images from P_u and P_l using the MSE loss (Eq. 5). This process allows the CAE to acquire knowledge of lower-dimensional features within the dataset. Once the network is trained, the resulting latent space provides a feature $z_i, \forall i \in P_l \cup P_u$. Subsequently, the cluster centroids in the clustering layer are initialized with the average values of the vectors in the latent space of each class in our labeled pool P_l as depicted in Eq. 6.

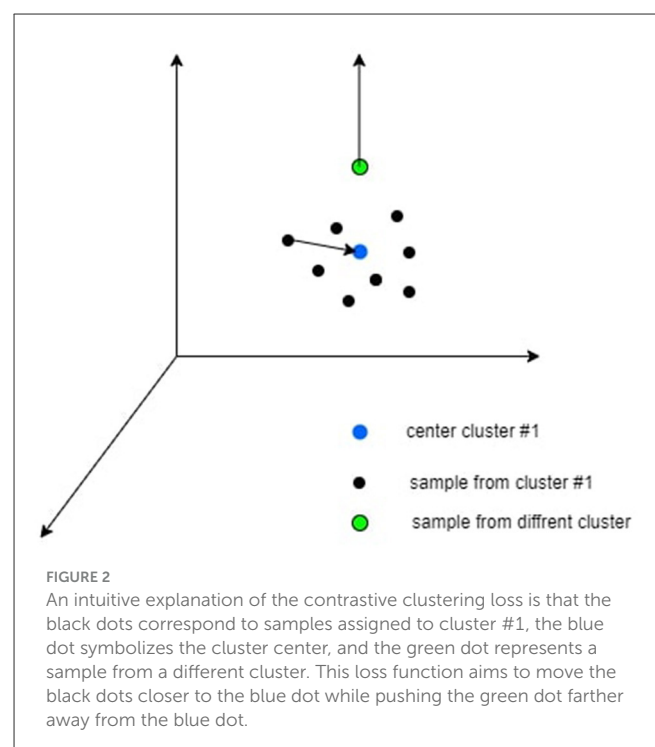
$$L_{rec} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (5)$$

$$\mu_c = \frac{1}{n_c} \sum_{i=1}^{n_c} z_c \quad (6)$$

Next, we incorporate clustering into the training of the CAE by clustering the acquired latent space with the utilization of a clustering layer (Guo et al., 2017) and employing a Kullback-Leibler divergence loss (Csiszár, 1975) as shown in Eq. 2. The primary objective of this stage is to organize the latent space into clusters, ensuring that similar image pairs produce proximate feature vectors within the latent space.

In the final stage, we incorporate the image labels from P_l . To utilize these labels effectively, we employ the suggested cluster contrastive loss L_{ccl} as shown in Eq. 7 on all vectors in the latent space derived from P_l , meaning that solely annotated images are taken into account by this loss. The CCL loss works by either pulling or pushing the feature vectors Z_i in the latent space toward their respective cluster center μ_i , or away from other cluster centers μ_j where $j \neq i$. This method allows us to enhance the purity of clusters while using a limited number of labeled images from P_l , during this stage we continue to make use of the previous clustering stage. Finally, we add all those losses and update the parameters of the model. The process is reiterated until reaching convergence or utilizing the entire annotation budget.

At the end of every active learning iteration, we perform query sampling to choose the n -th image that exhibits features with the highest entropy compared to all other clusters. These features are the most ambiguous in terms of their cluster assignment, and by labeling them, we gain valuable insights that the model failed to generalize. Algorithm 1 presents a generic pseudo-code for this approach, in Table 1 the symbols used in the algorithm are elucidated, providing clarity on their respective meanings and roles within the context of the algorithm.



3.2.1 Cluster contrastive loss

The cluster contrastive loss (CCL) is a revised variant of the supervised contrastive loss introduced in Khosla et al. (2020). To enhance the purity of the clusters, the proposed approach incorporates the labeled images from P_l into the clustering procedure. Consequently, this results in the adoption of the proposed CCL. The mathematical expression for the CCL is displayed below:

$$L_{ccl} = - \sum_{c \in C} \sum_{i \in I_c} \log \frac{\exp(z_i \cdot \mu_c / \tau)}{\sum_{z' \in I_{c'}} \exp(z' \cdot \mu_c / \tau)} \quad (7)$$

Where $c \in C$ is the class index, I_c is the set of all the samples indexes in class c , $I_{c'}$ is the set of all the samples indexes in all the classes beside class c . z_i is the i -th sample in the latent space and μ is the center of the cluster, $\tau \in R^+$ is a scalar temperature parameter. An intuition of the loss can be shown in Figure 2.

This loss involves both pulling samples toward their cluster center and pushing from other unmatched centroids centers simultaneously. It specifically affects the labeled data points. The CCL serves as a complementary approach to the unsupervised methods we currently employ, and empirical experiments indicate their mutual benefit. Figure 2 provides a visual representation of CCL as defined in Eq. 7.

3.2.2 The need for the contrastive clustering loss

During the training for CAE, we are provided with representation vectors in the latent space. In order to group the latent space into clusters corresponding to each class, as elaborated in Section 2.1, the clustering layer is utilized. This layer aims to streamline the process of image classification. Nevertheless, the clustering mechanism is proficient in grouping vectors with high certainty, which may result in certain images not being grouped together, particularly those from the same class that map to distant vectors in the latent space. Therefore, the integration of the suggested contrastive clustering loss becomes essential. This suggested CCL loss function works on adjusting vectors that were not properly aligned by the clustering process. Through this loss function, we can enhance the separation of classes in the latent space, even when dealing with a limited number of labeled images or when images are challenging to cluster due to the low confidence in the P-distribution of the clustering process.

3.2.3 Pre-training

During the initial phase, we train the convolutional autoencoder. We are using all the images from the unlabeled data pool P_u and the labeled data pool P_l . Each image $x_i \sim P_l \cup P_u$ inferences through the encoder and provides z_i a lower dimension latent vector $z_i = \sigma(x_i * W)$ where w is the weights of the encoder layers, σ is a nonlinear activation function, and $*$ is a convolution operation. The latent vector z_i is inference through the decoder

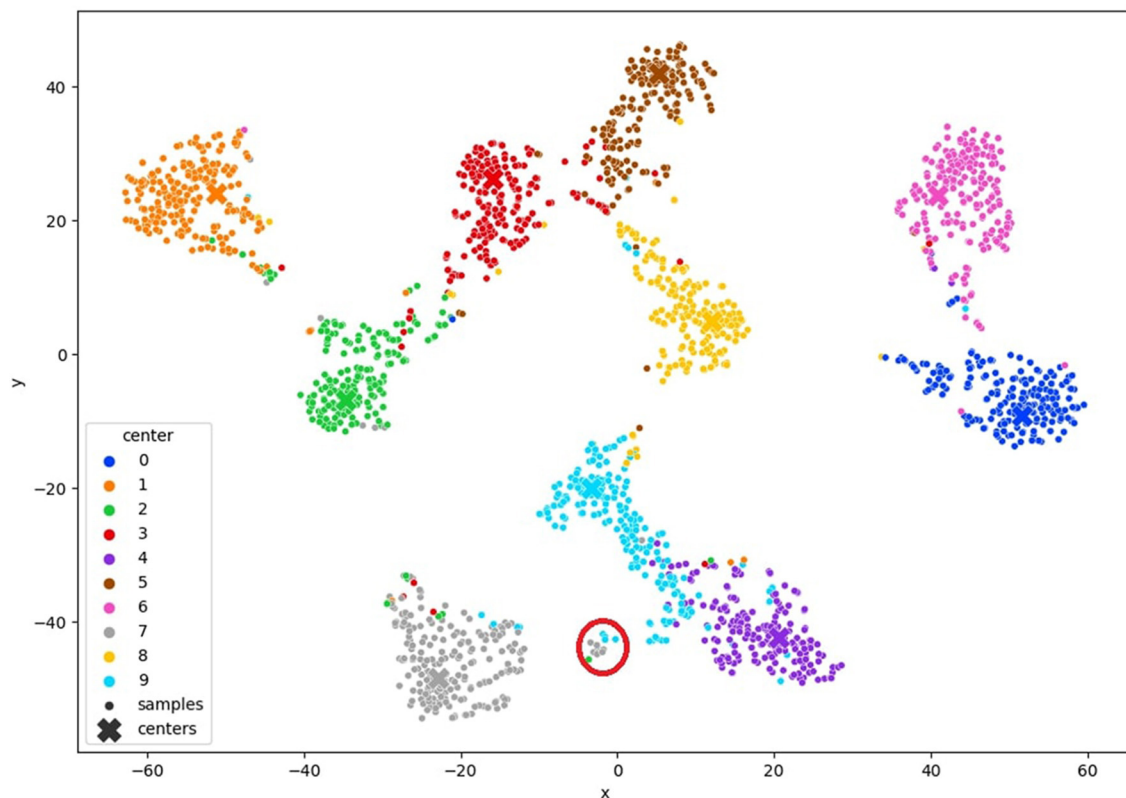


FIGURE 3
TSNE visualization of the query method the red circle represents samples with high entropy.

which provides an \hat{x} which is a reconstruction of the original image x_i . $\hat{x} = \sigma(z_i * U)$ where U is the weight for the decoder. \hat{x}_i and x_i are entered to MSE loss (Eq. 5) which provides a high loss when x_i looks different from \hat{x}_i and a low loss when they are similar. At the end of this step, the CAE has trained weights W and U .

3.2.4 Initialization and update centroids

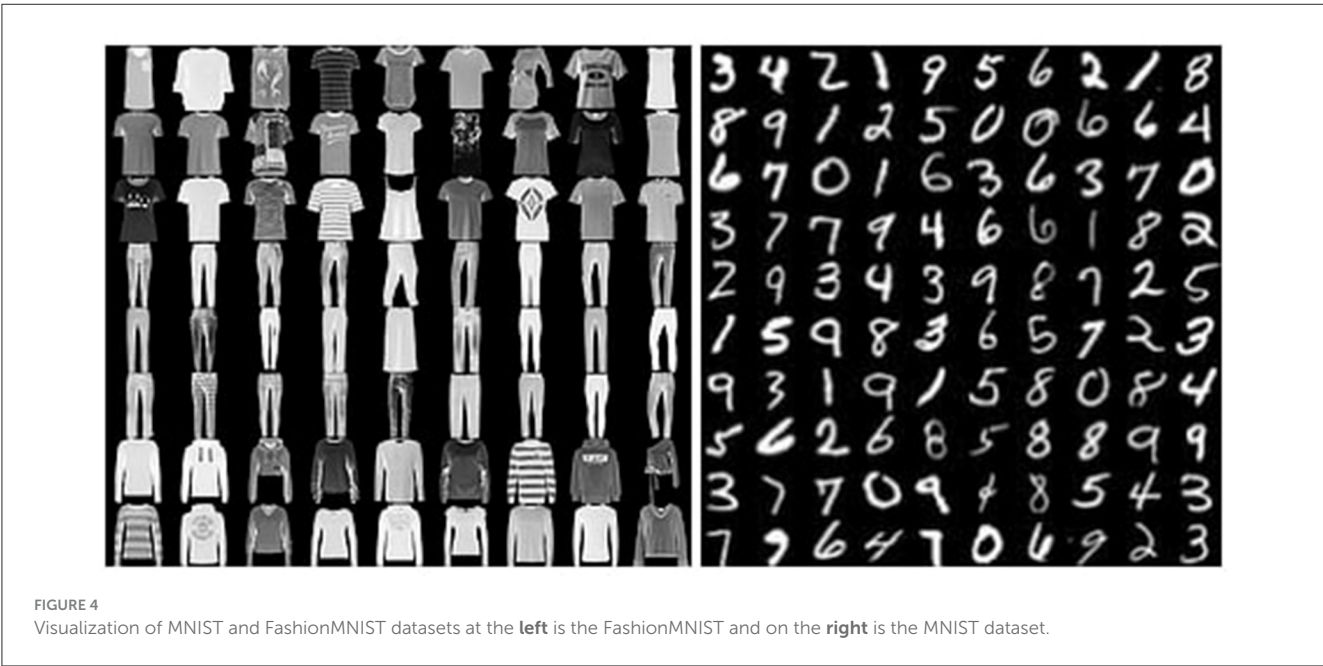
Once the CNN is pre-trained, the centroids in the clustering layer are initialized using the average value of each class projection from P_l in the latent space. Subsequently, every 80 iterations, the distribution of P is updated by the following (Eq. 3). As detailed in Section 2.1, the centroids represent the weights of the clustering layer, and therefore they are adjusted during each training iteration.

3.2.5 Query samples

In this stage, our objective is to acquire image annotations by engaging a human annotator in the active learning procedure. At this point, we have already acquired a clustered latent space generated by the model itself. Any vectors within the latent space that are not clustered or are distant from the cluster center are identified as hard examples, representing images that require annotation. We select samples linked to vectors in the latent space that do not clearly belong to any cluster and annotate them based on the uncertainty criterion detailed in Eq. 4. More specifically, we target the vectors that exhibit the highest entropy in the cluster distribution. A visual representation of this approach is shown in Figure 3. By focusing on a small number of samples associated with feature vectors located far from the cluster center, we gain insight into these samples and the clusters they are associated with, thereby enhancing the overall clustering process.

3.3 Combination of contrastive learning and clustering

When the suggested clustering method is applied to the latent space, there may be instances where some feature vectors are not accurately clustered. This situation can arise when feature vectors within the latent space that should belong to the same cluster are spatially distant from each other. As a result, the clustering layer may encounter challenges in grouping these feature vectors effectively. To address this issue, we introduce our proposed CCL,



which works to minimize the distance between distant feature vectors that belong to the same cluster while maximizing the separation between those that do not. Furthermore, we incorporate a query mechanism to select challenging examples (i.e., samples that are significantly distant from their corresponding cluster center) for manual annotation. By integrating these strategies and progressively bringing the feature vectors closer together in a semi-supervised fashion, followed by clustering using the clustering layer, we improve the purity of the clustering outcomes.

3.4 Implementation details

In this work, we used a convolutional autoencoder for our model. The encoder consists of 3 convolutional layers, a batch normalization layer, and a linear embedding layer with a size of 10. The decoder consists of a linear de-embedding layer, 3 deconvolutional layers, and a batch normalization layer. The clustering layer weights are initialized with the mean of the latent space clusters using the starting labeled images in P_l , and are then updated with the kl-loss using the Q and P distribution as described earlier. The P-distribution, or target distribution, is initialized every 80 steps. Each benchmark dataset is split into a 20% validation set and 80% training set, which is further divided into two data pools: a labeled data pool P_l and an unlabeled data pool P_u . First, we pre-trained the model for 50 epochs. Then each active learning training

iteration was set to 10 epochs and for the duration of overall 20 active learning loops. In each active learning loop, we query 250 image samples using the uncertainty strategy for annotation.

4 Experiments and results

4.1 Datasets

We have evaluated our method in image classification tasks. We have used MNIST (LeCun, 1998), FashionMNIST (Xiao et al., 2017), and USPS (Hull, 1994) datasets. Both the MNIST and the FashionMNIST datasets have 60K grayscale images of size 28x28. Examples of MNIST and FashionMNIST datasets can be viewed at Figure 4, and USPS has 9298 grayscale images of 16x16 size. An example of USPS dataset can be viewed at Figure 5.

4.2 Performance measurement

We evaluate the performance of our method with the image classification task by measuring the accuracy over different amounts of labeled images from 500 to 5k images with a raising of 250 images from query to query. The results of all our experiments are averaged over 3 runs.

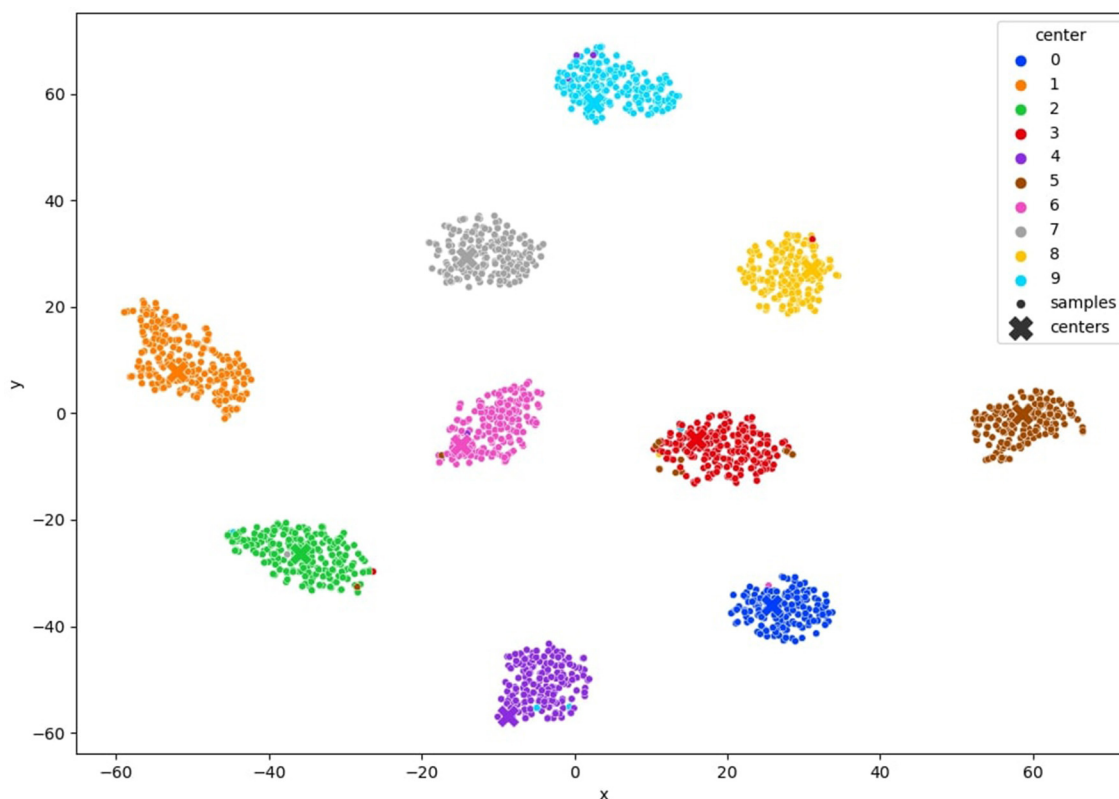


FIGURE 6

TSNE visualization of the clustered MNIST latent space after convergence of our method with 10% of annotated samples.

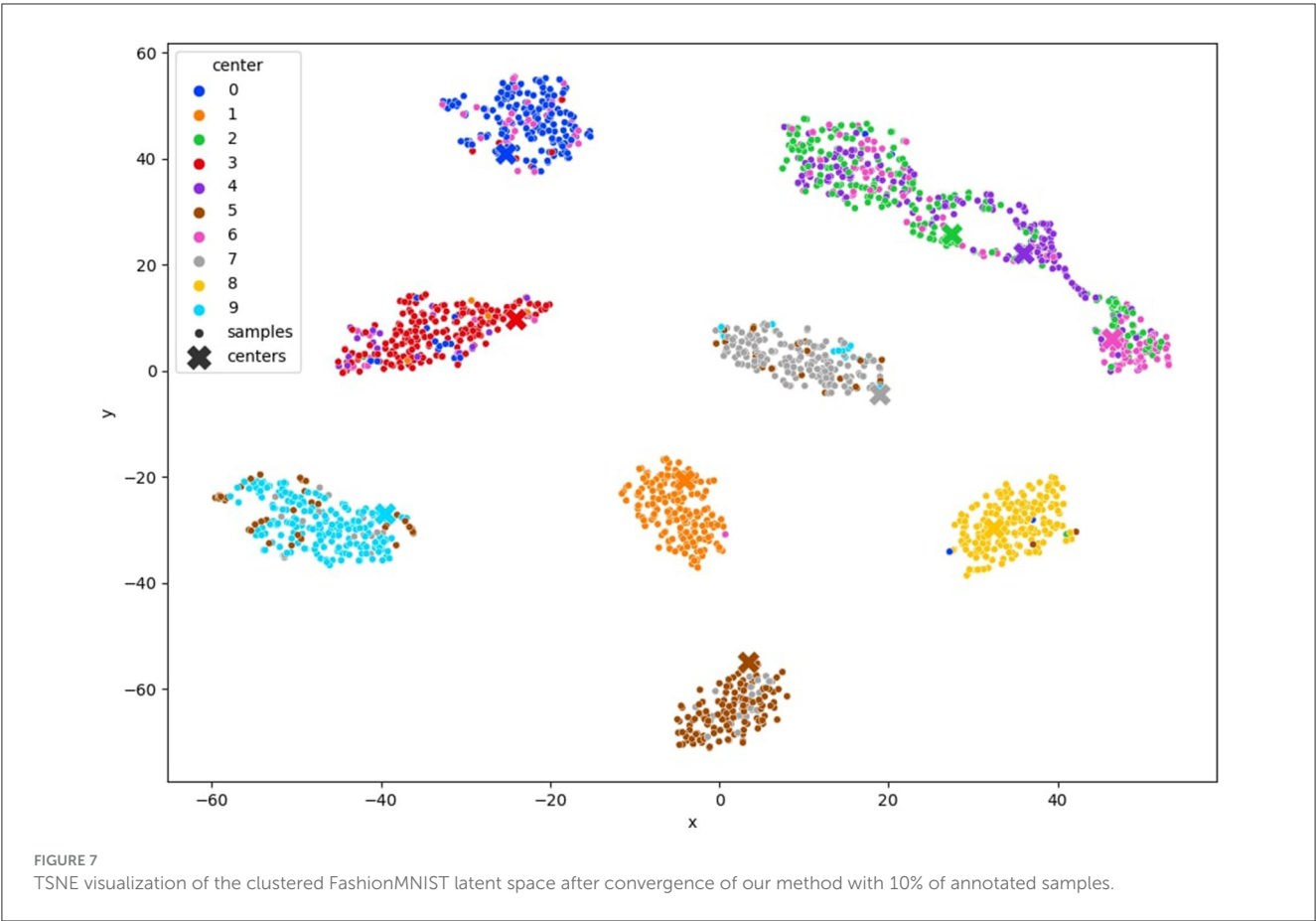


TABLE 2 Ablation study: clustering vs. clustering + CCL (3% of annotated data).

Dataset	Method	
	Clustering	Clustering + CCL
MNIST	81.6%	91.0%
USPS	68.7%	86.5%

4.3 Experiments details

We begin our experiments with an initial labeled pool of the size of 250 and in each iteration of the training loop we provided another 250 images that were annotated by the human oracle and added to the initial labeled pool P_l . Training is repeated on the new training set with the new labeled images. We assume that the dataset is balanced and the oracle annotations are ideal.

In Figure 6 MNIST result. In Figure 7 FashionMNIST result.

4.4 Effectiveness of the CCL

In Table 2, we present an ablation study comparing our proposed method with the use of clustering alone. The study evaluates the performance of both approaches on the Mnist

and USPS datasets. The results demonstrate that integrating the CCL with clustering, using only 3% of labeled data, significantly improves model performance. The CCL operates by encouraging the model to learn discriminative representations within clusters while simultaneously enforcing compactness among cluster centroids. By incorporating this loss function into our framework, we guide the clustering process to yield clusters that not only capture inherent data structures but also ensure inter-class separability. This results in more coherent and well-separated clusters, facilitating better decision boundaries and ultimately leading to improved classification accuracy. Additionally, Figure 8 visually illustrates the difference between using clustering alone and incorporating the CCL into the clustering process.

4.5 Comparing with other methods

We conducted a comprehensive evaluation of our proposed method across multiple datasets, including MNIST, FashionMNIST, and USPS, as detailed in Tables 3–5. Our results showcase significant performance improvements over baseline methods, particularly evident in scenarios with limited labeled data. When compared to state-of-the-art techniques such as Core-Set Approach (Sener and Savarese, 2017), Variational

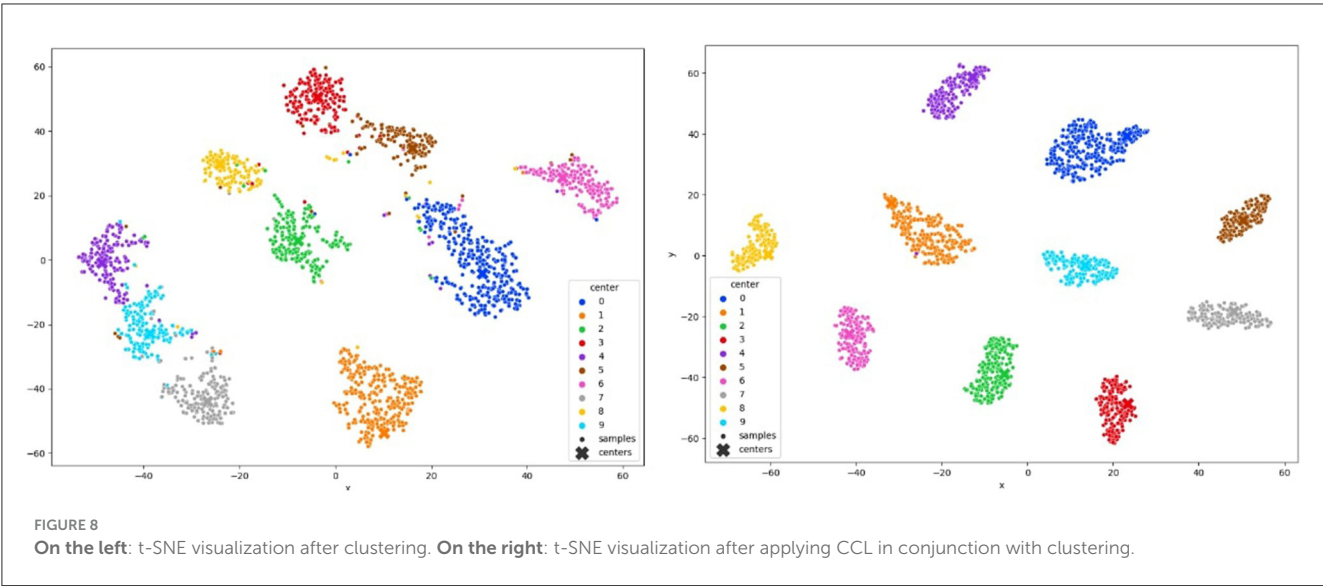


TABLE 3 MNIST accuracy results on entropy sampling (Wang and Shang, 2014) BALD (Gal et al., 2017) Vaal (Sinha et al., 2019) Core-set (Sener and Savarese, 2017) and our method with 1, 3, 5, and 10% of the data labeled.

Percentage of labeled data	Entropy	BALD	Vaal	Core-set	Ours
1%	0.151	0.251	0.255	0.336	0.832
3%	0.600	0.701	0.735	0.805	0.910
5%	0.805	0.813	0.810	0.888	0.948
10%	0.935	0.945	0.917	0.928	0.983

TABLE 4 Fashion MNIST accuracy results on entropy sampling (Wang and Shang, 2014) BALD (Gal et al., 2017) Vaal (Sinha et al., 2019) Core-set (Sener and Savarese, 2017) and our method with 1, 3, 5, and 10% of the data labeled.

Percentage of labeled data	Entropy	BALD	Vaal	Core-set	Ours
1%	0.318	0.264	0.189	0.305	0.490
3%	0.468	0.360	0.520	0.627	0.671
5%	0.556	0.616	0.602	0.679	0.697
10%	0.637	0.703	0.673	0.729	0.758

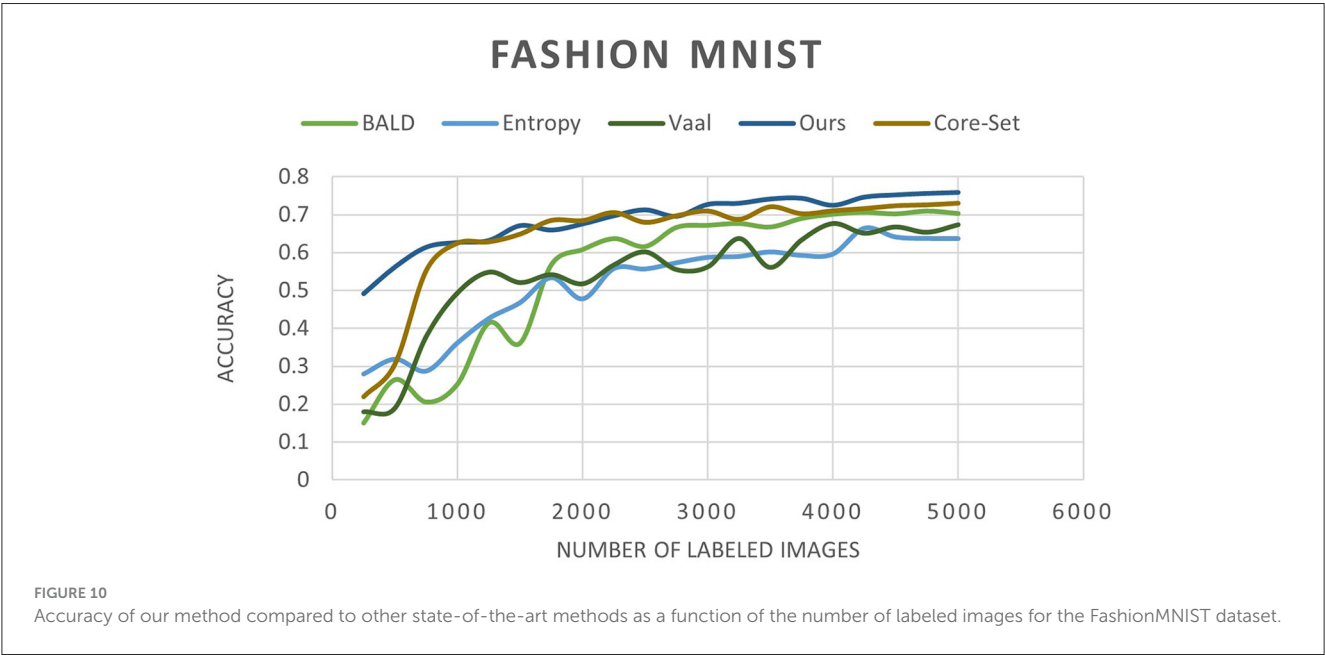
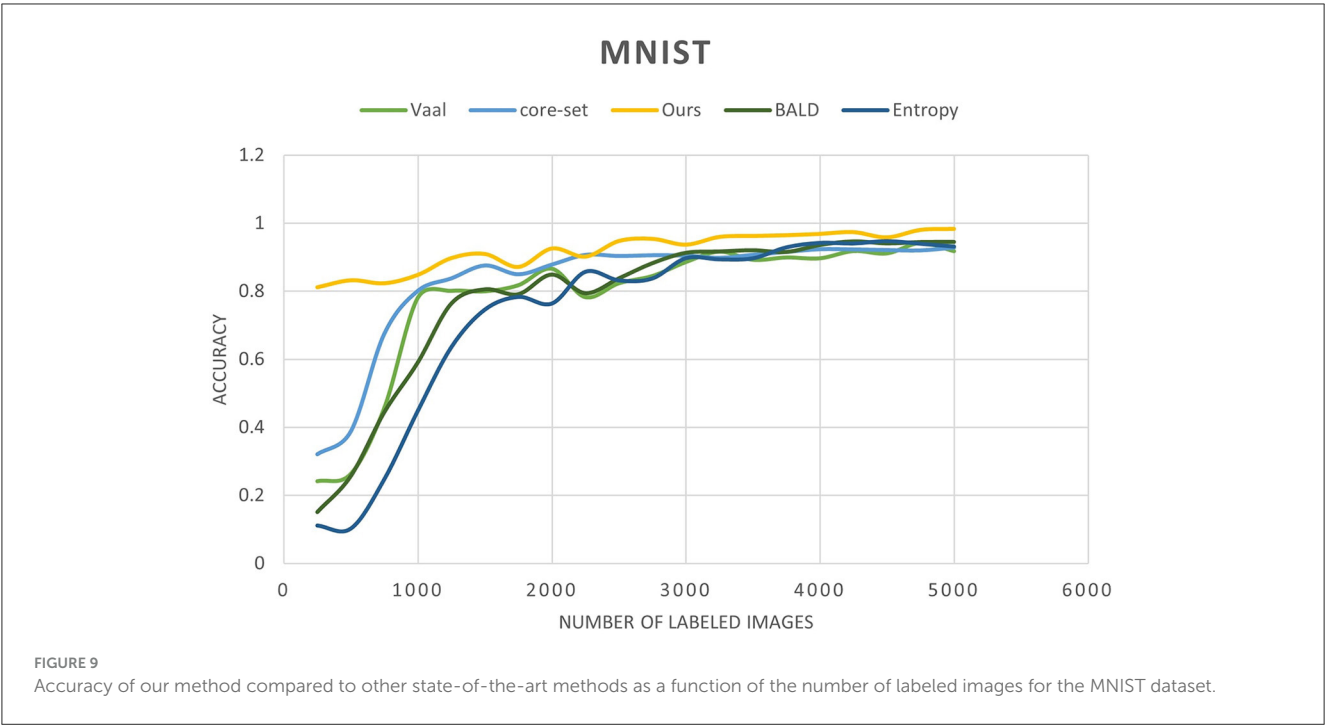
TABLE 5 USPS accuracy results on entropy sampling (Wang and Shang, 2014) BALD (Gal et al., 2017) Vaal (Sinha et al., 2019) random sampling and our method with 3, 5, and 10% of the data labeled.

Percentage of labeled data	Entropy	BALD	Vaal	Random sampling	Ours
3%	0.770	0.821	0.836	0.797	0.865
5%	0.855	0.860	0.876	0.858	0.895
10%	0.909	0.896	0.926	0.894	0.933

Adversarial Active Learning (VAAL) (Sinha et al., 2019), and Bayesian Active Learning by Disagreement (BALD) (Houlsby et al., 2011), our approach consistently demonstrates competitive performance. Figures 9–11 showing our method comparing to the others (Notably, leveraging pre-trained) Notably, leveraging pre-trained clustering models contributes to achieving relatively high accuracy, particularly in scenarios with a scarcity of labeled samples.

4.6 Experiment analysis

To comprehensively validate the efficacy of our approach, we conducted an in-depth analysis of clustering quality throughout the training process. We monitored the evolution of clustering performance and visualized the t-SNE projections of learned latent space representations, as depicted in Figures 6, 7, 12. These visualizations offer insights into the structure of the learned



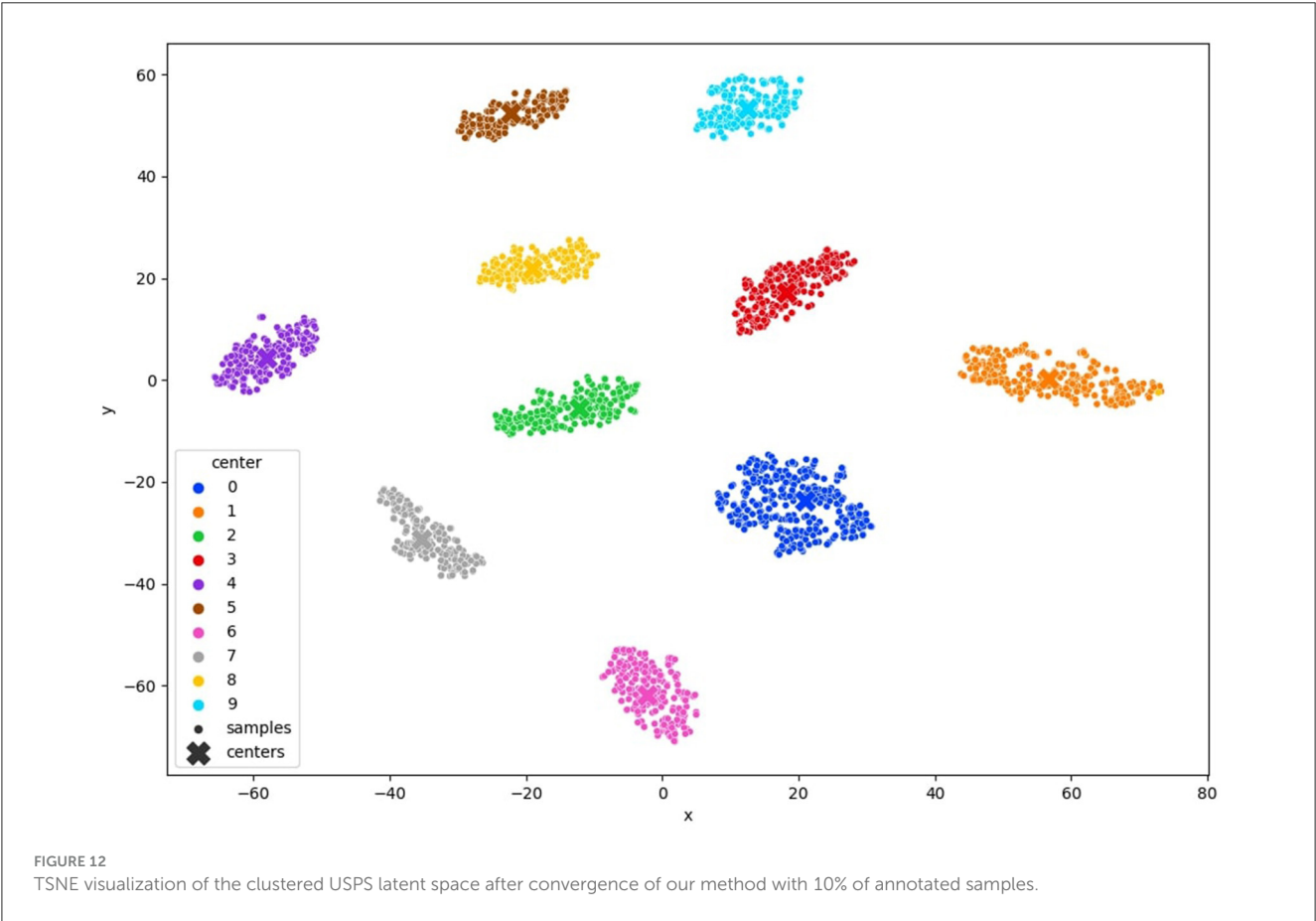
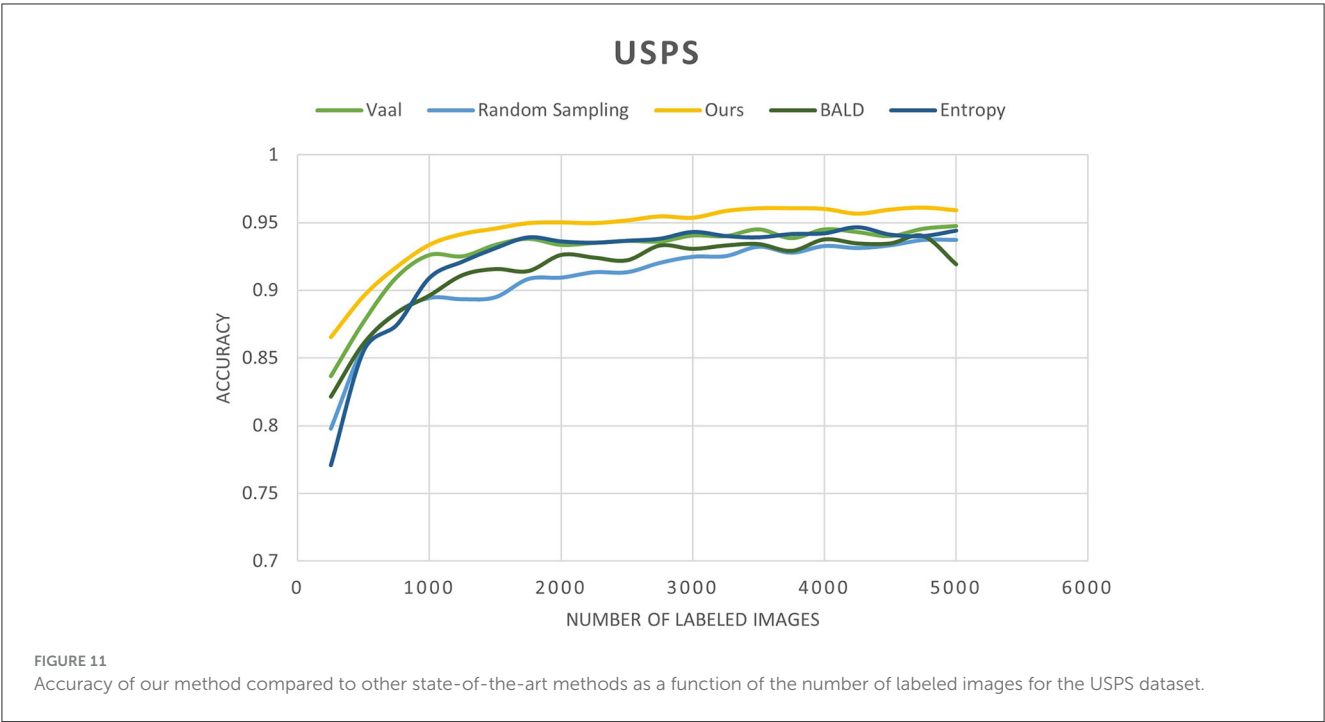
representations, revealing distinct clusters corresponding to each class. The observed trends in clustering align well with the accuracy improvements reported in [Tables 3–5](#), corroborating the effectiveness of our method.

In addition to accuracy comparisons, it's imperative to delve deeper into the performance metrics of our approach compared to baseline methods. For instance, on the MNIST dataset, our method achieves an accuracy of 91% with only 3% labeled data, outperforming the Core-Set Approach, which attains 80.5% accuracy. This notable performance gain underscores the superiority of our method in leveraging limited labeled data effectively.

5 Discussion

The integration of convolutional autoencoders, clustering, and a novel clustering contrastive loss in our semi-supervised active learning approach presents a unique and promising strategy for leveraging both labeled and unlabeled data in image classification tasks. By combining clustering with active learning, our method offers a distinctive approach that distinguishes it from previous methodologies.

A significant strength of our approach lies in its ability to extract valuable insights from unlabeled data by organizing it into clusters, thereby guiding the query selection process in



active learning. However, the effectiveness of our method may depend on the quality of clustering initialization, which could potentially limit performance, particularly in scenarios involving

complex, high-dimensional data. Exploring the applicability of our approach beyond image classification domains warrants further investigation.

Despite these potential limitations, our research represents a notable advancement in the realm of semi-supervised active learning. By integrating deep clustering, active learning, and contrastive learning principles, we address challenges associated with data scarcity, thereby enhancing model performance in resource-constrained settings. Moving forward, future research endeavors could explore the development of more robust clustering techniques, alternative representation learning methods, and synergistic combinations with other active learning strategies to further enhance performance and generalization capabilities.

Theoretically, the clustered representations derived by our approach hold promise for facilitating various downstream tasks, including data augmentation, domain adaptation, and the incorporation of weak or noisy labels. Such capabilities could prove invaluable in addressing the challenges posed by limited annotation scenarios. While our work contributes to the field, it also underscores the inherent challenges and opportunities associated with semi-supervised learning in real-world applications, paving the way for continued advancements and innovation in this domain.

It is essential to acknowledge the use of a smaller model architecture in our experiments. The complexity introduced by clustering necessitated the use of a smaller model to maintain tractability and computational efficiency. While this choice may have influenced our absolute performance metrics, it enabled us to explore the feasibility and efficacy of our approach within practical constraints. It is plausible that in subsequent studies, researchers may employ larger, more complex models to further improve performance.

6 Conclusions and future work

In this study, we have introduced a novel approach to image classification through a pool-based semi-supervised active learning technique. By integrating deep clustering and deep active learning, we aim to enhance classification accuracy by using fewer labeled images. Our method involves clustering feature vectors in the latent space that corresponds to images from P_l and P_u , thereby obtaining a more informative representation of the latent space to support the active learning procedure. We have also incorporated a clustering contrastive loss to enhance the clustering of the latent space even with a limited number of labeled images. Cases where feature vectors in the latent space are not well grouped together or are far from their respective cluster centers are recognized as hard examples and are then queried for annotation by a human oracle.

Our empirical experiments demonstrated that our method achieves high classification accuracy even with a small number of annotations. The iterative combination of clustering with the suggested contrastive learning and query method leads to a more separated latent space, which in turn facilitates the classification process. Thanks to the clustering step, our method achieves high accuracy from the beginning. However, the clustering step may have a drawback for complicated datasets, as it can be challenging

to cluster them effectively. We believe that future work can improve the clustering process to provide better clustering initialization even for complex datasets.

We used a convolutional autoencoder (CAE) to map samples to the latent space, but future work could explore more robust methods like a variational autoencoder that creates smoother and more connected latent spaces, which will help to improve clustering. Furthermore, our method is currently designed for image classification tasks, but it could be extended to other computer vision tasks such as semantic segmentation and object detection by inserting a suitable network head to the model for the requested task.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

HR: Conceptualization, Data curation, Investigation, Methodology, Resources, Software, Visualization, Writing—original draft, Writing—review & editing. AG: Conceptualization, Supervision, Validation, Writing—original draft, Writing—review & editing.

Funding

The authors declare that financial support was received for the research, authorship, and/or publication of this article. The publication fee for this paper was paid by InnerEye Ltd. The funder was not involved in the study design, collection, analysis, interpretation of data, the writing of this article, or the decision to submit it for publication.

Conflict of interest

AB was employed by InnerEye Ltd CTO.

The remaining author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Angluin, D. (1988). Queries and concept learning. *Mach. Learn.* 2:319–342. doi: 10.1007/BF00116828
- Atlas, L., Cohn, D., and Ladner, R. (1989). “Training connectionist networks with queries and selective sampling,” in *Advances in Neural Information Processing Systems*, 2 (NIPS).
- Chopra, S., Hadsell, R., and LeCun, Y. (2005). “Learning a similarity metric discriminatively, with application to face verification,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)* (San Diego, CA: IEEE), 539–546.
- Csiszár, I. (1975). I-divergence geometry of probability distributions and minimization problems. *Ann. Probabil.* 3, 146–158.
- Gal, Y., Islam, R., and Ghahramani, Z. (2017). “Deep bayesian active learning with image data,” in *International Conference on Machine Learning* (New York: PMLR), 1183–1192.
- Guo, X., Liu, X., Zhu, E., and Yin, J. (2017). “Deep clustering with convolutional autoencoders,” in *International Conference on Neural Information Processing* (Cham: Springer), 373–382.
- Houlsby, N., Huszár, F., Ghahramani, Z., and Lengyel, M. (2011). “Bayesian active learning for classification and preference learning,” in *arXiv [Preprint]*. arXiv:1112.5745.
- Hull, J. J. (1994). A database for handwritten text recognition research. *IEEE Trans. Pattern Anal. Mach. Intell.* 16, 550–554. doi: 10.1109/34.291440
- Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., et al. (2020). Supervised contrastive learning. *Adv. Neural Inf. Process. Syst.* 33, 18661–18673.
- Kim, Y.-Y., Song, K., Jang, J., and Moon, I.-C. (2021). Lada: Look-ahead data acquisition via augmentation for deep active learning. *Adv. Neural Inf. Process. Syst.* 34, 22919–22930.
- LeCun, Y. (1998). *The Mnist Database of Handwritten Digits*. Available online at: <http://yann.lecun.com/exdb/mnist/> (accessed May 15, 2024).
- Lewis, D. D. (1995). “A sequential algorithm for training text classifiers: Corrigendum and additional data,” in *Acm Sigir Forum* (New York, NY: ACM), 13–19.
- Lewis, D. D., and Gale, W. A. (1994). “A sequential algorithm for training text classifiers,” in *SIGIR’94* (Cham: Springer), 3–12.
- Liu, S., and Li, X. (2023). “Understanding uncertainty sampling,” in *arXiv [Preprint]* (Wiley Online Library), arXiv:2307.02719.
- Masci, J., Meier, U., Cireşan, D., and Schmidhuber, J. (2011). “Stacked convolutional auto-encoders for hierarchical feature extraction,” in *International Conference on Artificial Neural Networks* (Cham: Springer), 3–12.
- Mukherjee, S., Asnani, H., Lin, E., and Kannan, S. (2019). “Clustergan: Latent space clustering in generative adversarial networks,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 4610–4617.
- Peng, X., Xiao, S., Feng, J., Yau, W.-Y., and Yi, Z. (2016). “Deep subspace clustering with sparsity prior,” in *IJCAI*, 1925–1931.
- Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Gupta, B. B., et al. (2021). A survey of deep active learning. *ACM Comp. Surveys (CSUR)* 54, 1–40. doi: 10.1145/3472291
- Sener, O., and Savarese, S. (2017). “Active learning for convolutional neural networks: a core-set approach,” in *arXiv*.
- Settles, B. (2009). *Active Learning Literature Survey*. Madison: Wisconsin/Madison.
- Seung, H. S., Oppor, M., and Sompolinsky, H. (1992). “Query by committee,” in *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, 287–294.
- Shannon, C. E. (1948). A mathematical theory of communication. *Bell Syst. Techn. J.* 27, 379–423. doi: 10.1002/j.1538-7305.1948.tb01338.x
- Sinha, S., Ebrahimi, S., and Darrell, T. (2019). “Variational adversarial active learning,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (IEEE/CVF)*, 5972–5981.
- Tian, F., Gao, B., Cui, Q., Chen, E., and Liu, T.-Y. (2014). “Learning deep representations for graph clustering,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 1.
- Van der Maaten, L., and Hinton, G. (2008). Visualizing data using t-sne. *J. Mach. Learn. Res.* 9, 27.
- Wang, D., and Shang, Y. (2014). “A new active labeling method for deep learning,” in *2014 International Joint Conference on Neural Networks (IJCNN)* (Beijing: IEEE), 112–119.
- Xiao, H., Rasul, K., and Vollgraf, R. (2017). “Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms,” in *arXiv*.
- Xie, J., Girshick, R., and Farhadi, A. (2016). “Unsupervised deep embedding for clustering analysis,” in *International Conference on Machine Learning* (New York: PMLR), 478–487.
- Yang, J., Parikh, D., and Batra, D. (2016). “Joint unsupervised learning of deep representations and image clusters,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5147–5156.
- Zhang, B., Li, L., Yang, S., Wang, S., Zha, Z.-J., and Huang, Q. (2020). “State-relabeling adversarial active learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (IEEE/CVF)*, 8756–8765.



OPEN ACCESS

EDITED BY

Deepika Koundal,
University of Petroleum and Energy Studies,
India

REVIEWED BY

M. A. Khan,
HITEC University, Pakistan
Vatsala Anand,
Chitkara University, India
Shankar Shambhu,
Chitkara University, India

*CORRESPONDENCE

Ziming Shi
✉ szm_zz@163.com
Pan Zheng
✉ pan.zheng@canterbury.ac.nz

RECEIVED 01 March 2024

ACCEPTED 08 April 2024

PUBLISHED 10 June 2024

CITATION

Hou P, Wang Y, Shi Z and Zheng P (2024) An improved Dijkstra cross-plane image encryption algorithm based on a chaotic system.
Front. Artif. Intell. 7:1394101.
doi: 10.3389/frai.2024.1394101

COPYRIGHT

© 2024 Hou, Wang, Shi and Zheng. This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

An improved Dijkstra cross-plane image encryption algorithm based on a chaotic system

Pijun Hou¹, Yuepeng Wang², Ziming Shi^{3*} and Pan Zheng^{4*}

¹The Key Laboratory of Advanced Design and Intelligent Computing, School of Software Engineering, Dalian University, Dalian, China, ²DHC IT Company, Dalian, China, ³Experimental Center, Dalian University, Dalian, China, ⁴Information Systems, University of Canterbury, Christchurch, New Zealand

While encrypting information with color images, most encryption schemes treat color images as three different grayscale planes and encrypt each plane individually. These algorithms produce more duplicated operations and are less efficient because they do not properly account for the link between the various planes of color images. In addressing the issue, we propose a scheme that thoroughly takes into account the relationship between pixels across different planes in color images. First, we introduce a new 1D chaotic system. The performance analysis shows the system has good chaotic randomness. Next, we employ a shortest-path cross-plane scrambling algorithm that utilizes an enhanced Dijkstra algorithm. This algorithm effectively shuffles pixels randomly within each channel of a color image. To accomplish cross-plane diffusion, our approach is then integrated into the adaptive diffusion algorithm. The security analysis and simulation results demonstrate that the approach can tackle the issue of picture loss in telemedicine by encrypting color images without any loss of quality. Furthermore, the images we utilize are suitable for both standard RGB and medical images. They incorporate more secure and highly sensitive keys, robustly withstanding various typical ciphertext analysis attacks. This ensures a reliable solution for encrypting original images.

KEYWORDS

cross-plane scrambling, adaptive diffusion, image encryption, chaotic system, Dijkstra algorithm

1 Introduction

Image encryption technology is gaining popularity due to its ability to enhance the security of image communication. This is especially crucial as people become increasingly aware of security issues during image transmission (Liang et al., 2022). Image encryption can storage by converting it from significant plaintext into purposeless ciphertext to defend it against permission access and malicious attacks (Huang et al., 2022).

To maintain digital images' security, researchers have proposed many attack-resistant techniques, including data hiding (Ahmadian and Amirmazlaghani, 2019), image encryption (Hu, 2021; Huang et al., 2022; Li et al., 2023), digital watermarking (Zhang X et al., 2020), and compressive sensing (Wang and Su, 2021; Chai et al., 2022; Sarangi and Pal, 2022). Of such techniques, image encryption is often known for being a direct and significant technique, and utilizing the proper key is the only method to recover the original image data. Over the last several years, a number of approaches have been used to build plenty of digital image encryption algorithms, such as the DNA coding encryption scheme (Liang and Zhu, 2023),

the quaternion technique (Wang X. et al., 2022; Wang Y. et al., 2022), and the scheme using block compressive sensing and elementary cellular automata (Chai et al., 2018), it uses cellular automata scrambling to achieve the goal of making pixel values more difficult to predict, and the new zigzag global scrambling scheme designed (Li H et al., 2022). These programs offer multiple benefits and a high level of security.

Chaotic systems have complex dynamic characteristics, unique inherent randomness, control parameters, initial value sensitivity, traversal, and long-term unpredictability, making them appropriate for application in digital image encryption. Andono and Setiadi (2022) introduce several common chaotic systems and utilize multiple multidimensional chaotic systems, such as Lorenz system and Henon map to complete the image encryption. Researchers (Mansouri and Wang, 2020) improved the 2D Arnold mapping by obtaining a scrambled Arnold mapping. Hua et al. (2018) used sine and logistic mappings to produce a new chaotic 2D system. Although this algorithm has high complexity and hyper-chaotic behavior, most multidimensional chaotic systems have high computational costs. In addition, a 1D chaotic system (Wang et al., 2021) was developed and the designed system has the advantages of fast computation and fast image encryption, resulting in time savings.

While color photos are more information-dense than grayscale images, the majority of color image encryption techniques now in use have certain clear shortcomings. The algorithm in Li Q et al. (2022) uses a self-designed inter-plane rule, which requires the calculation of the pixel inter-plane position each time, leading to repeated calculations and a failure to maximize the relationship between pixels and planes. Furthermore, the algorithm in Hua et al. (2021) uses a Latin cube to design a set of scrambling rules for RGB images. For developing the encryption results and safety, the scheme blurs the original image's pixel values, making the decrypted image inconsistent with the initial image and impossible to fully recover from the initial image. In the later study (Zhou et al., 2021), an RGB image is divided into three planes for independent encryption, and a color image is reconstructed from the encrypted result. In the password system, the security level is low because when a pixel on a plane change, it cannot change quickly enough to extend to three planes. Furthermore, inefficient is this encryption scheme, which ignores the relationship between a color image's three planes, as a result, real-time encryption systems that demand great security and efficiency are not appropriate for this encryption technique. Images were encrypted using a discrete chaotic system and S-box in the algorithm (Liu et al., 2022), which required over 100 iterations of the S-box and consumed a lot of processing resources. The algorithm (Zheng et al., 2022) use DNA coding to encrypt a portion of the image many times, leading to a poor level of efficiency in the image encryption process.

It is evident from the explanation above that a large number of current encryption techniques for chaotic and color pictures have serious fundamental problems. We provide a fresh approach to color image encryption that makes use of a unique one-dimensional chaotic system for purposed of overcome these problems. The creation of a unique 1D chaotic system with enhanced chaotic performance and a broader parameter range is a main component of this technique. We have developed an improved Dijkstra algorithm that considers the properties of color pictures, building upon the new 1D chaotic system. Rather of encrypting each color plane independently, we accomplish

pixel scrambling across color planes. Next, we perform adaptive diffusion based on plane distribution to further alter pixel values and enhance the safety of the encryption method.

The following are this study's primary contributions:

- 1 A performance analysis shows that the 1D chaotic system we present eliminates several shortcomings of current chaotic mappings, such as restricted parameters, inadequate nonlinear behavior, and poor unpredictability. According to the analysis of the security performance of Chaos, the new 1D chaotic system proposed by us meets the security requirements, is evenly distributed, and can generate keys that meet the security standards;
- 2 Many color image encryption techniques have flaws in their architecture. The design of several color picture encryption systems is incorrect. Three distinct gray planes are processed for the majority of color pictures. Using the design of a novel 1D chaotic system and an improved Dijkstra algorithm as the foundation for a cross-plane color encryption technique. Pixels will appear anywhere on any plane, and Adaptive Diffusion Based on Plane Distribution will vary the value of each pixel sufficiently. In contrast to previous color image encryption techniques, our proposed diffusion and permutation operate simultaneously on all three planes, rather than individually on each;
- 3 Simulation findings and implementation analyses show that our proposed system outperforms several current image encryption techniques in various data aspects and can withstand chosen plaintext attacks.

This essay's remaining sections are as follows: Chaotic system with a performance study covered in Section 2. The creation of keys and certain encryption procedures, such as diffusion and scrambling methods, are covered in Section 3. Section 4 presents method's security analysis and simulation findings. Paper's conclusion is given in Section 5.

2 Related work

2.1 1D-SASCS chaotic system

1D-SASCS chaotic system (Wang and Liu, 2022) is presented Eq. 1:

$$x_{n+1} = |\sin(100\mu / \arcsin x_n)| \quad (1)$$

λ is a parameter of control, $\lambda \in (0, +\infty)$. The chaotic system possesses good chaotic characteristics, but the chaotic range of 1D-SASCS is relatively small.

2.2 Ill-conditioned matrix

When the data are significantly disrupted, an ill-conditioned matrix exhibits significant oscillations in the solutions of an equation system. Solving linear equation $Ax = b$, one such matrix is as Eq. 2:

$$\begin{pmatrix} R & 201 \\ -800 & 401 \end{pmatrix} \begin{pmatrix} K_1 \\ K_2 \end{pmatrix} = \begin{pmatrix} 200 \\ -200 \end{pmatrix} \quad (2)$$

For example, when $R=400$, $K_1=-100$, and $K_2=-200$; when $R=402$, $K_1=99.5025$, and $K_2=198.01$.

2.3 1D chaotic system

The formula for 1D chaotic system is as Eq. 3:

$$\begin{cases} A = \begin{pmatrix} 400.5 + X(i) - 201 \\ -800 & 401 \end{pmatrix} \\ B = \begin{bmatrix} 200 \\ -200 \end{bmatrix} \\ A * \begin{pmatrix} K_1 \\ K_2 \end{pmatrix} = B \\ X(i+1) = \text{mod}(|\sin(K_1 / \arcsin(X(i))) + K_2|, 1) \end{cases} \quad (3)$$

When $X(i) \in (0, +\infty)$, the mapping demonstrates good chaotic behavior. Compared with certain standard 1D chaotic mappings, our suggested 1D mapping has a wider parameter range. The chaotic system formed when $X(1)=0.5$ is adopted by our method, which contains two parameters that vary with each repetition of the $X(i)$ value. Our scheme also widens the chaotic system's beginning value range.

2.4 Diagram of bifurcation

To ensure that the pseudo-random sequence values of chaotic system iteration are evenly distributed throughout a range, bifurcation diagram can be used to visualize the distribution of function values. Figure 1 shows parameter μ range of mapping is represented by the x-axis of the bifurcation diagram, while the values produced by the mapping are represented by the y-axis (Kaçar et al., 2022). One may judge the quality of a chaotic mapping using the bifurcation diagram. 1D chaotic system's sequence may be examined using the bifurcation diagram to see if it is randomly distributed. In Figure 1A, $K_1 = -465.7689$. The logistic mapping bifurcation diagram is displayed in Figure 1B, with parameter $\mu \in [0, 4]$. In Figures 1C,D, $\mu \in [0, 100]$, K_1 and K_2 are set to -465.7689 , respectively. The uniform distribution of values within the range of $[0,1]$ is evident, suggesting that the suggested. The chaotic behavior of a 1D system is good. These demonstrate its complicated properties and continuous chaotic range when seen from the perspective of bifurcation trajectory and diagram.

2.5 Lyapunov exponent

Lyapunov exponent (LE) is one of crucial reference indices to determining if the chaotic system has especially chaotic qualities. The following formula explains how the LE is Eq. 4:

$$LE = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln |\dot{f}(x_i)| \quad (4)$$

The representation of a chaotic system is $f(x_i)$. The value of LE may be found in the formula by calculating the derivative of $f(x)$ and averaging the logarithms. A system is considered chaotic when the LE value is higher than 0. Conversely, a system is considered stable when the LE value is smaller than 0. We will determine whether a chaotic system is in a chaotic state within the parameter range by looking at the positive and negative LE values.

Figure 2 displays $\mu \in (0, 1]$ LE diagrams for 1D-SASCS chaotic system, the Logistic map and 1D map. We select $X(1)=0.1$ in this case, $K_1=465.7689$ and $K_2=465.7689$, our chaotic system has a large range of control settings since it consistently maintains a positive LE value. When $K_2=465.7689$ our LE values are the greatest, suggesting that our chaotic scheme has more complicated nonlinear behavior and superior unpredictability.

2.6 Sample entropy

The accuracy of sample entropy (SE) (Richman and Moorman, 2000) is higher than that of approximation entropy. The complexity of the output produced by chaotic systems during iteration is measured quantitatively. A positive SE shows chaotic behavior in the created sequence, which deviates from conventional regularity. A higher SE value denotes less regularity in the sequence, which suggests that the chaotic system's behavior is more complicated. The SE of various chaotic systems is calculated using the computation technique outlined. The SE of our new chaotic system that we have presented is compared with other 1D chaotic systems in Figure 3 and we set the initial value $X(1)=0.5$ for all chaos. As can be seen, our suggested 1D chaotic system achieves positive SE values for all control parameters. The outcomes of our experiments show that our chaotic system operates effectively. The computation equations for SE are as Eq. 5:

$$SE(m, r, N) = -\log \frac{A}{B} \quad (5)$$

In which A and B denote two successive random sequences of chaos, respectively, and m, the array's dimension, N, the sequence length, and r, the threshold. The Chebyshev distance between A and B is computed, and it is not more than the threshold's percentage. We set our chaotic, 1D-SASCS and Logistic, $X(1)=0.9$, $m=1$, $r=0.2$. As can be observed, the SE value is somewhat larger than the SE value of other 1D chaos when $K_1=465.7689$ and is comparatively steady. The SE value is larger than 0 for $K_2=465.7689$, which satisfies all safety standards.

3 Related algorithms

We introduce a cross-plane color image encryption scheme in the section. The architecture of cross-plane encryption technique is shown in Figure 4. The picture is converted into a 384-bit key using SHA-384. This key and the chaotic matrix produced by a 1D chaotic system are combined to make the encryption key. The image's three planes are split up, and each plane is simultaneously according to chaotic system and

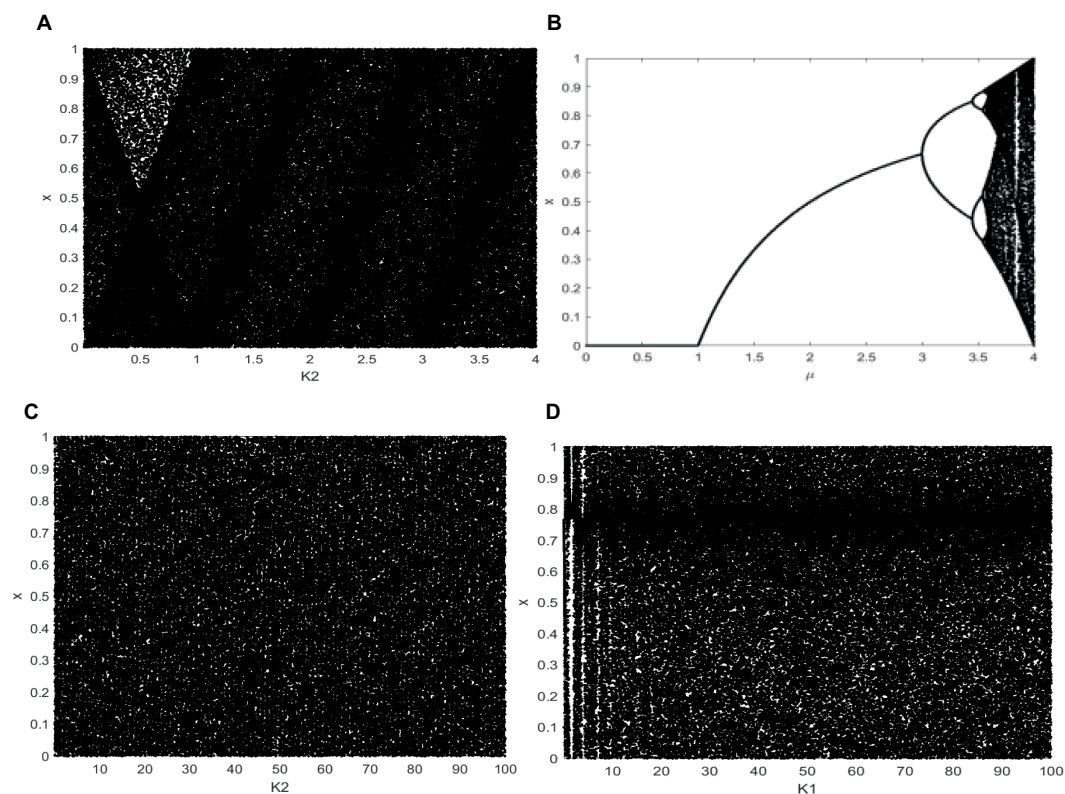


FIGURE 1

Bifurcation diagram. (A) When $K_1 = -465.7689$, bifurcation diagram of 1D chaotic map. (B) Logistic map, (C) When $K_1 = -465.7689$, larger range of 1D chaotic map. (D) When $K_2 = -465.7689$ larger range of 1D chaotic map.

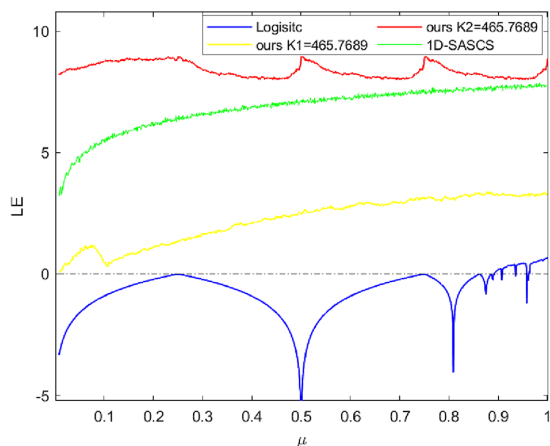


FIGURE 2

The LE results of logistic map, 1D-SASCS and our method with K_1 and K_2 .

an improved Dijkstra algorithm for cross-plane scrambling. This allows the original image's pixel to appear at any location in any plane, making it more difficult for an attacker to anticipate where a pixel would appear. An adaptive diffusion approach is used after obtaining the scrambled matrix. This algorithm starts with bidirectional diffusion on the rows and columns, and then moves on to random diffusion over the color planes. Finally, the planes of the image were merged to obtain the final

encrypted image. By modifying pixel values to improve security, and because both the improved Dijkstra method and the adaptive diffusion based on plane distribution are reversible, the algorithm can retrieve the original image information using the proper key.

3.1 Key generation

The research suggests a key generation process that generates four chaotic sequences using a 1D chaotic system. Because these sequences leverage chaos' unpredictable nature. To enhance unpredictability, we omit the first 1,000 iterations of the chaotic iterations. Moreover, this key generation mechanism makes ordinary images highly sensitive. The four generated chaotic sequences are denoted as V_1 , V_2 , V_3 , and V_4 . D_1 , D_2 , and D_3 are matrices generated from the chaotic sequences, with the size of $M \times N$.

RGB image P to be encrypted is first input into SHA-384 to obtain the 384-bit key Z . Z is the key shifted into 96 decimal numbers, each of which has a length of four digits. Z can be represented as $Z = h_1, h_2, h_3, \dots, h_{96}$. Next, use Z to obtain parameters $C_1 \dots C_{12}$. Then, 3 chaotic sequences of U_1 , U_2 , and U_3 are generated using the specific generation method, as Eqs. 6, 7:

$$\begin{cases} C_1 = h_1 + h_2 + \dots + h_8 \\ C_2 = h_9 + h_{10} + \dots + h_{16} \\ \vdots \\ C_{12} = h_{89} + h_{90} + \dots + h_{96} \end{cases} \quad (6)$$

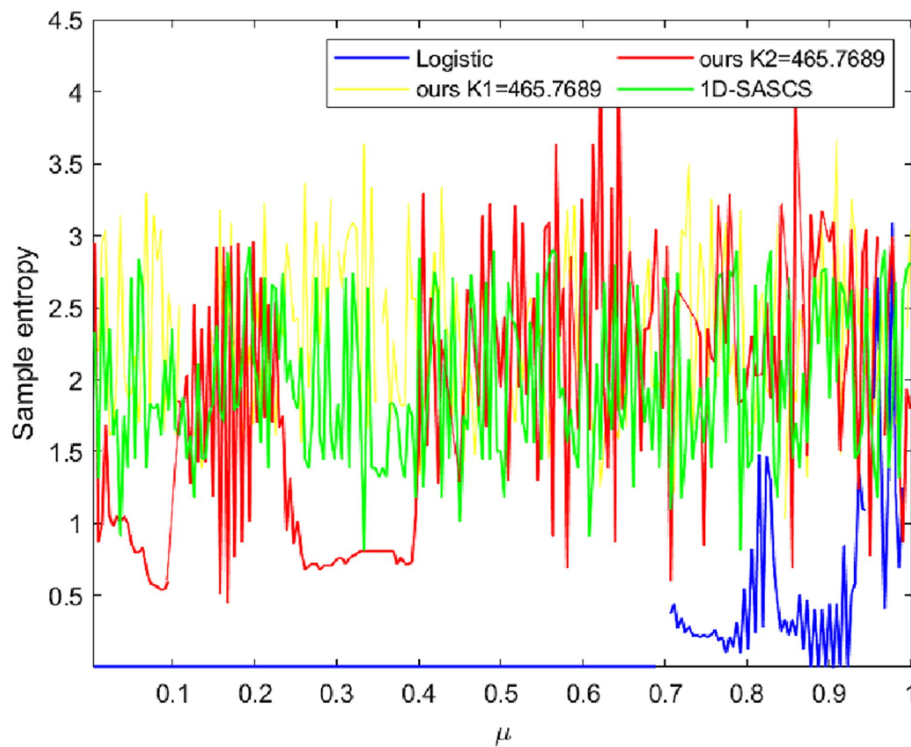


FIGURE 3

The sample entropy comparison on logistic map 1D-SASCS and our method with K_1 and K_2 .

$$\begin{cases} U_1(i,j) = \text{floor} \left(\text{mod} \left(\begin{matrix} V_1(i,j) * 10000 - \\ \text{floor}(V_1(i,j) * 10000), 256 \end{matrix} \right) \right) \\ U_2(i,j) = \text{floor} \left(\text{mod} \left(\begin{matrix} V_2(i,j) * 10000 - \\ \text{floor}(V_2(i,j) * 10000), 256 \end{matrix} \right) \right) \\ U_3(i,j) = \text{floor} \left(\text{mod} \left(\begin{matrix} V_3(i,j) * 10000 - \\ \text{floor}(V_3(i,j) * 10000), 256 \end{matrix} \right) \right) \end{cases} \quad (7)$$

3.2 Dijkstra algorithm

One kind of greedy method for determining the shortest path for a single source in weighted networks is the Dijkstra algorithm. It can be applied to both directed and undirected graphs. It is used here to resolve the shortest path issue with directed and undirected graphs. Figure 5 shows that the algorithm starts from vertex A and eventually obtains the set $U \{A, C, F, B, E, D\}$.

3.3 Improved Dijkstra algorithm

Only pixels on the same plane or multiple operations can be scrambled using conventional color image schemes. Therefore, it is crucial to create a scrambling algorithm that is both effective and secure. This research enhances the position updating procedure to better satisfy the demands of image encryption. As for the pixel weight, which influences both the layer value and the pixel's coordinates in the plane, we utilize a chaotic matrix. By essentially

removing the link between pixel locations and lowering the correlation between neighboring pixels, this method makes it more difficult to anticipate the position of pixels.

Our improved Dijkstra algorithm efficiently makes use of the inter-plane interactions between pixels, shuffle the image pixel position, arrange it across planes, in contrast to conventional color image scrambling techniques. The spatial associations of pixels can be more randomly shuffled, enabling them to appear at random on any plane. This algorithm only requires a single operation to complete the encryption process, rather than encrypting the three planes of an RGB image separately multiple times. It can better leverage the relationships between pixels across different planes, allowing pixels to quickly appear at any position on any plane. Original image P with the size of $M \times N$, this scheme for scrambling $H_1(a, b)$, $H_2(a, b)$, and $H_3(a, b)$ obtained by scanning P from left to right is as shown below.

Step 1: The four chaotic sequences V_1 , V_2 , V_3 , and V_4 are taken with lengths of $M \times N$, $M \times N$, $M \times N$, and $3 \times M \times N$, respectively.

Step 2: Three chaotic matrices are reshaped by processed the V_1 , V_2 , and V_3 chaotic sequences, denoted as D_1 , D_2 , and D_3 , respectively. Where 'sort' means to sort the elements of an array. Obtain the index matrices I_1 , I_2 , and I_3 for the three chaotic matrices, as Eq. 8:

$$[\sim, I_i] = \text{sort}(D_i) \quad (8)$$

Step 3: The three planes of image P — H_1 , H_2 , and H_3 —are scrambled to obtain P_1 , P_2 , and P_3 according to the three index matrices a and b , D_i acts as the pixel's weight to guide pixel movement, 'find' represents a vector that returns a linear index, as Eq. 9:

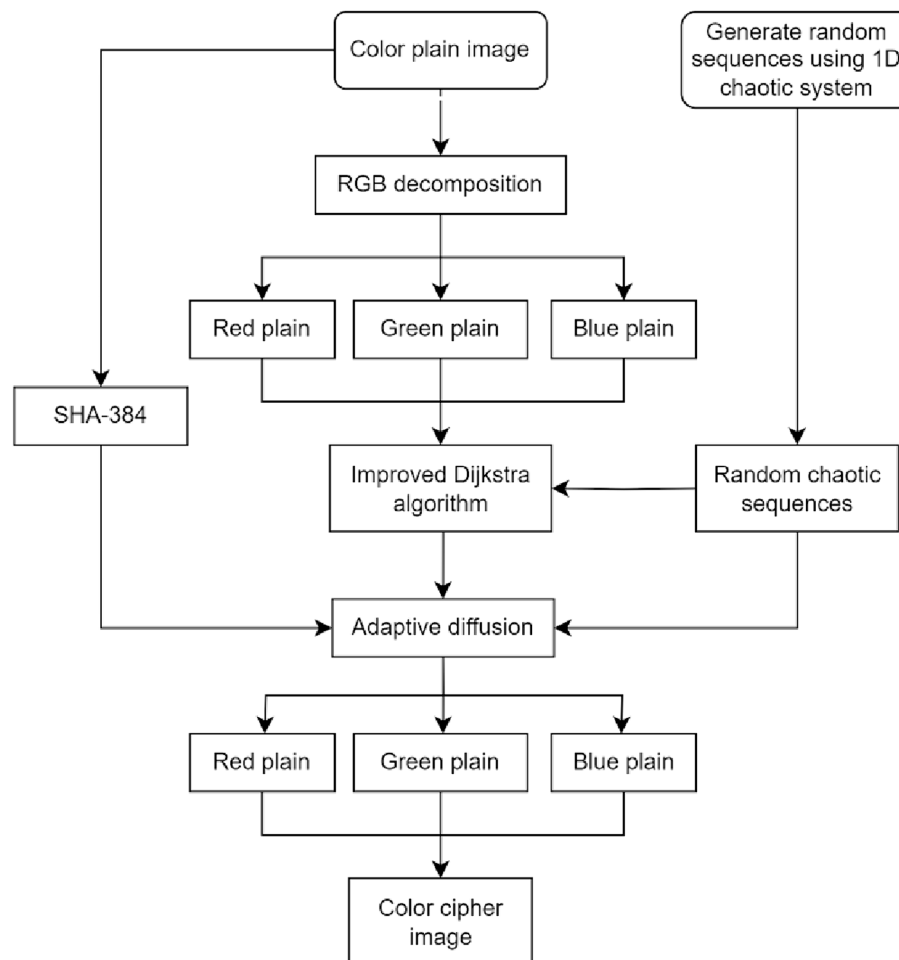


FIGURE 4
The encryption process for a flowchart.

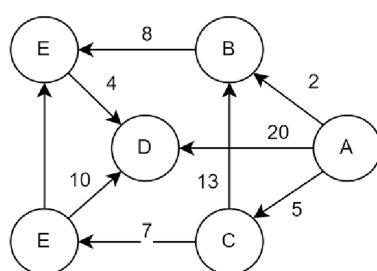


FIGURE 5
An example of in a directed graph.

$$\begin{cases} [m,n] = \text{find}(D_i = N * (a-1) * b) \\ P_i(m,n) = H_i(a,b) \end{cases} \quad (9)$$

Step 4: Reshape V_4 into a matrix with a row length of 3 and a column length of $M \times N - 1$, obtaining matrix I_4 . The I_4 index is sorted by row priority and P1, P2 and P3 are scrambled across planes according to the improved Dijkstra algorithm I_4 will guide the pixel to which level of the R, G, B plane, 1, 2, 3 stand for R, G, and B, respectively. Columns' indicates the plane where the pixel values are located.

Both the original Dijkstra algorithm and the improved Dijkstra algorithm are methods for determining the shortest path. The shortest path cross-plane scrambling algorithm is random, and the image pixel is determined by the point-to-point position of the chaotic system, which ensures that each pixel of the image can determine the final position, and ensures the integrity and randomness of the pixel. The magnitude of the comparison weight affects how far pixels shift in relation to their ultimate location. A cross-plane configuration for a $3 \times 3 \times 3$ colored image is shown in Figure 6. Our planes are initially positioned as follows: $R(1,1)=1$, $G(1,1)=1$, $B(1,1)=1$. The positions are changed into $R_1(3,1)=1$, $G_1(2,2)=1$, $B_1(3,3)=1$ based on our input data: $I_1(3,1)=1$, $I_2(2,2)=1$, $I_3(3,3)=1$. This completes the first step of shuffling. The value of V_4 specifies the plane into which the pixel will be shuffled, and it indicates the weights allocated to the pathways used to shuffle the image. The row-wise sorting of V_4 is I_4 . For instance, 2, 1, and 3 are in the first row of I_4 . $R_1=1$ positions are positioned in the second plane, $G_1=1$ positions are positioned in the first plane, and $B_1=1$ positions are positioned in the third plane. After guidance, we obtain $R_2(1,2)=1$, $G_2(1,2)=1$, and $B_2(2,1)=1$ based on the index order established: $1 \rightarrow 2, 2 \rightarrow 3 \dots \rightarrow 3 \times M \times N, 3 \times M \times N \rightarrow 1$. We obtain the final shuffled image when the three planes have finished shuffling. The distribution of each element in the sequence is uniform and random.

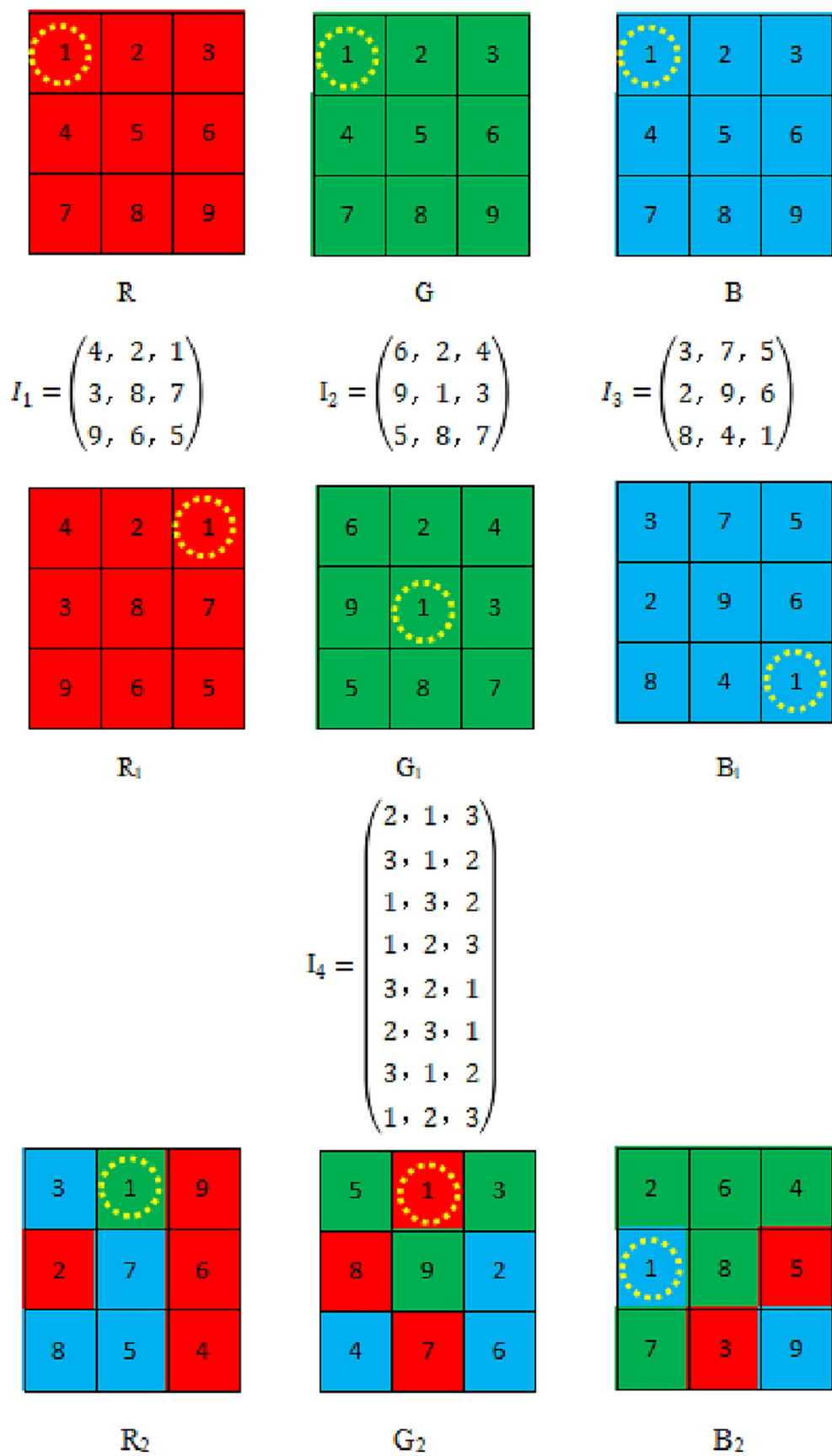


FIGURE 6
Example of an improved Dijkstra Algorithm.

3.4 Adaptive diffusion based on plane distribution

Encrypted pixels typically solely pertain to the current pixel; they have no effect on following pixels. Even if the current pixel undergoes slight changes in the image. The adaptive diffusion strategy proposed in the paper is based on plane distribution, that is an encryption scheme that utilizes the image's R, G, and B layers' pixel values as keys for one another. The image encryption task can be successfully completed with just one diffusion operation on scrambled image. Modifying pixel value of image increases its security and makes it harder for attackers to obtain the original. Specifically, row-column diffusion takes place inside each of the pixels' individual planes first, followed by diffusion between planes. As a result, pixels differ from one plane to the next. The values of succeeding pixels shift significantly when one does. After the original image has been disturbed by the trans-plane scrambling of improved Dijkstra algorithm. Since neighboring pixels in scrambled image originate from several color planes, The scrambled image is then placed in adaptive diffusion based on plane distribution, the processing sequence is arbitrary and kept a secret, pixel value is severely destroyed by our algorithm, the safety of proliferation is further enhanced. The technique creates a consistent pixel distribution and one-step encryption based on protecting private information, as (Eqs. 10, 11):

$$\left\{ \begin{array}{l} R(1, j) = R(1, j) \oplus U_1(1, j) \\ R(i, j) = R(i-1, j) \oplus I_1(1, j) \oplus R(i, j) \\ G(1, j) = G(1, j) \oplus U_2(1, j) \\ G(i, j) = G(i-1, j) \oplus I_2(1, j) \oplus G(i, j) \\ B(1, j) = B(1, j) \oplus U_3(1, j) \\ B(i, j) = B(i-1, j) \oplus I_3(1, j) \oplus B(i, j) \\ R(i, 1) = R(i, 1) \oplus U_1(i, 1) \\ R(i, j) = R(i, j-1) \oplus I_1(i, j) \oplus R(i, j) \\ G(i, 1) = G(i, 1) \oplus U_2(i, 1) \\ G(i, j) = G(i, j-1) \oplus I_2(i, j) \oplus G(i, j) \\ B(i, 1) = B(i, 1) \oplus U_3(i, 1) \\ B(i, j) = B(i, j-1) \oplus I_3(i, j) \oplus B(i, j) \end{array} \right. \quad (10)$$

$$\left\{ \begin{array}{l} k = \text{mod}(k, 12) + 1 \\ R(i, j) = \text{mod} \left(\left(\text{double} \left(\left(\text{bitxor} \left(R(i, j), \left((i+j)^2 * R(i, j) \right) \right) \right) \right), 256 \right) \right) \\ G(i, j) = \text{mod} \left(\left(\text{double} \left(\left(\text{bitxor} \left(G(i, j), \left((i+j)^2 * R(i, j) \right) \right) \right) \right), 256 \right) \right) \\ B(i, j) = \text{mod} \left(\left(\text{double} \left(\left(\text{bitxor} \left(B(i, j), \left((i+j)^2 * G(i, j) \right) \right) \right) \right), 256 \right) \right) \end{array} \right. \quad (11)$$

In this case, $M \times N$ represents the encrypted image P's size. In addition, the image consists of three layers: R, G, and B. The modulo operation is denoted by 'mod', the bitwise XOR operation by 'bitxor', and the key generation set in Section 2 is denoted by $p(k)$, where k ranges from 1 to 12.

4 Simulation results and security analysis

To address the requirements of many situations, we discuss the results of simulations using a variety of image formats. We also detail a significant amount of security research to show the safety and effectiveness of our approach. All experiments are conducted and simulated using MATLAB 2021a on the laptop with an i7-10710U CPU. In this paper, two sets of ablation experiments are set up, when the encryption algorithm is only named EX1 using the improved Dijkstra algorithm, and when the encryption algorithm is only used adaptive diffusion based on plane distribution, it is named EX2.

This part shows the simulation and testing of the encryption technique provided in part 3. We perform tests on the original images by employing different-sized standard test images and using the encryption method suggested in this study as Figure 7 shown. No meaningful data are present in the encrypted image in Figure 7A. The contrast between original image and encrypted image, the latter of which is a completely black image, is also shown in Figure 7D. This result indicates that our method applies to image encryption and retrieves images without any loss.

4.1 Simulation results and histogram analysis

Histogram analysis is a highly effective means of presenting data in a cryptographic system because it provides a visual presentation of the statistical information contained within image pixels. Regarding cryptography, the distribution of the cipher in the histogram must be as uniform as possible because any deviations can provide attackers with valuable statistical information that can be used to compromise the system's security. As seen in Figure 8, we compared different images using histograms. Figures 8B,C show histograms of plaintext and ciphertext, respectively, demonstrating that our encryption scheme produces a relatively flat histogram. This result suggests that there is some degree of assault resistance in our design.

4.2 Key space analysis

A wider key area is necessary to successfully deter attackers from acquiring the correct key. A secure cryptographic system (Chai et al., 2017) often requires a key space greater than 2^{100} . Key space for SHA-384 of the scheme is 2^{192} and has 12 keys, which is far greater than 2^{100} to effectively fend against brute-force attack.

4.3 Information entropy

A signal source for distribution can be quantitatively described using information entropy. Moreover, 8 is optimal value of entropy for



FIGURE 7

Simulation results of the image encryption algorithm proposed are as follows: (A) and (C) Initial and decrypted images, (B) Encrypted images, and (D) Difference between initial images and decrypted image (A–C). The images ‘House’, ‘Couple’ and ‘Female’ have been downloaded from [USC-SIPI Image Database](#).

an image of an 8-bit binary. The formula for computing information entropy as Eq. 12:

$$E(a) = -\sum_{i=0}^{255} Q(a_i) \log_2 p(a_i) \quad (12)$$

Where a_i represents a pixel’s value, and $Q(a_i)$ stands for the frequency of a_i . When each value is equally likely to occur, the maximum value will be reached by information entropy. In an 8-bit image, 256 is gray level, and when each pixel appears with a probability of $1/256$, the maximum information entropy can be obtained. In this section, we present entropy testing on the Lena ($256 \times 256 \times 3$) image, and Table 1 provide a comparison of the test data utilized to develop our approach. Even if our values are not the highest, they are similar and adhere to security standards, which shows our scheme has good performance.

Table 2 shows our scheme results about the entropy for diverse image. Because neighboring pixels are associated, rather than random, the plaintext image has a low entropy. This shows the validity of our hypothesis and comes close to the predicted maximum value of 8. The encrypted images suggested in this paper have erratic distributional properties, from which no usable data can be derived. Table 3 shows the information entropy ablation experiment, and it can be seen that the EX1 and EX2 values are low and do not meet the safety criteria.

4.4 Analysis of adjacent pixel correlation

The initial pixels’ regular distribution usually creates a stable correlation between them, which can negatively impact the quality of the ciphertext when introduced in encryption. For evaluating the correlation between relevance pixels in our proposed encryption system, for test items, we select 3,000 pairs of pixels with the formula expressed as Eq. 13:

$$C(x,y) = \frac{\sum_{i=1}^N (X_i - L(a))(Y_i - L(b))}{\sqrt{\frac{1}{N} \sum_{i=1}^N (X_i - L(a))^2} \sqrt{\frac{1}{N} \sum_{i=1}^N (Y_i - L(b))^2}} \quad (13)$$

Where $L(A)$ and $L(B)$ are the sequences ‘a’ and ‘b’ respectively, in mathematical expectations. A greater correlation between the sequences ‘a’ and ‘b’ is indicated by a larger correlation coefficient, while a correlation coefficient that is closer to zero suggests less correlation.

In contrast to the accompanying ciphertext image, which is evenly scattered over the plane in a diagonal orientation. Figure 9 displays the pixel distribution in the test image and its surroundings. In Table 4, the correlation coefficients are displayed.

Between proposed scheme and the corresponding ciphertext images of different plaintext images, owing to the large data redundancy

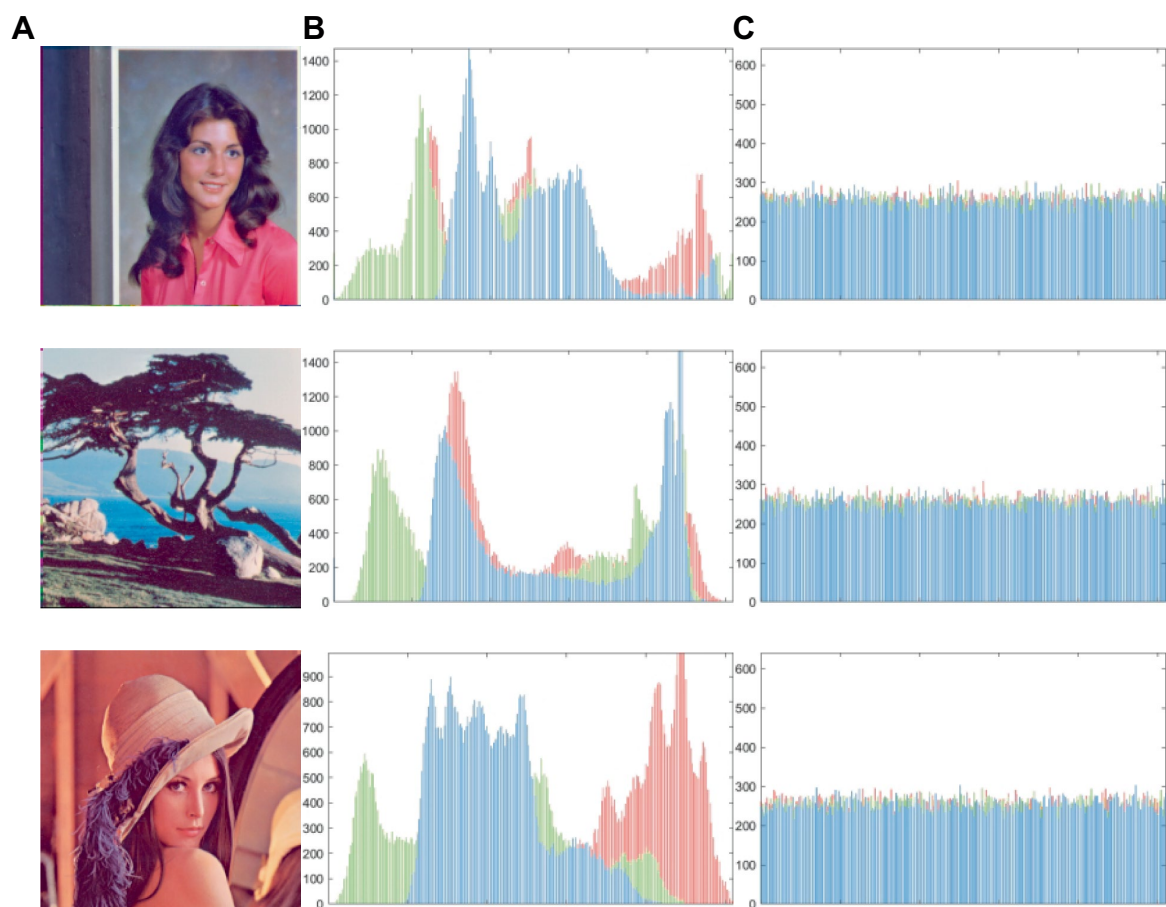


FIGURE 8 (A) The plaintext image, (B) and (C) the histogram of plaintext and cipher images. The images ‘Female’, ‘Lena’ and ‘Tree’ have been downloaded from USC-SIPI Image Database.

TABLE 1 Information entropy compares the ‘Lena’ (256 × 256 × 3) image with other schemes.

Encryption schemes	R	G	B	Avg
Lena	7.2353	7.5683	6.9176	7.2404
proposed	7.9969	7.9974	7.9970	7.9971
EX1	7.7253	7.7305	7.7292	7.7283
EX2	7.7974	7.7973	7.7972	7.7973
Zhang Y. Q. et al. (2020)	7.9917	7.9912	7.9917	7.9915
Wang X. et al. (2022) and Wang Y. et al. (2022)	7.9973	7.9971	7.9971	7.9972
(Chai et al. (2019)	7.9973	7.9969	7.9971	7.9971
(Hosny et al. (2021)	7.9956	7.9949	7.9953	7.9953

of the plaintext image, the nearby pixels exhibit a high correlation coefficient. Since the ciphertext image’s correlation coefficient is practically 0, the suggested approach can be successful in eliminating the substantial relationship between adjacent pixels in plaintext image. Here, we investigate correlation coefficients of ciphertext images using various encryption techniques. Three planes of the test image Lena, which has dimensions of 256 × 256 px, are used to determine correlation coefficients. Table 5 displays data for the correlation coefficient comparison of various ciphertext images. The values of our

scheme are closer to zero, EX1 and EX2 have high correlation between adjacent pixels.

4.5 Differential attack experiment

Differential attack is a extensive used and powerful attack strategy. By evaluating the impact of the change rate of each pixel between original and encrypted images, we find that the best performance indicators for judging differentiated attacks are the number of pixels change rate (NPCR) and unified average changed intensity (UACI). K_1 and K_2 are two encrypted outputs of the same plaintext image produced after fine-tuning, NPCR and UACI (Gao et al., 2022) calculated as Eqs. 14, 15:

$$NPCR(C_1, C_2) = \sum_{i=1}^M \sum_{j=1}^N \frac{U(i,j)}{D} \times 100\%, \tag{14}$$

$$UACI(C_1, C_2) = \sum_{i=1}^M \sum_{j=1}^N \frac{|K_1(i,j) - K_2(i,j)|}{D * F} \times 100\%, \tag{15}$$

U is the difference between K_1 and K_2 , F is the greatest pixel value, D is a total number of color plane pixels. $K_1(i, j) = K_2(i, j)$ if $U(i, j) = 0$; otherwise,

TABLE 2 Information entropy of different size and different images.

Image size	Images	Plain images			Cipher images		
		R	G	B	R	G	B
256×256×3	4.1.01	6.4200	6.4457	6.3807	7.9971	7.9972	7.9967
	4.1.02	6.2499	5.9642	5.9309	7.9971	7.9973	7.9961
	4.1.03	5.7150	5.3738	5.7117	7.9970	7.9973	7.9972
	4.1.04	7.2549	7.2704	6.7825	7.9974	7.9973	7.9974
	4.1.05	6.4311	6.5389	6.2320	7.9977	7.9969	7.9975
	4.1.06	7.2104	7.4136	6.9207	7.9970	7.9971	7.9973
	4.1.07	5.2626	5.6947	6.5464	7.9972	7.9970	7.9971
512×512×3	4.2.05	6.7178	6.7990	6.2138	7.9993	7.9994	7.9993
	4.2.06	7.3124	7.6429	7.2136	7.9993	7.9992	7.9994
	4.2.07	7.3255	7.3912	6.9169	7.9993	7.9993	7.9994

TABLE 3 Different images of different images of ablation experiments have different sizes of information entropy.

Image size	Images	EX1			EX2		
		R	G	B	R	G	B
256×256×3	4.1.01	7.7972	7.797	7.7971	7.8972	7.8970	7.8971
	4.1.02	6.2948	6.2921	6.2927	7.8970	7.8973	7.8971
	4.1.03	5.9691	5.9628	5.9749	7.8976	7.8973	7.8975
	4.1.04	7.4229	7.4276	7.4255	7.8970	7.8973	7.8971
	4.1.05	7.0711	7.0615	7.0676	7.8972	7.8970	7.8971
	4.1.06	7.5335	7.5341	7.5377	7.8967	7.8971	7.8969
	4.1.07	6.5855	6.5814	6.5797	7.8972	7.8975	7.8972
512×512×3	4.2.05	6.6623	6.6639	6.6642	7.9974	7.9973	7.9972
	4.2.06	7.7605	7.7613	7.7632	7.9972	7.9970	7.9971
	4.2.07	7.5839	7.5824	7.5820	7.9893	7.9893	7.9893

$U(i, j)=1$. As shown in Table 6, we perform a comparison test of our method EX1 and EX2 against others. Using a Lena image (256×256×3 px). The NPCR and UACI are found to be extremely near to the theoretical maximums of 99.61 and 33.46%, respectively (Kumar et al., 2018). We also observe that our UACI values meet the safety standards and that the NPCR values are higher than those of other methods. EX1 and EX2 does not meet safety standards. Table 7 shows our scheme's NPCR and UACI values for various image sizes are near the theoretical value, demonstrating the system's strong potential for differential protection.

4.6 Resistance to data loss and noise

The risk of data loss or noise contamination exists while sending data over the internet. Images that have lost data or are tainted by noise must be able to retrieve most of their information when using a trustworthy encryption technique. To evaluate our system's resistance to these dangers, we simulate data loss and noise pollution on ciphertext image. As shown in Figure 10, we tested different attacks, the experiment proved our method successfully retrieves most of the information while reconstructing an ordinary, visually clear image. Our suggested system can therefore successfully withstand data loss and noise pollution.

Peak signal-to-noise ratio (PSNR), a statistic measures the degree of visual distortion, is objective. When the PSNR is high, we might get

results that are closer to original image. The computation equations for PSNR and MSE are as Eqs. 16, 17:

$$\text{PSNR} = 10 \times \log_2 \left(\frac{\text{MAX}_I^2}{\text{MSE}} \right), \quad (16)$$

$$\text{MSE} = \frac{1}{A * B} \sum_{i=0}^{A-1} \sum_{j=0}^{B-1} [M(i,j) - N(i,j)]^2, \quad (17)$$

For plaintext and ciphertext images, $M(i,j)$ and $N(i,j)$ are the values of pixel, respectively. Maximum pixel value for images is MAX_I , Table 8 shows PSNR values larger than 10 dB this technique outperforms previous attack techniques in terms of resistance to Gaussian noise. We may thus draw the conclusion that this plan can ensure security and maintain a strong connection to typical images.

4.7 Image autocorrelation test

2D image autocorrelation compares all possible pairs of two pixels which shows likelihood of having similar values based on

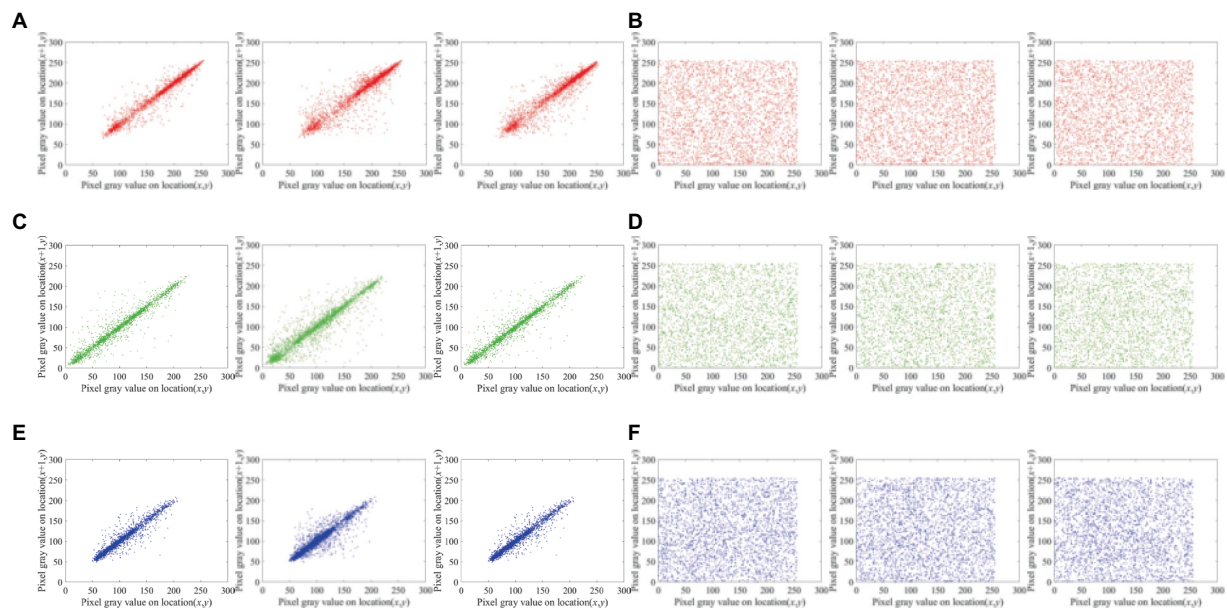


FIGURE 9

Lena (256 × 256 × 3 px) horizontal, diagonal and vertical distribution of adjacent pixels: (A) and (B) Distribution of adjacent red pixels in the plaintext and ciphertext, (C) and (D) distribution of adjacent green pixels in the plaintext and ciphertext, and (E) and (F) distribution of adjacent blue pixels in plaintext and ciphertext.

TABLE 4 Correlation among adjacent pixels in different sizes and different images.

Images	Directions	Plain images			Cipher images		
		R	G	B	R	G	B
4.1.01	H	0.9593	0.9678	0.9462	−0.0051	−0.0033	−0.0028
	D	0.9476	0.9563	0.9398	−0.0056	0.0006	−0.0064
	V	0.9766	0.9715	0.9585	−0.0068	−0.0068	−0.0057
4.1.02	H	0.9610	0.9511	0.9491	0.0029	−0.0015	0.0019
	D	0.9167	0.9049	0.8844	0.0062	0.0031	0.0009
	V	0.9588	0.9320	0.9092	0.0079	0.0072	−0.0086
4.1.03	H	0.9453	0.9226	0.8936	0.0060	0.0061	0.0064
	D	0.9125	0.9066	0.8705	−0.0071	0.0008	0.0007
	V	0.9739	0.9752	0.9711	0.0068	−0.0067	0.0049
4.2.05	H	0.9537	0.9678	0.9237	−0.0061	0.0046	−0.0025
	D	0.9354	0.9287	0.9123	−0.0031	−0.0026	0.0041
	V	0.9720	0.9560	0.9648	−0.0035	−0.0053	0.0066
4.2.06	H	0.9576	0.9707	0.9665	−0.0056	−0.0003	0.0026
	D	0.9417	0.9506	0.9515	−0.0015	−0.0005	−0.0056
	V	0.9568	0.9720	0.9731	0.0062	−0.0036	0.0006
4.2.07	H	0.9656	0.9781	0.9671	0.0046	−0.0028	0.0057
	D	0.9533	0.9712	0.9573	−0.0024	−0.0033	0.0045
	V	0.9605	0.9816	0.9628	−0.0057	0.0056	−0.0009

distance and separation direction. Generally, the autocorrelation of a planar image is visualized as a wave and cone shape in the spatial domain, whereas the autocorrelation of a cipher image appears as a uniform and level surface. Equation is used for the image autocorrelation is calculated as in Eq. 18:

$$'(x,y) = D^{-1}D[O(M,N)] * \bar{D}[O(M,N)] \quad (18)$$

In this case, D^{-1} stands for the conjugate Fourier transform, $O(M, N)$ is pixel's value at position (M, N) in picture, D is the

TABLE 5 Comparison of correlation coefficients with different methods using the image 'Lena'.

Planes	Directions	Plane image	Our scheme	EX1	EX2	Hosny et al. (2021)	Hosny et al. (2022)	Zheng et al. (2023)
R	H	0.9746	−0.0064	0.0150	0.0055	0.0064	−0.0154	0.0071
	D	0.9406	−0.0007	0.0114	0.0119	−0.0026	0.0159	−0.0006
	V	0.9558	0.0039	0.0120	−0.3333	0.0160	−0.0102	0.0089
G	H	0.9722	0.0013	0.0070	−0.0046	0.0009	−0.0096	−0.0012
	D	0.9102	0.0015	−0.0316	0.4410	0.0125	−0.0162	−0.0043
	V	0.9458	0.0045	−0.0101	−0.0371	0.0034	0.0027	−0.0018
B	H	0.9478	0.0030	−0.0213	0.0314	0.0091	−0.0030	−0.0015
	D	0.8776	0.0017	0.0204	−0.0092	−0.0090	−0.0026	−0.0019
	V	0.9318	−0.0063	0.0119	−0.0056	−0.0045	0.0117	0.0041

TABLE 6 NPCR and UACI data testing using image of 'Lena'.

Lena	NPCR				UACI			
	R	G	B	Avg	R	G	B	Avg
Ours	99.6323	99.6338	99.6124	99.6261	33.5163	33.4215	33.4666	33.4681
EX1	3.3707	3.3707	3.3707	3.3707	1.8079	1.7963	1.7838	1.7960
EX2	96.7209	82.5531	63.6673	80.9804	32.9611	29.0607	22.1373	28.0530
Hosny et al. (2022)	99.6017	99.6124	99.6368	99.6149	33.4128	33.4980	33.4974	33.4694
Hosny et al. (2021)	99.6094	99.6124	99.6307	99.6175	33.4666	33.4241	33.4212	33.4373
Gao et al. (2022)	99.6180	99.6376	99.6003	99.6189	33.4285	33.4549	33.4275	33.4399

TABLE 7 NPCR and UACI of different images and different planes.

Images	NPCR (%)				UACI (%)			
	R	G	B	Avg	R	G	B	Avg
4.1.01	99.6567	99.5972	99.5697	99.6078	33.4495	33.5133	33.5127	33.4918
4.1.02	99.6216	99.6155	99.6140	99.6170	33.5080	33.5635	33.6437	33.5717
4.1.03	99.5911	99.5850	99.5865	99.5875	33.5879	33.4915	33.4298	33.5030
4.1.04	99.6231	99.6017	99.6048	99.6098	33.4754	33.5302	33.5021	33.5025
4.2.05	99.6181	99.6117	99.6193	99.6163	33.4725	33.5218	33.4894	33.4945
4.2.06	99.6151	99.6155	99.6014	99.6106	33.4426	33.4549	33.4847	33.4607
4.2.07	99.6231	99.5914	99.6220	99.6121	33.5338	33.4807	33.4993	33.5046

Fourier transform, and $P(x, y)$ is the autocorrelation function. According to Figure 11, we utilize 'Tree' as the test image with an encrypted image across the R, G, B color channels using it as our benchmark. The figure depicts our experimental results. The autocorrelation of the planar image shown in Figures 11B–D demonstrates a wave-like pattern, indicating that the probability of pixel pairs with the same pixel value is higher in planar images. By contrast, the cipher image is smoother according to the test results of autocorrelation (Figures 11F–H), reflecting that our proposed method effectively reduces the probability of equal pixel values.

4.8 Floating frequency test

The plain image should uniformly encrypt all rows and columns using a good image encryption technique. A key

indicator for assessing an encryption method that can generate stochastic data for all rows and columns and analyze the vulnerabilities in the encrypted image is the floating frequency test (Murillo-Escobar et al., 2019). For example, below is the procedure for determining the row and column floating frequencies for a 256×256 -px image.

Step 1: Set the 256-element image as a window in each row and column.

Step 2: Count the number of diverse components in every window.

Step 3: Determine a number of different items in each window, as well as the row and column floating frequency values.

Step 4: Determine the average values of the floating frequency for rows and columns.

Here is a sample of the selected color image 'Lena'. The frequency float test as shown in Figure 12, the row and column floating frequency values for the original image are relatively low

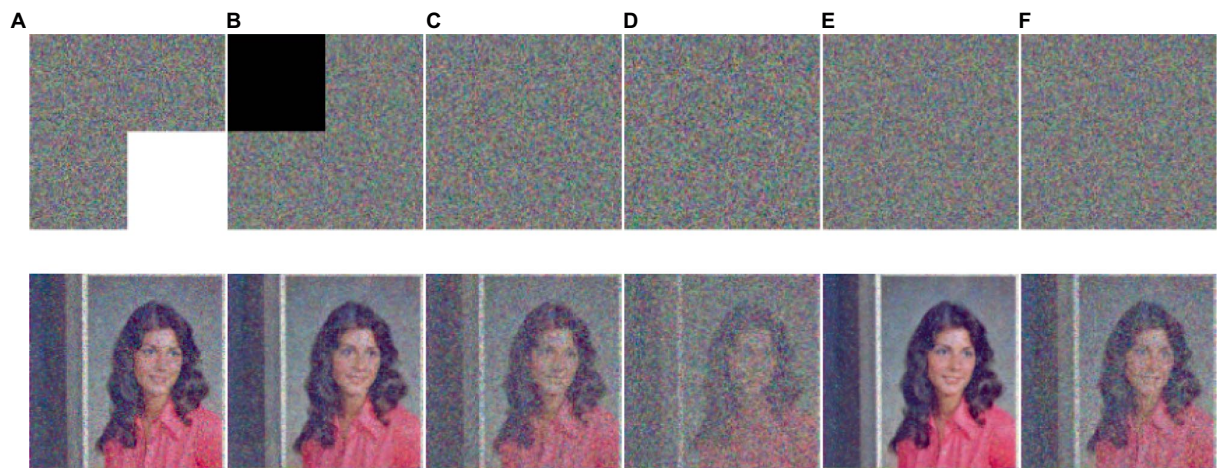


FIGURE 10
The first row shows the cipher images with data loss and different levels of noise, respectively, (A) is missing in the bottom right corner, (B) is missing in the top left corner, (C) is 0.1 density of salt and pepper noise, (D) is 0.2 density of salt and pepper noise, (E) is 0.000001 density of Gaussian noise, and (F) is 0.000002 density of Gaussian noise, While the second row shows the matching decrypted data. The image ‘Female’ has been downloaded from [USC-SIPI Image Database](#).

TABLE 8 MSE and PSNR are compared under different attack data.

Cipher image	MSE			PSNR (dB)		
	Red	Green	Blue	Red	Green	Blue
1/4 Data loss at the bottom-right corner	5,415	5,402	5,459	10.7945	10.8045	10.7591
1/4 Data loss at the top-left corner	5,456	5,445	5,402	10.7616	10.7705	10.8046
Gaussian noise = 0.000001	0.0493	0.0497	0.0492	61.1982	61.1701	61.2103
Gaussian noise = 0.000003	0.2601	0.2579	0.2575	53.9789	54.0165	54.0232
Salt&Pepper noise = 0.1	2,207	2,131	2,146	14.6927	14.8440	14.8135
Salt&Pepper noise = 0.2	4,309	4,432	4,365	11.7861	11.6647	11.7309

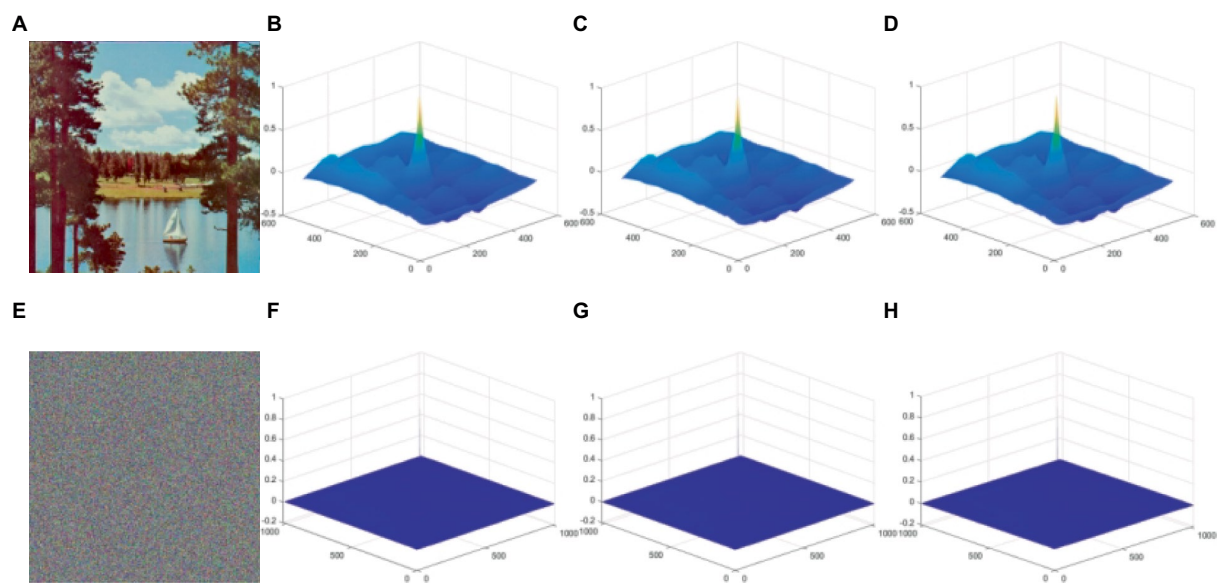


FIGURE 11
Test for graphic autocorrelation. (A) is the original image and (E) is the corresponding decrypted image. Plaintext images in the R, G, B planes are subjected to a 3D graphic autocorrelation test for (B–D), and (F–H) ciphertext image 3D graphic autocorrelation test in R, G, B planes. The image ‘Sailboat on lake’ has been downloaded from [USC-SIPI Image Database](#).

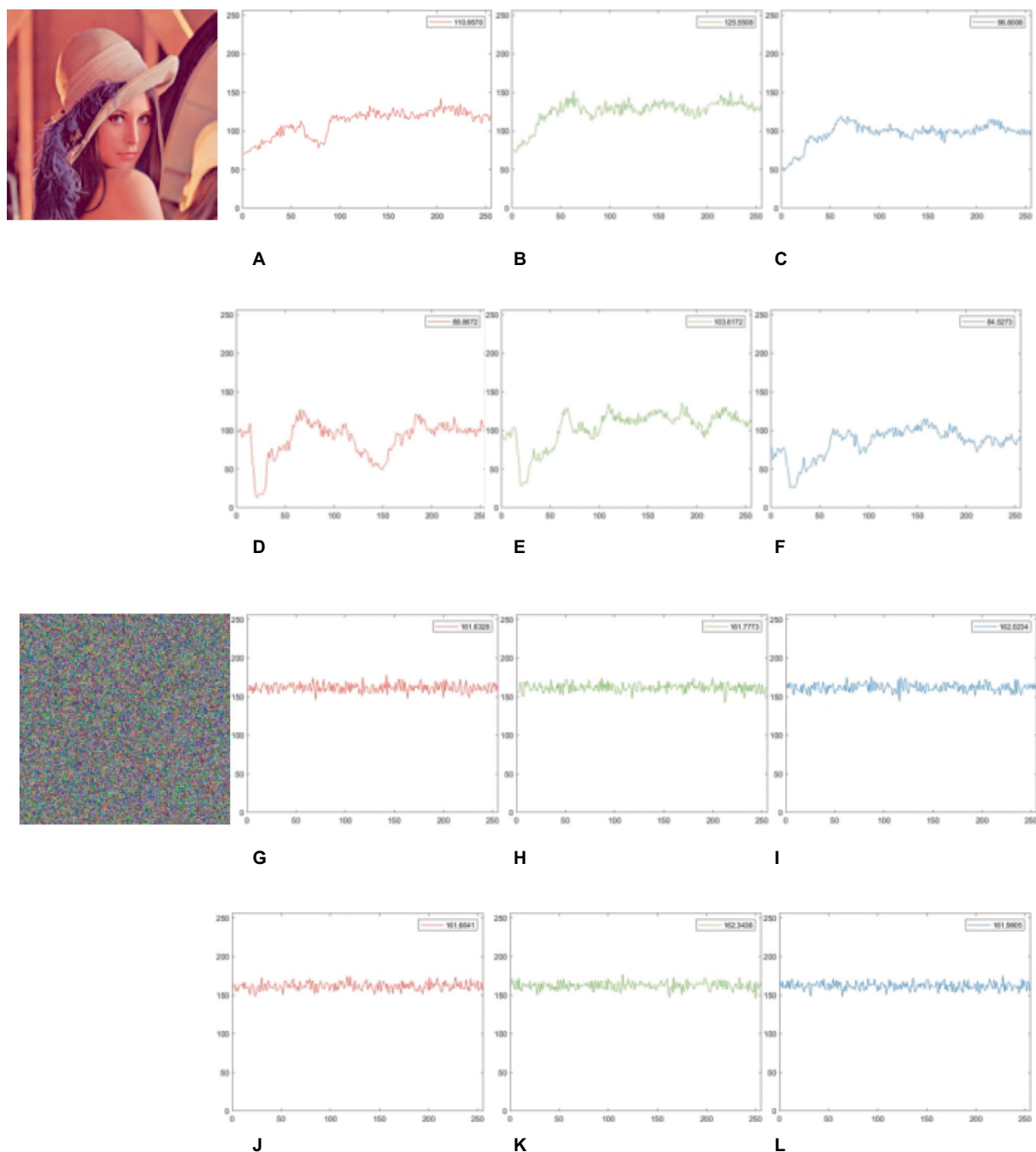


FIGURE 12

Floating frequency test for plain and cipher images. (A–C) and (G–I) show the row floating frequency of the plain and cipher image 'Lena' in the R, G, B channels. (D–F) and (J–L) show the column floating frequency of the plain and cipher image 'Lena' in R, G, B channels. The image 'Lena' have been downloaded from [USC-SIPI Image Database](https://www.usc-sipi.org/).

(Figures 12A–F), indicating the plain image's pixel distribution is uneven (with numerous repeated elements). Figures 12G–L displays the cipher image's row and column floating frequency values, both of which are rather high, at about 161, indicating that nearly 63% of the 256 elements in each column and row are unique. This implies our scheme generates a cipher image and a more uniform component distribution.

4.9 χ^2 test

χ^2 test provides a quantitative analysis of the homogeneity of the image pixel distribution. We calculate the image's χ^2 value (Liu et al., 2023) g formula and compare it with the benchmark value. The distribution of the image's pixels seems to be more uniform when the calculated value is lower than the standard value as Eqs. 19, 20:

$$\chi^2 = \sum_{i=0}^{255} \frac{(p_i - p)^2}{f} \quad (19)$$

$$p = \frac{M * N}{256} \quad (20)$$

Where the appearance's pixel frequency value i in image is represented by p_i , and the average frequency is represented by p . The benchmark for ciphertext images in the second test of this technique is 293.24783. As demonstrated in Table 9, the outcomes of this approach for numerous images are provided, and our ciphertext is fairly evenly distributed. EX1 does not meet safety standards, EX2 meets the safety standards but has a higher value than the scenario in this article.

Lena is then compared in Table 10 between our plan and other plans. It is clear that our system produces far less data than other systems, demonstrating our technique's superior resilience to attacks based on pixel feature distribution.

5 Conclusion

The present color image encryption techniques either encrypt each of the three planes independently or they include repetitive processes that reduce the algorithm's performance. To get beyond these problems, the paper has introduced a novel 1D chaotic system. By utilizing the new 1D chaotic system and Dijkstra algorithm, we have proposed a new improved Dijkstra algorithm and an adaptive diffusion cross-plane color encryption technique. We propose an image pixel that can make full use of the pixels of different planes and can directly process the three color planes of the color image to complete the cross-plane scrambling. A unique cross-plane permutation strategy has been suggested to increase the encryption system's security and effectiveness. In the process of chaotic scrambling using cross-planes, we make great use of the relationship between different planar pixels, which makes the pixels very shuffled in order, pixels can appear at arbitrary coordinates on any plane,

making it disrupting correlation between adjacent pixels and more difficult to predict pixel positions. Adaptive diffusion based on plane distribution utilizes the method of cross-plane diffusion, where any change in pixel values will result in a significant change in a large number of subsequent pixel values. According to the simulation results and security analysis in Chapter 4, it shows that our solution complies with various security standards, and most of the test indicators show that our solution is higher than the current popular image encryption schemes, it has been found to have stronger robustness and higher security. In this paper, differential attack experiment and resistance to data loss and noise simulated attack test are used respectively, and the experimental results show that our scheme is used that protects against attacks using specific plaintext and known plaintext, and compared to other schemes, our NPCR value is higher than other schemes, and the UACI value meets the safety standards. The original image is used to generate SHA-384 and a new chaotic system to compose the key, and the key space analysis shows that the key space size meets the security standards. The suggested approach has been demonstrated by simulation and security analysis to be successful, indicating that its security can render many attack schemes ineffective.

The proposed encryption technique avoids repeatedly encrypting the same areas of the image by making greater use of the correlation between pixels in distinct planes to encrypt the image just once. The improved Dijkstra algorithm used in this paper is a point-to-point encryption scheme. It avoids repeatedly encrypting the same areas of the image by making greater use of the correlation between pixels in distinct planes to encrypt the image just once. No new pixels are generated during the encryption process, and no pixels are lost, ensuring that the decrypted image is lossless. Color medical image is a special kind of RGB image, which has high privacy, and ciphertext security is related to the privacy and security of patients. Our scheme have been tested to the safety standards of Histogram Analysis, information entropy, analysis of adjacent pixel correlation, floating frequency test, image autocorrelation test, and χ^2 test, data analysis has shown that our protocols meet safety standards and protect patient privacy. However, currently, this scheme is only applicable to RGB images since only the position relationship between the three planes of the color image is considered in the design, the encryption scheme of single-channel or multi-channel image is not considered and is not suitable for grayscale images or special images. Compared with other popular schemes, the encryption scheme proposed in this paper is normal in terms of speed and efficiency, but with the enlargement of image size, the number of chaotic iterations and the computation of the final position of the pixel are getting larger and larger, the time required by the proposed scheme is also increasing, and the time cost is higher when large-size image encryption is required, so it is not suitable for encryption scheme. In the future, we will attempt to develop schemes suitable for multichannel image encryption and remote image encryption.

TABLE 9 χ^2 test between different images and planes.

Image	χ^2 test			
	Plain	Cipher	EX1	EX2
4.1.04	81,482	237.3906	46396.2734	259.9679
4.1.05	317,260	238.7396	108529.8567	260.8698
4.1.06	89,401	260.8932	45279.6093	279.6484
4.1.07	486,578	261.7838	199795.3515	264.3880
4.2.05	822,925	249.7903	770085.6176	267.7802
4.2.06	223,807	247.9980	74588.0410	277.8831
4.2.07	389,487	244.8919	182028.0351	282.3417

TABLE 10 χ^2 test comparison of our algorithm with other algorithms.

Algorithm	Proposed	EX1	EX2	Richman and Moorman (2000)	Liu et al. (2018)	Asgari-Chenaghlu et al. (2019)
χ^2	239.8008	21630.2343	261.5618	254.18	244.9922	262.054

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: (<https://sipi.usc.edu/database/database.php>).

Author contributions

PH: Conceptualization, Data curation, Investigation, Methodology, Writing – original draft. YW: Conceptualization, Data curation, Formal analysis, Methodology, Writing – original draft. ZS: Conceptualization, Formal analysis, Funding acquisition, Resources, Supervision, Writing – review & editing. PZ: Resources, Validation, Writing – review & editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work is

supported by the Dalian Outstanding Young Science and Technology Talent Support Program (No. 2022RJ08), the Fundamental Research Funds for the Central Universities under grant (No. DUT23YG122).

Conflict of interest

YW was employed at DHC IT Company.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Ahmadian, A. M., and Amirmazlaghani, M. (2019). A novel secret image sharing with steganography scheme utilizing optimal asymmetric encryption padding and information dispersal algorithms. *Signal Process. Image Commun.* 74, 78–88. doi: 10.1016/j.image.2019.01.006
- Andono, P. N., and Setiadi, D. R. (2022). Improved pixel and bit confusion-diffusion based on mixed chaos and hash operation for image encryption. *IEEE Access* 10, 115143–115156. doi: 10.1109/access.2022.3218886
- Asgari-Chenaghlu, M., Balafar, M.-A., and Feizi-Derakhshi, M.-R. (2019). A novel image encryption algorithm based on polynomial combination of chaotic maps and dynamic function generation. *Signal Process.* 157, 1–13. doi: 10.1016/j.sigpro.2018.11.010
- Chai, X., Fu, X., Gan, Z., Lu, Y., and Chen, Y. (2019). A color image cryptosystem based on dynamic DNA encryption and Chaos. *Signal Process.* 155, 44–62. doi: 10.1016/j.sigpro.2018.09.029
- Chai, X., Fu, J., Gan, Z., Lu, Y., and Zhang, Y. (2022). An image encryption scheme based on multi-objective optimization and block compressed sensing. *Nonlinear Dyn.* 108, 2671–2704. doi: 10.1007/s11071-022-07328-3
- Chai, X., Fu, X., Gan, Z., Zhang, Y., Lu, Y., and Chen, Y. (2018). An efficient chaos-based image compression and encryption scheme using block compressive sensing and elementary cellular automata. *Neural Comput. & Applic.* 32, 4961–4988. doi: 10.1007/s00521-018-3913-3
- Chai, X., Gan, Z., Yang, K., Chen, Y., and Liu, X. (2017). An image encryption algorithm based on the memristive hyperchaotic system, cellular automata and DNA sequence operations. *Signal Process. Image Commun.* 52, 6–19. doi: 10.1016/j.image.2016.12.007
- Gao, X., Mou, J., Xiong, L., Sha, Y., Yan, H., and Cao, Y. (2022). A fast and efficient multiple images encryption based on single-channel encryption and chaotic system. *Nonlinear Dyn.* 108, 613–636. doi: 10.1007/s11071-021-07192-7
- Hosny, K. M., Kamal, S. T., and Darwish, M. M. (2021). A color image encryption technique using block scrambling and Chaos. *Multimed. Tools Appl.* 81, 505–525. doi: 10.1007/s11042-021-11384-z
- Hosny, K. M., Kamal, S. T., and Darwish, M. M. (2022). Novel encryption for color images using fractional-order hyperchaotic system. *J. Ambient. Intell. Humaniz. Comput.* 13, 973–988. doi: 10.1007/s12652-021-03675-y
- Hu, G., and Li, B. (2021). A uniform chaotic system with extended parameter range for image encryption. *Nonlinear Dyn.* 103, 2819–2840. doi: 10.1007/s11071-021-06228-2
- Hua, Z., Jin, F., Xu, B., and Huang, H. (2018). 2d logistic-sine-coupling map for image encryption. *Signal Process.* 149, 148–161. doi: 10.1016/j.sigpro.2018.03.010
- Hua, Z., Zhu, Z., Chen, Y., and Li, Y. (2021). Color image encryption using orthogonal Latin squares and a new 2D chaotic system. *Nonlinear Dyn.* 104, 4505–4522. doi: 10.1007/s11071-021-06472-6
- Huang, L., Sun, Y., Xiang, J., and Wang, L. (2022). Image encryption based on a novel memristive chaotic system, grain-128A algorithm and dynamic pixel masking. *J. Syst. Eng. Electron.* 33, 534–550. doi: 10.23919/jsee.2022.000053
- Kaçar, S., Konyar, M. Z., and Çavuşoğlu, Ü. (2022). 4D chaotic system-based secure data hiding method to improve robustness and embedding capacity of videos. *J. Inf. Secur. Appl.* 71:103369. doi: 10.1016/j.jisa.2022.103369
- Kumar, M., Mohapatra, R. N., Agarwal, S., Sathish, G., and Raw, S. N. (2018). A new RGB image encryption using generalized Vigenère-type table over symmetric group associated with virtual planet domain. *Multimed. Tools Appl.* 78, 10227–10263. doi: 10.1007/s11042-018-6586-0
- Li, D., Li, J., di, X., and Li, B. (2022). Design of cross-plane colour image encryption based on a new 2D chaotic map and combination of ECIES framework. *Nonlinear Dyn.* 111, 2917–2942. doi: 10.1007/s11071-022-07949-8
- Li, H., Hu, Y., Shi, Z., Wang, B., and Zheng, P. (2022). An image encryption algorithm based on improved lifting-like structure and cross-plane zigzag transform. *IEEE Access* 10, 82305–82318. doi: 10.1109/access.2022.3194730
- Li, Q., Wang, X., Ma, B., Wang, X., Wang, C., Gao, S., et al. (2022). Concealed attack for robust watermarking based on generative model and perceptual loss. *IEEE Trans. Circuits Syst. Video Technol.* 32, 5695–5706. doi: 10.1109/TCSVT.2021.3138795
- Li, Y., You, X., Lu, J., and Lou, J. (2023). A joint image compression and encryption scheme based on a novel coupled map lattice system and DNA operations. *Front. Inf. Technol. Electron. Eng.* 24, 813–827. doi: 10.1631/fitee.2200645
- Liang, Z., Qin, Q., and Zhou, C. (2022). An image encryption algorithm based on Fibonacci Q-matrix and genetic algorithm. *Neural Comput. Applic.* 34, 19313–19341. doi: 10.1007/s00521-022-07493-x
- Liang, Q., and Zhu, C. (2023). A new one-dimensional chaotic map for image encryption scheme based on random DNA coding. *Opt. Laser Technol.* 160:109033. doi: 10.1016/j.optlastec.2022.109033
- Liu, X., Tong, X., Wang, Z., and Zhang, M. (2022). Uniform non-degeneracy discrete chaotic system and its application in image encryption. *Nonlinear Dyn.* 108, 653–682. doi: 10.1007/s11071-021-07198-1
- Liu, P., Wang, X., Su, Y., Liu, H., and Unar, S. (2023). Globally coupled private image encryption algorithm based on infinite interval spatiotemporal chaotic system. *IEEE Trans. Circuits Syst. I Regul. Pap.* 70, 2511–2522. doi: 10.1109/tcsi.2023.3250713
- Liu, D., Zhang, W., Yu, H., and Zhu, Z. L. (2018). An image encryption scheme using self-adaptive selective permutation and inter-intra-block feedback diffusion. *Signal Process.* 151, 130–143. doi: 10.1016/j.sigpro.2018.05.008
- Mansouri, A., and Wang, X. (2020). Image encryption using shuffled Arnold map and multiple values manipulations. *Vis. Comput.* 37, 189–200. doi: 10.1007/s00371-020-01791-y
- MATLAB (2021a). MATLAB version: 9.10.0 (R2021a), Natick, Massachusetts: The MathWorks Inc. <https://www.mathworks.com>

- Murillo-Escobar, M. A., Meranza-Castillón, M. O., López-Gutiérrez, R. M., and Cruz-Hernández, C. (2019). Suggested integral analysis for chaos-based image cryptosystems. *Entropy* 21:815. doi: 10.3390/e21080815
- Richman, J. S., and Moorman, J. R. (2000). Physiological time-series analysis using approximate entropy and sample entropy. *Am. J. Phys. Heart Circ. Phys.* 278, H2039–H2049. doi: 10.1152/ajpheart.2000.278.6.h2039
- Sarangi, P., and Pal, P. (2022). Measurement matrix design for sample-efficient binary compressed sensing. *IEEE Sig. Pro. Lett.* 29, 1307–1311. doi: 10.1109/lsp.2022.3179230
- Wang, X., Chen, X., Feng, S., and Liu, C. (2022). Color image encryption scheme combining cross-plane zigzag scrambling and pseudo-random combination RGB component diffusion. *Optik* 269:169933. doi: 10.1016/j.ijleo.2022.169933
- Wang, X., Chen, S., and Zhang, Y. (2021). A chaotic image encryption algorithm based on random dynamic mixing. *Opt. Laser Technol.* 138:106837. doi: 10.1016/j.optlastec.2020.106837
- Wang, X., and Liu, H. (2022). Cross-plane multi-image encryption using chaos and blurred pixels. *Chaos Solitons Fractals* 164:112586. doi: 10.1016/j.chaos.2022.112586
- Wang, Y., Shang, Y., Shao, Z., Zhang, Y., Coatrieux, G., Ding, H., et al. (2022). Multiple color image encryption based on cascaded quaternion gyrator transforms. *Signal Process. Image Commun.* 107:116793. doi: 10.1016/j.image.2022.116793
- Wang, X., and Su, Y. (2021). Image encryption based on compressed sensing and DNA encoding. *Signal Process. Image Commun.* 95:116246. doi: 10.1016/j.image.2021.116246
- Zhang, Y. Q., He, Y., Li, P., and Wang, X. Y. (2020). A new color image encryption scheme based on 2DNLCML system and genetic operations. *Opt. Lasers Eng.* 128:106040. doi: 10.1016/j.optlaseng.2020.106040
- Zhang, X., Su, Q., Yuan, Z., and Liu, D. (2020). An efficient blind color image watermarking algorithm in spatial domain combining discrete Fourier transform. *Optik* 219:165272. doi: 10.1016/j.ijleo.2020.165272
- Zheng, H., Li, G., Xu, W., Zhong, H., and Xu, X. (2023). A compressive sensing encryption scheme for dual color images based on discrete memristor map and Rubik's cube scramble. *Optik* 286:170991. doi: 10.1016/j.ijleo.2023.170991
- Zheng, W., Yan, L., Gou, C., and Wang, F. Y. (2022). An ACP-based parallel approach for color image encryption using redundant blocks. *IEEE Trans. Cybern.* 52, 13181–13196. doi: 10.1109/tcyb.2021.3105568
- Zhou, Y., Li, C., Li, W., Li, H., Feng, W., and Qian, K. (2021). Image encryption algorithm with circle index table scrambling and partition diffusion. *Nonlinear Dyn.* 103, 2043–2061. doi: 10.1007/s11071-021-06206-8



OPEN ACCESS

EDITED BY
Jussi Tohka,
University of Eastern Finland, Finland

REVIEWED BY
JiQian Zhang,
Anhui Normal University, China
Changming Wang,
Capital Medical University, China

*CORRESPONDENCE
Haifeng Li
✉ lihaifeng@hit.edu.cn

RECEIVED 28 February 2024

ACCEPTED 20 May 2024

PUBLISHED 19 June 2024

CITATION
Sun C, Xu C, Li H, Bo H, Ma L and Li H (2024)
A novel multi-feature fusion attention neural
network for the recognition of epileptic EEG
signals.
Front. Comput. Neurosci. 18:1393122.
doi: 10.3389/fncom.2024.1393122

COPYRIGHT
© 2024 Sun, Xu, Li, Bo, Ma and Li. This is an
open-access article distributed under the
terms of the [Creative Commons Attribution
License \(CC BY\)](#). The use, distribution or
reproduction in other forums is permitted,
provided the original author(s) and the
copyright owner(s) are credited and that the
original publication in this journal is cited, in
accordance with accepted academic
practice. No use, distribution or reproduction
is permitted which does not comply with
these terms.

A novel multi-feature fusion attention neural network for the recognition of epileptic EEG signals

Congshan Sun¹, Cong Xu¹, Hongwei Li¹, Hongjian Bo², Lin Ma¹
and Haifeng Li^{1,2*}

¹Faculty of Computing, Harbin Institute of Technology, Harbin, China, ²Shenzhen Academy of Aerospace Technology, Shenzhen, China

Epilepsy is a common chronic brain disorder. Detecting epilepsy by observing electroencephalography (EEG) is the main method neurologists use, but this method is time-consuming. EEG signals are non-stationary, nonlinear, and often highly noisy, so it remains challenging to recognize epileptic EEG signals more accurately and automatically. This paper proposes a novel classification system of epileptic EEG signals for single-channel EEG based on the attention network that integrates time-frequency and nonlinear dynamic features. The proposed system has three novel modules. The first module constructs the Hilbert spectrum (HS) with high time-frequency resolution into a two-channel parallel convolutional network. The time-frequency features are fully extracted by complementing the high-dimensional features of the two branches. The second module constructs a grayscale recurrence plot (GRP) that contains more nonlinear dynamic features than traditional RP, fed into the residual-connected convolution module for effective learning of nonlinear dynamic features. The third module is the feature fusion module based on a self-attention mechanism to assign optimal weights to different types of features and further enhance the information extraction capability of the system. Therefore, the system is named HG-SANet. The results of several classification tasks on the Bonn EEG database and the Bern-Barcelona EEG database show that the HG-SANet can effectively capture the contribution degree of the extracted features from different domains, significantly enhance the expression ability of the model, and improve the accuracy of the recognition of epileptic EEG signals. The HG-SANet can improve the diagnosis and treatment efficiency of epilepsy and has broad application prospects in the fields of brain disease diagnosis.

KEYWORDS

epilepsy, EEG, Hilbert spectrum, grayscale recurrence plot, self-attention mechanism

1 Introduction

Epilepsy is a kind of brain disease caused by the abnormal hypersynchronous firing of neurons in the brain, which poses a great threat to the life and health of patients (Acharya et al., 2013). Therefore, an accurate epilepsy diagnosis is of great clinical significance in reducing the harm caused by epileptic seizures to patients. Electroencephalography (EEG)

is the most commonly used and effective procedure for diagnosing epilepsy (Noachtar and Rémi, 2009). The diagnosis of epilepsy is a continuous and long-term process (Sazgar and Young, 2019; Jang and Lee, 2020). Moreover, the characteristic pattern of epileptic seizures varies greatly among different patients and even within the same patient (Ren et al., 2023). Therefore, the diagnosis of epilepsy and the pattern analysis of epileptic seizures are usually carried out by neurologists through the detailed analysis of a large number of EEG data by visual detection and manual annotation (Peng et al., 2022). Since EEG signals are nonlinear, non-stationary, highly noisy, and tend to be of long duration, manual judgment to analyze EEG signals is very time-consuming and subject to the subjective judgment of the clinician (Andrzejak et al., 2001; San-Segundo et al., 2019; Hamavar and Asl, 2021). Therefore, more efficient automated detection and analysis methods have received much attention recently. This work will explore automatic and accurate recognition techniques of epileptic EEG signals to assist neurologists in analyzing EEG signals, reduce the burden of neurologists, and improve the efficiency of epilepsy diagnosis and treatment.

For the classification methods of epileptic EEG signals, scholars mainly use statistical analysis-based methods, traditional machine learning and deep learning methods. Gao et al. (2018) propose a statistical analysis-based method to detect seizures. First, they compute joint time-domain features and use the auto-regressive (AR) linear model to model the data. Then, based on the non-parametric statistical test of random power martingale (RPM), the decision is made. Das et al. (2018) extracted time-domain and frequency-domain features of EEG signals based on variational mode decomposition (VMD) and then detected epileptic seizure events by thresholding. Chen et al. (2019) used various distance measurement methods, such as Bhattacharyya distance, to solve the feature similarity of the power spectrum features based on short-time Fourier transform (STFT) of EEG signals at different moments and then detected the EEG signals by null hypothesis test. The above method has the advantages of easy implementation and fast detection speed. Since EEG signals are non-stationary signals, they are easily disturbed by noise generated by brain activity, and the extracted features are easily statistically unstable, leading to inaccurate detection results. In addition, scholars have conducted a lot of research on the classification of epileptic EEG signals based on machine learning and deep learning. Wang et al. (2017) extracted time-domain, frequency-domain, and time-frequency-domain features of EEG signals based on wavelet transform (WT), extracted nonlinear features based on information theory, and then combined the two types of features for epileptic seizure detection by machine learning methods such as k-nearest neighbor classification (KNN) and support vector machine (SVM). Lu et al. (2021) extracted several nonlinear features, such as sample entropy and Higuchi's fractal dimension, and combined them with SVM for epileptic EEG classification. Then, they found that phase space reconstruction and Poincaré section can improve the recognition accuracy of epileptic EEG signals. Jang and Lee (2020) use the wavelet transform (WT) and phase space reconstruction (PSR) to extract features and then input features to the neural network with weighted fuzzy membership (NEWFM) to detect seizure. Sui et al. (2021) proposed a time-frequency hybrid network (TFHybridNet) based on STFT and a convolutional neural network (CNN) for epileptic focus localization. Varlı and Yılmaz (2023) propose a

combined deep learning model based on CNN and long short-term memory (LSTM) to detect seizures. This model uses continuous wavelet transform (CWT) and STFT methods to input the signal conversion time-frequency image to the CNN module and the raw EEG signal to the LSTM module. Compared with traditional machine learning models and statistical analysis-based methods, deep learning models have stronger learning ability and better performance. Current deep learning methods mainly focus on the construction of deep network structures. Combining the non-stationary and nonlinear inherent signal characteristics of EEG with deep learning technology to improve detection accuracy needs further research.

Empirical mode decomposition (EMD) is a non-stationary signal analysis method widely used in the study of epileptic EEG recognition (Mahjoub et al., 2020; Lu et al., 2023). EMD decomposes EEG signals into several linear combinations of intrinsic mode functions (IMF). However, due to the mode mixing problem in EMD, false components in the obtained IMF will adversely affect the EEG analysis. In our previous work, we proposed an improved EMD method named adaptively optimized masking empirical mode decomposition (AOMEMD) (Sun et al., 2024). AOMEMD can effectively alleviate the mode mixing problem of EMD so that the obtained IMFs can effectively capture the underlying physics of EEG. By applying the Hilbert transform (HT) to the IMFs, the Hilbert spectrum (HS) of the EEG can be constructed for high-resolution time-frequency representation of EEG signals. Compared with STFT and CWT methods, this method does not need to set the basis function in advance and has high adaptability and flexibility. Therefore, in this paper, time-frequency features of EEG are represented based on AOMEMD and HT.

The recurrence plot (RP) is a nonlinear time series analysis method that can reveal hidden dynamic characteristics in EEG signals in the form of images (Eckmann et al., 1987; Huang et al., 2023). The traditional RP is a binary symmetric square matrix, usually using the recurrence quantification analysis (RQA) method to extract the structural features of RP for classification recognition. Since the traditional RP cannot reflect detailed time series information, scholars have proposed various improved RP methods. Hatami et al. (2017) skipped the threshold segmentation step in the process of RP construction and combined the gray-level texture image of RP with CNN to classify the time series. Khosla et al. (2022) proposed an un-thresholded recurrence plot (URP) and used the fractal weighted local binary pattern (URP-FWLBP) method to extract the texture features to classify epileptic seizure types. Experiments show that the URP-FWLBP method is better than the traditional method based on RQA. Considering the nonlinear, dynamic, and complex EEG signal, this paper combines the time-frequency feature based on HT with the nonlinear and non-stationary features based on RP to classify epileptic EEG signals.

Therefore, in this paper, we propose a novel system combining nonlinear dynamic features of EEG and time-frequency features extracted by non-stationary time-frequency analysis methods with deep learning techniques to classify epileptic EEG signals automatically. The proposed system is based on a self-attention mechanism to fuse time-frequency features of the HS and nonlinear dynamic features of the grayscale recurrence plot (GRP) to detect epileptic EEG signals for single-channel EEG. So, we call the proposed system HG-SANet. Several classification tasks on the

TABLE 1 The details of five sets in the Bonn EEG time series.

Set	New name	Subjects	Conditions	Electrodes
A	EO	Healthy volunteers	Eyes open	Surface
B	EC	Healthy volunteers	Eyes closed	Surface
C	SOE	Epilepsy patients	Seizure-free interval from outside the epileptogenic zone	Intracranial
D	SFE	Epilepsy patients	Seizure-free interval from epileptogenic zone	Intracranial
E	ES	Epilepsy patients	Epileptic seizure	Intracranial

Bonn EEG database and the Bern-Barcelona EEG database verify the performance of the proposed system for the classification of epileptic EEG signals.

2 Materials and methods

In this section, the public dataset used in this paper is first introduced. Secondly, the proposed approach of seizure detection in EEG signals is elaborated. Finally, the experimental setup of this paper is introduced.

2.1 Dataset and data pre-processing

In this paper, two datasets are used. The first dataset is the Bonn EEG time series (Andrzejak et al., 2001). The dataset consists of five sets (denoted A, B, C, D, and E in the original reference) of single-channel EEG segments from healthy volunteers and epilepsy patients, with a signal sampling frequency of 173.61 Hz and a duration of 23.6 s per sample. In order to better distinguish the five subsets, the names of the five subsets are changed to A (denoted EO), B (denoted EC), C (denoted SOE), D (denoted SFE), and E (denoted ES). Each set has 100 recordings and is described in Table 1. Some samples are shown in Figure 1. All EEG signals are digitally band-pass filtered over a range of 0.53~40 Hz. We used all the samples in this database for experiments to verify the effectiveness of the proposed method in epilepsy detection. We split the data to expand the size of the dataset (Varlı and Yilmaz, 2023). The data is divided into a segment of 512 sample points; the distance between segments is 128 sample points, the last one sample points of the data are deleted, and the final data is divided into 29 segments.

The second dataset is the Bern-Barcelona EEG database (Schindler et al., 2012). The dataset consists of focal and non-focal EEG segments during seizure-free periods from five epilepsy patients, with a signal sampling frequency of 1,024 Hz and a duration of 20 s per sample. Each class has 3,750 samples. If the channel is in the epileptogenic region, its label is focal; otherwise, its label is non-focal. The database is preprocessed as follows: (1) Samples are down-sampled to 512 Hz; (2) All EEG signals are digitally band-pass filtered over a range of 0.5~150 Hz

using a fourth-order Butterworth filter and phase distortions are minimized using forward filtering and backward filtering (Schindler et al., 2012). We used all the samples in this database for experiments to verify the effectiveness of the proposed method in epileptic focus localization. Some samples are shown in Figure 2. According to the previous works (Fasil and Rajesh, 2019), the data is divided into a non-overlapping segment of 1,024 sample points to expand the size of the dataset, and the final data is divided into 10 segments.

All the EEG signals in two datasets are normalized by the following Equation 1 to keep all data at the same scale, helping to improve recognition performance.

$$\tilde{x} = \frac{x - \mu}{\sigma} \quad (1)$$

where x is the input signal, μ is the mean of the signal, and σ is the standard deviation of the signal.

2.2 The proposed framework

The overview of the system based on the proposed HG-SANet is shown in Figure 3. The HG-SANet consists of three modules: EEG time-frequency feature extraction module based on HS and two-channel parallel convolutional neural network (HS-PCNet), nonlinear dynamic feature extraction module based on GRP and residual networks (GRP-ResNet), and multi-domain feature fusion module based on self-attention mechanism (MF-SANet). Below, we first introduce the construction method of HS and GRP and then introduce the network structure of each module.

2.2.1 AOMEMD-based Hilbert spectrum

In this part, we use AOMEMD and HT to construct Hilbert spectrum. For a single-channel EEG signal $x(t)$, the AOMEMD is first used to decompose $x(t)$ into a finite number of IMFs and a residue. Therefore, $x(t)$ can be represented as Equation 2:

$$x(t) = \sum_{k=1}^{n_{imf}} c_k(t) + r(t) \quad (2)$$

where $c_k(t)$ ($k = 1, 2, \dots, n_{imf}$) is the k th IMF and $r(t)$ represents the residue. The frequency of the n_{imf} IMFs decreases from the first to the n_{imf} th in order. In this work, we use the AOMEMD without the optimization strategy, which can save computation time while maintaining performance (Sun et al., 2024). The AOMEMD obtains IMFs through the following sifting process and the details of EMD are referred to the work of Huang et al. (1998).

Step 1: Input the signal $x(t)$. Initialize $k = 1$ and $r_{k-1}(t) = x(t)$. The number of phases is n_p .

Step 2: Determine the amplitude \bar{a}_k and frequency \bar{f}_k of the k th group masking signal $v_k(t)$ with resulted IMFs by applying EMD to $r_{k-1}(t)$.

Step 3: Construct the k th group masking signal $v_{kj}(t) = \bar{a}_k \cos [2\pi \bar{f}_k t + 2\pi(j-1)/n_p]$, ($j = 1, 2, \dots, n_p$). Obtain the k th IMF $c_k(t) = [\sum_{j=1}^{n_p} \text{EMD}_1(r_{k-1}(t) + v_{kj}(t))]/n_p$, where $\text{EMD}_1(\cdot)$ represents to obtain the first IMF using EMD.

Step 4: Update $r_k(t) = r_{k-1}(t) - c_k(t)$ and $k = k+1$. If $r_{k-1}(t)$ fulfils termination criterion, $r(t) = r_{k-1}(t)$; otherwise, go to step 2 and execute the loop.

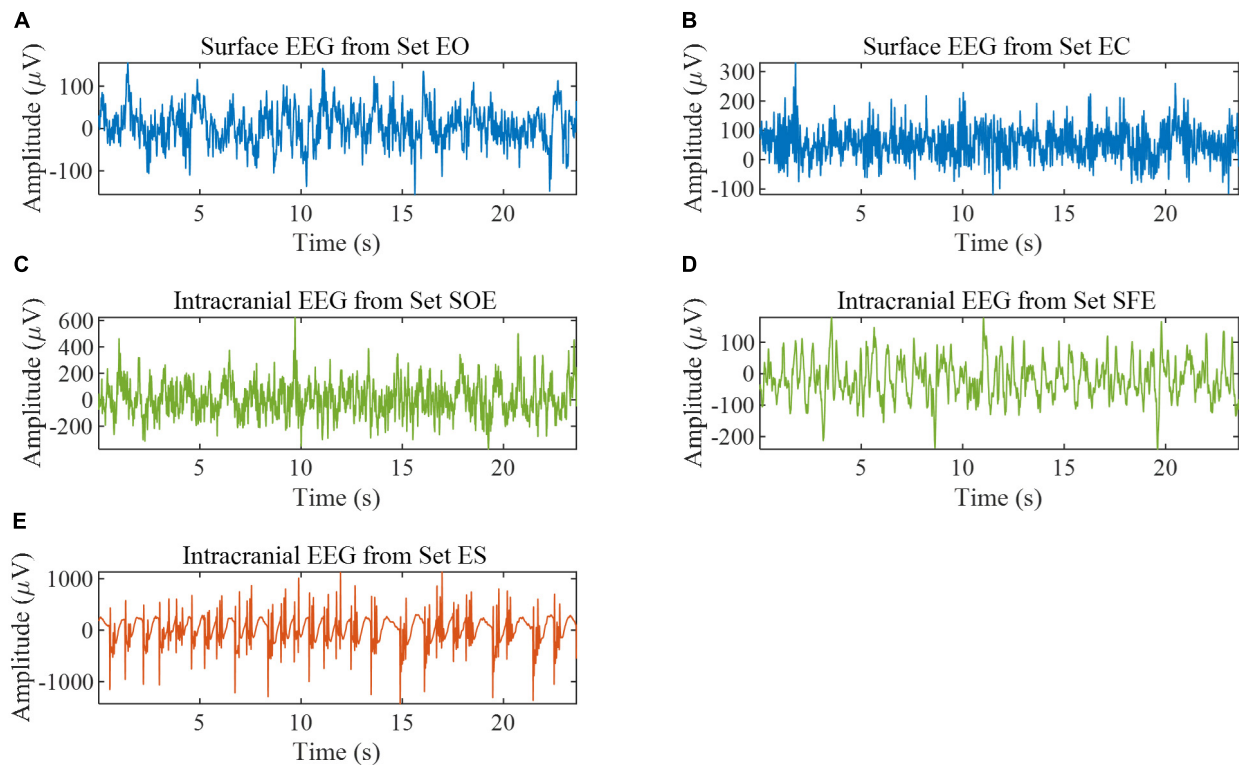


FIGURE 1

EEG samples from the Bonn EEG database. (A) Example of set EO. (B) Example of set EC. (C) Example of set SOE. (D) Example of set SFE. (E) Example of set ES.

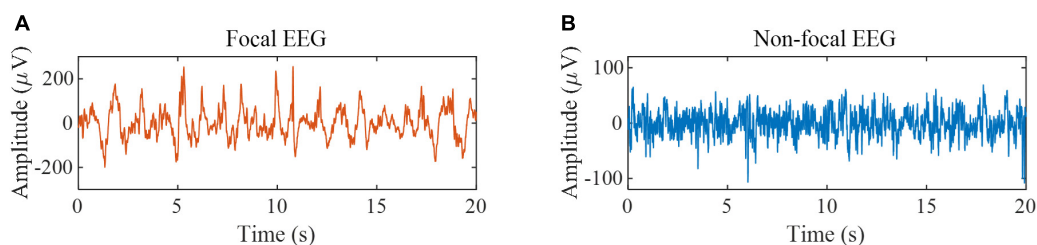


FIGURE 2

EEG samples from the Bern-Barcelona EEG database. (A) Example of focal EEG signals. (B) Example of non-focal EEG signals.

For the obtained $c_k(t)$ ($k = 1, 2, \dots, n_{inf}$) by AOMEMD, we use the HT to obtain the instantaneous frequency $f_k(t)$ and instantaneous amplitude $a_k(t)$ of $c_k(t)$. The formula for $y_k(t)$ obtained by applying the HT to $c_k(t)$ is shown in Equation 3 (Huang et al., 1998):

$$y_k(t) = \frac{1}{\pi} \text{p.v.} \int_{-\infty}^{+\infty} \frac{c_k(\tau)}{\tau - t} d\tau \quad (3)$$

where p.v. is the cauchy principal value. Then, $f_k(t)$ and $a_k(t)$ are solved as shown in Equations 4, 5:

$$f_k(t) = \frac{1}{2\pi} \cdot \frac{d}{dt} \left(\arctan \frac{y_k(t)}{c_k(t)} \right) \quad (4)$$

$$a_k(t) = \sqrt{c_k^2(t) + y_k^2(t)} \quad (5)$$

Then the amplitude distribution of $x(t)$ with frequency and time is the Hilbert spectrum (HS), denoted as $HS(f, t)$, expressed as follows Equation 6:

$$HS(f, t) = \text{Re} \left(\sum_{k=1}^{n_{inf}} a_k(t) e^{i \int_{-\infty}^t 2\pi f_k(\tau) d\tau} \right) \quad (6)$$

Where Re represents the real part and i is the imaginary unit. $HS(f, t)$ is a two-dimensional matrix with a time resolution equal to the sampling period (Molla and Hirose, 2007). Examples of HS are shown in Figure 4. As shown in Figure 4, the time-frequency distribution of samples from set EC and set ES is quite different.

2.2.2 Grayscale recurrence plot

For a single-channel EEG signal $x(t)$ of length T_{EEG} , the RP is computed as the following. First, according to Takens' embedding theory (Takens, 1985), a phase space is reconstructed for $x(t)$, and

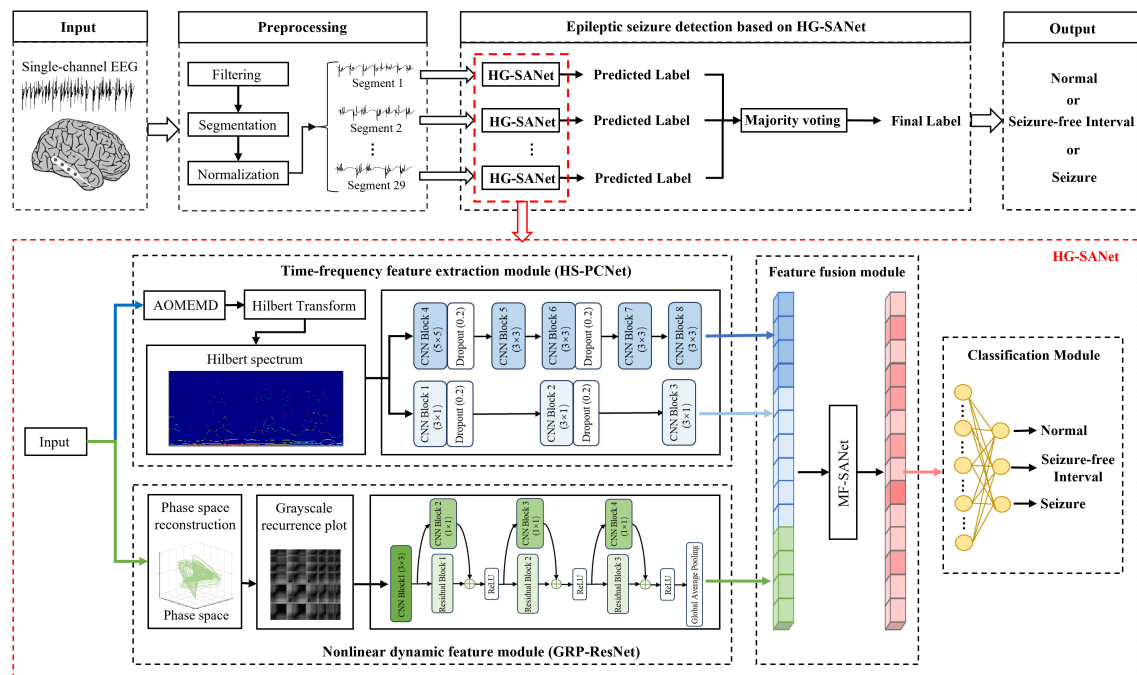


FIGURE 3

The overview of the proposed epileptic seizure detection system. The cortical model in the figure is from the literature (Andrzejak et al., 2001).

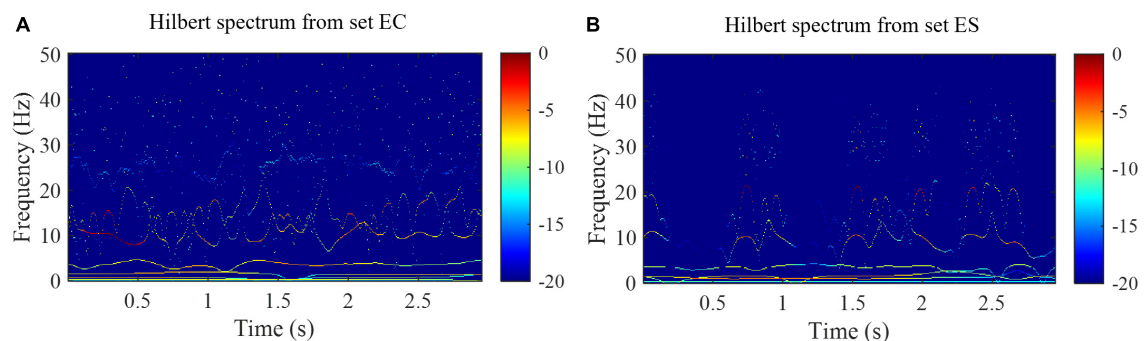


FIGURE 4

Hilbert spectra from the set EC and set ES of the Bonn EEG database. (A) Hilbert spectrum from the set EC. (B) Hilbert spectrum from the set ES.

a phase point in this space is denoted as s_n and $n = 1, 2, \dots, T_{EEG} - T_{ps}(m-1)$, where T_{ps} is the time delay and m is the embedding dimension. T_{ps} and m can be selected using mutual information (MI) and false nearest neighbor (FNN) methods, respectively (He et al., 2023). Second, the RP is defined according to Equation 7 below:

$$RP(n, j) = \begin{cases} 1, & \varepsilon \geq \|s_n - s_j\| \\ 0, & \varepsilon < \|s_n - s_j\| \end{cases}, \quad (7)$$

$$n, j = 1, 2, \dots, T_{EEG} - T_{ps}(m-1)$$

where ε is the distance threshold and $\|\cdot\|$ is the Euclidean norm. By assigning a black dot to the RP element (n, j) of $RP(n, j) = 1$ and a white dot to the RP element (n, j) of $RP(n, j) = 0$, a binary square image of an RP can be obtained, as shown in Figures 5A,C.

Binary square images constructed using the threshold method lose a lot of information, so we convert the RP to a grayscale

intensity image (named grayscale RP, GRP). The examples of GRP are shown in Figures 5B,D. The GRP is defined according to Equation 8 below (Chen and Shi, 2019):

$$GRP(n, j) = \frac{\|s_n - s_j\| - \min(\|s_n - s_j\|)}{\max(\|s_n - s_j\|) - \min(\|s_n - s_j\|)}, \quad (8)$$

$$n, j = 1, 2, \dots, T_{EEG} - T_{ps}(m-1)$$

2.2.3 Network structure of the HG-SANet

In this section, the network structure in each module of the HG-SANet is described in detail.

The first HS-PCNet module inputs the HS built in section 2.2.1 into a parallel two-channel CNN network containing different convolutional kernels. CNN overcomes the limitation of insufficient feature extraction ability of machine learning methods through simultaneous shift calculation of convolutional kernel in

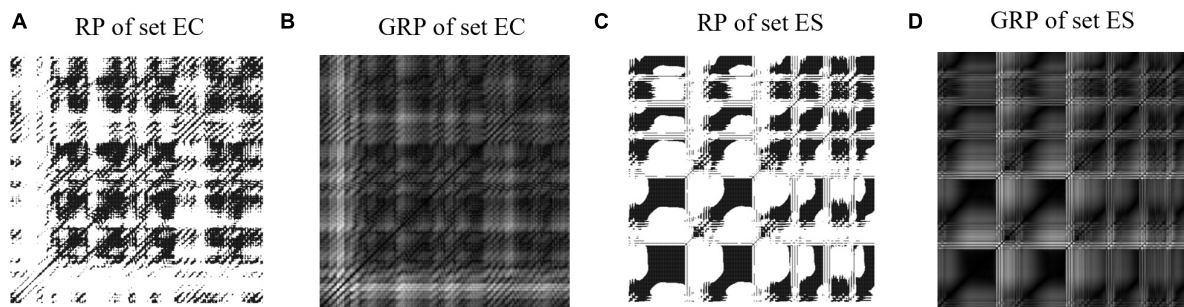


FIGURE 5

Examples of recurrence plots and grayscale recurrence plots from the set EC and set ES. (A) Recurrence plot from set EC. (B) Grayscale recurrence plot from set EC. (C) Recurrence plot from set ES. (D) Grayscale recurrence plot from set ES.

the time and frequency dimensions of feature maps (Zhang et al., 2015). It has been used in time-frequency feature extraction of EEG signals (Sui et al., 2021). Therefore, in this section, we use CNN to further extract the high-level time-frequency features of HS. For HS, we design a parallel two-channel CNN network containing different types of convolutional kernels for feature extraction. Two types of convolution kernels are set as $[N_{kernel}, 1]$ and $[N_{kernel}, N_{kernel}]$. As EEG signals comprise time-series data, we construct a convolution kernel of size $[N_{kernel}, 1]$ to make feature extraction pay more attention to changes in the time domain. The convolution kernel of size $[N_{kernel}, N_{kernel}]$ slides synchronously in the time domain and frequency domain dimensions of the HS to retain its original time-frequency characteristics. The time-frequency features are fully extracted by complementing the high-dimensional features of the two branches. The structure and details of the HS-PCNet module are shown in Figure 6A. The structure and parameter settings in each CNN block of the HS-PCNet module are shown in Table 2. In Table 2, the serial number corresponds to the serial number in Figure 6A. Each CNN block has a batch normalization layer and a ReLU activation layer between the 2D convolution (Conv 2D) and max pooling layers, which are omitted to save space. For the HS-PCNet module, the batch normalization layer normalizes the input data in small batches to speed up the training of the HS-PCNet and reduce the sensitivity to the network initialization. The max pooling layer performs downsampling by dividing the feature map into rectangular pooling regions and calculating the maximum value for each region, which helps reduce overfitting. The dropout layer makes the activation value of a certain neuron stop working with a certain probability, helping to prevent the HS-PCNet from overfitting (Krizhevsky et al., 2017).

A large number of studies have proved the advantage of residual networks in the field of image recognition (He et al., 2015). Therefore, the second GRP-ResNet module inputs the GRP in section 2.2.2 into a CNN with residual connections to fully learn the nonlinear dynamic features in the GRP. The convolutional module in the GRP-ResNet can use the receptive field of neurons to extract high-level local feature representation of the GRP, and the residual module allows cross-layer propagation, which can avoid overfitting caused by too many layers in the network, and will not lose important information in the feature (He et al., 2015). The overall structure of the GRP-ResNet module is shown in Figure 6B. The structure and parameter settings in each residual block and CNN block are shown in Table 3, and the serial number corresponds

to the serial number in Figure 6B. In each block, there is a batch normalization layer after the 2D convolution (Conv 2D) layers, which is omitted to save space.

Research shows that the self-attention mechanism (Vaswani et al., 2017) can help the network select important features and assign higher weights to these important features to improve the performance of downstream tasks (Lin et al., 2022; Yang Q. et al., 2023). Therefore, in the third MF-SANet, a feature fusion module based on a multi-head self-attention mechanism is proposed to assign optimal weights to different types of features obtained by the HS-PCNet module and GRP-ResNet module to enhance the information extraction capability of HG-SANet further. The feature fusion formulas are calculated as follows. First, the features extracted from the HS-PCNet module and GRP-ResNet module are concatenated and the concatenated features are denoted as *Feature_initial*. In the self-attention mechanism, there are three kinds of important input queries, keys and values, denoted as *QUE*, *KEY*, and *VAL*, respectively. They are calculated as Equations 9–11 (Lin et al., 2022):

$$QUE_j = \text{Feature_initial} \times W_j^{QUE} \quad (9)$$

$$KEY_j = \text{Feature_initial} \times W_j^{KEY} \quad (10)$$

$$VAL_j = \text{Feature_initial} \times W_j^{VAL} \quad (11)$$

where $j = 1, 2, \dots, N_{head}$ and N_{head} is the number of attention heads. W_j^{QUE} , W_j^{KEY} , and W_j^{VAL} are the parameter matrices. Then, the features of the final output are calculated as Equation 12:

$$\text{Feature}_{final} = \text{Concat}(\text{HEAD}_1, \text{HEAD}_2, \dots, \text{HEAD}_{N_{head}}) W^o \quad (12)$$

where W^o is a parameter matrix and $\text{Concat}(\cdot)$ is the concatenating operation. The HEAD_j is calculated as Equation 13

$$\text{HEAD}_j = \text{softmax}\left(\frac{QUE_j KEY_j^T}{\sqrt{d_{KEY}}}\right) VAL_j \quad (13)$$

where d_{KEY} is the dimension of keys.

In the classification layer based on the full connection layer, the activation function after the full connection layer is the softmax function.

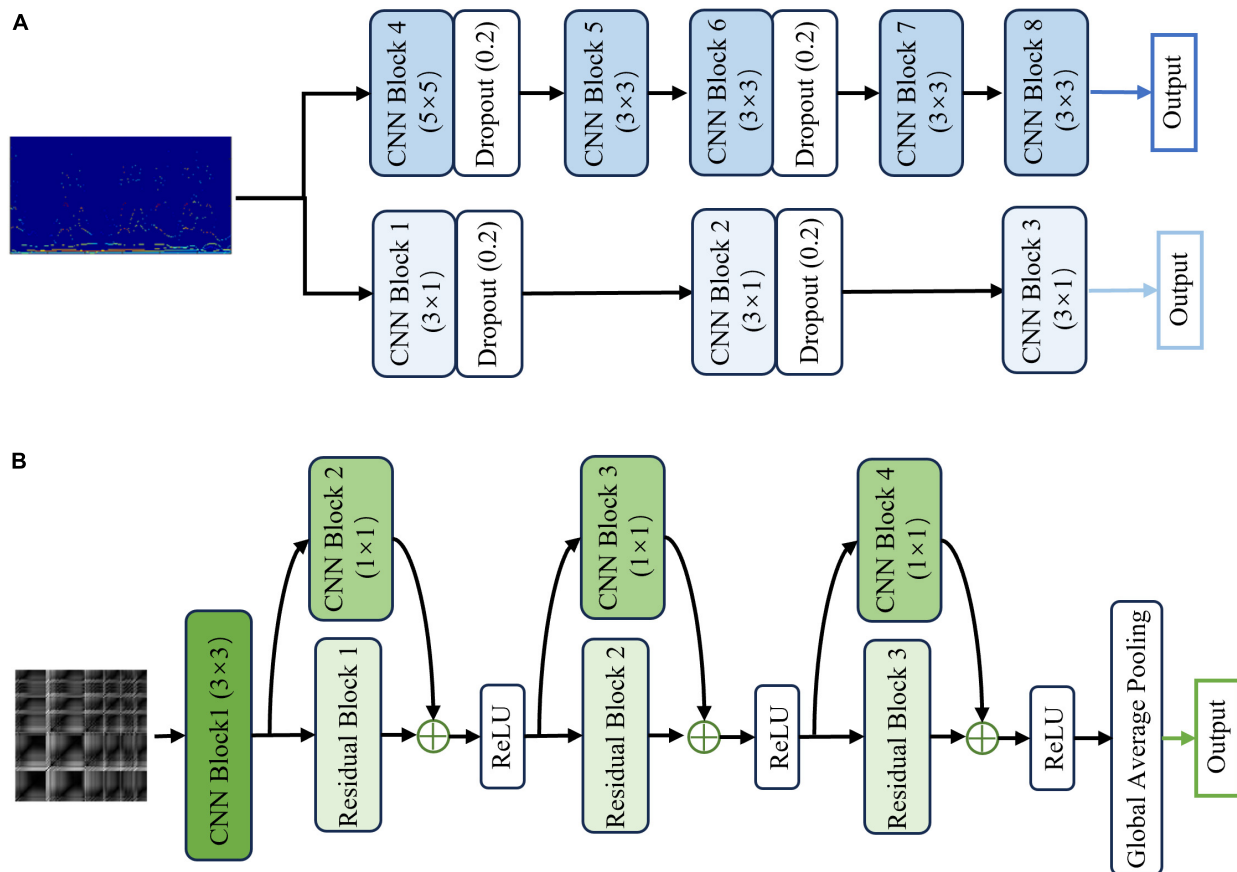


FIGURE 6

The structure and details of the HS-PCNet module and GRP-ResNet module. (A) The overall structure of the HS-PCNet module. (B) The overall structure of the GRP-ResNet module.

2.3 Experiment configurations

2.3.1 Evaluation metrics

In this paper, epileptic EEG recognition is evaluated using precision (P), recall (R), accuracy (Acc), and specificity (SP) (Sriram and Raghu, 2017; Gao et al., 2018). The sensitivity and recall are calculated using the same formula, so we no longer calculate sensitivity separately. Precision focuses on evaluating the percentage of true positive samples in all predicted positive samples. Recall focuses on the percentage of all positive samples that are successfully predicted to be positive. Accuracy is the proportion of correctly classified samples in total samples. The specificity is the proportion of all negative samples predicted correctly to all actual negative samples. These metrics are calculated as shown in Equations 14–17.

$$P = \frac{N_{TP}}{N_{TP} + N_{FP}} \quad (14)$$

$$R = \frac{N_{TP}}{N_{TP} + N_{FN}} \quad (15)$$

$$Acc = \frac{N_{TP} + N_{TN}}{N_{TP} + N_{TN} + N_{FN} + N_{FP}} \quad (16)$$

$$SP = \frac{N_{TN}}{N_{TN} + N_{FP}} \quad (17)$$

where N_{TP} is the number of true positive (TP) samples, N_{TN} is the number of true negative (TN) samples, N_{FP} is the number of false positive (FP) samples, and N_{FN} is the number of false negative (FN) samples.

In the decision stage, the HG-SANet gives prediction labels for all short segments of each sample. Finally, based on the prediction labels of short segments, the majority voting method is used to make the final prediction for the category of each test sample.

2.3.2 Model parameter setting

Parameters of the HG-SANet in the training process are set as follows. Adaptive moment estimation (Adam) optimizer is used to train the HG-SANet. The epoch used for training is 30, and the mini-batch size used for each training iteration is 32. The learning rate is 0.001. The cross-entropy loss function is used as the loss function. We reduce the overfitting of the HG-SANet by adding the regularization term of the weight to the loss function. The number of heads in the attention module is set to 2. In the testing process, the testing sample is input into the proposed system trained by the training set as shown in Figure 3 to obtain the final recognition result. The ten-fold cross-validation is used to obtain an unbiased evaluation of classification performance.

3 Results and discussion

3.1 Analysis of the proposed model

In this part, we designed several ablation experiments to analyze the effects of each module of the model. First, based on clinical applications and experiments conducted by scholars in the Bonn dataset (Ma et al., 2021), we selected three typical detection tasks to analyze our approach. The three typical tasks are: (1) Two-class detection task: distinguish between set EO and set ES, comparing the performance of methods to distinguish between healthy subjects and epileptic patients. (2) Two-class detection task: distinguish between set SOE and set ES, comparing the performance of methods to distinguish between non-epileptic interictal EEG and seizures in epileptic patients. (3) Three-class detection task: distinguish between normal (include set EO and EC), interictal activities (include set SOE and set SFE), and epileptic seizures (include set ES). This three-class task can be used not only to find epilepsy patients but also to automatically diagnose their symptoms, which is of great significance.

TABLE 2 The structure and parameter settings in each CNN block of HS-PCNet module.

Index	CNN block
1	Conv 2D: Size (3×1), Stride (1×1), Filters (8)
	Max Pooling: Size (3×1), Stride (2×2)
2	Conv 2D: Size (3×1), Stride (2×1), Filters (16)
	Max pooling: Size (3×1), Stride (2×1)
3	Conv 2D: Size (3×1), Stride (2×1), Filters (8)
	Max pooling: Size (3×1), Stride (2×2)
4	Conv 2D: Size (5×5), Stride (2×2), Filters (16)
	Max Pooling: Size (5×5), Stride (1×1)
5	Conv 2D: Size (3×3), Stride (1×1), Filters (32)
	Max pooling: Size (1×1), Stride (1×1)
6	Conv 2D: Size (3×3), Stride (1×1), Filters (64)
	Max pooling: Size (3×3), Stride (2×2)
7	Conv 2D: Size (3×3), Stride (1×1), Filters (32)
	Max pooling: Size (2×2), Stride (2×2)
8	Conv 2D: Size (3×3), Stride (1×1), Filters (16)
	Max pooling: Size (2×2), Stride (2×2)

TABLE 3 The structure and parameter settings in each residual block of GRP-ResNet module.

Name	Residual block 1	Residual block 2	Residual block 3	CNN block 1
Details	Conv 2D: Size (3×3), Stride (1×1), Filters (32)	Conv 2D: Size (3×3), Stride (2×2), Filters (64)	Conv 2D: Size (3×3), Stride (2×2), Filters (128)	Conv 2D: Size (3×3), Stride (2×2), Filters (16)
	ReLU	ReLU	ReLU	ReLU
	Conv 2D: Size (3×3), Stride (1×1), Filters (32)	Conv 2D: Size (3×3), Stride (1×1), Filters (64)	Conv 2D: Size (3×3), Stride (1×1), Filters (128)	Max pooling size (3×3), Stride (2×2)
Name	CNN block 2	CNN block 3	CNN block 4	–
Details	Conv 2D: Size (1×1), Stride (1×1), Filters (32)	Conv 2D: Size (1×1), Stride (2×2), Filters (64)	Conv 2D: Size (1×1), Stride (2×2), Filters (128)	–

In order to verify the performance of each module, we designed the following experiments: (1) Use the RQA method to extract the structural features of RP (Pham, 2020) and input these features into a SVM to classify three-class detection task (denotes as RQA-SVM). (2) A fully connected classification layer is added to the back of the GRP-ResNet module to classify the three-class detection task (denoted as GRP-ResNet). (3) A fully connected classification layer is added to the back of the HS-PCNet module to classify the three-class detection task (denoted as HS-PCNet). (4) The Hilbert Spectrum of the HS-PCNet module is replaced with a CWT-based scalogram (denoted as CWT-PCNet). Then, a fully connected classification layer is added to the back of the CWT-PCNet module to classify the three-class detection task. The Morlet wavelet is used as the mother wavelet (Varlı and Yilmaz, 2023). CWT is an important method for EEG signal analysis. We designed the fourth experiment to compare AOMEMD method and CWT method. (5) The features extracted from the HS-PCNet module and GRP-ResNet module are concatenated and the concatenated features are input to a fully connected classification layer to classify three-class detection task, denotes as HG-SANet without self-attention mechanism (HG-SANet-wo). (6) Use the HG-SANet to classify all three typical tasks. The classification results are shown in Table 4 and Figure 7.

Figure 7 shows the results of the ablation experiments designed in this section for the three-class detection task. The average results of the ten-fold cross-validation method are shown in Figure 7. As shown in Figure 7, each module can detect seizures, and the HG-SANet gives the best results in terms of overall performance. The best result of all the 10-fold cross-validation results is 100%. Combining the nonlinear features based on GRP-ResNet with the time-frequency features based on HS-PCNet improves the average accuracy, precision, and recall of the model. Moreover, the average accuracy, precision, and recall of the fusion model with added attention mechanism are increased by 0.8%, 0.67%, and 0.7%, respectively, compared with the fusion model without added attention mechanism. The results in Figure 7 demonstrate the validity of the proposed HG-SANet. The performance of RQA-SVM is the worst. The dimension of the RQA features is only eight. The information expression ability of RQA features is limited. The performance of CWT-PCNet is worse than HS-PCNet. For set ES, the recall of CWT-PCNet is the worst, only 86%. In Table 4, we compare the average performance of the proposed HG-SANet under different classification tasks. As shown in Table 4, in the two-class detection task of identifying set EO and set ES, our method achieves 100% recognition rate.

TABLE 4 Classification results of the proposed HG-SANet for the three typical tasks.

Cases	Class	P (%)	R (%)	Acc (%)	Mean P (%)	Mean R (%)
Set EO vs. Set ES	Set EO	100	100	100	100	100
	Set ES	100	100			
Set SOE vs. Set ES	Set SOE	99	100	99.50	99.50	99.55
	Set ES	100	99.09			
Set (EO, EC) vs. Set (SOE, SFE) vs. Set ES	Set (EO, EC)	98	98.54	98.20	98	98.56
	Set (SOE, SFE)	99	97.15			
	Set ES	97	100			

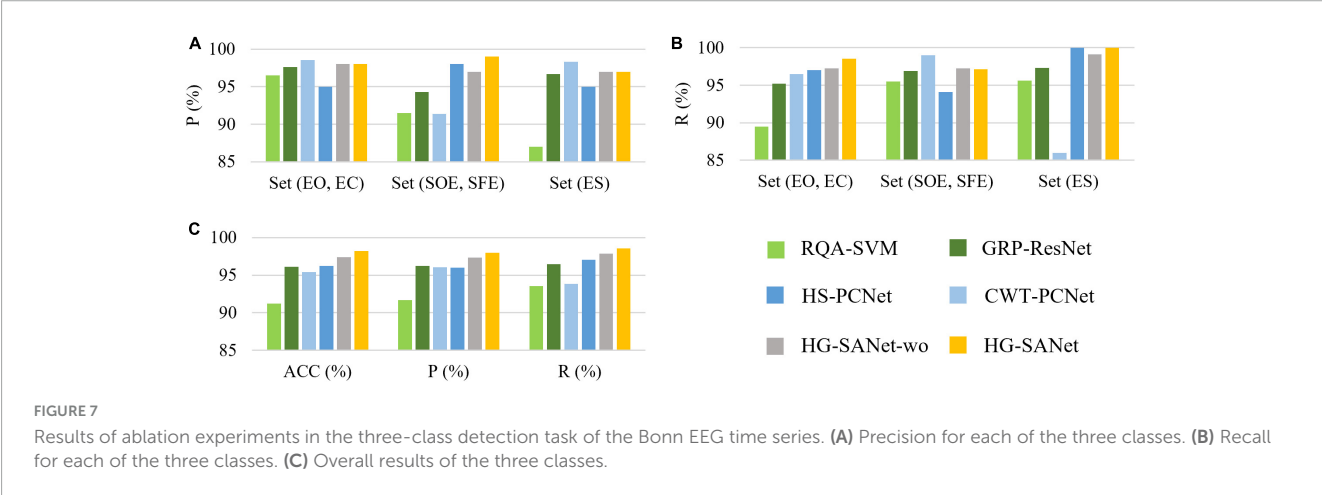


TABLE 5 Comparison of different methods on the Bonn EEG time series database.

Case	References	Methods	Acc (%)	P (%)	R (%)	SP (%)
Set SOE vs. Set ES	Zhao et al. (2020)	Raw EEG + CNN	98.02	–	–	–
	Zeng et al. (2019)	Entropy of visibility heights of hierarchical neighbors +LS-SVM	98.5	–	–	–
	Türk and Özerdem (2019)	CNN + Scalogram	98.5	–	98.01	98.98
	Peng et al. (2021)	Dictionary learning with homotopy	99	–	98	100
	Proposed	HG-SANet	99.50	99.50	99.55	99.50
Set EO vs. Set ES	Jang and Lee (2020)	Wavelet transform+ PSR+ neural network with weighted fuzzy membership	97.5	–	95	100
	Varlı and Yılmaz (2023)	2D CNN + CWT + LSTM	98.97	98.98	98.97	98.97
	Fu et al. (2015)	HHT+SVM	99.13	–	–	–
	Türk and Özerdem (2019)	CNN + Scalogram	99.5	–	99.0	100
	Zhao et al. (2020)	Raw EEG + CNN	99.52	–	–	–
	Proposed	HG-SANet	100	100	100	100
Set (EO, EC) vs. Set (SOE, SFE) vs. Set ES	Ullah et al. (2018)	Pyramidal one-dimensional CNN	96.27	97.00	95.00	98.00
	Khan et al. (2021)	Hilbert vibration decomposition +LSTM	96.00	95.77	95	–
	Zhao et al. (2020)	Raw EEG + CNN	96.97	–	–	–
	Varlı and Yılmaz (2023)	2D CNN + CWT + LSTM	97.3	97.31	97.30	98.35
	Proposed	HG-SANet	98.20	98	98.56	98.55

TABLE 6 Comparison of different methods on the Bern-Barcelona EEG database.

References	Methods	Acc (%)	P (%)	R (%)
Sharma et al. (2015)	Entropy +EMD + SVM	87.00	87.20	90.00
Fasil and Rajesh (2019)	Exponential energy features + SVM	89.00	–	–
Sriraam and Raghu (2017)	Multi-features + SVM	92.15	89.21	94.56
Gao et al. (2018)	Joint time-domain features + auto-regressive linear model + Randomized Power Martingale	–	93.75	93.75
Chen et al. (2019)	STFT + Bhattacharyya distance	–	88.68	94.00
Zhao et al. (2021)	Multi-feature Fusion + FCNN	93.44	94.28	92.50
Sui et al. (2021)	Time-Frequency Hybrid Network	94.30	94.30	94.30
Yang Y. et al. (2023)	Multi-level temporal-spectral features + FCNN	94.50	94.20	95.00
Proposed	HG-SANet	95.60	95.61	95.60

3.2 Comparison with SOTA methods for the classification of epileptic EEG signals

To further validate the effectiveness of the proposed method, we compare the proposed HG-SANet with other state-of-the-art (SOTA) methods on the Bonn EEG time series and the Bern-Barcelona EEG database. The results of the Bonn EEG time series are shown in Table 5. All the comparison methods include deep learning methods and traditional machine learning methods. The results of the proposed HG-SANet in Table 5 are the mean of the 10-cross validation results. As shown in Table 5, the proposed HG-SANet performs best on all the tasks. The proposed HG-SANet has a high recall value, which indicates that the method proposed in this paper can detect the seizure signal as much as possible, which is of great significance for diagnosing the disease. The proposed model can distinguish not only the EEG data of epileptic patients and non-epileptic persons but also the EEG data from epileptic seizures and seizure-free intervals in epileptic patients. When conducting comparative experiments, it was also found that deep learning-based methods outperformed other types of methods.

The results of the Bern-Barcelona EEG database are shown in Table 6. A binary classification task is performed on this database (focal vs. non-focal). The comparison methods include deep learning, traditional machine learning, and statistical modeling methods. The results of the proposed HG-SANet in Table 6 are the mean of the 10-cross validation results. As seen from Table 6, the performance of the proposed method in epileptic focal location is better than that of all the compared methods. It is also seen on the Bern-Barcelona EEG database that deep learning methods outperform other methods. The results of the two datasets show that the proposed method can classify multiple brain states associated with epilepsy. The proposed method can be used in automatic epileptic seizure detection, the epileptic focal location, and other related applications in diagnosing epilepsy diseases.

4 Conclusion

In this study, a novel model named HG-SANet is developed for the automated detection of epileptic EEG signals. This innovative

model proposes a multi-channel parallel feature extraction module based on multi-domain features and a feature fusion module based on an attention mechanism. Through many experiments, the proposed network structure can capture the non-stationary nonlinear properties of epilepsy EEG well and realize the automatic and high-accuracy detection of epileptic seizures, epileptic focus localization, and EEG classification. The method proposed in this paper is of great significance to detecting and warning brain disease. In the future, we will research other epilepsy-related issues, such as seizure prediction, and further reduce the time complexity of the method and make the method better applied to real-time seizure prediction.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Ethics statement

Ethical approval was not required for the study involving humans in accordance with the local legislation and institutional requirements. Written informed consent to participate in this study was not required from the participants or the participants' legal guardians/next of kin in accordance with the national legislation and the institutional requirements.

Author contributions

CS: Methodology, Writing – original draft, Writing – review & editing, Conceptualization, Data curation, Formal analysis, Validation, Visualization. CX: Writing – review & editing, Methodology, Formal analysis. HoL: Data curation, Writing – review & editing, Formal analysis. HB: Validation, Writing – review & editing. LM: Supervision, Writing – review & editing, Conceptualization. HaL: Supervision, Writing – review & editing, Conceptualization, Funding acquisition, Resources.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This study was supported by the National Natural Science Foundation of China (Grant No. U20A20383), Basic and Applied Basic Research of Guangdong (Grant No. 2021B1515120052), and Shenzhen Foundational Research Funding (Grant No. JCYJ20200109150814370).

Acknowledgments

The authors are grateful for the reviewers who made constructive comments.

References

- Acharya, U. R., Vinitha Sree, S., Swapna, G., Martis, R. J., and Suri, J. S. (2013). Automated EEG analysis of epilepsy: A review. *Knowledge Based Syst.* 45, 147–165. doi: 10.1016/j.knsys.2013.02.014
- Andrzejak, R. G., Lehnertz, K., Mormann, F., Rieke, C., David, P., and Elger, C. E. (2001). Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Phys. Rev. E* 64:61907. doi: 10.1103/PhysRevE.64.061907
- Chen, G. Y., Lu, G. L., Shang, W., and Xie, Z. H. (2019). Automated change-point detection of EEG signals based on structural time-series analysis. *IEEE Access* 7, 180168–180180. doi: 10.1109/ACCESS.2019.2956768
- Chen, W., and Shi, K. (2019). A deep learning framework for time series classification using relative position matrix and convolutional neural network. *Neurocomputing* 359, 384–394. doi: 10.1016/j.neucom.2019.06.032
- Das, P., Manikandan, M. S., and Ramkumar, B. (2018). “Detection of epileptic seizure event in EEG signals using variational mode decomposition and mode spectral entropy,” in *Proceedings of the IEEE 13th international conference on industrial and information systems (ICIIS)*, (Rupnagar), 42–47.
- Eckmann, J. P., Oliffson Kamphorst, S., and Ruelle, D. (1987). Recurrence plots of dynamical systems. *Europhys. Lett.* 4:973. doi: 10.1209/0295-5075/4/9/004
- Fasil, O. K., and Rajesh, R. (2019). Time-domain exponential energy for epileptic EEG signal classification. *Neurosci. Lett.* 694, 1–8. doi: 10.1016/j.neulet.2018.10.062
- Fu, K., Qu, J., Chai, Y., and Zou, T. (2015). Hilbert marginal spectrum analysis for automatic seizure detection in EEG signals. *Biomed. Signal Process. Control* 18, 179–185. doi: 10.1016/j.bspc.2015.01.002
- Gao, Z., Lu, G. L., Yan, P., Lyu, C., Li, X. Y., Shang, W., et al. (2018). Automatic change detection for real-time monitoring of EEG signals. *Front. Physiol.* 9:325. doi: 10.3389/fphys.2018.00325
- Hamavar, R., and Asl, B. M. (2021). Seizure onset detection based on detection of changes in brain activity quantified by evolutionary game theory model. *Comput. Meth. Programs Biomed.* 199:105899. doi: 10.1016/j.cmpb.2020.105899
- Hatami, N., Gavet, Y., and Debayle, J. (2017). “Classification of time-series images using deep convolutional neural networks,” in *Proceedings of the international conference on machine vision*, (Cham).
- He, K., Zhang, X., Ren, S., and Sun, J. (2015). “Deep residual learning for image recognition,” in *Proceedings of the 2016 IEEE conference on computer vision and pattern recognition (CVPR)*, (Las Vegas, NV), 770–778.
- He, Q., Yu, F., Chang, J., and Ouyang, C. (2023). Fuzzy granular recurrence plot and quantification analysis: A novel method for classification. *Pattern Recogn.* 139:109456. doi: 10.1016/j.patcog.2023.109456
- Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., et al. (1998). The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* 454, 903–995. doi: 10.1098/rspa.1998.0193
- Huang, W., Yan, G., Chang, W., Zhang, Y., and Yuan, Y. (2023). EEG-based classification combining Bayesian convolutional neural networks with recurrence plot for motor movement/imagery. *Pattern Recogn.* 144:109838. doi: 10.1016/j.patcog.2023.109838
- Jang, S., and Lee, S. (2020). Detection of epileptic seizures using wavelet transform, peak extraction and PSR from EEG signals. *Symmetry* 12:1239. doi: 10.3390/sym12081239
- Khan, P., Khan, Y., Kumar, S., Khan, M. S., and Gandomi, A. H. (2021). HVD-ISTM based recognition of epileptic seizures and normal human activity. *Comput. Biol. Med.* 136:104684. doi: 10.1016/j.combiomed.2021.104684
- Khosla, A., Khandnor, P., and Chand, T. (2022). EEG-based automatic multi-class classification of epileptic seizure types using recurrence plots. *Expert Syst.* 39:e12923. doi: 10.1111/exsy.12923
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Commun. ACM* 60, 84–90. doi: 10.1145/3065386
- Lin, S., Wang, Y., Zhang, L., Chu, Y., Liu, Y., Fang, Y., et al. (2022). MDF-SA-DDI: Predicting drug–drug interaction events based on multi-source drug fusion, multi-source feature fusion and transformer self-attention mechanism. *Brief. Bioinf.* 23:bbab421. doi: 10.1093/bib/bbab421
- Lu, X., Wang, T., Ye, M., Huang, S., Wang, M., and Zhang, J. (2023). Study on characteristic of epileptic multi-electroencephalograph base on Hilbert-Huang transform and brain network dynamics. *Front. Neurosci.* 17:1117340. doi: 10.3389/fnins.2023.1117340
- Lu, X., Zhang, J., Huang, S., Lu, J., Ye, M., and Wang, M. (2021). Detection and classification of epileptic EEG signals by the methods of nonlinear dynamics. *Chaos Solitons Fract.* 151:111032. doi: 10.1016/j.chaos.2021.111032
- Ma, M., Cheng, Y., Wei, X., Chen, Z., and Zhou, Y. (2021). Research on epileptic EEG recognition based on improved residual networks of 1-D CNN and indRNN. *BMC Med. Inform. Decis. Mak.* 21:100. doi: 10.1186/s12911-021-01438-5
- Mahjoub, C., Le Bouquin, Jeannès, R., Lajnef, T., and Kachouri, A. (2020). Epileptic seizure detection on EEG signals using machine learning techniques and advanced preprocessing methods. *Biomed. Tech.* 65, 33–50. doi: 10.1515/bmt-2019-0001
- Molla, M. K. I., and Hirose, K. (2007). Single-mixture audio source separation by subspace decomposition of Hilbert spectrum. *IEEE Trans. Audio Speech Lang. Process.* 15, 893–900. doi: 10.1109/TASL.2006.885254
- Noachtar, S., and Rémi, J. (2009). The role of EEG in epilepsy: A critical review. *Epilepsy Behav.* 15, 22–33. doi: 10.1016/j.yebeh.2009.02.035
- Peng, H., Li, C., Chao, J., Wang, T., Zhao, C., Huo, X., et al. (2021). A novel automatic classification detection for epileptic seizure based on dictionary learning and sparse representation. *Neurocomputing* 424, 179–192. doi: 10.1016/j.neucom.2019.12.010
- Peng, R., Jiang, J., Kuang, G., Du, H., Wu, D., and Shao, J. (2022). EEG-based automatic epilepsy detection: Review and outlook. *Acta Autom. Sin.* 48, 335–350. doi: 10.16383/j.aas.c200745
- Pham, T. D. (2020). “Recurrence plots,” in *Proceedings of the Fuzzy recurrence plots and networks with applications in biomedicine*, (Cham: Springer), 9–16. doi: 10.1007/978-3-030-37530-0_2
- Ren, Q. R., Sun, X. F., Fu, X. Q., Zhang, S. D., Yuan, Y. Y., Wu, H., et al. (2023). A review of automatic detection of epilepsy based on EEG signals. *J. Semicond.* 44:121401. doi: 10.1088/1674-4926/44/12/121401

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

- San-Segundo, R., Gil-Martín, M., D'Haro-Enríquez, L. F., and Pardo, J. M. (2019). Classification of epileptic EEG recordings using signal transforms and convolutional neural networks. *Comput. Biol. Med.* 109, 148–158. doi: 10.1016/j.compbiomed.2019.04.031
- Sazgar, M., and Young, M. G. (2019). "Seizures and epilepsy," in *Proceedings of the absolute epilepsy and EEG rotation review*, (Cham: Springer), 9–46. doi: 10.1007/978-3-030-03511-2_2
- Schindler, K., Rummel, C., and Andrzejak, R. G. (2012). Nonrandomness, nonlinear dependence, and nonstationarity of electroencephalographic recordings from epilepsy patients. *Phys. Rev. E* 86:46206. doi: 10.1103/PhysRevE.86.046206
- Sharma, R., Pachori, R. B., and Acharya, U. R. (2015). Application of entropy measures on intrinsic mode functions for the automated identification of focal electroencephalogram signals. *Entropy* 17, 669–691. doi: 10.3390/e17020669
- Sriraam, N., and Raghu, S. (2017). Classification of focal and non focal epileptic seizures using multi-features and SVM classifier. *J. Med. Syst.* 41:160. doi: 10.1007/s10916-017-0800-x
- Sui, L., Zhao, X., Zhao, Q., Tanaka, T., Cao, J., and Fornaro, M. (2021). Hybrid convolutional neural network for localization of epileptic focus based on iEEG. *Neural Plast.* 2021:6644365. doi: 10.1155/2021/6644365
- Sun, C., Li, H., Xu, C., Ma, L., and Li, H. (2024). Adaptively optimized masking EMD for separating intrinsic oscillatory modes of nonstationary signals. *IEEE Signal Process. Lett.* 31, 216–220. doi: 10.1109/LSP.2023.3347146
- Takens, F. (1985). On the numerical determination of the dimension of an attractor. *Lect. Notes Math.* 1125, 99–106. doi: 10.1007/BFb00756
- Türk, Ö., and Özerdem, M. S. (2019). Epilepsy detection by using scalogram based convolutional neural network from EEG signals. *Brain Sci.* 9:115. doi: 10.3390/brainsci9050115
- Ullah, I., Hussain, M., Qazi, E., and Aboalsamh, H. (2018). An automated system for epilepsy detection using EEG brain signals based on deep learning approach. *Expert Syst. Appl.* 107, 61–71. doi: 10.1016/j.eswa.2018.04.021
- Varli, M., and Yilmaz, H. (2023). Multiple classification of EEG signals and epileptic seizure diagnosis with combined deep learning. *J. Comput. Sci.* 67:101943. doi: 10.1016/j.jocs.2023.101943
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). "Attention is all you need," in *Proceedings of the 31st international conference on neural information processing systems (NIPS'17)*, (Red Hook, NY: Curran Associates Inc), 6000–6010.
- Wang, L. N., Xue, W. N., Li, Y., Luo, M. L., Huang, J., Cui, W. G., et al. (2017). Automatic epileptic seizure detection in EEG signals using multi-domain feature extraction and nonlinear analysis. *Entropy* 19:222. doi: 10.3390/e19060222
- Yang, Q., Tang, B., Shen, Y., and Li, Q. (2023). Self-attention parallel fusion network for wind turbine gearboxes fault diagnosis. *IEEE Sens. J.* 23, 23210–23220. doi: 10.1109/JSEN.2023.3308971
- Yang, Y., Li, F., Luo, J., Qin, X., and Huang, D. (2023). Epileptic focus localization using transfer learning on multi-modal EEG. *Front. Comput. Neurosci.* 17:1294770. doi: 10.3389/fncom.2023.1294770
- Zeng, M., Zhao, C., and Meng, Q. (2019). Detecting seizures from EEG signals using the entropy of visibility heights of hierarchical neighbors. *IEEE Access* 7, 7889–7896. doi: 10.1109/ACCESS.2019.2890895
- Zhang, H., Mcloughlin, I., and Song, Y. (2015). "Robust sound event recognition using convolutional neural networks," in *Proceedings of the 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, (South Brisbane, QLD), 559–563. doi: 10.1109/ICASSP.2015.7178031
- Zhao, W., Zhao, W., Wang, W., Jiang, X., Zhang, X., Peng, Y., et al. (2020). A novel deep neural network for robust detection of seizures using EEG signals. *Comput. Math. Method Med.* 2020:9689821. doi: 10.1155/2020/9689821
- Zhao, X., Solé-Casals, J., Zhao, Q., Cao, J., and Tanaka, T. (2021). "Multi-feature fusion for epileptic focus localization based on tensor representation," in *Proceedings of the 2021 Asia-Pacific signal and information processing association annual summit and conference (APSIPA ASC)*, (Tokyo), 1323–1327.



OPEN ACCESS

EDITED BY

Deepika Koundal,
University of Petroleum and Energy Studies,
India

REVIEWED BY

Vatsala Anand,
Chitkara University, India
Manoj Diwakar,
Graphic Era University, India

*CORRESPONDENCE

Mohammed Alarfaj
✉ mkalarfaj@kfu.edu.sa

RECEIVED 27 April 2024

ACCEPTED 17 June 2024

PUBLISHED 27 June 2024

CITATION

Alarfaj M, Al Madini A, Alsafran A, Farag M,
Chtourou S, Afifi A, Ahmad A, Al Rubayyi O,
Al Harbi A and Al Thunaiyan M (2024) Wearable
sensors based on artificial intelligence models
for human activity recognition.
Front. Artif. Intell. 7:1424190.
doi: 10.3389/frai.2024.1424190

COPYRIGHT

© 2024 Alarfaj, Al Madini, Alsafran, Farag,
Chtourou, Afifi, Ahmad, Al Rubayyi, Al Harbi
and Al Thunaiyan. This is an open-access
article distributed under the terms of the
[Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/)
(CC BY). The use, distribution or reproduction
in other forums is permitted, provided the
original author(s) and the copyright owner(s)
are credited and that the original publication
in this journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

Wearable sensors based on artificial intelligence models for human activity recognition

Mohammed Alarfaj^{1*}, Azzam Al Madini¹, Ahmed Alsafran¹,
Mohammed Farag¹, Slim Chtourou¹, Ahmed Afifi², Ayaz Ahmad¹,
Osama Al Rubayyi¹, Ali Al Harbi¹ and Mustafa Al Thunaiyan¹

¹Department of Electrical Engineering, College of Engineering, King Faisal University, Al-Ahsa, Saudi Arabia, ²Department of Computer Science, College of Computer Science and Information Technology, King Faisal University, Al-Ahsa, Saudi Arabia

Human motion detection technology holds significant potential in medicine, health care, and physical exercise. This study introduces a novel approach to human activity recognition (HAR) using convolutional neural networks (CNNs) designed for individual sensor types to enhance the accuracy and address the challenge of diverse data shapes from accelerometers, gyroscopes, and barometers. Specific CNN models are constructed for each sensor type, enabling them to capture the characteristics of their respective sensors. These adapted CNNs are designed to effectively process varying data shapes and sensor-specific characteristics to accurately classify a wide range of human activities. The late-fusion technique is employed to combine predictions from various models to obtain comprehensive estimates of human activity. The proposed CNN-based approach is compared to a standard support vector machine (SVM) classifier using the one-vs-rest methodology. The late-fusion CNN model showed significantly improved performance, with validation and final test accuracies of 99.35 and 94.83% compared to the conventional SVM classifier at 87.07 and 83.10%, respectively. These findings provide strong evidence that combining multiple sensors and a barometer and utilizing an additional filter algorithm greatly improves the accuracy of identifying different human movement patterns.

KEYWORDS

human body motion, inertial measurement unit, barometer, fall detection, machine learning, convolutional neural network, sensors, sensor networks

1 Introduction

The elderly demographic is rapidly expanding and is expected to accelerate significantly in the 21st century. This projection is based on an analysis conducted by the United Nations (UN) examining global population aging trends from 1950 to 2050. Based on the UN, the population of Saudi Arabia will increase to 40 million by 2050, with a quarter of this population (i.e., 10 million individuals) aged 60 years or older. The population's age distribution in Saudi Arabia during the period 1950–2050 is depicted in [Figure 1](#).

Cohorts aged 60–79 years and those aged above 80 years are currently experiencing particularly pronounced growth. In addition, there has been a consistent increase of approximately 5% in the number of individuals aged 60 years and over from 1950 to 2015, as illustrated in [Figure 2](#).

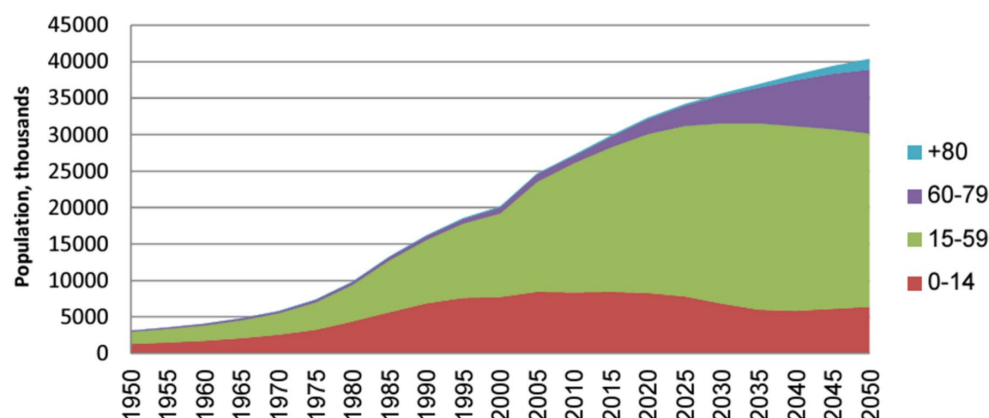


FIGURE 1
The Saudi Arabian population by age group is in the thousands (Abusaaq, 2015).

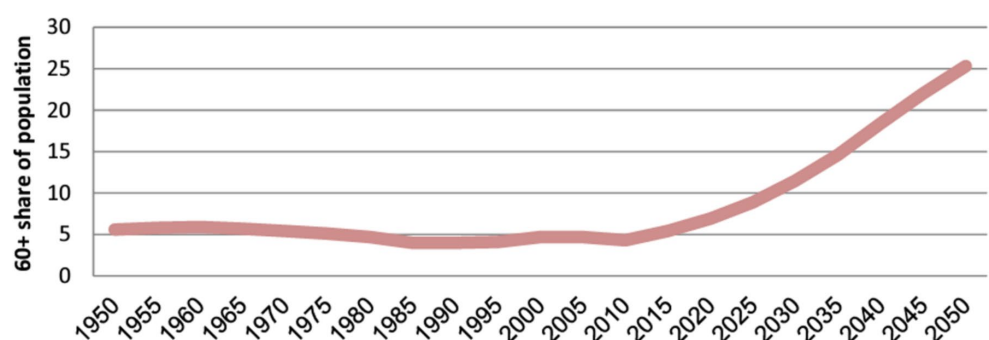


FIGURE 2
Share of the population aged over 60 years in Saudi Arabia (Abusaaq, 2015).

Human activity recognition (HAR), a research hotspot in academia and industry aiming to further ubiquitous computing and human–computer interactions, is utilized in healthcare, fitness, gaming, tactical military operations, and indoor navigation. Wearable sensors and external equipment (e.g., cameras and wireless RF modules) represent two basic HAR systems. In sensor-based HAR, sensors are worn on the body to capture segmented and precise sensor signal patterns (Alarfaj et al., 2021).

There are many proposed machine learning (ML) algorithms for HAR prediction, with the five main types of algorithms as follows: algorithms based on fuzzy logic (FL) (Medjahed et al., 2009; Schneider and Banerjee, 2021), algorithms based on probabilities (Maswadi et al., 2021; Schneider and Banerjee, 2021), algorithms based on rules (Hartmann et al., 2022; Radhika et al., 2022), algorithms based on distance (Agac et al., 2021; Fahad and Tahir, 2021), and optimization-based approaches (Muralidharan et al., 2021; Nguyen et al., 2021). The six actions recognized in HAR, including exercise, lying down, sitting, standing up, walking, and sleeping, are recognized by fuzzy rule-based inference systems using FL (Medjahed et al., 2009). Recently, a new method for HAR using first-person video and fuzzy rules for inference was reported (Schneider and Banerjee, 2021).

This study presents a novel methodology for enhancing HAR using sensor-specific convolutional neural networks (CNNs). Each CNN is designed to the unique data characteristics and

shape of a particular sensor type (accelerometer, gyroscope, or barometer), facilitating effective processing and accurate classification of a wide range of human activities. The methodology incorporates a late-fusion technique to integrate predictions from these diverse models, generating a comprehensive and accurate estimation of human activity. This approach addresses the limitations of single-model methods, using the strengths of individual sensor-specific CNNs for improved performance.

The novelty of this study lies in developing the sensor-specific CNN architecture, which enables the effective capture and utilization of distinctive features inherent to each sensor type, enhancing activity classification accuracy. This research overcomes the constraints of single-model approaches by implementing the late-fusion technique, which aggregates predictions from individual CNNs to comprehensively and accurately estimate human activity.

This study significantly contributes to the field of HAR by demonstrating the superior performance of the proposed late-fusion CNN model compared to the traditional support vector machine (SVM) classifier. This model's enhanced accuracy and robustness can revolutionize healthcare applications, enabling advanced monitoring, early detection of health issues, and personalized interventions for improved patient outcomes.

2 Study background

The increased utilization of wearable sensors has stimulated notable progress in HAR. Although early-fusion approaches have been prominent in industry, late-fusion methods are becoming more popular because of their potential for modularity, interpretability, and enhanced performance in specific situations. This section examines prominent late-fusion techniques for HAR and contrasts them with the CNN-based late-fusion method developed in this study. Many studies have investigated late-fusion methods for HAR, employing various sensor modalities and fusion algorithms. Hammerling and Rosipal (2013) used late fusion with support vector machines (SVMs) on accelerometer and gyroscope data for classification HAR, which resulted in good accuracy but limited interpretability. Yang et al. (2017) introduced a majority voting method for late fusion, which revealed promising outcomes but can have neglected intricate interconnections among different modalities. Wang et al. (2016) employed a layered generalization model to integrate data from an accelerometer, a gyroscope, and a barometer. Although this approach yielded better results than utilizing each model individually, more processing resources were required. Zhang et al. (2019) combined data from various modalities before inputting them into a deep neural network, resulting in high accuracy. However, this approach can have overlooked inter-modal relationships. Sun et al. (2018) introduced a hybrid method integrating early- and late-fusion techniques with deep learning (DL) models. This strategy demonstrated better results than fusion strategies; however, the fusion architecture must be meticulously designed. Yu et al. (2020) employed early fusion to extract features and late fusion for decision-making using a deep neural network. Although the model showed good accuracy and robustness, its complexity increased. The CNN-based late-fusion approach proposed in this study presents numerous advantages compared to previous research. Utilizing separate CNNs for each sensor modality enables customized extraction of features specific to each data type to capture more comprehensive and distinguishing information than generic techniques that fuse features at a higher level. The study utilizes a late-fusion technique where the predictions from separate CNN models for each sensor (accelerometer, gyroscope, and barometer) are combined at the decision level. Each CNN model processes its sensor input independently and generates predictions for human activity. The individual predictions are aggregated through a weighted average or voting mechanism to get the final prediction.

The widespread adoption and advancement of neural networks have led to the displacement of conventional methods by DL techniques in solving HAR problems. Many studies have employed CNNs to perform activity categorization tasks using sensor data (Moya Rueda et al., 2018; Demrozi et al., 2020; Mahmud et al., 2021; Sikder et al., 2021). In addition, Sikder et al. (2021) evaluated the effectiveness of one-dimensional and two-dimensional (2D) sequential CNN models for classifying HAR signals. The results indicated that 2D CNNs yield superior results and surpass traditionally created models. The DL models were developed to classify HAR tasks. Xu et al. (2019) introduced the InnoHAR model, which combines an inception neural network with a recurrent neural network. iSPLInception drew inspiration from Google's Inception-ResNet architecture and delivered superior predicted accuracy with reduced device resource requirements for signal-based HAR (Ronald et al., 2021). Hybrid models incorporating long short-term memory (LSTM)

and bi-directional LSTM have become increasingly popular in recent studies for human activity classification as they are adept at extracting spatial and temporal properties (Hayat et al., 2022; Khan et al., 2022; Li and Wang, 2022; Luwe et al., 2022). Zhao et al. (2022) used a hierarchical LSTM CNN to classify farmers' behavior in agriculture. Zhang et al. (2017) addressed gesture recognition by employing two types of neural networks: 3DCNN and ConvLSTM. In addition, many studies have used DL and ML to predict HAR (Almabdy and Elrefaei, 2019; Xu et al., 2019; Mutegeki and Han, 2020; Zheng et al., 2021).

3 Materials and methods

The primary objective of this study is to develop a continuous human movement monitoring system capable of acquiring user movement data and accurately and efficiently transmitting them to a remote server. A wearable device in the form of a bracelet is designed to serve three primary functions: monitoring human body movement, fall detection, and localization. In addition, the bracelet can measure heart rate, pulse oximetry, and body temperature. In addition, an alarm system is integrated to become activated in response to concerns regarding declines in the user's vital signs. This bracelet-type wearable device is selected for several compelling reasons: first, its accuracy remains unaffected by external factors such as weather, location, and time; second, the utilization of compact electronic components contributes to its low-power consumption; and third, it bears extensive adaptability, including minimal distance limitations, the capability to process and analyze substantial volumes of data, and user-friendly portability. Figure 3 displays the proposed framework of the HAR.

3.1 Hardware

As the detection of movement patterns can be enhanced by combining multiple sensors, this study employs five distinct sensors: an inertial measurement unit (IMU), a barometer, a human body temperature sensor, a pulse-oximeter sensor, and an active buzzer. Each sensor is assigned a specific role with the goal of increasing the accuracy and precision of pattern detection. All these sensors are interconnected with a single microcontroller. The hardware block diagram is depicted in Figure 4.

3.1.1 Arduino Nano RP2040 connect

The Arduino Nano RP2040 Connect device is designed to encapsulate the Raspberry PiRP2040 microcontroller in a compact nano-sized form. The device uses both Bluetooth® and WiFi connectivity and possesses an accelerometer and gyroscope. In addition, it incorporates artificial intelligence technologies. Figure 5 displays the Arduino Nano RP2040 type utilized to develop the proposed system.

3.1.2 IMU

IMUs are primarily employed in various devices for measuring velocity, orientation, and gravitational force. In its prior technological iteration, an IMU comprises two sensor types: accelerometers and gyroscopes. Accelerometers are utilized to quantify inertial acceleration, whereas gyroscopes measure angular rotation. Typically, both sensors provide three degrees of freedom to measure along three axes.

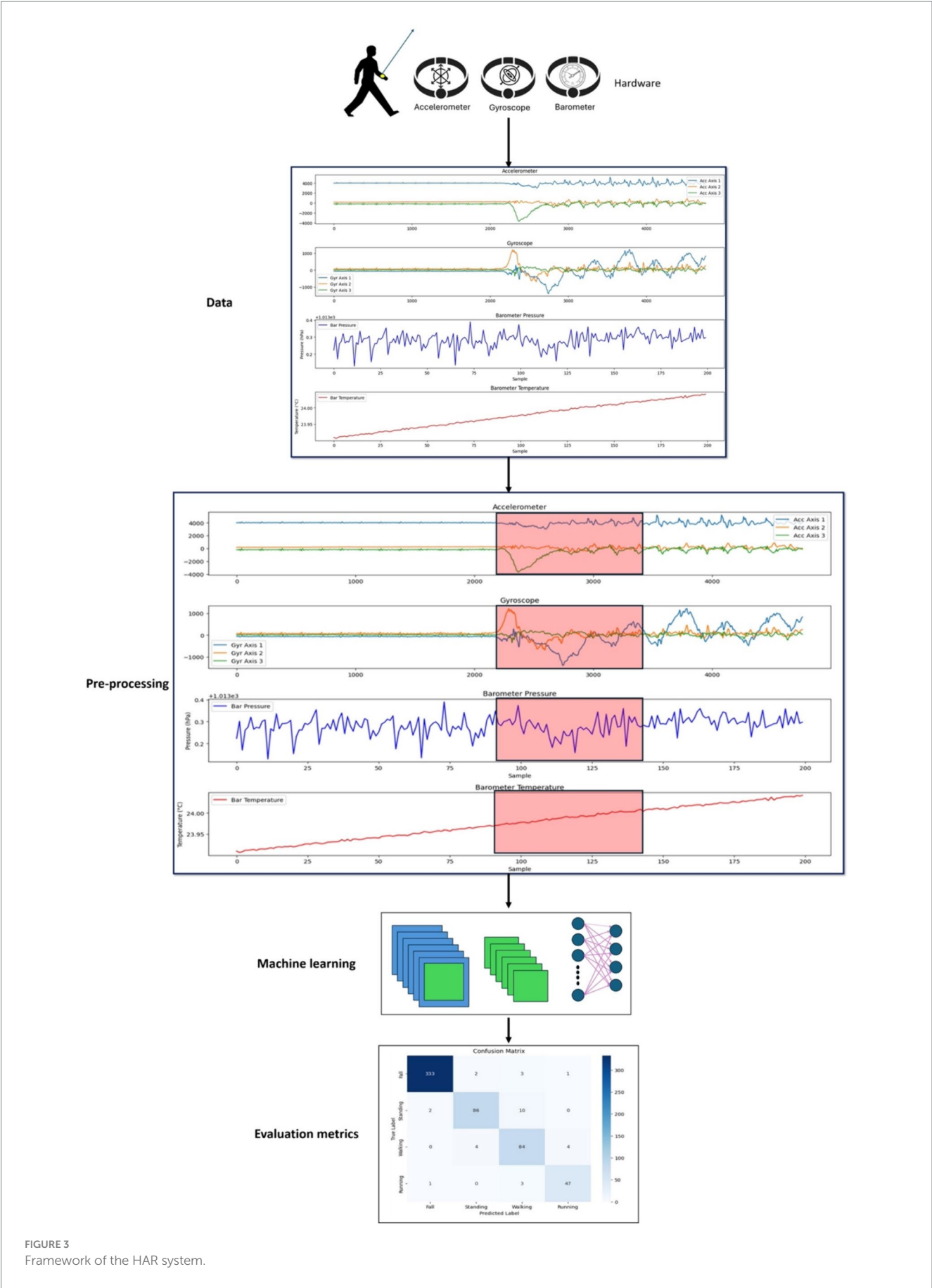


FIGURE 3
Framework of the HAR system.

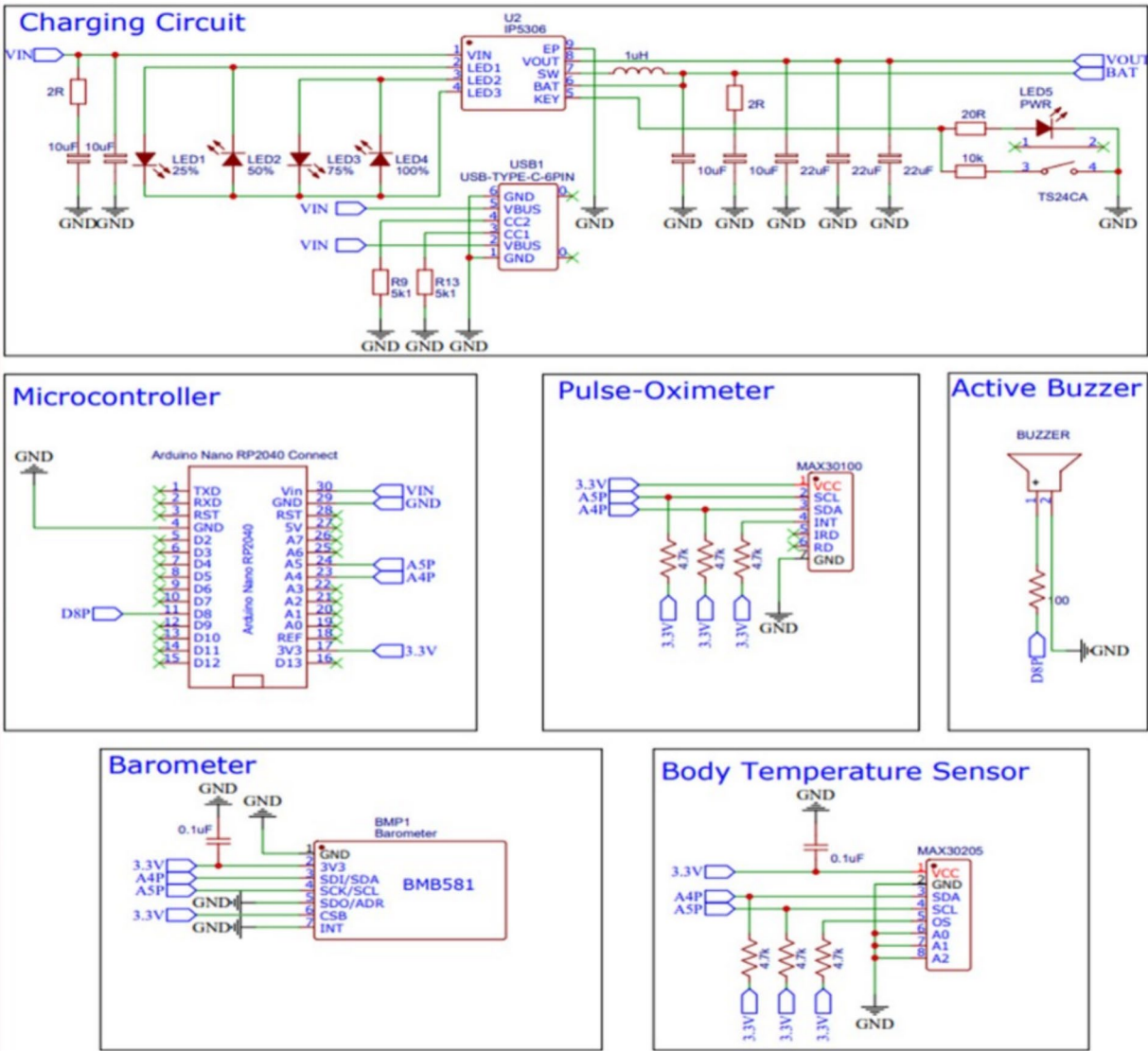


FIGURE 4
Hardware block diagram.

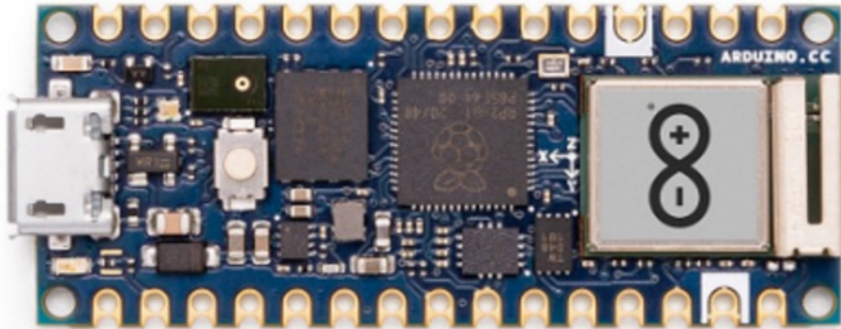


FIGURE 5
Arduino Nano RP2040.

Capacitive accelerometers, the most frequently used type, rely on changes in electrical capacitance to determine acceleration. When subjected to acceleration, the distance between the capacitor plates within the sensor changes as the diaphragm moves. Within the IMU, the gyroscope quantifies instantaneous angular velocity, typically expressed in units of degrees per second. The IMU device that is utilized in the framework of the proposed system is presented in Figure 6.

3.1.3 Barometer

Barometers are highly responsive devices employed to measure atmospheric pressure at a given location, in which the fluctuations in air pressure at varying altitudes are employed to determine the changes in elevation at specified points. The ability of an IMU to precisely assess changes in height is susceptible to the influence of weight. Therefore, using a barometer facilitates quantifying vertical displacement within the system. The barometer device used in the proposed system is depicted in Figure 7.

3.1.4 Temperature sensor (MAX30205) device

MAX30205 employs a negative temperature coefficient thermistor to measure the temperature by detecting variations in resistance in response to temperature fluctuations. This thermistor is placed in direct contact with the target object, typically the skin, and its resistance is measured by passing a small current through it and recording the resultant reduction in voltage. In addition, the sensor incorporates a digital filter and integrator to process the thermistor output, yielding a high-resolution digital representation of the measured temperature. The digital filter and integrator employ oversampling and noise-shaping techniques to enhance the precision and resolution of the temperature measurement. The MAX30205 sensor used to measure the temperature in the proposed system is depicted in Figure 8.



FIGURE 6
Inertial measurement unit.



FIGURE 7
Barometer device.

3.1.5 Oximeter pulse sensor device

The oximeter pulse sensor operates on photoplethysmography (PPG) principles, a volumetric measurement technique achieved through optical means. PPG quantifies oxygen volume by analyzing variations in light absorption within the body. The device aids in monitoring respiratory levels and various circulatory parameters in the blood. In addition, it enables the calculation of heart rate based on peaks detected in the signal (Yang et al., 2017). Figure 9 illustrates the oximeter pulse sensor used in the proposed system.

3.2 Datasets

3.2.1 FallAllID: movement pattern detection standard data

FallAllID constitutes a comprehensive open dataset that encompasses human falls and activities of daily living, as simulated by 15 participants (Saleh et al., 2021). The dataset comprises 26,420 files, collected via three data loggers worn on the users' waist, wrist, and neck. The motion signals were captured using an accelerometer (Acc), gyroscope (Gyr), and barometer (Bar); the magnetometer was excluded from this study. These sensors were efficiently configured to

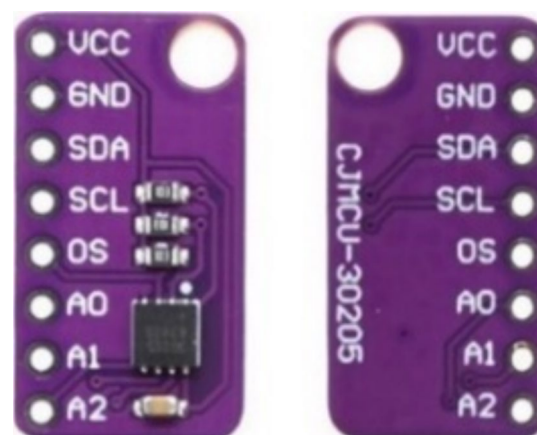


FIGURE 8
MAX30205 device.

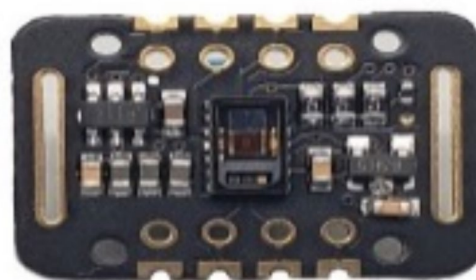


FIGURE 9
Sensor (MAX30102).

align with potential applications such as fall detection, prevention, and HAR (Saleh et al., 2021). Table 1 lists the features of the dataset.

3.2.2 Custom collected data: localization

A total of 260 samples were collected from the three sensor types: Acc, Gyr, and a received signal strength indicator (RSSI). Every sample included a sequence of sensor readings with corresponding timestamps. In addition, the collection consisted of 111 examples linked to specific locations, as localization depended on these labeled instances as a definitive data source. The features of the customized dataset collected from the proposed framework are listed in Table 2.

3.3 Preprocessing

This section comprehensively explains the feature engineering and preprocessing procedures employed in the current methodology, emphasizing the transformation of raw sensor data into meaningful and actionable features. Figure 10 displays the preprocessing approach for enhancing the proposed system. The data from the various sensors was processed to account for differing data shapes and sensor-specific characteristics:

- 1 **Data cleaning:** The raw sensor data was first cleaned by converting string representations of lists into actual lists using the `ast.literal_eval` function.
- 2 **Feature extraction:** Statistical features (mean, standard deviation, and range) were obtained from the accelerometer, gyroscope, and barometer data. This was done using separate functions for each sensor type:

- o `calculate_features`: Used for accelerometer and gyroscope data, which have X, Y, and Z axes.
- o `calculate_features_rssi`: Used for barometer data, which has pressure and temperature readings.

- 3 **Combined features:** The extracted features from all three sensors were then combined into a single feature array for each sample. This allowed the data to be input for the ML algorithms.

The preprocessing did differ slightly between sensor types due to the different data shapes and characteristics:

- **Accelerometer and gyroscope:** These sensors have three axes (X, Y, and Z), so the `calculate_features` function calculated the mean, standard deviation, and range for each axis.
- **Barometer:** This sensor has two readings (pressure and temperature), so the `calculate_features_rssi` function calculated the mean, standard deviation, and range for each reading.

However, the overall preprocessing approach was similar for all sensor types, involving data cleaning and feature extraction to prepare the data for analysis by the ML models.

3.3.1 Data cleaning

The `ast.literal_eval` function is employed to convert textual representations of lists in the “Acc,” “Gyr,” and “RSSI” columns back into actual lists. The `calculate_features` function is defined and implemented to obtain statistical features (mean, standard deviation, and range) from accelerometer and gyroscope data. In addition, the `calculate_features_rssi` function is defined and applied to extract the same statistical features from the RSSI data.

TABLE 1 Features of the standard FallAID dataset.

Feature name	Data type	Feature type	Description	Sensor
Acc X	Float	Numerical, Continuous	X-axis acceleration	Accelerometer
Acc Y	Float	Numerical, Continuous	Y-axis acceleration	Accelerometer
Acc Z	Float	Numerical, Continuous	Z-axis acceleration	Accelerometer
Gyr X	Float	Numerical, Continuous	X-axis rotational speed	Gyroscope
Gyr Y	Float	Numerical, Continuous	Y-axis rotational speed	Gyroscope
Gyr Z	Float	Numerical, Continuous	Z-axis rotational speed	Gyroscope
RSSI	Integer	Numerical, Continuous	Received signal strength indicator (RSSI) (Wi-Fi/Bluetooth signal power)	Wireless Communication

TABLE 2 Features of the customized FallAID dataset.

Feature name	Data type	Feature type	Description	Sensor
Acc X	Float	Numerical, Continuous	X-axis acceleration	Accelerometer
Acc Y	Float	Numerical, Continuous	Y-axis acceleration	Accelerometer
Acc Z	Float	Numerical, Continuous	Z-axis acceleration	Accelerometer
Gyr X	Float	Numerical, Continuous	X-axis rotational speed	Gyroscope
Gyr Y	Float	Numerical, Continuous	Y-axis rotational speed	Gyroscope
Gyr Z	Float	Numerical, Continuous	Z-axis rotational speed	Gyroscope
Bar Pressure	Float	Numerical, Continuous	Atmospheric pressure	Barometer
Bar Temperature	Float	Numerical, Continuous	Temperature	Barometer



FIGURE 10
Preprocessing steps.

3.3.2 Feature extraction approach

The `calculate_features` function is created, which derives statistical features from the sensor data, including the mean, standard deviation, and range for each axis (X, Y, and Z for Acc and Gyr; X and Y for Bar). This function is used on the preprocessed Acc, Gyr, and Bar data to derive their characteristics.

3.3.3 Combined features

The retrieved features from the Acc, Gyr, and Bar data are combined into one feature array to be used as an input for ML algorithms.

3.4 Classification algorithms

3.4.1 SVM

SVM is a widely used supervised learning method that can be applied to classification and regression tasks. Primarily, it is utilized for classification tasks in the field of ML. The objective of the SVM method is to establish an optimal line or decision boundary that can divide an n -dimensional space into various classes, enabling accurate categorization of incoming data points in the future. The optimal decision boundary is referred to as a hyperplane. SVM aims to identify a hyperplane with the largest margin, namely, the greatest distance between data points from different classes. Increasing the margin distance enhances the classification confidence of subsequent data points.

A standard SVM classifier using the one-vs-rest methodology was employed as a baseline for evaluating the performance of the proposed late-fusion CNN model in HAR. The SVM was selected as the baseline due to specific considerations.

- **SVM is a well-established** and commonly utilized ML technique for classification tasks, such as HAR. Its performance characteristics are widely recognized, making it a suitable benchmark for assessing new techniques.
- **SVM is easier** to develop and understand than more complex DL models such as CNNs. This facilitates comprehension of the factors contributing to performance disparities between the two strategies.
- **The one-vs-rest methodology** is a popular technique for modifying binary classifiers such as SVM for multiclass tasks like HAR. This enables a balanced comparison between SVM and the late-fusion CNN model, both intended for multiclass classification.

3.4.2 Random forest tree

Random forest tree (RFT) is an ML method utilized to address regression and classification tasks. It employs ensemble learning,

which integrates multiple classifiers to address intricate issues. The RFT algorithm comprises several decision trees. The “forest” created by the RF algorithm is trained using bagging or bootstrap aggregating. Bagging is an ensemble meta-algorithm that enhances the precision of ML methods. The RF algorithm builds an ensemble of decision trees, typically created via a method called “bagging” or “bootstrap aggregating.” This process involves creating numerous subsets of the original dataset (with the potential for duplication) and training a decision tree on each subset. Each tree in the forest is built using a bootstrap sample, where a sample is selected from the training set with replacement. In addition, when a node is divided during the tree construction, the selected split is no longer the most optimal among all the features. Instead, the selected split is the most efficient among a randomly selected subset of the attributes. Utilizing random subsets for training, encompassing both samples and characteristics, ensures that the trees within the forest are uncorrelated. By utilizing a forest model instead of individual decision trees, the resilience and accuracy of the model are improved.

3.4.3 K-nearest neighbors algorithm

The K-nearest neighbors (K-NN) method categorizes new cases by comparing their resemblance to existing cases and placing the former in the most similar category. The K-NN algorithm retains all the existing data and categorizes a new point by assessing its similarity. When fresh data are introduced, they can be efficiently categorized into a suitable group by utilizing the K-NN method. First, the value K is chosen for the neighbors. Then, the Euclidean distance of K neighbors is computed. The K-nearest neighbors are selected based on the computed Euclidean distance. K-NN functions by determining the data points in the training set closest to the new point requiring classification. The letter “K” in K-NN represents the nearest neighbors to consider. For example, when the value of K is set to 5, the algorithm looks for the five nearest neighbors of the new data point. Once the nearest neighbors are identified, the algorithm performs a majority vote for categorization purposes, allocating the new point to the class most frequently observed among its neighboring points. When performing regression tasks, it is feasible to determine the mean or median of the adjacent data points. The word “nearest” commonly refers to calculating the distances among locations utilizing metrics such as Euclidean, Manhattan, or Hamming distances.

3.4.4 CNNs

The studies were conducted utilizing two distinct datasets: the FallAIID dataset and a custom dataset. The FallAIID dataset, referred to as the standard dataset, was primarily used for HAR. When developing the CNN models for HAR, the raw sensor readings from the Acc, Gyr, and Bar were used directly without feature extraction. In contrast, when using the FallAIID dataset for SVM comparison, feature extraction was performed. The custom dataset was specifically

used for localization tasks, where feature extraction was also applied for traditional machine learning algorithms.

The adapted CNNs were created to capture the distinct characteristics of each sensor type:

Accelerometers measure acceleration to detect changes in speed and direction. The CNN model for accelerometers was developed to detect variations crucial for recognizing actions such as walking, running, and falling.

Gyroscopes measure angular velocity to detect rotational movements. The CNN model for gyroscopes was created to detect rotational movements, which is crucial for recognizing actions such as turning and twisting.

Barometers measure air pressure to detect variations in height. The CNN model for barometers was created to detect variations in height, which is crucial for recognizing actions such as ascending stairs or descending.

Each CNN model was specifically constructed to efficiently process the specific data shapes and characteristics associated with its respective sensors. The primary goal of these models was to accurately classify a broad spectrum of human activities. A window of 13 s instead of 20 s was selected for several reasons. An excessively long sliding window is at risk of encompassing extraneous behaviors, potentially confusing the classifier. In contrast, an extremely short window can fail to adequately capture all stages of falls. The suggested duration of 13 s is optimal, as it offers a reasonable timeframe for capturing all stages of falls and HAR activities (Zhang et al., 2019).

The design of the CNN models was impacted by these characteristics in multiple ways:

- **The input shape** of each CNN model was designed to correspond with the data shape of the specific sensor it was built for. The input shape of the accelerometer CNN model was (2,899, 260, 3), representing 2,899 samples of 260 time steps with three axes (x, y, and z).
- **The filter size** of each CNN model was selected to capture the pertinent properties of its corresponding sensor. The filter size of the accelerometer CNN model was selected to capture the brief alterations in acceleration typical of human motion.
- **The number of layers** in each CNN model was selected to strike a compromise between the model's complexity and its capacity to learn the pertinent information. The accelerometer CNN model featured fewer layers than the gyroscope CNN model due to the simpler nature of the accelerometer data.

The data initially obtained from the Acc and Gyr sensors exhibited a sampling frequency of 238 Hz. A deliberate decision was made to reduce the sampling frequency of these sensors to 20 Hz to enhance the ability of the system to detect temporal variations and facilitate a more detailed feature analysis. This adjustment extended the duration of each sensor measurement by an equivalent of approximately 13 s of recorded data. Hence, the dimensions of the Acc and Gyr data matrices transformed to (2,899, 260, 3), with each of the 260 samples representing a duration of 13 s at a sampling rate of 20 Hz. This modification of the CNNs facilitated their ability to analyze sequences of sensor data over a longer timeframe, yielding an enhanced capacity to detect nuanced activity patterns. The Bar sensor was designed to measure the barometric pressure and temperature, acquiring data at a sampling frequency of 10 Hz. Hence, each recorded sensor reading

corresponded to the selected time window of 13 s. This produced a modified data matrix with dimensions (2,899, 130, 2). In this case, the 130 samples represented a duration of 13 s at a frequency of 10 Hz, creating 130 samples. This adjustment enabled the CNNs to focus on variations in barometric pressure and temperature within the specified timeframe. This study ensured that the CNN models can efficiently process and extract significant information from the sensor readings by employing this method to modify the sensor data. This conversion was essential for data preprocessing, enhancing the effectiveness of the HAR system. Figure 11 presents the structure of the CNN model.

3.5 Late fusion technique

To capitalize on the strengths of different sensor modalities, namely the Acc, Gyr, and Bar, a late-fusion approach was employed in our CNN models. Each sensor type was assigned a dedicated CNN model specifically trained to capture the unique data characteristics pertinent to that sensor. The Acc model was designed to detect linear motion, the Gyr model focused on capturing rotational movements, and the Bar model aimed to identify changes in altitude. These models generated predictions in the form of class probabilities, reflecting the likelihood of each activity.

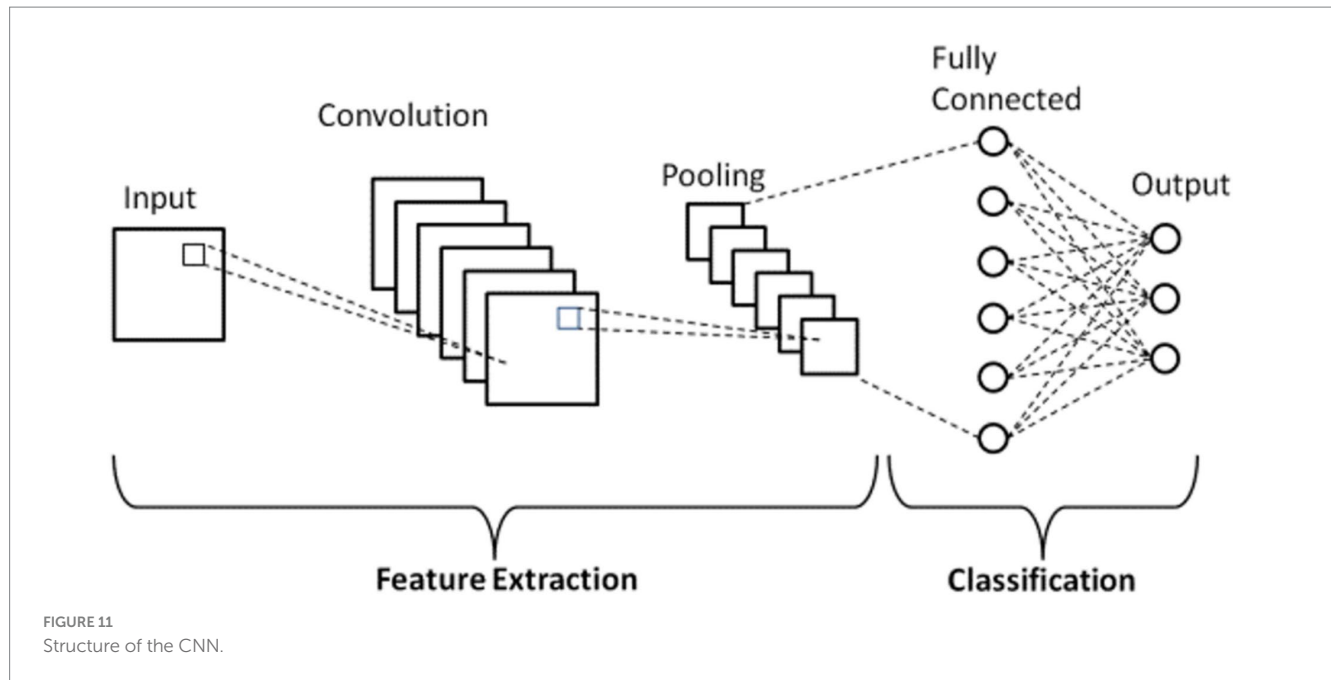
In the late-fusion approach, we combined these individual predictions using SVM. This process involved aggregating the class probabilities from each sensor-specific CNN model and feeding them into an SVM to form a final, comprehensive prediction. The SVM leveraged the strengths of each model's predictions, ensuring a robust and accurate classification.

This technique effectively preserved the unique features captured by each sensor, thereby enhancing the overall accuracy and robustness of the HAR system. By integrating predictions at the decision level with the SVM, the late-fusion method provided a more accurate and reliable estimation of human activities compared to single-model approaches. The late-fusion CNN and SVM model demonstrated superior performance in activity classification, thereby validating the efficacy of this multisensory integration strategy.

The choice to utilize late fusion was made due to its various benefits compared to other fusion techniques.

- 1 **Modularity:** Late fusion enables the separate development and optimization of each sensor-specific CNN model, promoting modularity. The system's modularity enhances its flexibility and adaptability to various sensor setups or data types.
- 2 **Interpretability:** Late fusion simplifies the assessment of each sensor's impact on the final prediction. This is beneficial for comprehending the significance of various sensor modalities for specific activities.
- 3 **Improved performance:** Late fusion can sometimes enhance performance compared to early fusion, where sensor data is merged before inputting into a single model. Late fusion enables each model to concentrate on extracting features from its unique sensor data, which can be more effective than attempting to learn features from mixed data with diverse properties.

The late-fusion technique was selected for its ability to capitalize on the advantages of several sensor modalities and merge their predictions



to create a more precise and resilient HAR system. The late-fusion CNN model's performance was assessed based on specific metrics and criteria:

- **Validation accuracy:** The precision of the model on a validation subset utilized to assess the model's performance during training.
- **Final test accuracy:** The precision of the model on a final test set, a subset of the dataset not utilized for training or validation, is employed to evaluate the model's performance on new data.
- **A classification report** is a detailed analysis of a model's performance, including precision, recall, and F1-score for each class (i.e., each type of human activity).
- **A confusion matrix** is a tabular representation that displays the counts of true positives, false positives, true negatives, and false negatives for each class.

The metrics and criteria were utilized to evaluate the model's efficacy in categorizing human actions precisely. The late-fusion CNN model demonstrates good validation and test accuracies and strong performance in the classification report and confusion matrix, indicating its effectiveness for HAR.

3.6 Evaluation metrics

Evaluation metrics are essential for evaluating the effectiveness of ML and DL models. Evaluation metrics also assist in choosing models and adjusting hyperparameters. As various jobs necessitate specific measures, using the appropriate metrics is crucial for accurately interpreting model outcomes. In this study, we employed the following evaluation metrics (Equations 1–4):

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \times 100\%. \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \times 100\%. \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \times 100\%. \quad (3)$$

$$Fscore = \frac{2 * precision * Sensitivity}{precision + Sensitivity} \times 100\%. \quad (4)$$

Where True Positive (TP) indicates a correct positive prediction; False Positive (FP) indicates an incorrect positive prediction; False Negative (FN) indicates an incorrect negative prediction; and True Negative (TN) indicates a correct negative prediction. These metrics provide a comprehensive understanding of the models' accuracy, precision, recall, and overall effectiveness in classifying human activities. By employing these metrics, the study ensured robust and reliable performance evaluation, highlighting the strengths and weaknesses of both deep learning and traditional machine learning approaches.

4 Experimental results

This section presents the proposed wearable system for human motion sensing technologies.

4.1 Experimental setup

Developing a wearable system for human motion sensing technologies requires substantial hardware and software. Tables 3, 4 present these hardware and software requirements, respectively.

TABLE 3 Hardware requirements.

Devices	Type
Arduino® Nano RP2040 Connect	Pull up Resistors (4.7kΩ)
Barometric pressure sensor (BMP581 Qwiic)	Active buzzer (LTE12-03)
Human body temperature sensor (MAX30205)	Battery (3.7 V)
Pulse sensor & oximeter pulse (MAX30102)	Circuit charger + boost voltage
PCB	3D Model

TABLE 4 Software requirements.

Library	Modules/functions
Pandas	"pd.read_excel," "pd.sample," "pd.reset_index," "pd.DataFrame.apply," "pd.concat"
Numpy	"np.mean," "np.std.," "np.max," "np.min," "np.array," "np.hstack"
Ast	"ast.literal_eval"
Sklearn.model_selection	"train_test_split"
Sklearn.ensemble	"RandomForestClassifier"
Sklearn.metrics	"classification_report," "accuracy_score"
Sklearn.neighbors	"KNeighborsClassifier"
Sklearn.preprocessing	`StandardScaler`
Sklearn.svm	"SVC"
Sklearn.multiclass	"OneVsRestClassifier"
Keras.models	"load_model"
Keras	"Sequential"
Google.colab	"drive.mount"

4.2 Splitting data

Splitting a dataset into two sections enables the assessment of the performance of ML and CNN models, aiding in model selection, hyperparameter tuning, and early halting decisions. Table 5 displays the standard and customized datasets for splitting.

4.3 Hyperparameter optimization for all sensors

Achieving optimal performance for CNN models across different sensor types—Acc, Gyr, and Bar—requires a comprehensive hyperparameter optimization strategy. This strategy involves tuning key parameters such as epochs, batch sizes, learning rates, dropout rates, and regularization techniques to enhance model accuracy and robustness.

4.3.1 Accelerometer model

For the accelerometer model, the optimization focused on epochs, batch sizes, and learning rates. Combinations of 10 and 20 epochs, batch sizes of 32 and 64, and learning rates of 0.0001 and 0.001 were evaluated. The optimal configuration, consisting of 20 epochs, a batch

TABLE 5 Splitting the datasets.

Split	Number of samples
Standard dataset	
Training 60%	1739
Validation 20%	579
Test 20%	581
Custom dataset	
Training 60%	66
Validation 20%	22
Test 20%	23

size of 64, and a learning rate of 0.001, resulted in a test accuracy of 86.20%. This configuration ensured sufficient training duration and stability while balancing convergence speed and precision.

4.3.2 Gyroscope model

The hyperparameter optimization for the gyroscope model incorporated L2 regularization and a dynamic learning rate schedule. The model utilized 20 epochs, a batch size of 16, and an initial learning rate of 0.0005. A learning rate scheduler was applied to halve the learning rate after 5 epochs, enhancing the model's fine-tuning capability during later training stages. Additionally, dropout layers with a rate of 0.5 were used to mitigate overfitting. This combination of regularization, dynamic learning rate adjustment, and early stopping produced a robust model capable of effectively interpreting gyroscope data.

4.3.3 Barometer model

The optimization process for the barometer model included a deeper network architecture with multiple convolutional and dense layers, each followed by Leaky ReLU activations and dropout regularization. An initial learning rate of 0.001, which decayed exponentially every 10,000 steps, allowed for gradual refinement of the learning process. The model was trained for up to 50 epochs with a batch size of 32, employing early stopping and model checkpointing to prevent overfitting and to save the best-performing model. This comprehensive approach ensured that the model effectively captured the nuances of barometric data, thereby enhancing its predictive accuracy.

4.4 Results of standard data

4.4.1 Results of the CNN model

The results obtained from the late-fusion CNN-based model exhibited remarkable utility, demonstrating its substantial potential to enhance both the accuracy and precision of HAR systems. Throughout the experiments, the late-fusion CNN-based model consistently delivered outstanding performance metrics. The validation accuracy was 98.35%, with a final test accuracy of 94.83%. For a more comprehensive evaluation of the model's performance, please refer to the classification report in Table 6 and the confusion matrix illustrated in Figure 12. The outcomes presented in this study provide compelling evidence of the effectiveness of the late-fusion CNN model in accurately

classifying human activities. The model achieved exceptional accuracy, recall, and F1 scores across various activity classes, emphasizing its ability to accurately distinguish various activities.

4.4.2 Results of the SVM model

Employing a traditional SVM classifier with a one-vs-rest approach led to lower performance metrics, as the validation set accuracy was 87.07%, and test accuracy was 83.10%. The 262 performance details of the SVM model are reported in the classification report in Table 7, and the confusion matrix is displayed in Figure 13. Although the SVM model demonstrates satisfactory performance, it is significantly outperformed by the late-fusion CNN-based model, with the latter achieving higher validation accuracy. This outcome emphasizes the superior

ability of the CNN model to accurately classify human activities. The present findings unequivocally establish that the developed CNN model, when combined with the late-fusion technique, substantially enhances the accuracy of HAR. In addition, the classification report provides empirical evidence of its effectiveness in distinguishing a wide range of activities. The impressive accuracy of the CNN model at 95% is somewhat constrained by the limited availability of datasets featuring diverse sensor types and the relatively small dataset size, comprising only 2,899 samples. To further advance HAR systems, future investigations can explore utilizing larger and more diverse datasets to continue improving the accuracy and robustness of these models.

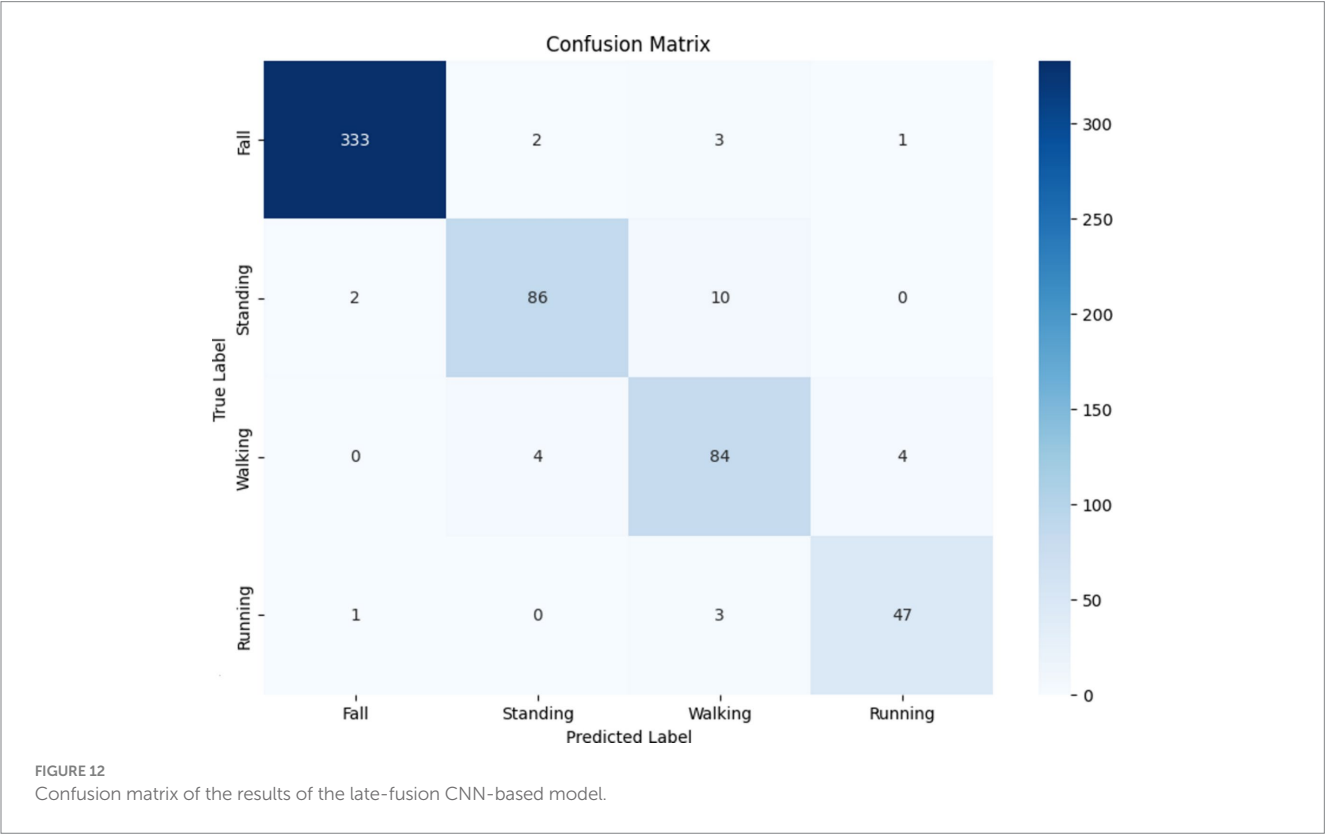
4.5 Results of the custom collected data

4.5.1 Results of the RFT

The results provide encouraging possibilities in the realm of indoor localization via the use of ML methodologies. Although the dataset was limited, with only 111 examples, the principal model used (the RF classifier) revealed strong performance in accurately identifying the position of users utilizing sensor data. The model demonstrated a commendable overall accuracy of 91.30%. The classification report offers comprehensive metrics for each class, further clarifying the model's performance. The metrics of precision, recall and F1-score were calculated for each location class (Room1, Room2, and Room3), as presented in the classification report provided in Table 8. The confusion matrix of the RF model is illustrated in Figure 14.

TABLE 6 Classification report for the late-fusion CNN model.

	Precision	Recall	F1-score	Support
Fall	0.99	0.98	0.99	339
Standing	0.93	0.88	0.91	98
Walking	0.84	0.91	0.87	92
Running	0.90	0.92	0.91	51
Accuracy			0.95	580
Macro average	0.92	0.92	0.92	580
Weighted average	0.95	0.95	0.95	580



4.5.2 Results of the K-NN model

These metrics provide essential insight into the model’s capacity to accurately categorize each site. The confusion matrix offers a graphical depiction of the model’s predictions compared to the actual ground truth, facilitating the evaluation of true positives, false positives, true negatives, and false negatives. The findings illustrate the resilience of the RF model in indoor localization, indicating its potential for practical applications. With a larger dataset, the model’s performance can improve in robustness and accuracy. The classification report and confusion matrix of the K-NN model are presented in Table 9 and Figure 15, respectively.

TABLE 7 Classification report for the traditional SVM classifier.

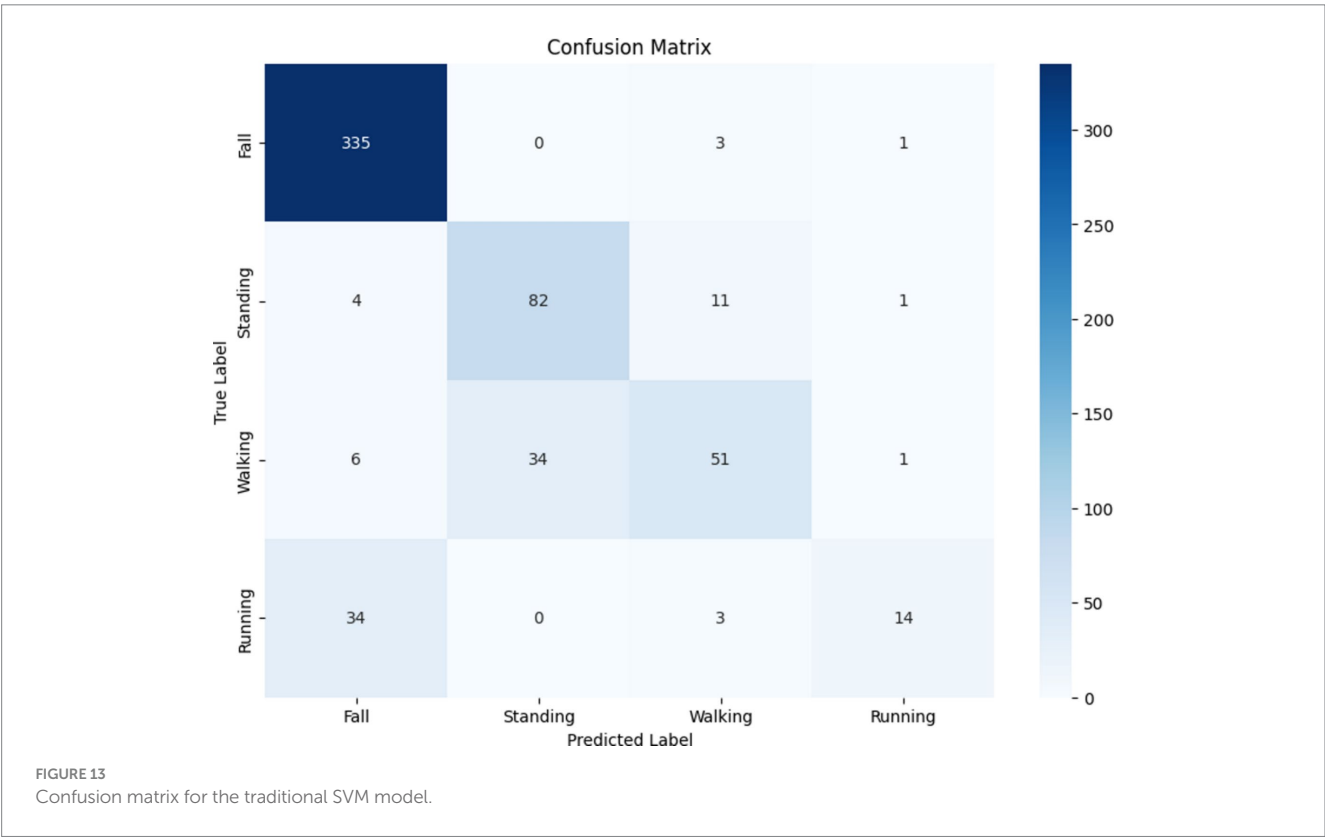
	Precision	Recall	F1-score	Support
Fall	0.88	0.99	0.93	339
Standing	0.71	0.84	0.77	98
Walking	0.75	0.55	0.64	92
Running	0.82	0.27	0.41	51
Accuracy			0.83	580
Macro average	0.79	0.66	0.69	580
Weighted average	0.83	0.83	0.81	580

5 Discussion

Wearable devices have enabled a range of functions, including recording activities, monitoring wellbeing, and interacting with computers, all aimed at evaluating and improving users’ everyday habits. These applications make use of low-power sensors on mobile and wearable devices to facilitate HAR. The system proposed in this study utilizes CNNs within a late-fusion framework to analyze and integrate data from various sensors for precise HAR specifically designed for healthcare applications. Processing inputs from accelerometers, gyroscopes, and other sensors provide a comprehensive and dynamic representation of patient movements, facilitating accurate and real-time monitoring of physical activities.

The advanced approach to HAR provides significant benefits in the healthcare sector by enabling continuous, non-invasive monitoring of users’ physical activities, contributing to personalized healthcare plans, early detection of potential health issues, and enhanced user care. The proposed system’s high accuracy and reliability in activity recognition can support healthcare professionals in making informed decisions, optimizing treatment plans, and monitoring user recovery processes, improving overall user outcomes.

This study conducted extensive preprocessing to prepare the dataset for training the ML model. The preparation procedures included importing the dataset from an Excel file and performing random shuffling to provide impartial training data. The string representations of the lists in the “Acc,” “Gyr,” and “RSSI” columns were transformed into concrete lists. Then, the sensor data were analyzed to obtain important statistical parameters (e.g., mean,



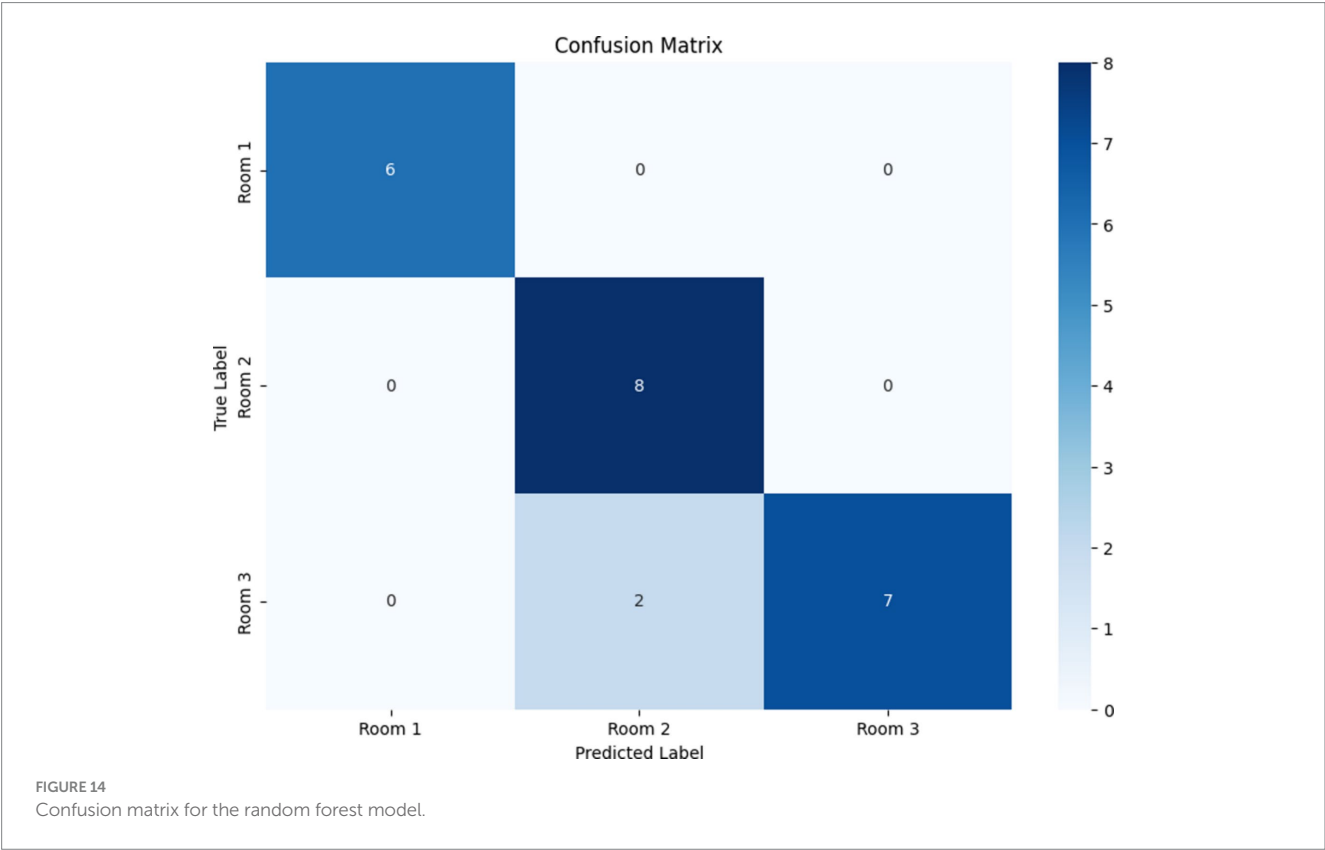
standard deviation, and range) like those for the feature engineering and traditional SVM preprocessing section above. The features were employed as input variables to train the model. The features from “Acc,” “Gyr,” and “RSSI” were merged to form a single array of features for each sample. Table 10 lists the performance of ML and the standard and customized dataset of the CNN model. The study demonstrated that the late-fusion approach utilizing CNNs outperformed traditional HAR methods, with the former achieving a test accuracy of 94.83% compared to that of the SVM classifier at 83.10%. These findings highlighted the effectiveness of using multisensory data through advanced DL techniques, indicating a substantial advancement in accurately classifying diverse human activities. They also emphasized the potential of CNN-based models in setting new standards for HAR applications and the importance of integrating complex sensor data for enhanced performance.

TABLE 8 Classification report for the random forest model.

	Precision	Recall	F1-score	Support
Room 1	1.0	1.0	1.0	6.0
Room 2	0.8	1.0	0.89	8.0
Room 3	1.0	0.78	0.88	9.0
Accuracy				0.91
Macro average	0.93	0.93	0.92	23
Weighted average	0.93	0.91	0.91	23

This study recommends, in light of the system’s exceptional performance and the accuracy of the sensors used, that future research efforts focus on the following:

- 1 **Dataset size:** The accumulation of more diverse and extensive datasets. Such endeavors will bolster the system’s robustness across various scenarios and facilitate the exploration of new dimensions in HAR. In addition, the research community is encouraged to explore integrating these refined datasets with the system to enhance its efficacy and applicability in real-world contexts. This collaborative approach promises to set new benchmarks in the field, extending the frontiers of HAR technology.
- 2 **Sensor fusion challenges:** Combining data from various sensors such as accelerometer, gyroscope, and barometer can be difficult because of differences in sample rates, data formats, and sensor-specific traits. The study addressed this issue by creating specialized CNN models for individual sensor types to capture their distinct characteristics and merge the data successfully in a subsequent phase.
- 3 **Integration of additional sensors:** Enhancing the process by integrating other sensors like heart rate monitors or electromyography (EMG) sensors can provide a more thorough understanding of human mobility and physiological reactions.
- 4 **Computational complexity:** CNNs can be costly in terms of the computer resources required for training and deployment. In the future, this issue can be resolved by improving the architecture of the CNN models using methods such as pruning or quantization to decrease the model size or utilizing cloud computing resources for training and inference.



5 Real-world applicability: The model's performance in practical situations can vary from its performance on the test dataset due to differences in sensor placement, user behavior, and ambient variables. It can collect and test the model using a more diverse dataset that accurately reflects real-world scenarios to address this issue.

The applicability of the results in the research to different HAR applications and datasets is contingent on certain factors:

- **Activity similarity:** The behaviors discussed in the study, such as walking, running, ascending stairs, and falling, are frequently used in various HAR applications. The techniques and models presented in the research can be directly applied or readily adjusted for similar HAR circumstances.

- **Sensor configuration:** The sensor configuration utilized in the paper, consisting of an accelerometer, gyroscope, and barometer, is frequently employed in various HAR applications. If the sensor setup is substantially different, such as using varied sensor kinds or a variable quantity of sensors, the models can require adjustments or retraining to accommodate the new data.
- **Data quality and quantity:** The performance of models can be considerably affected by the quality and quantity of data used for training and testing. The FallAllID dataset in the paper was minimal, perhaps restricting the models' applicability to bigger, more varied datasets. The models' generalizability can be enhanced by retraining them on a larger and more diverse dataset.
- **Variability:** The study recognizes that the model's performance can vary in real-world situations compared to its performance on the test dataset due to differences in sensor placement, user behavior, and ambient variables. Hence, it is crucial to consider these elements when using the methodology in various situations.

TABLE 9 Classification report of the K-NN model.

	Precision	Recall	F1-score	Support
Room 1	0.56	0.83	0.67	6
Room 2	0.71	0.56	0.63	9
Room 3	0.86	0.75	0.80	8
Accuracy			0.70	23
Macro average	0.71	0.71	0.70	23
Weighted average	0.72	0.70	0.70	23

6 Conclusion

The primary objective of this study is to investigate the capabilities and effectiveness of a multisensory approach, specifically the combination of an IMU and a barometer, to observe and track human movement. The empirical findings support the idea that integrating a triaxial accelerometer, a triaxial gyroscope, and a barometer enhances precision in recognizing various human movement patterns. This enhancement is further reinforced by incorporating an additional filter

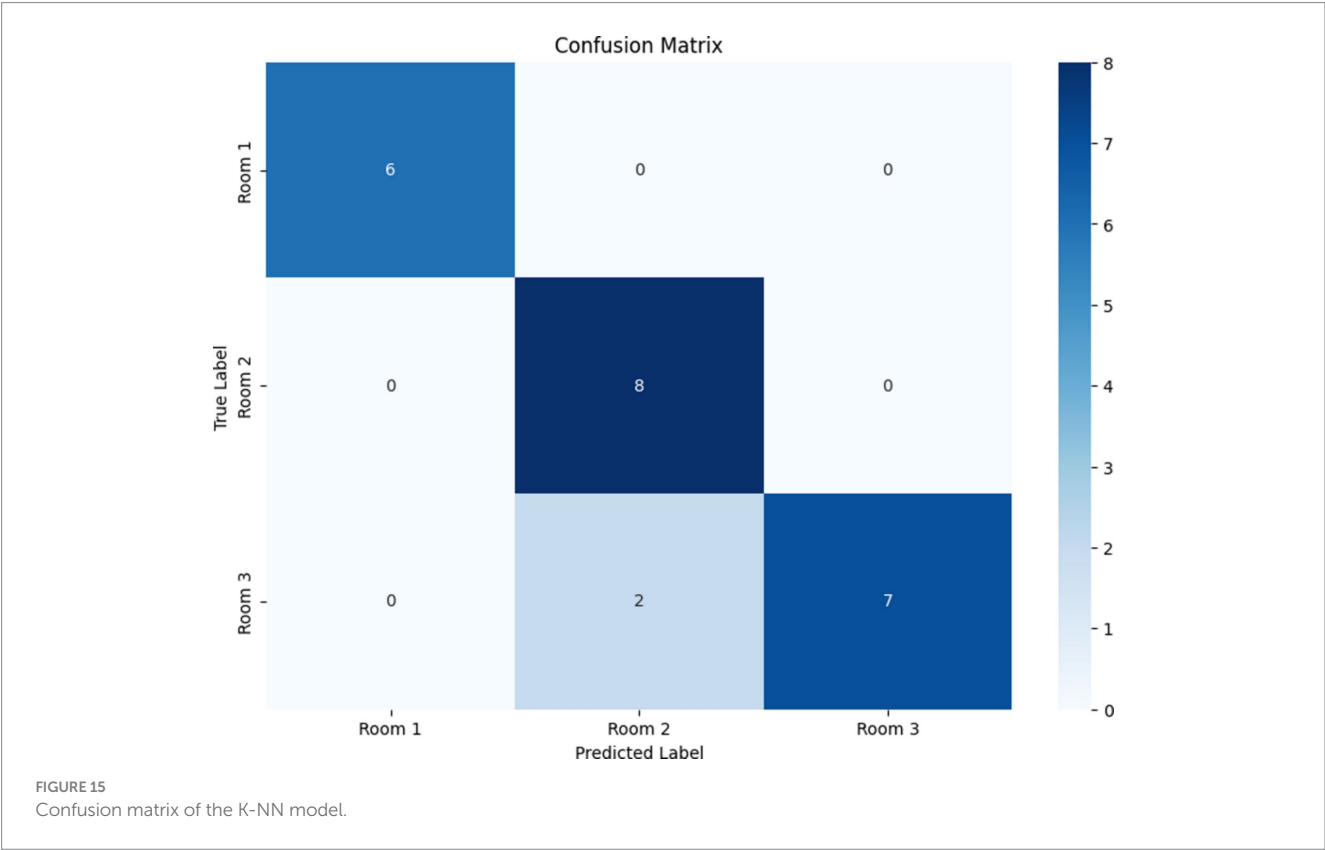


TABLE 10 Test and validation accuracies of the four models.

	Standard data for HAR				Custom data for localization			
	CNN	SVM	Random forest	K-NN	CNN	SVM	Random forest	K-NN
Validation %	98.35	87.07	96.12	92.03	94.44	72.22	89	70
Testing %	95	83	95.52	89.31	82.60	52.17	91	70

algorithm, effectively distinguishing between diverse movement patterns, such as standing, falling, running, and walking. In addition, the comprehensive monitoring of various physiological indicators (e.g., cardiovascular rate, sphygmomanometer readings, and thermal body states) provides an additional layer of diagnostic accuracy. This array of capabilities represents a significant advancement in the field of geriatric care, with the potential to mitigate adverse consequences associated with unforeseen movement-related incidents, including falls.

The late-fusion convolutional neural network model in this study improves HAR by achieving a final test accuracy of 94.83%, outperforming the standard SVM classifier using a one-vs-rest approach, which had an accuracy of 83.10%. Using customized CNNs for each sensor type and employing the late-fusion strategy to combine their predictions has proven beneficial. The improved precision in HAR, mainly in distinguishing between behaviors such as falling and regular everyday activities, has significant implications for fall detection systems, personalized health monitoring, and sports performance analysis. Future research will focus on improving the methodology using larger and more diverse datasets, adding further sensors, enhancing real-time processing, and introducing explainable AI techniques. This research ultimately seeks to enhance persons' quality of life by developing more precise and efficient HAR systems that can be integrated into wearable devices.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

MA: Conceptualization, Investigation, Methodology, Resources, Supervision, Validation, Writing – original draft, Writing – review & editing. AM: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Supervision,

Validation, Writing – original draft, Writing – review & editing. AAl: Methodology, Supervision, Writing – review & editing. MF: Methodology, Validation, Writing – review & editing. SC: Conceptualization, Investigation, Software, Writing – review & editing. AAF: Data curation, Methodology, Writing – review & editing. AAh: Writing – review & editing. OR: Conceptualization, Investigation, Methodology, Software, Writing – original draft, Data curation. AH: Conceptualization, Investigation, Methodology, Writing – original draft, Writing – review & editing, Software. MT: Writing – review & editing, Conceptualization, Data curation, Investigation, Methodology, Writing – original draft.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. The authors acknowledge the Deanship of Scientific Research at King Faisal University for the financial support under the annual research project (Grant No. KFU241307).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Abusaaq, H. I. (2015). Population aging in Saudi Arabia. *Saudi Arabian Monitory. Agency* 4-5:374.

Agac, S., Shoaib, M., and Incel, O. D. (2021). Context-aware and dynamically adaptable activity recognition with smart watches: a case study on smoking. *Comput. Electr. Eng.* 90:106949. doi: 10.1016/j.compeleceng.2020.106949

Alarfaj, M., Qian, Y., and Liu, H. (2021). Detection of human body movement patterns and barometers, in *Proceedings of the International Conference on Communications, Signal Processing, and Their Application*. 2021.

Almabdy, S., and Elrefaei, L. (2019). Deep convolutional neural network-based approaches for face recognition. *Appl. Sci.* 9:4397. doi: 10.3390/app9204397

Demrozi, F., Pravadelli, G., Bihorac, A., and Rashidi, P. (2020). Human activity recognition using inertial, physiological and environmental sensors: a comprehensive survey. *IEEE Access* 8, 210816–210836. doi: 10.1109/access.2020.3037715

Fahad, L. G., and Tahir, S. F. (2021). Activity recognition in a smart home using local feature weighting and variants of nearest-neighbors classifiers. *J. Ambient. Intell. Humaniz. Comput.* 12, 2355–2364. doi: 10.1007/s12652-020-02348-6

Hammerling, M., and Rosipal, F. (2013). Sensor fusion for activity recognition from accelerometer and gyroscope data, in *International conference on artificial neural networks*. Berlin: Springer, 419, 355–364.

Hartmann, Y., Liu, H., and Schultz, T. (2022). Interactive and interpretable online human activity recognition, in *Proceedings of the 2022 IEEE international conference on pervasive computing and communications workshops and other affiliated events*, Pisa, Italy, 109–111.

- Hayat, A., Morgado-Dias, F., Bhuyan, B. P., and Tomar, R. (2022). Human activity recognition for elderly people using machine and deep learning approaches. *Information* 13:275. doi: 10.3390/info13060275
- Khan, I. U., Afzal, S., and Lee, J. W. (2022). Human activity recognition via hybrid deep learning based model. *Sensors* 22:323. doi: 10.3390/s22010323
- Li, Y., and Wang, L. (2022). Human activity recognition based on residual network and BiLSTM. *Sensors* 22:635. doi: 10.3390/s22020635
- Luwe, Y. J., Lee, C. P., and Lim, K. M. (2022). Wearable sensor-based human activity recognition with hybrid deep learning model. *Informatics* 9:56. doi: 10.3390/informatics9030056
- Mahmud, T., Sazzad Sayyed, A. Q. M., Fattah, S. A., and Kung, S.-Y. (2021). A novel multi-stage training approach for human activity recognition from multimodal wearable sensor data using deep neural network. *IEEE. Sens. J.* 21, 1715–1726. doi: 10.1109/JSEN.2020.3015781
- Maswadi, K., Ghani, N. A., Hamid, S., and Rasheed, M. B. (2021). Human activity classification using decision tree and naive Bayes classifiers. *Multimed. Tools Appl.* 80, 21709–21726. doi: 10.1007/s11042-020-10447-x
- Medjahed, H., Istrate, D., Boudy, J., and Dorizzi, B. (2009). Human activities of daily living recognition using fuzzy logic for elderly home monitoring, in Proceedings of the 2009 IEEE international conference on fuzzy systems, Jeju, Republic of Korea, 2001–2006.
- Moya Rueda, F., Grzeszick, R., Fink, G. A., Feldhorst, S., and ten Hompel, M. (2018). Convolutional neural networks for human activity recognition using body-worn sensors. *Informatics* 5:26. doi: 10.3390/informatics5020026
- Muralidharan, K., Ramesh, A., Rithvik, G., Prem, S., Reghunaath, A. A., and Gopinath, M. P. (2021). 1D convolution approach to human activity recognition using sensor data and comparison with machine learning algorithms. *Int. J. Cogn. Comput. Eng.* 2, 130–143. doi: 10.1016/j.ijcce.2021.09.001
- Mutegeki, R., and Han, D. S. (2020). A CNN-LSTM approach to human activity recognition, in Proceedings of the 2020 international conference on artificial intelligence in information and communication (ICAIIIC), Fukuoka, Japan. Piscataway, NJ, USA: IEEE, 362–366.
- Nguyen, B., Coelho, Y., Bastos, T., and Krishnan, S. (2021). Trends in human activity recognition with focus on machine learning and power requirements. *Mach. Learn. Appl.* 5:100072. doi: 10.1016/j.mlwa.2021.100072
- Radhika, V., Prasad, C. R., and Chakradhar, A. (2022). Smartphone-based human activities recognition system using random forest algorithm, in Proceedings of the 2022 international conference for the advancement in technology, Goa, India, 1–4.
- Ronald, M., Poulouse, A., and Han, D. S. (2021). iSPLInception: an inception-ResNet deep learning architecture for human activity recognition. *IEEE. Access* 9, 68985–69001. doi: 10.1109/ACCESS.2021.3078184
- Saleh, M., Abbas, M., and Le Jeannes, R. B. (2021). FallAllID: an open dataset of human falls and activities of daily living for classical and deep learning applications. *Sens. J.* 21, 1849–1858. doi: 10.1109/JSEN.2020.3018335
- Schneider, B., and Banerjee, T. (2021). Bridging the gap between atomic and complex activities in first person video, in Proceedings of the 2021 IEEE International Conference on Fuzzy Systems, Luxembourg, 1–6.
- Sikder, N., Ahad, M. A. R., and Nahid, A.-A. (2021). Human action recognition based on a sequential deep learning model, in Proceedings of the 2021 joint 10th international conference on informatics, Electronics & Vision (ICIEV) and 2021 5th international conference on imaging, Vision & Pattern Recognition (icIVPR), Kitakyushu, Japan, 1–7.
- Sun, S., Cao, J., Xu, Y., and He, Z. (2018). A hybrid deep learning framework for human activity recognition using wearable sensors. *IEEE Trans. Cybern.* 49, 3900–3910.
- Wang, Z., Chen, Y., Fu, X., and Hu, W. (2016). A stacked generalization based late fusion approach for human activity recognition with multi-sensor wearable devices. *Pervasive Mob. Comput.* 29:423.
- Xu, C., Chai, D., He, J., Zhang, X., and Duan, S. I. H. (2019). InnoHAR: a deep neural network for complex human activity recognition. *IEEE. Access* 7, 9893–9902. doi: 10.1109/ACCESS.2018.2890675
- Yang, A., Xu, Y., Zhou, X., and Jia, W. (2017). Late fusion of low-level features for action recognition with two-stream convolutional networks. *Pattern Recogn. Lett.* 88:421.
- Yu, H., Chen, W., Zhang, S., and Guan, L. (2020). A hybrid architecture with early and late fusion for human activity recognition using wearable sensors. *Sensors* 20:339.
- Zhang, W., Liu, Y., Li, H., and Li, R. (2019). Deep learning based late fusion for human activity recognition with wearable sensors. *Sensors* 19, 424–425.
- Zhang, L., Zhu, G., Shen, P., Song, J., Shah, S. A., and Bennamoun, M. (2017). Learning spatiotemporal features using 3DCNN and convolutional LSTM for gesture recognition, in Proceedings of the I.E.E.E. International Conference on Computer Vision Workshops, Venice, Italy, 3120–3128.
- Zhao, W., Xu, J., Li, X., Chen, Z., and Chen, X. (2022). Recognition of farmers' working based on HC-LSTM model. *Neurocomputing* 813, 77–86. doi: 10.1007/978-981-16-6963-7_7
- Zheng, W., Yin, L., Chen, X., Ma, Z., Liu, S., and Yang, B. (2021). Knowledge base graph embedding module design for visual question answering model. *Pattern Recogn.* 120:108153. doi: 10.1016/j.patcog.2021.108153



OPEN ACCESS

EDITED BY

Deepika Koundal,
University of Petroleum and Energy Studies,
India

REVIEWED BY

Tariq Hussain,
Zhejiang Gongshang University, China
Rohan Borgalli,
University of Mumbai, India

*CORRESPONDENCE

Shtwai Alsubai
✉ sa.alsubai@psau.edu.sa

RECEIVED 21 May 2024

ACCEPTED 04 October 2024

PUBLISHED 30 October 2024

CITATION

Alsubai S, Alqahtani A, Alanazi A, Sha M and
Gumaei A (2024) Facial emotion recognition
using deep quantum and advanced transfer
learning mechanism.
Front. Comput. Neurosci. 18:1435956.
doi: 10.3389/fncom.2024.1435956

COPYRIGHT

© 2024 Alsubai, Alqahtani, Alanazi, Sha and
Gumaei. This is an open-access article
distributed under the terms of the [Creative
Commons Attribution License \(CC BY\)](#). The
use, distribution or reproduction in other
forums is permitted, provided the original
author(s) and the copyright owner(s) are
credited and that the original publication in
this journal is cited, in accordance with
accepted academic practice. No use,
distribution or reproduction is permitted
which does not comply with these terms.

Facial emotion recognition using deep quantum and advanced transfer learning mechanism

Shtwai Alsubai^{1*}, Abdullah Alqahtani¹, Abed Alanazi¹,
Mohemmed Sha² and Abdu Gumaei¹

¹Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia, ²Department of Software Engineering, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia

Introduction: Facial expressions have become a common way for interaction among humans. People cannot comprehend and predict the emotions or expressions of individuals through simple vision. Thus, in psychology, detecting facial expressions or emotion analysis demands an assessment and evaluation of decisions for identifying the emotions of a person or any group during communication. With the recent evolution of technology, AI (Artificial Intelligence) has gained significant usage, wherein DL (Deep Learning) based algorithms are employed for detecting facial expressions.

Methods: The study proposes a system design that detects facial expressions by extracting relevant features using a Modified ResNet model. The proposed system stacks building-blocks with residual connections and employs an advanced extraction method with quantum computing, which significantly reduces computation time compared to conventional methods. The backbone stem utilizes a quantum convolutional layer comprised of several parameterized quantum-filters. Additionally, the research integrates residual connections in the ResNet-18 model with the Modified up Sampled Bottle Neck Process (MuS-BNP), retaining computational efficacy while benefiting from residual connections.

Results: The proposed model demonstrates superior performance by overcoming the issue of maximum similarity within varied facial expressions. The system's ability to accurately detect and differentiate between expressions is measured using performance metrics such as accuracy, F1-score, recall, and precision.

Discussion: This performance analysis confirms the efficacy of the proposed system, highlighting the advantages of quantum computing in feature extraction and the integration of residual connections. The model achieves quantum superiority, providing faster and more accurate computations compared to existing methodologies. The results suggest that the proposed approach offers a promising solution for facial expression recognition tasks, significantly improving both speed and accuracy.

KEYWORDS

facial expressions, artificial intelligence, deep learning, quantum computing, ResNet model

1 Introduction

Facial expressions are a form of non-verbal communication that arise from the movement of facial muscles to convey emotions or gestures (Khan, 2022). They serve as a means of expressing emotions, such as opinions, goals, intentions, and feelings. However, predicting human expression is challenging. Currently, computer applications are widely used to calculate

facial expression scores. Facial emotion recognition (FER) is essential for computer vision-aided applications to enhance human-computer interactions.

Human faces exhibit a heterogeneous nature, with image variations caused by factors such as lighting and poses, which pose challenges for computer models to achieve robust and accurate predictions (Kaur and Singh, 2022). In FER, the process of associating different facial expressions with their corresponding emotions involves several steps, including image pre-processing, feature selection, and feature classification.

In traditional computer-based models, feature extraction and noise reduction have been carried out using polyp (Tsuneki, 2022) computer-aided classification models. Various feature extraction techniques have been used in existing research, such as principle component analysis (PCA) (Sachadev and Bhatnagar, 2022), linear discriminant analysis (LDA), individual component analysis (ICA), local dynamic pattern (LDP) (Makhija and Sharma, 2019), geometric feature mapping (Rosen et al., 2021), and elastic bunch graph mapping (EBGM) (Oloyede et al., 2020). Machine learning (ML)-based algorithms can be used in the classification process. However, an additional feature engineering process is required for feature extraction. Deep learning (DL) (Karnati et al., 2023), a sub-domain of ML algorithms, has been widely used in image classification tasks for enhanced accuracy. The training time for DL algorithms has been less than for ML algorithms. Convolutional Neural Network (CNN) (Mohan et al., 2020) is a significant algorithm used for image classification as part of ML and deep learning-based neural networks (Mungra et al., 2020; Mohan et al., 2021). Unlike the traditional models, CNN can extract abstract and accurate features. Automatic learning can be enhanced with CNN by adopting depth features (Karnati et al., 2022) and block architectures. Traditional CNN algorithms perform better for many image classification tasks like SVNN (Ghasemi et al., 2020), CIFAR (Yang et al., 2020), and MNIST (Kadam et al., 2020).

Quantum-based principles can be integrated into ML models across various domains. Quantum-enabled ML models have been used in various algorithms such as quantum neural networks, quantum generative models, and quantum support vector machines. Artificial intelligence (AI)-based algorithms can be seen as a resemblance of the human brain with highly abstract functions. Significant AI models include capsule neural networks (Jiang et al., 2020), recurrent neural networks (RNN) (Mei et al., 2019), feedforward neural networks, (Tacchino et al., 2020) and CNN. Quantum neural networks (QNN) employ quantum mechanisms to enhance the structure of neural networks (Wang et al., 2022). The architecture can be improved through the concepts of quantum interference attributes, quantum entanglement, and quantum parallelism. The performance of a traditional neural network can be enhanced by implementing a conventional neural network with a quantum neural network. The hybrid architectures thus formed can be trained and tested on IBM Quantum Experience through Qiskit-enabled quantum computers.

QNNs have similarities with traditional neural models and have variation parameters. QNNs have several potential advantages. Quantum computers can outperform traditional models in speed for Fourier transform based on Shor's factoring technique. Various computational issues can be efficiently resolved with quantum contextuality and non-locality. Moreover, the learning process from a quantum dataset created by a quantum process is more efficient than

a traditional dataset. In large-scale exponential datasets such as Hilbert space, the ability of QNN to extract adequate data from the quantum state is difficult (Li et al., 2022).

Moreover, quantum networks can perform massive parallel calculations and provide high-performance speed. An attention mechanism has recently been used in QNN models. An enhanced CNN model has been used in a DL computer vision application named AlexNet. It has performed data augmentation, convolutions, ReLU activations, max pooling, stochastic gradient descent (SGD) (Zheng et al., 2019), and dropout. The issue with deep network training can be mitigated by implementing modified blocks that ignore and leap over layers. This enhanced the training of large networks with fewer training errors.

Another ResNet model has been implemented for deep-coupled low-resolution neural networks (Kavitha et al., 2022). The ResNet model has selected dissimilar features in various facial images. The image features have been projected with training from coupled mappings of branch networks. The models have been evaluated with SCface datasets and LFW datasets and have achieved remarkable accuracy for face verification (Singhal et al., 2021). Even though various face recognition models have been developed, high recognition rates are difficult to achieve with traditional feature classification algorithms.

Moreover, convolutional layers have the ability to handle only spatial features in images. Subtle and depth features are not properly recognized with CNN models. Furthermore, the abstract features extracted in the deep CNN model suffer from vanishing gradient issues as the number of layers increases. QNN algorithms provide correlated and probabilistic components, whereas performance is limited by dimensionality issues and computational bottlenecks. To resolve all the above issues, the MuS-BNP with ResNet-18 model named MuS-BNP is proposed.

The model uses the FER 13 dataset to predict the facial emotions in the images. Unlike traditional CNN and ResNet architecture, both shallow and deep features are extracted using a backbone stem integrated with a quantum convolutional layer. This layer incorporates various parameterized quantum filters, which replace the conventional kernel in traditional convolutional layers. The parameterized quantum filter is used to obtain quantum bit information in the local data space. It includes a double-bit gate that performs quantum entanglement on other quantum bits, enhancing the interaction between data points.

In this process, pixel value information is converted into quantum state information through quantum state encoding, achieved via a quantum rotation gate. The model retains the weight-sharing mechanism of the traditional kernel while incorporating quantum parameters to boost computational capabilities. Furthermore, the filter connection phases in the ResNet-18 model are linked with the MuS-BNP through residual connections, which significantly enhance computational performance. The major contributions of the proposed model, combining the MuS-BNP with the ResNet-18 architecture, are as follows:

- To perform shallow and deep feature extraction through a backbone stem network and a modified quantum convolutional layer with parameterized quantum filters.
- To perform facial emotion classification through the proposed MuS-BNP with the ResNet-18 model in less computation time.
- To evaluate the efficacy of the proposed model with performance metrics such as accuracy, F1-score, recall, and precision.

1.1 Contributions

QNNs are typically designed to handle large data efficiently, unlike conventional NNs (neural networks), which permit them to accomplish better classification. The present study proposes ResNet18 architecture with a Modified Sampled Bottleneck Process for FER. Accordingly, residual connections have been utilized to associate the filter connection phase in the ResNet-18 model with the MuS-BNP. This architecture helps manage computational efficiency while leveraging the benefits of residual connections. Moreover, the residual version of the ResNet-18 model with the MuS-BNP has employed a simplified module.

Furthermore, the filter expansion layer that follows each module has been enlarged with the dimensions of the filter bank. For matching the input, it has been integrated before. Thus, it reimburses the minimization of dimensionality that is available in an n block.

Feature extraction has also been accomplished with the quantum convolutional layer. This is encompassed with various parameterized quantum filters. Similar to the convolution kernel present in the conventional convolutional layer, the parameterized quantum filter finds utility for information extraction that is present in individual quantum bits. In an image, the pixel value corresponding to the information has been altered into the quantum state information (that utilizes quantum state encoding) with the means of the quantum rotational gate $R(\theta)$. In accordance with this process, the procured information regarding the features of the image has been modified into the angle of the quantum rotatory gate.

Furthermore, for the quantum rotatory gate, the corresponding parameters have been afforded by each pixel value. The proposed method comprises exclusive quantum mechanical features and retains the weight sharing in the convolutional kernel. In the proposed technique, individual blocks have a self-regulating convolutional way of delivering information in the prior and middle layers.

The strategy introduces the concept of “pass-over,” a modification from the ResNet model that builds on modest blocks containing residual connections. The traditional residual building block has not utilized the information accessible in the middle layer. However, the proposed model incorporates pass-over information to capture all relevant features.

Thus, the proposed ResNets with QNNs possess the ability to generalize. Furthermore, by leveraging the effects of quantum-like superposition and entanglement, QNNs obtain several complex associations amongst the input features, resulting in model robustness and better generalization. The proposed QNN could effectively use quantum hardware, leading to the count of quantum gates needed for computation. Through this system, quantum gates needed for computation are also minimized. The proposed framework finds more complex and subtle features of an image than traditional algorithms, resulting in robust and optimal classification. Moreover, the proposed system performs functions on multiple qubits at concurrent times, permitting the effective parallel processing of the features from the images.

1.2 Paper organization

Section II of the paper deals with the review of existing literature for image recognition and classification through various ML models,

DL models, and quantum-based DL models. The problems identified from the existing literature have also been discussed. Section III deals with the proposed flow, architecture, and mathematical formulations. Section IV deals with the dataset description, performance results, comparative results, and discussions. Section V deals with the conclusions and future recommendations of the work.

2 Review of literature

Image classification and emotion recognition can be performed in literature through various ML algorithms, DL algorithms, and enhanced quantum-based ML and DL algorithms. The section briefly deals with all conventional models, along with the gaps identified from the state of artworks.

A human emotion identification model has been proposed in the study (Alreshidi and Ullah, 2020) using two ML algorithms for image classification and detection. The model has been trained for real-time implementations offline. The faces in the image are initially recognized with AdaBoost cascade algorithms (Chen et al., 2019). The facial features denoted by localized appearance data named Neighborhood Difference Features (NDF) (Kaplan et al., 2020) have been extracted. The association among various NDF patterns has been considered rather than intensity data. Even though the study calculates only seven facial emotions, it can be extended to more facial feature recognition. The model has been invariant to skin color, gender, orientation, and illumination. The evaluation results on Real-World Affective Faces (RAF) (Jiang et al., 2020) and Static Facial Expressions in the Wild (SFEW) (Liu, 2020) datasets have exhibited 24 and 13% accuracy enhancement, respectively.

Another study has been designed to identify microexpressions in human faces. Unsupervised micro-expression detection models based on ML algorithms have been suggested with extreme learning machines (ELMs). The algorithm offers higher performance and faster training ability than conventional algorithms. The ELM model has been compared with the Support Vector Machine (SVM) (Okwuashi and Ndehedehe, 2020) benchmark model for training time efficacy. Feature extraction has been performed through Local Binary Pattern (LBP) (Zhao et al., 2019) on apex-micro expression frame and Local Binary Pattern on Three Orthogonal Planes (LBP-TOP)-based division of image segments from video through spatiotemporal features. The model has been evaluated using a dataset from the Chinese Academy of Sciences (CASME II). The results indicate that ELM has a better prediction rate and less computation time than SVM (Adegun and Vadapalli, 2020).

The facial emotion intensity has been encoded by considering multimodal facial behavior for recognizing emotions from intensities. The intensity extraction has been performed with ML algorithms like Random Forest (RF) (Speiser et al., 2019), SVM, and K-Nearest Neighbor (KNN) (Ma et al., 2020). Three feature extraction methods, namely local binary pattern (LBP), histogram of oriented gradients (HOG) (Zhou et al., 2020), and Gabor features (Munawar et al., 2021), have been implemented. Intensity calculation and emotion identification have been performed through a comparative analysis of three algorithms on CK, B DFE, JAFEE, and private datasets. Emotion recognition and facial intensity detection have been analyzed from the three algorithms (Mehta et al., 2019).

Another fake image detection model has been developed with generative adversarial networks (GANs) that create fake images with low-dimension noise. Fake images have created various issues in social media networks. Contrastive loss-based fake image detection has been implemented using the DL-based DenseNet model. Pairwise information has been fed as input through a two-streamed network model. The training has been performed on the pairwise information to identify the fake input image (Hsu et al., 2020). DL-based CNN models have exhibited high computational efficiency and unsupervised feature extraction. CNN-based image prediction has been performed on the FER 2013 dataset. The visual geometric group (VGG) algorithm (Deepan and Sudha, 2020) has been used to design the model with various learning schedulers and optimization techniques. The model's hyperparameters have been tuned, and the accuracy is 73.28% (Khairuddin and Chen, 2021).

High-level feature identification from facial images has been performed with a two-layer CNN model and sparse representation. The training data independent of feature space has been used to sparsely denote the facial features in the proposed Sparse Representation Classifier (SRC). Real-world classification and feature recognition depend on the proper details extracted from the faces of images. The results of the SRC-based feature selector have proved superior to other traditional classifiers (Cheng et al., 2019). The transfer learning (TL)-based deep CNN (DCNN) model has been developed for accurate classification of images, considering shallow and depth features. The pertained DCNN model has been modified with a FER-compatible upper dense layer fine-tuned to recognize facial emotion. The pipelining technique has been adopted after dense layer training and tuning. The model has been tested on pertained DCNN models like DenseNet-161 (Song et al., 2019), Inception-v3, ResNet-152 (Gour et al., 2020), ResNet-50, ResNet-34, ResNet-18, VGG19 and VGG-16, along with JAFFE and KDEF, using a 10-fold cross-validation approach (Akhand et al., 2021).

Another study identified facial emotion from video sequences with global and local networks (Hu et al., 2019). The cascaded CNN-LSTM networks and Local Enhanced Motion History Image (LEMHI) (Gavade et al., 2022) have been implemented for the above feature extraction. LEMHI has been used to aggregate the video frames as a single frame, which has been fed into the CNN for prediction. The global features have been extracted through an enhanced CNN-LSTM model as a classifier and feature extractor. The final prediction was performed using a late fusion fashion-based random search summation model. The information to decode the features from facial images has been obtained from each CNN layer. The experiments on MMI, CK+, and AFEW datasets have exhibited better integrated model performance than the individual model. The complexity of the CNN (Jing et al., 2022) model depends on the activation function.

Although the ReLU activation function outperforms tanh and sigmoid in many cases, it still has limitations. The ReLU model returns zero value on negative inputs, which is termed neuronal necrosis. This has been eliminated by implementing a piecewise activation function in CNN. The new function has been compared with other functions such as softplus-ReLU, leaky ReLU, tanh, and Sigmoid (Zhang et al., 2022). The comparison of results on the Keras framework utilizing the FER13 and JAFFE datasets exhibited better activation function performance (Wang et al., 2020). Another deep CNN-based model has been implemented with residual blocks for enhanced performance.

The image labels have been initiated, followed by training on the proposed DNN model. Japanese Female Facial Expression (JAFPE) and Extended Cohn-Kanade (CK+) datasets have been used to test the accuracy of the model (Jain et al., 2019). Computational issues have been optimized through an unsupervised ensemble model of hybrid deep neural networks (HDNN) and an improved quantum-inspired gravitational search algorithm (IQI-GSA). Quantum computing and gravitational search algorithm (GSA) have been combined to form IQI-GSA. The local trapping and stochastic features have been handled with the enhanced model. The temporal and relational components have been optimized by hybridizing recurrent and convolutional (HDCR-NN) neural models. The experimental analysis has been performed on KDEF and JAFPE datasets to exhibit the model's efficacy (Kumar et al., 2021).

Transfer learning (Tammina, 2019) with a quantum-based hybrid approach has been implemented to ensure security and reliability. The fake images have been classified using the ResNet-18-based quantum neural model. The model has been trained on various depths, and the reliability of vision-based models is tested (Ciylan and Ciylan, 2021; Kumar et al., 2022). The kernel-based quantum CNN model has been implemented to diagnose pneumonia early. The hybrid model can detect pneumonia from chest X-ray images obtained from a public repository. High classification accuracy has been obtained with the inclusion of a quantum model (Tayba et al., 2022). A parameterized circuit-based quantum deep convolutional neural network (QDCNN) model has been proposed in another study to classify image emotions. Quantum-classical training has been implemented through variational quantum algorithms. Parameters have been updated through QDCNN, and complexity has been analyzed using GTSRB and MNIST datasets to evaluate validity and feasibility (Li et al., 2020).

Tensorflow quantum-based (Lazzarin et al., 2022) QCNN models have been implemented for binary image classification. Box-counting-based fractal features, multi-scale entanglement, and the renormalization ansatz model have been used for downscaling, followed by classification through hybrid QCNN on the breast cancer dataset (Chen et al., 2022). Particle swarm optimization with binary encoding (BQPSO) based on quantum principles has been adopted to perform binary encoding of image emotions. A CNN model has been used to classify the features extracted from the hybrid model. The efficacy has been tested with seven benchmark datasets (Li et al., 2019). A quantum Hopfield network has been designed by combining quantum principles with traditional neural networks. The model has been applied to image recognition in a conventional computer, and its feasibility has been validated (Liu et al., 2020). Quantum Neural Networks (QNNs) have been evaluated for negational summary and binary classification in another algorithm on Google's quantum computing platform (Dong et al., 2022).

Moreover, the FER is considered critical for several implementations. However, existing studies have shown better results in facial recognition. Moreover, the FER systems have shown enhanced accuracy in ML and DL methods compared to the conventional FER methods (Borgalli and Surve, 2022).

2.1 Problem identification

Various problems identified from the extensive literature have been discussed as follows:

- ML algorithms for facial expression recognition suffer from dynamic head motion, illumination variants, and noise sensitivity. Moreover, spatial and temporal features have not been integrated in the study. Furthermore, the work has not considered facial deformation and geometric features (Alreshidi and Ullah, 2020).
- Deep CNN-based models can handle spatial features alone in the FER 13 dataset (Jain et al., 2019). The vanishing gradient problem has occurred with an increase in the number of CNN layers. Training CNN-based models such as VGG, ResNet, and Inception requires significant computational power and large datasets (Akhand et al., 2021).
- Feature extraction capability in conventional shallow CNN models has been limited in the case of high-resolution images (Li et al., 2019).

3 Proposed methodology

The proposed study aimed to recognize facial expressions by employing quantum computing alongside the ResNet-18 model and the MuS-BNP architecture. However, many existing studies have intended to perform facial expression recognition. The accuracy of the already existing study is less and needs further improvement. In the present study, the information present in the qubits has been manipulated so that it is capable of producing more quality solutions to complex problems quickly. Hence, it is clear that quantum computing has been used to address difficult problems. The classification of quantum images based on facial expressions using modified ResNet architecture is shown in Figure 1.

The FER 13 dataset has been loaded and preprocessed. The process of preprocessing transformed the raw data into a usable format. The transformed data were then split into training and testing sets. A train test split has been used for the model validation procedure, which stimulates the model's performance for new and unseen data, and the outcome of the train test split is trained data. The trained data was classified using the proposed ResNet-18 model with the MuS-BNP, which produces the trained model. Both the trained model and test data were used to predict the result. Performance metrics such as precision, recall, F-measure, and accuracy were used to assess the proposed model.

3.1 Quantum architecture

When QCF is exercised on an input tensor, a feature map is produced by each QCF due to the spatial transformation of local subsections present in the input tensor using QCF. However, in contrast to the modest element-wise matrix multiplication that traditional convolutional filters have applied, QCF has used a quantum circuit to transform structured and random input data. In the present study, a quantum circuit, which is randomly generated, has been used in QCF, which is different from the designed structure. By using QCF, the process can be formalized and transforms the classical data as mentioned below:

1. Single QCF, which used random quantum circuit 'q' and a local subsection of images, has been taken as input from the

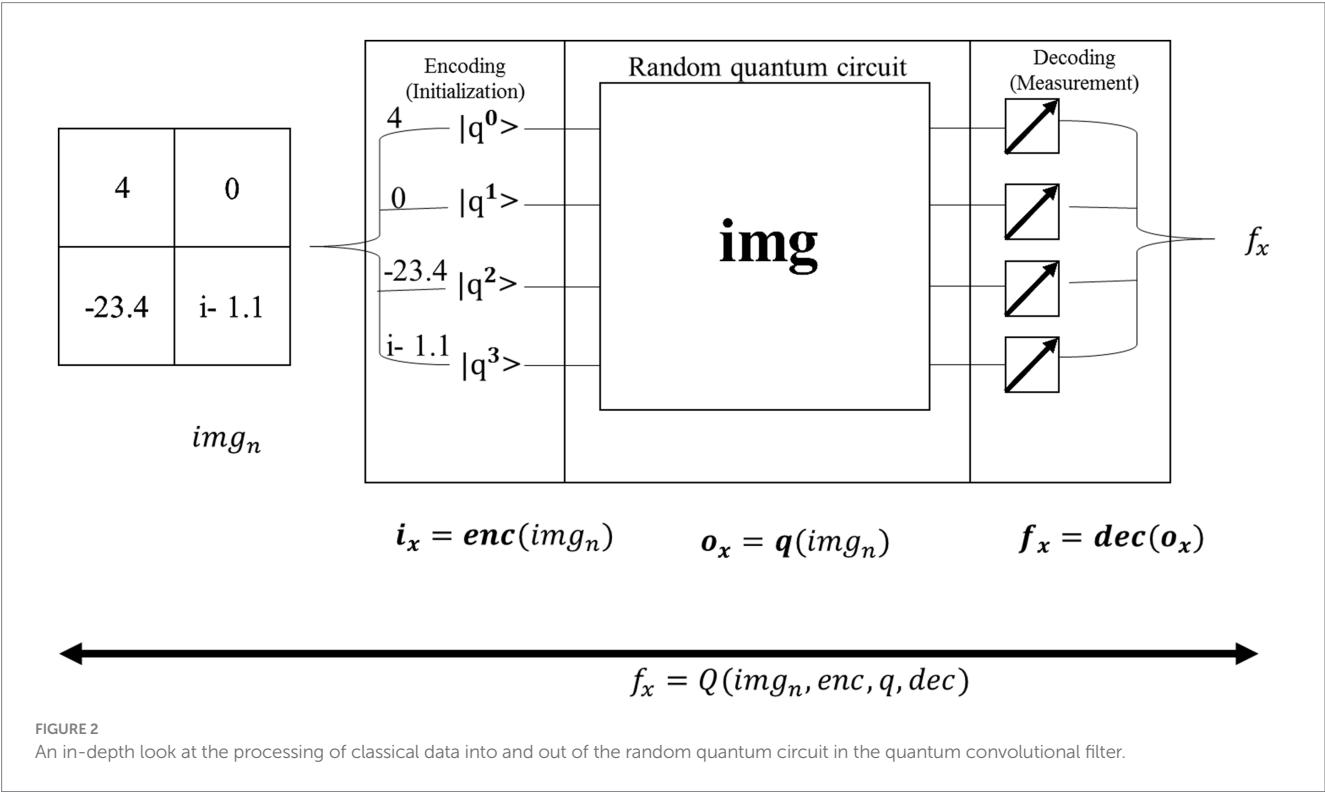
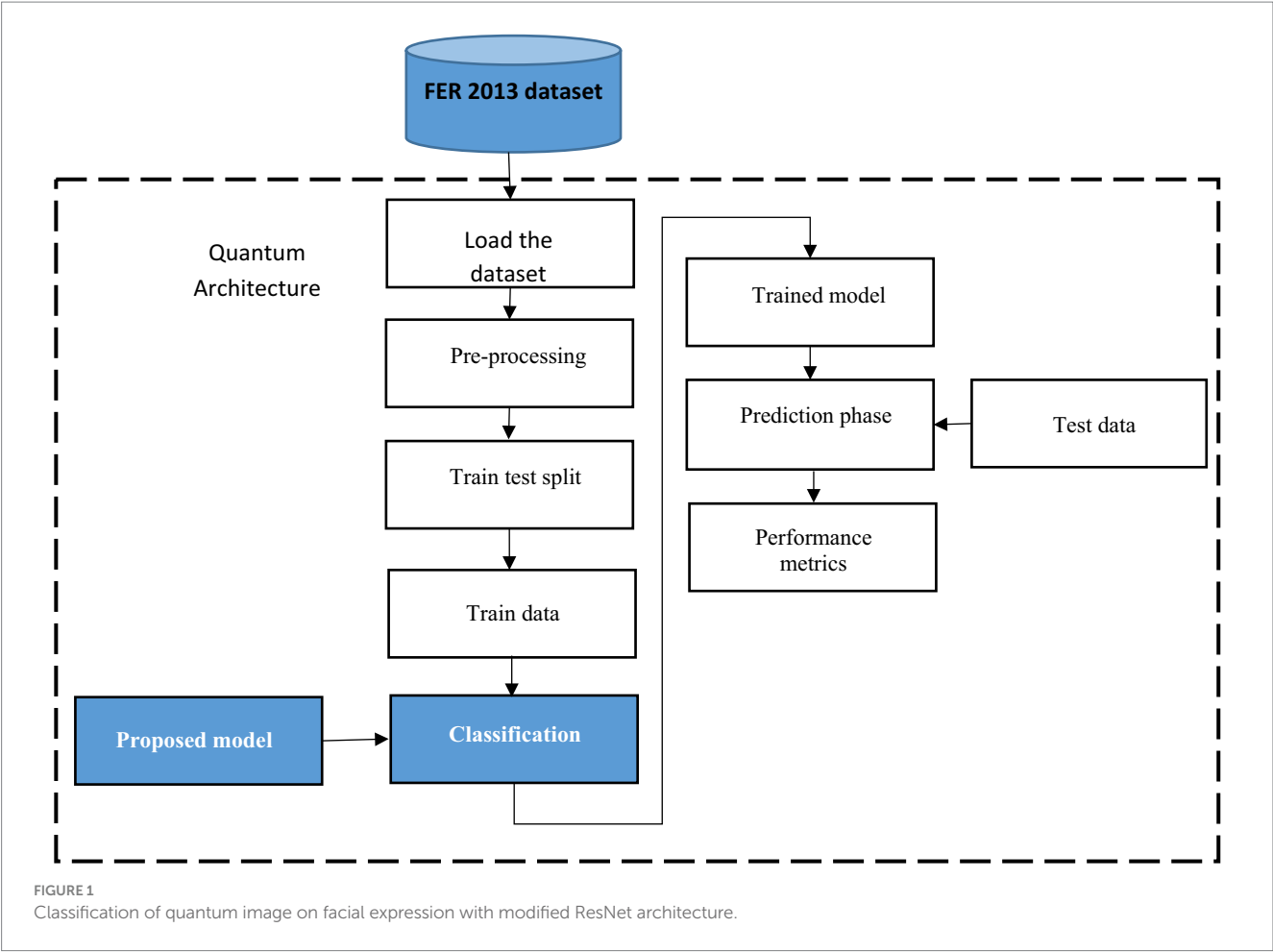
dataset u . Each input has been defined as u_x , and the matrix size of each u_x is n by n , where $n > 1$.

2. Though many ways are available to encode u_x at the initial state of q , for each QCF, one specific encoding function e has been chosen, the encoded initialization state ix as $ix = enc(img_n)$ has been defined.
3. After applying the quantum circuit to the initialized state ix , an output quantum state ox has been attained, which is the result of quantum computation where the relationship between ix and ox is given as $ox = q(ix) = q(enc(img_n))$.
4. Though many ways are available with a finite number of measurements to decode the information of ox , to confirm the consistency of QCF output with other similar output taken from regular classical convolution, the final decoded state has been given as $f_x = dec(ox) = dec(q(enc(img_n)))$ where d refers to the decoding function, and f_x refers to a scalar value.
5. The complete transformation of $dec(q(enc(img_n)))$ has been defined as QCF transformation at this point, in which Q of u_x , aka $f_x = Q(img_n, enc, q, dec)$. A single QCF visualization has been shown in Figure 2, which exhibits the process of encoding, applied circuits, and decoding.
6. The number of classifications that happened when the classical convolutional filter was applied as an input from dataset u , the required number of computations is given as $O(n^2)$, placing the computational complexity squarely in P . It is not considered in the case of computational complexity Q . It has emerged from the complexity of random quantum circuit transformation q , where e and d show efficient performance on classical devices. Figure 2 illustrates the step-by-step QCF procedure in detail.

The present study has highlighted the novelties obtained from the QNN algorithm: the quantum convolutional layer generalizability inside a usual CNN architecture, the quantum algorithm's ability to be used on practical datasets, and the efficient use of features presented by quantum convolution transformation. Later, research was conducted in the field of using quantum circuits in ML applications, in which randomly parameterized quantum circuits were used to process classical data and linear models were trained using the output. Quantum transformations have built the model and shown more benefits in comparison with further linear models, which are directly built on the dataset itself, but the level of performance is not the same when compared with other classical models. The experiments in the present study have been built on these results, in which quantum feature detection has been integrated into more difficult neural network architecture since the QNN framework introduced classical models that contain non-linearities.

3.2 ResNet18 architecture with modified up-sampled bottleneck process

A residual network employs residual blocks, which allow additive interaction between the input and output present in the two convolutional layers. The advantage of ResNet is given as a gradient that flows directly on identity function from future layers to past layers, which has partially solved the disappearing gradient problem. To improve the flow of information between the layers, original blocks



replace the cascade blocks. Two Conv-BatchNorm-ReLU layers are used to build every cascade block, two in-out lines, and a shortcut connection line. However, the deep layer network contains many feature map inputs. To increase computational efficiency, the cascade block has been modified into a cascade bottleneck block, which uses four four-layer stacks instead of two.

In the present research, residual connections have been used to link the filter connection stage in the ResNet-18 model with the MuS-BNP. Therefore, the architecture allows for the maintenance of computational efficiency, which attains the advantages of the residual connection process. A residual version of the ResNet-18 model with the MuS-BNP has used a more simplified module. The filter expansion layer follows each module in which the dimensions of the filter bank have been enlarged. To match the input, it has been added before. Hence, it reimburses the reduction of dimensionality available in the n block.

Feature extraction was done through the quantum convolutional layer, which is composed of several parameterized quantum filters. Like the convolution kernel present in the traditional convolutional layer, the parameterized quantum filter has been used to extract the information present in every quantum bit, which exists in the data local space. A quantum filter consists of a double-bit gate in which quantum bit unitary conversion can be performed, and a double-bit gate is enforced on neighboring quantum bits, which leads to quantum entanglement present in neighboring quantum bits. In the image, the pixel value of the information has been changed into quantum state information (which uses quantum state encoding) using quantum rotation gate $R(\theta)$. Based on the process, the information attained about the features of the image has been altered to the angle of the quantum rotatory gate. Each pixel value has provided the corresponding parameters for the quantum rotatory gate. The quantum bit initial state $|0\rangle$ has been acted by different quantum rotatory gates, and the quantum state stores the feature information. It can be utilized as model input to QNN. For instance, by considering $n \times n$, initially, the function of quantum feature extraction is encoded into the quantum state by coding the quantum bit. Furthermore, the quantum state has evolved by using a parameterized quantum circuit and, finally, by using expected value measurement outputs a real number. The method possesses both exclusive quantum mechanics properties and retains the sharing of weights in the convolutional kernel. Figure 3 shows the quantum convolutional layer.

The present study has introduced the quantum circuit with parameters to enhance the network's performance. Quantum filters include a rotary gate R_θ and a CNOT gate. Figure 4 shows the quantum circuit diagram.

ResNet has been used in computer vision applications as a DL model. Many convolutional layers have been supported by CNN architecture. ResNet-18 is a CNN that consists of 18 layers deep. The vanishing of the gradient has been improved by using the network. The improved algorithm has used ResNet-18. The existing study has optimized the input present in the network. The input features were extracted in parallel, and feature fusion was performed at the termination of the parallel structure. A specific method has been used to accept the three parallel routes. In the convolutional operations present in the multi-feature fusion, to confirm the integrity of the input image size, the step has been set to 1.

Figure 5 has been used to better understand the process. Similarly, when applying the initial residual unit, the number of feature layers

is increased, and a better interpretation of dimensionalities is presented. In the end, the outcomes of three parallel routes were used for feature fusion, which extracts the features of the image and, in turn, improves the performance of the proposed model. The proposed QNN efficiently utilizes quantum hardware and reduces the number of quantum gates needed for a particular calculation. Moreover, the model outperforms traditional algorithms in identifying complex image features, improving classification accuracy and reliability. It also performs tasks on several qubits simultaneously, allowing for efficient parallel processing of image feature datasets. Figure 6 illustrates the modified up-sampled bottleneck process with the ResNet-18 architecture.

To prevent gradients from vanishing and exploding, the residual gradient structure has been used. Feature reuse is helpful for feature extraction, and residual units have been improved. During the feature extraction process, 128×128 feature information is present as the first residual block output, which has been given as the input for the 3rd residual block using downsampling, and the input scale has been changed to 75×75 . Similarly, the first residual block output feature information has been sent as input, multiple downsampling has been used for the fourth residual block output, and feature size has been given as 38×38 and 19×19 , respectively. The method that was used in the 1st residual block was the same as the second residual block output, which was 50×50 . The subsampled output has been given to the input and output present in the fourth residual block. The residual block output is subsampled, and it has been given to the fourth residual block output. The complete representation of the modified up-sampled bottleneck process is shown in Figure 7.

In the proposed method, every block has an independent convolutional way to deliver the information present in the previous and middle layers. The strategy exhibits the concept of "pass-over," which has been varied from ResNet, which loads the modest building blocks that contain residual connections. The classical residual building block does not use the information available in the middle layer. However, the proposed model has cached the pass-over information to obtain complete features.

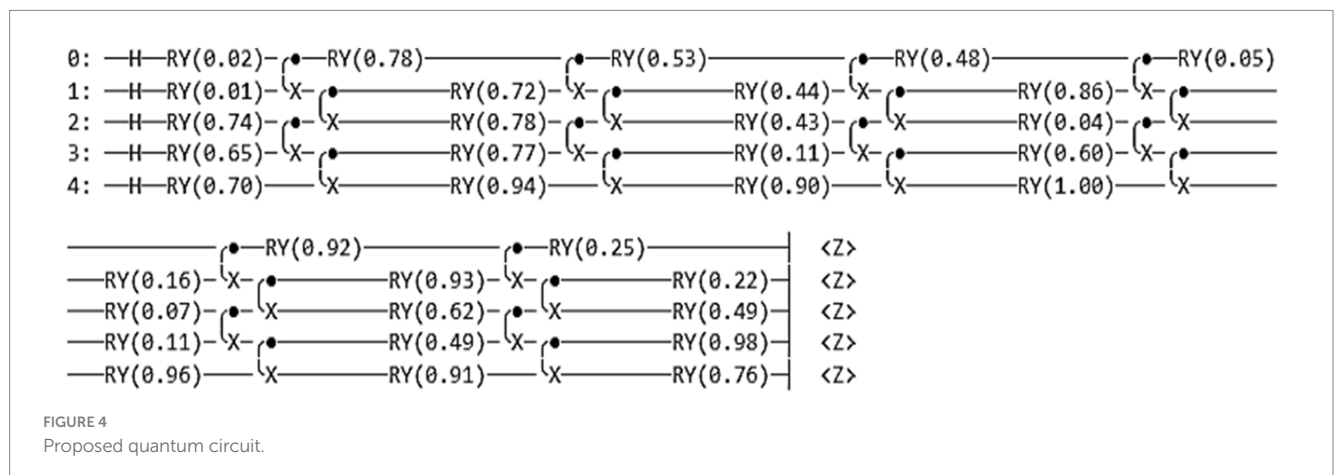
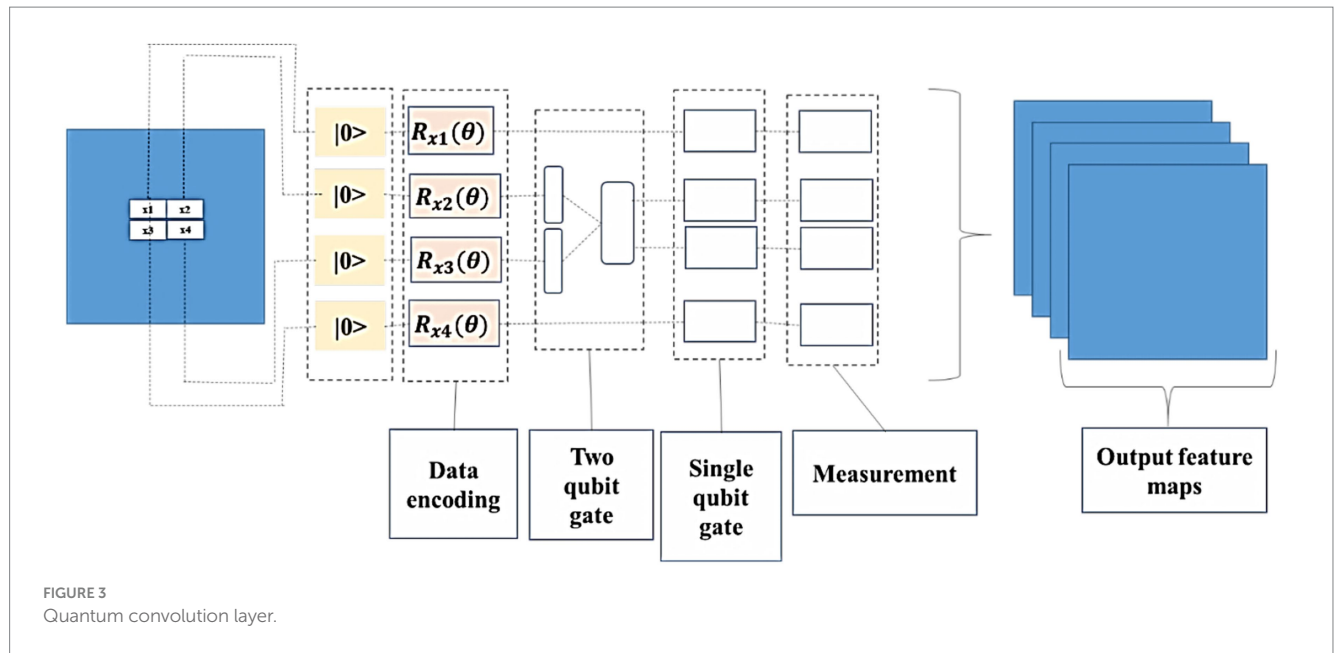
The proposed model structure has been designed to achieve many features. The pass-over way leads to various feature fields, which generate features at various levels of abstraction. Moreover, it supported the ensemble effects and showed improved performance in classification.

Proposed general form of function, given in Equations 1, 2:

$$g(x) = \text{softmax}(\quad) \quad (1)$$

$$g(a) - h(a) = \begin{cases} a \cdot \frac{\ln(e^a + 1)}{1 + \ln(e^a + 1)}, a \in (-\infty, 0); \\ k_{m+1}a + b_{m+1}, a \in [0, a_{m+1}); \\ \vdots \\ k_na + b_n, a \in [a_n, \infty). \end{cases} \quad (2)$$

During the training process of CNN, it was observed that the piecewise point of activation function was set between values of 0 and



1, greatly influencing the backward propagation of gradient, forward propagation of feature, and curve change. At point 0, the function has differentiated, and the slope of the function has been changed to 1 immediately. After conducting many tests, the piecewise function has been set as 0.1, and the function is given below in Equation 3:

$$\left\{ \begin{array}{l} a \cdot \frac{\ln(e^a + 1)}{1 + \ln(e^a + 1)} a\varepsilon(-\infty, 0); \\ a \cdot \frac{\ln 2}{1 + \ln 2} a\varepsilon[0, 0.1); \\ a + \frac{0.1 \ln 2}{1 + \ln 2} - 0.1, a\varepsilon[0.1, +\infty); \end{array} \right. \quad (3)$$

At the initial stage of the test, the model exhibited overfitting directly. It was observed that the slope of the function altered quickly, and the transition of the curve's slope from $\ln 2 / (1 + \ln 2)$ to 1 could not occur directly.

To address this, a linear function was introduced at the range (0.1, 1), acting as a buffer to stabilize the slope changes. After extensive testing, the optimal range was refined to (0.1, 0.2), which effectively mitigated the overfitting issue while preserving the model's performance.

The modified function is as follows in Equation 4:

$$\left\{ \begin{array}{l} a \cdot \frac{\ln(e^a + 1)}{1 + \ln(e^a + 1)} a\varepsilon(-\infty, 0); \\ a \cdot \frac{\ln 2}{1 + \ln 2} a\varepsilon[0, 0.1); \\ \left(2 - \frac{\ln 2}{1 + \ln 2} \right) x + \frac{0.2 \ln 2}{1 + \ln 2} - 0.2, a\varepsilon[0.1, +0.2); \\ a, a\varepsilon[0.2, +\infty). \end{array} \right. \quad (4)$$

The mean value outcome of ReLU has been compared with a new function, and the probability model of the parameter has been set as $p(a, \alpha)$, a^+ refers to the positive input, a^- refers to the negative input, α refers to the probability of input a . The new function output

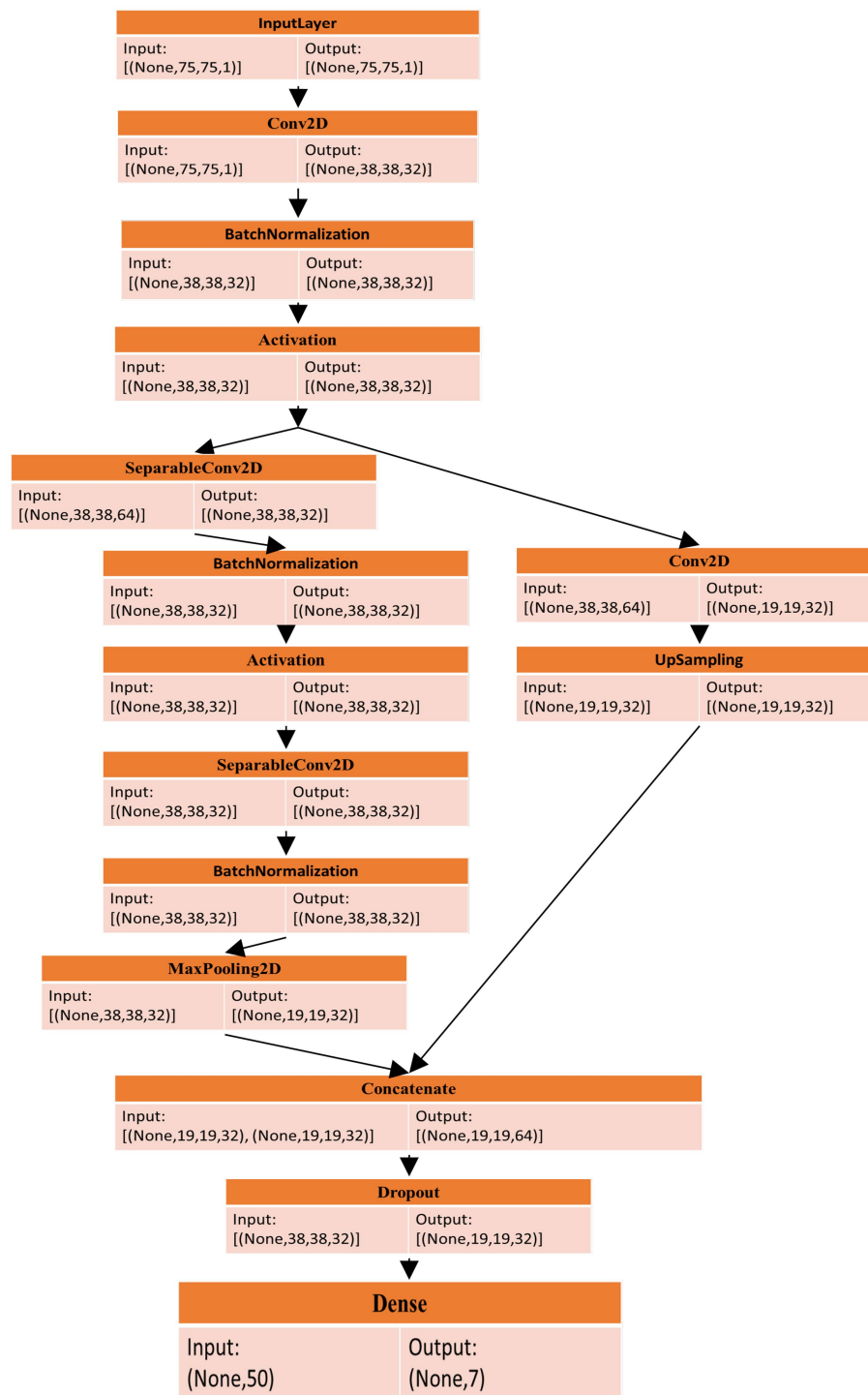


FIGURE 5
The flow of the ResNet-18 model with the MuS-BNP.

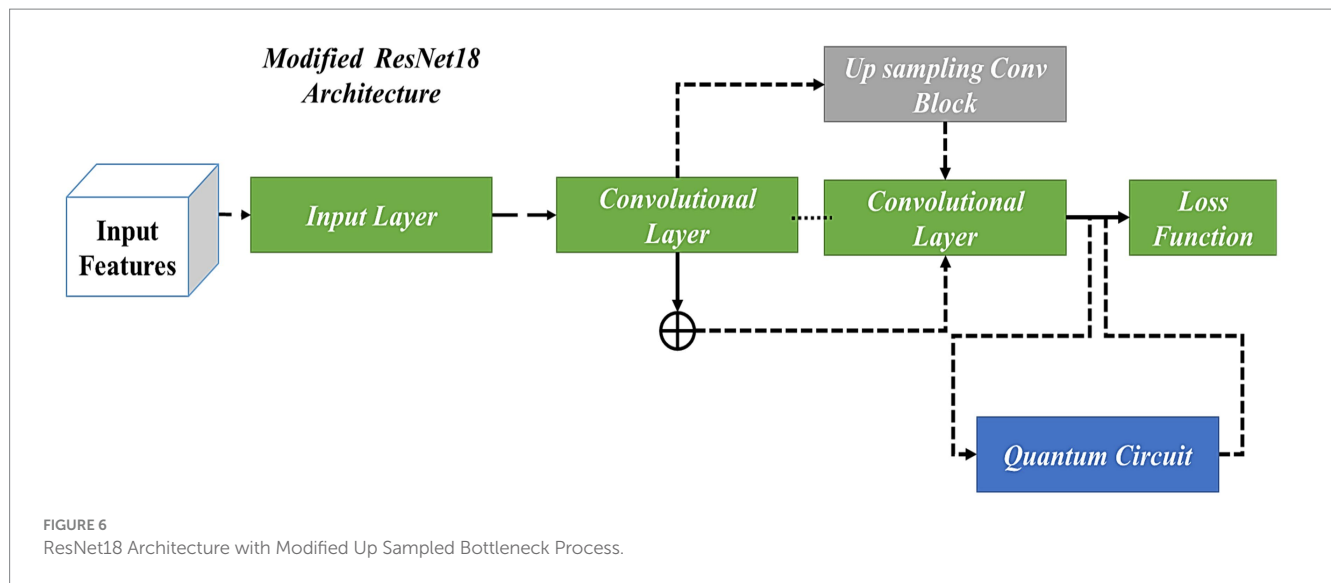
mean value after non-linear transformation is given as follows in Equations 5–8:

$$E_{ours}(a) = \sum \beta f_{ours}(a) = \sum \beta a^+ + \sum \beta a^-, \quad (5)$$

where

$$\sum \beta a^+ = \sum \beta a^+ + \sum \frac{\beta \ln 2}{(1 + \ln 2) a^+} + \sum \beta \left[\left(2 - \frac{\ln 2}{1 + \ln 2} \right) a^+ + \frac{0.2 \ln 2}{1 + \ln 2} - 0.2 \right] \quad (6)$$

$$\sum \beta a^- = \sum \beta a^- \cdot \ln(\exp[a^-] + 1) / (1 + \ln(\exp[a^-] + 1)) \quad (7)$$



The output mean value of ReLU is

$$E_{ReLU}(a) = \sum \beta f_{ReLU}(a) = \sum wa^+ + 0 \quad (8)$$

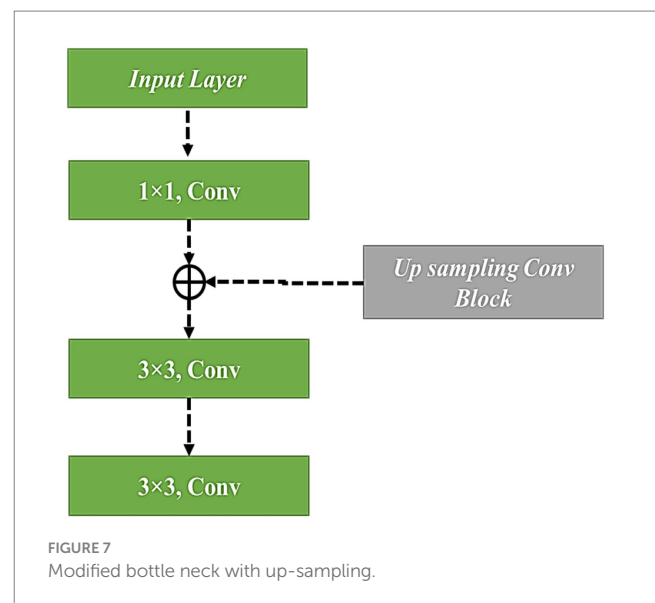
where, $E_{ReLU}(a)$ always has a positive value and the result of the new function $E_{ours}(a)$ has both +ve and -ve values that make the mean value close to 0. It has accelerated the convergence of the model and updated parameters.

Figure 8 illustrates the workflow of the proposed QCNN, where QNNs utilize quantum convolution layers and activation layers to extract features from the input images. The process begins with data encoding, converting actual images into the required quantum state. Quantum convolution is achieved by applying a series of quantum gates to the encoded state. The process continues through quantum pooling and fully connected layers, where neurons are interconnected in a feed-forward configuration, linking preceding neurons with subsequent ones. The model's performance is evaluated, and the final quantum state is delivered as the output result.

However, integrating conventional CNN with the QCNN framework creates a hybrid model that capitalizes on the strengths of both technologies. This approach diverges from usual QCNN formats, venturing into new areas of neural network configurations as an experimental model. Furthermore, utilizing a quantum simulator to run the model and generate results represents significant progress in the practical applications of QML. The findings from the proposed study indicate that employing a quantum strategy yields superior outcomes compared to traditional techniques, as demonstrated by improved precision rates when examining face images. These findings contribute to the growing knowledge of QML, opening the door to further research and experimentation, including the application of quantum methods to tackle more complex tasks.

4 Results and discussion

The results that have been obtained by implementing the proposed system are included in this section, along with a dataset description,



performance metrics, experimental results, performance analysis, and comparative analysis.

4.1 Dataset description

The study used the FER-2013 dataset, which consists of greyscale images, each with dimensions of 48*48 pixels. The images are automatically registered, meaning the faces are generally centered, and each image occupies a consistent volume of space. The goal of the study was to classify the emotions displayed in the facial expressions into one of seven categories: Neutral, Surprise, Sad, Happy, Fear, Disgust, and Angry. The dataset includes approximately 28,709 examples in the training set and 3,589 examples in the public test set. The dataset was sourced from <https://www.kaggle.com/datasets/msmbare/fer2013>.

The total images that are considered in the FER-2013 dataset are tabulated in Table 1 with sample images as shown in Figure 9.

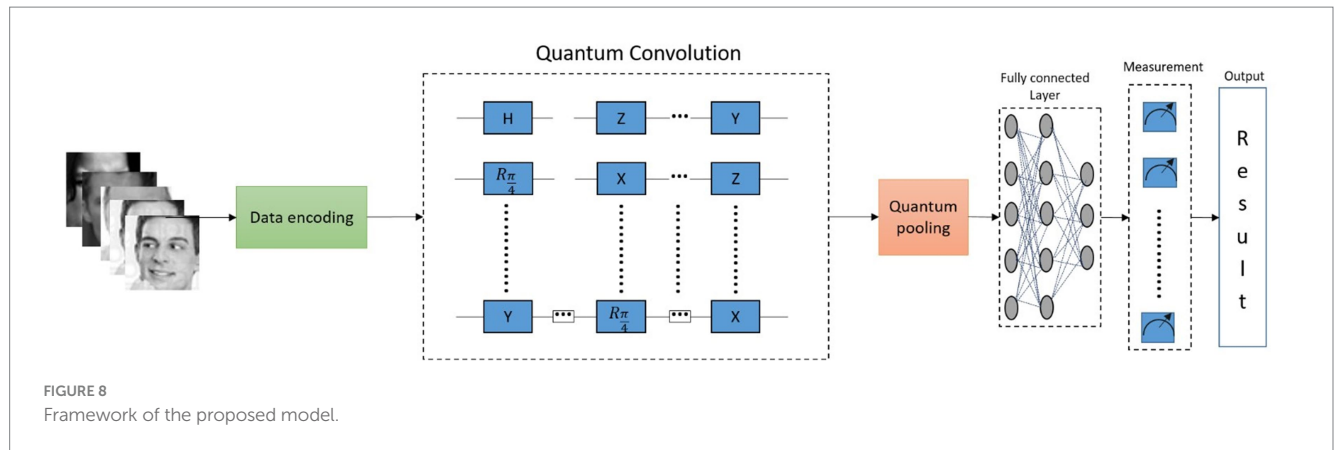


TABLE 1 Total images in the FER-13 dataset.

FER2013	Total number of images
Anger	4,953
Happy	8,989
Disgust	547
Surprise	4,012
Neutral	6,198
Sad	6,077
Fear	5,121

$$F - \text{measure} = \frac{2 * \text{Rec} * \text{Prec}}{\text{Rec} + \text{Prec}} \quad (11)$$

iv) Recall

The term recall quantifies the amount of correct positive classifications made out of all the positive classifications that are done. It is computed with the following Equation 12:

$$(\text{Rec}) \text{ Recall} = \frac{\text{Tru}_{\text{po}}}{\text{Fal}_{\text{Ne}} + \text{Tru}_{\text{po}}} \quad (12)$$

4.2 Performance metrics

Performance metrics are generally used to determine the performance of the proposed model, which is calculated based on the accuracy, precision, recall, and f1-score. Performance metrics are also used to determine the proposed model's efficiency.

i) Accuracy

The term accuracy can be referred to as the model classification rate that is provided through the proportion of correctly classified instances ($\text{Tru}_{\text{po}} + \text{Tru}_{\text{Ne}}$) to the sum of instances in the dataset ($\text{Tru}_{\text{po}} + \text{Fal}_{\text{po}} + \text{Tru}_{\text{Ne}} + \text{Fal}_{\text{Ne}}$). The succeeding equation can be used to estimate the accuracy range as given in Equation 9:

$$\text{Accuracy} = \frac{\text{Tru}_{\text{Ne}} + \text{Tru}_{\text{po}}}{\text{Tru}_{\text{Ne}} + \text{Tru}_{\text{po}} + \text{Fal}_{\text{Ne}} + \text{Fal}_{\text{po}}} \quad (9)$$

ii) Precision

The term precision is defined as the degree of covariance of the system, which results from the correctly identified instances Tru_{po} to the total number of instances that are correctly classified ($\text{Tru}_{\text{po}} + \text{Fal}_{\text{po}}$). It is measured by Equation 10:

$$\text{Precision} = \frac{\text{Tru}_{\text{po}}}{\text{Tru}_{\text{po}} + \text{Fal}_{\text{po}}} \quad (10)$$

In this equation, the variables are defined as Fal_{Ne} -False Negative, Fal_{po} -False Positive, Tru_{Ne} -True Negative, and Tru_{po} -True Positive.

iii) F-Measure

F1-score denotes the weighted harmonic mean value of (Rec) recall and (Prec) precision. It is calculated with the following Equation 11:

4.3 Exploratory data analysis (EDA)

In general, EDA indicates the critical procedure of performing primary investigations on the data, realizing patterns, verifying assumptions, and spotting anomalies with the help of graphical representations and summary statistics. This section deliberates on the EDA of the proposed models in the present study for the datasets FER-13. The training and test data for different emotions are mentioned in Figure 9 for better understanding.

For the FER-2013 dataset, sample images for some common emotions like happy, neutral, disgust, sad, angry, fear, and surprise have been shown in Figure 10. Based on the images in the dataset, the emotions are classified.

The test data for the FER-2013 dataset for the mentioned emotions, such as neutral, disgust, fear, anger, sadness, surprise, and happiness, has been shown in the graphical representation in Figure 11 to obtain more clarity.

The considered train and test data for the FER-2013 dataset for the mentioned emotions like neutral, disgust, fear, anger, sad, surprise, and happy has been shown in the graphical representation in Figures 12, 13.

4.4 Experimental results

The test results for the proposed model are shown in Figure 14. The proposed system, which used quantum computing and the ResNet18 architecture with modified-Up Sampled Bottle Neck Process for the FER-2013 dataset, produced the exact predictions. Figures 14, 15 clearly show that the original emotion and predicted emotions are

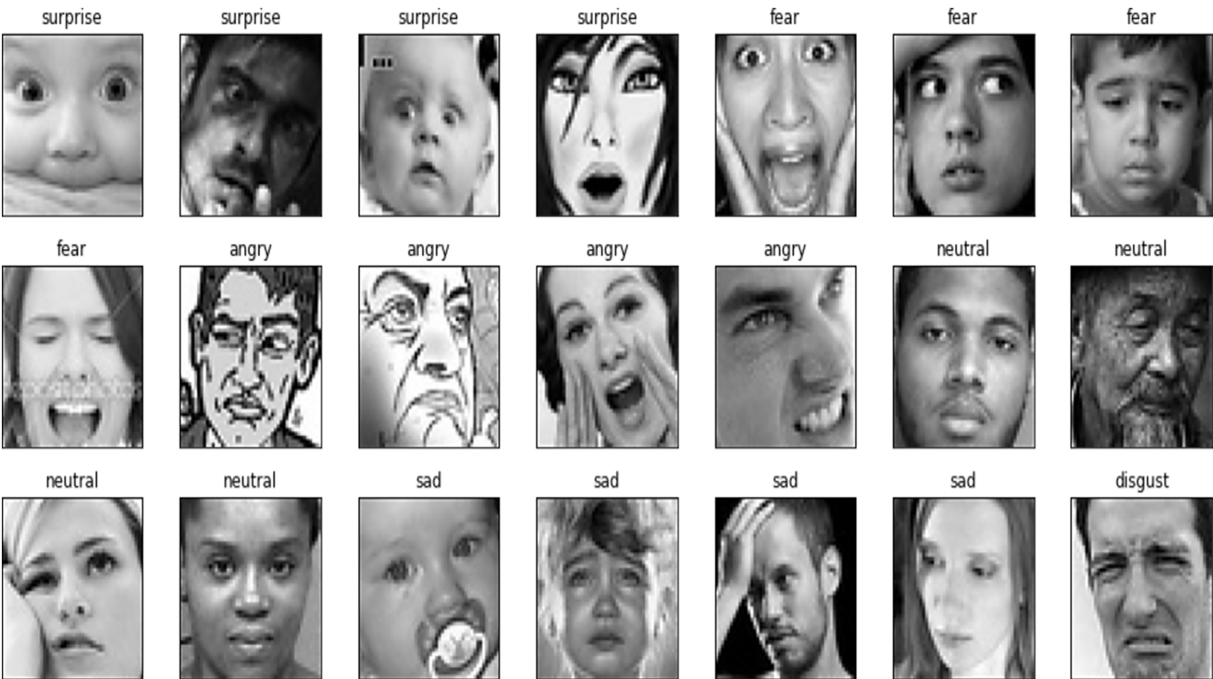


FIGURE 9
Sample images from the dataset.

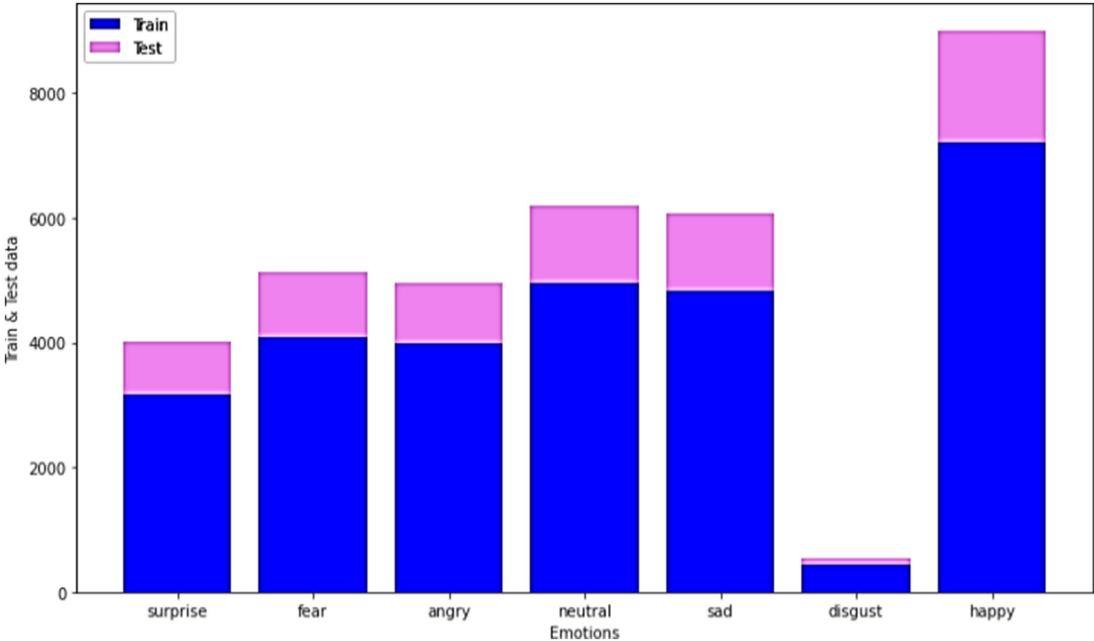


FIGURE 10
Train and test data for the FER-2013 dataset for different datasets.

the same. Thus, the proposed method recognizes facial emotions with utmost accuracy. The proposed method has classified the emotions into seven categories: neutral, surprised, sad, happy, fearful, disgusted, and angry. From Figure 6, it is clear that the proposed method has predicted all seven emotions correctly. On the contrary, the misclassification results are shown in Figure 15.

From Figure 16, it was found that the misclassification rate of the proposed model was 5 for the original 2.

4.4.1 Statistical tests

Distribution tests have been considered in this case. When the dataset pursues normal distribution, it could be found that most of the

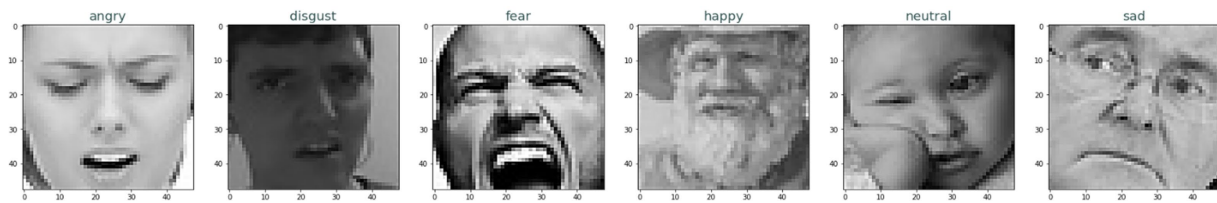


FIGURE 11
Sample images for the FER-2013 dataset with different emotions.

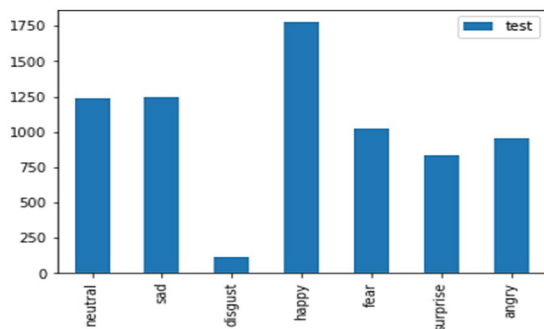


FIGURE 12
Test data for the FER-2013 dataset.

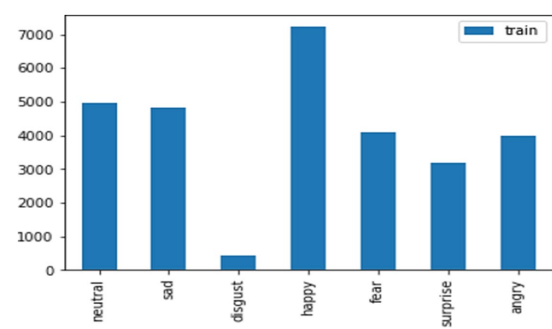


FIGURE 13
Train data for the FER-2013 dataset.

images fall within a certain SD (standard deviation) of the mean. When distribution seems to be not normal, it might be found that distribution is either skewed or possesses a heavy tail. Additionally, it is probable to evaluate if the dataset approximately pursues normal distribution with the creation of a data histogram and a visually performed inspection. Typically, a normal distribution possesses a bell-shaped curve with most of the data points clustered about the mean. When it has been assumed that FER-13 is a persistent variable (for instance, facial expression intensity), then a data histogram could be created and visually inspected for normality. When the histogram roughly pursues a bell-shaped curve, this could recommend that the dataset pursue a normal distribution. The corresponding histogram plot is shown in Figure 17.

In addition, the Shapiro–Wilk test was undertaken, which is a statistical test utilized to determine if sample data is typically distributed or not distributed. Moreover, the proposed work has used the FER 2013 BENCHMARK dataset, and the results for Shapiro–Wilk test statistics corresponding to the proposed work give 0.9844 with a p -value equal to 0, and it is clearly found that pixel values are not normally distributed.

4.5 Performance analysis

The performance of the proposed system has been analyzed, and the corresponding outcomes are discussed in this section.

Figure 18 shows the confusion matrix for the proposed model, illustrating the accuracy of emotion predictions. The model has successfully predicted the true labels, with “surprise” being the most accurately predicted emotion (1755 instances). In contrast, the predictions for other emotions were as follows: “neutral” (1251), “sad” (1243), “disgust” (994), “anger” (967), “fear” (762), and “happy” (96), with “happy” being the least predicted emotion. This analysis

reveals that “surprise” was the most frequently and accurately identified emotion, while “happiness” had the fewest correct predictions. Moreover, Figure 19 represents the accuracy analysis of FER-2013 and shows both trained and validated accuracy.

From Figure 19a, it is clearly visible that both train and validated accuracy have some differences until epoch 10. Train and validated accuracy have a closer match on 20, 25, and 30 epochs. Moreover, from Figure 19b, it is clearly found that both train and validated loss have some differences in epoch 0 and epoch 5. In 10, 15, 20, 25, and 30 epochs, both train loss and validated loss have a closer match. Figure 20 visualizes the performance curves of precision-recall and receiver operating characteristics (ROC) of the proposed model on the FER-2013 dataset.

Figure 20a shows that the proposed model achieved an AUC value of 0.99 for the Precision-Recall curve for surprise, neutral, sad, and fear; 0.98 for disgust and anger; and 0.82 for happiness. The AUC curve confirms that surprise, neutral, sad, and fear have achieved high values, whereas happiness had lower prediction accuracy for the FER-2013 dataset.

Figure 20b shows that the ROC curve reached a value of 1.00 for anger, disgust, neutral, sad, and fear; 0.99 for surprise and happiness. Moreover, the performance metrics of the proposed model are tabulated in Table 2.

For instance, the proposed model demonstrates strong performance in detecting emotions such as anger, disgust, surprise, neutral, sad, and fear, achieving precision, recall, and F1 scores close to 0.99 for each, indicating high accuracy and consistency in predicting these emotions. However, for the “happy” class, the model exhibits a distinction with a precision accuracy of 0.97 but a reduced recall rate of 0.84, leading to a slightly lower F1-Score of 0.90.



FIGURE 14
Experimental results for correct classification of the proposed model.



FIGURE 15
Experimental results for the correct classification of the proposed model.



FIGURE 16
Experimental results for misclassification of the proposed model.

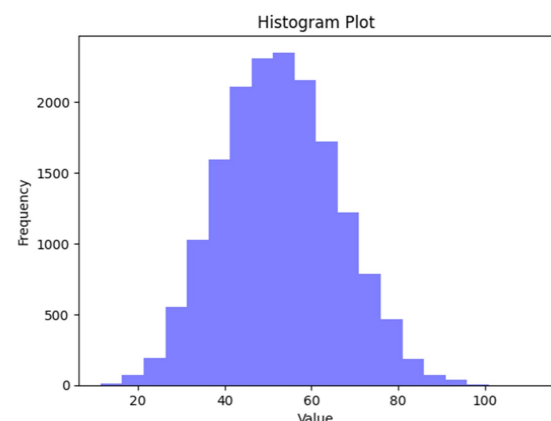


FIGURE 17
A histogram plot.

Moreover, it shows that while the model is generally accurate in predicting happiness, it fails to account for a significant number of actual happy instances. These metrics highlight the model's strengths in most emotional categories but highlight the need for improvement in predicting happiness.

Additionally, the model achieved a kappa coefficient of 0.9899, an overall accuracy of 0.99, a macro average of 0.99 for precision, 0.97 for recall, and 0.98 for the F1-score. The weighted averages for precision, recall, and F1-score were all 0.99, further confirming the model's robust performance.

Based on the performance analysis, the performance of the proposed system that has used quantum computing is found to be more efficient. In order to gauge its outstanding performance, the proposed system was compared with the conventional system, for which a comparative analysis was carried out. The results are discussed in the succeeding section.

4.6 Comparative analysis

The proposed system has been compared with four conventional studies, and the respective results are discussed in this section. The existing study has used various models such as DCNN Model1, DCNN Model2, EmNet (average fusion), and EmNet (weighted maximum fusion), and their corresponding outcomes are given in Table 3.

When compared with the existing study, we can observe that the proposed model has attained a higher accuracy of 98.19%, which is clearly shown in Table 3. The existing study (Zahara et al., 2020) has been compared with the proposed model, which used quantum computing, and the outcomes are 65.97% accuracy for the existing model and 98.19% for the proposed model. Hence, it

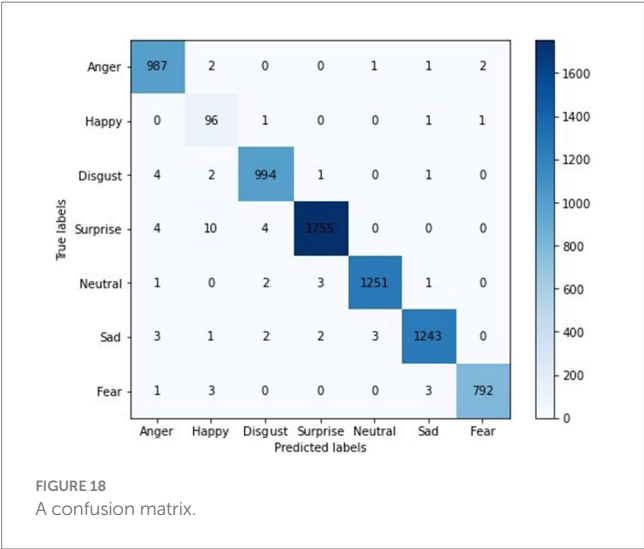


FIGURE 18
 A confusion matrix.

is clear that the proposed model has better accuracy, as shown in Table 3.

The train and test accuracy of the proposed method has been compared with the existing study (Bodavarapu and Srinivas, 2021), which has used various models like FERConvNet_Gaussian, FERConvNet_Nonlocal Means, FERConvNet_Bilateral, and FERConvNet_HDM, and the outcomes are shown in Table 4.

From Table 4, it is clear that the proposed method has attained higher train accuracy at 99%, and the test accuracy value is given as 98%, compared with the existing methods used in the existing study.

The performance metrics of the proposed method, which used quantum computing, have been compared with the existing study (Kim et al., 2021), which has used the SGD and Adam models, and it is shown that the proposed model achieves 98.19% of accuracy, 98% of precision, recall, and f1_score, compared with 76.17 and 77.17% of accuracy, 63.0118 and 66.6236% of precision, 61.0729 and 66.8845% of recall, as well as 61.0932 and 66.6779% of f1_score, respectively, for

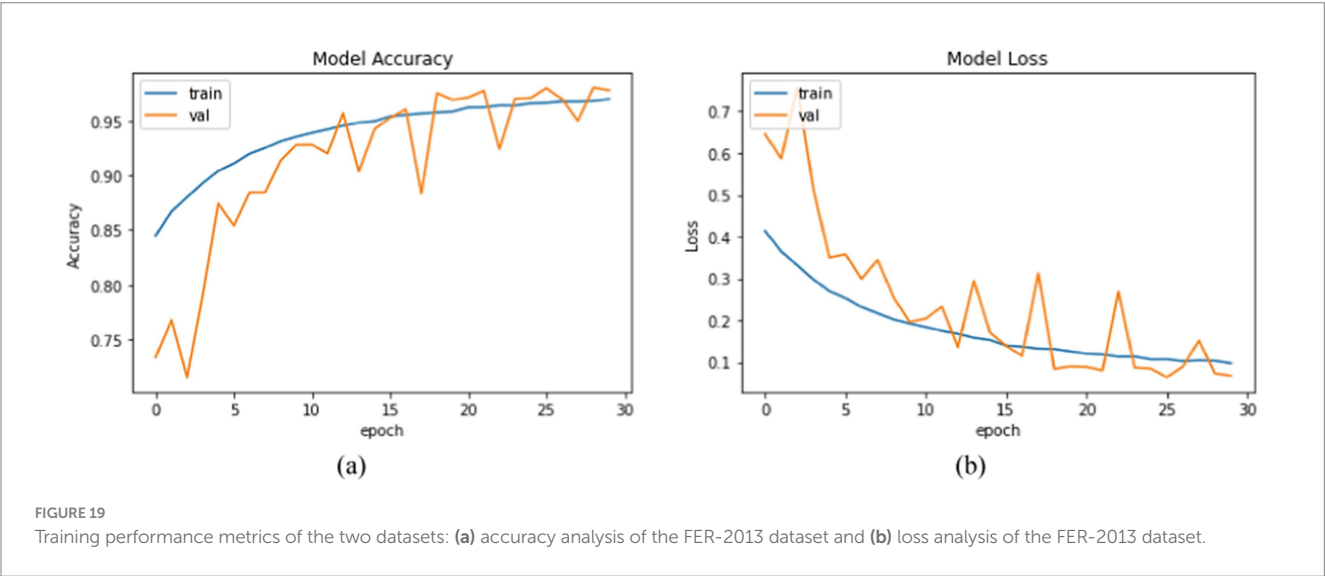


FIGURE 19
 Training performance metrics of the two datasets: (a) accuracy analysis of the FER-2013 dataset and (b) loss analysis of the FER-2013 dataset.

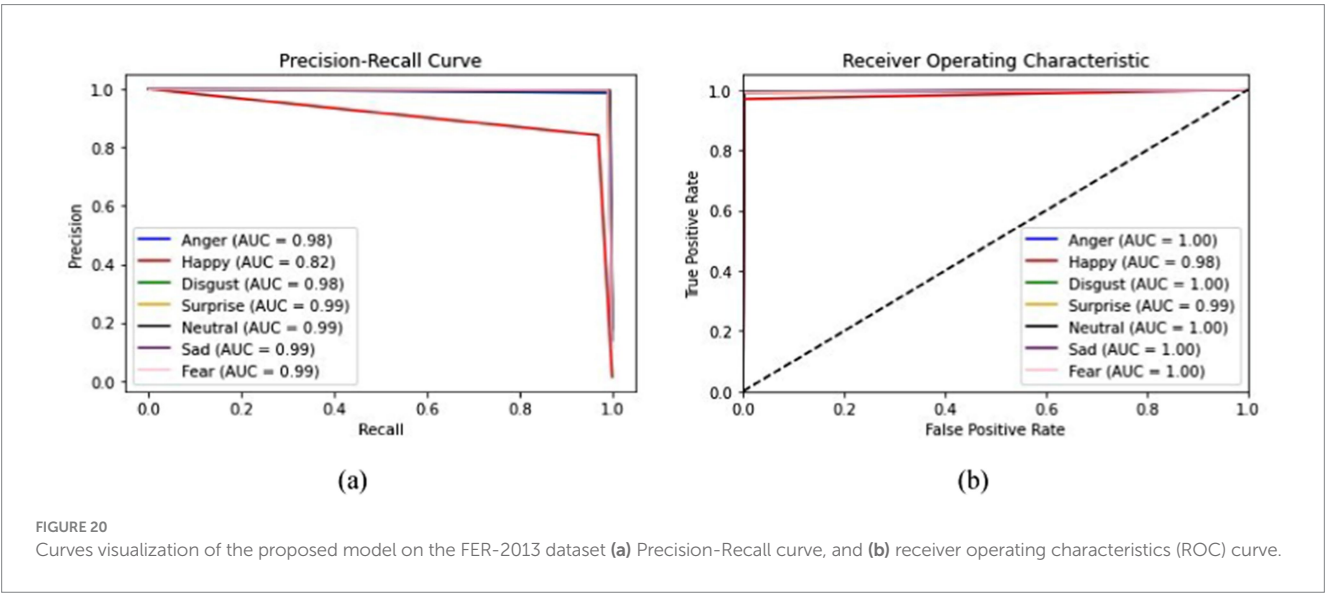


FIGURE 20
 Curves visualization of the proposed model on the FER-2013 dataset (a) Precision-Recall curve, and (b) receiver operating characteristics (ROC) curve.

TABLE 2 Performance metrics of the proposed model.

Class	Precision	Recall	F1-score
Anger	0.99	0.99	0.99
Happy	0.97	0.84	0.9
Disgust	0.99	0.99	0.99
Surprise	0.99	1	0.99
Neutral	0.99	1	1
Sad	0.99	0.99	0.99
Fear	0.99	1	0.99
Accuracy			0.99
Marco Avg	0.99	0.97	0.98
Weighted Avg	0.99	0.99	0.99

TABLE 3 Comparative analysis of accuracy (Saurav et al., 2021).

Model	Accuracy (%)
DCNN Model 1	72
DCNN Model 2	72.02
EmNet (average fusion)	74.11
EmNet (weighted maximum fusion)	74.06
Proposed	98.19

the SGD and Adam optimizers. Hence, it is clearly found that the proposed method has higher values in all performance metrics. Furthermore, a comparison has been undertaken between proposed and conventional methods by considering the JAFFE dataset. The respective outcomes are shown in Table 5.

From Table 5, it can be observed that existing algorithms such as VGG-16 have revealed an accuracy rate of 97.62%, DenseNet-161 has exposed an accuracy of 99.52%, and the Inception-v3 algorithm has shown 99.05% accuracy. However, the proposed model has explored a high accuracy rate of 99.68%. Similarly, the proposed system has been compared with conventional models for the CK+ dataset (Shanthi and Nickolas, 2021), and the corresponding outcomes are 97.86% for the existing model and 98.19% for the proposed model. Hence, it can be concluded that the proposed model has been confirmed to be more effective than conventional models when considering challenging datasets like the CK+ dataset and the JAFFE dataset. Hence, from the experimental results, performance analysis, and comparative analysis, it is clearly shown that the proposed model, which used quantum computing and ResNet18 Architecture with Modified Up Sampled Bottleneck Process, shows enhanced performance with higher accuracy due to effective feature extraction.

5 Discussion

The study (Bursic et al., 2020) considered two models, GRU-Cell RNN and spatio-temporal CNN. These have been initially trained upon the facial features alone. It has been found that including information associated with language articulation has enhanced the accuracy rate to approximately 12%. However,

TABLE 4 Comparative analysis of train and test accuracy (Bodavarapu and Srinivas, 2021).

Model	Train accuracy (%)	Test accuracy (%)
FERConvNet_Gaussian	98	58
FERConvNet_Bilateral	98	63
FERConvNet_Nonlocal Means	93	61
FERConvNet_HDM	98	95
Proposed	99	98

TABLE 5 Analysis in accordance with an accuracy rate (Akhand et al., 2021).

Pre-trained deep CNN model	Accuracy (%)
VGG-16	97.62
VGG-19	98.41
ResNet-18	98.09
ResNet-34	98.57
ResNet-50	99.05
ResNet-152	99.52
Inception-v3	99.05
DenseNet-161	99.52
Proposed model	99.68

the enhancement in accuracy rate has been highly reliant on the consecutive frames that have been afforded as input. Though the accuracy rate has been satisfactory, there is scope for further enhancement. Following this, the research (Qin et al., 2020) has aimed at an issue that conventional FER has not been accurate, for which CNN and GWT (Gabor Wavelet Transform) have been integrated. Initially, histogram equalization, cropping, face positioning, and several pre-processing stages were undertaken for expression images. Subsequently, keyframes corresponding to the expression sequences have been extracted. In this case, GWT was used to procure phase features, while CNN was utilized for training purposes. Experimentation has accomplished an accuracy rate of 96.81%. Furthermore, this study (El Dahshan et al., 2020) aimed to perform FER in accordance with QPSO (Quantum Particle Swarm Optimization) and DBN (Deep Belief Network). The suggested system has encompassed four stages. Initially, pre-processing has been undertaken by cropping region of interest (ROI) to attain the preferred region, thereby eliminating non-essential parts. Furthermore, image downsampling has been adapted to reduce the new sub-image size and enhance the performance of the system. Emotion class has been found with DBN. Rather than adapting the parameters of DBN manually, QPSO has been utilized to optimize DBN parameter values automatically. The suggested method has been employed in datasets including FER-2013. With the employment of the suggested system, the accuracy rate has been found to be 68.1% for the FER-2013 dataset. Furthermore, the article (Liu et al., 2020) has encompassed three major phases: frontal face identification module, feature extraction, and classification. Feature extraction encompasses dual channels. In this case, one is

for raw facial images, while the other one seems to be for the extraction of features from the images. LBP images have been utilized to extract texts to enrich the facial features, thereby improving the performance of the network. Furthermore, an attention mechanism has been adopted. Moreover, the arc-face loss function has been included for improvising the distance of the inter class and minimizing the distance of the inner class. Experimentations have been undertaken on two accessible datasets, namely CK+ and FER-2013. Outcomes have revealed an accuracy rate of 94.24% for the CK+ dataset and 72.56% for the FER-2013 dataset. In spite of various endeavors undertaken by existing works, it has been clearly found that there is a scope for enhancement with regard to accuracy. Accordingly, the proposed system has shown better results in accordance with accuracy (98.19%) than conventional systems.

5.1 Ethical implications of FER

Ethical concerns tied to FER technology, such as privacy, consent, and potential abuse, are significant. FER technology could enhance user interactions in various fields, such as healthcare and security, but it also poses risks like privacy invasion and the possibility of misidentification or bias, especially toward marginalized groups. To encourage ethical use, it is crucial to set up protocols such as obtaining consent before collecting emotional data, explaining the data's purpose, and conducting regular assessments to detect and correct algorithm biases. Additionally, the establishment of regulatory frameworks can help monitor the deployment of FER technologies, ensuring their ethical application and preventing infringements on fundamental rights. By prioritizing these approaches, individuals can reap FER's advantages, minimize its drawbacks, and establish trust with the public.

6 Conclusion

This study aimed to detect emotions from facial expressions using quantum computing. The experimental results showed that quantum computing performs more effectively, even with large and complex datasets. The FER-2013 dataset used in the research and ResNet18 Architecture with Modified Up-Sampled Bottleneck Process were used to classify emotion types from the provided emotions, such as neutral, disgust, anger, sad, happy, surprise, and fear. The proposed system performance was evaluated based on four performance metrics, and the outcomes were found to be 98.19% accuracy, 98% recall, 98% f1-score, and 98% precision. Furthermore, comparative analyses were undertaken with four recent studies to confirm the efficacy of the proposed system. The outcomes of the analysis showed that the proposed model had better values in the performance metrics when compared with the existing models. The results showed the efficient performance of the proposed system over the existing models, and the proposed method achieved 98.19% accuracy. Furthermore, the standard deviation of the proposed system was determined from the execution of the proposed system and was found to be 52.69816460460272. Moreover, the computational complexity for QNNs typically relies on the depth and size of the circuit, the dimensionality of input, and

the number of training samples. Accordingly, for ResNet18, the computational complexity is $O(n^2 \cdot d)$, where n represents the length of image features and d corresponds to the quantum bit dimension. With the integration of position encoding, computational complexity increases to $O(n^2 \cdot d + n \cdot d^2)$. Future studies should further explore the power of quantum computing in machine learning applications.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found at: <https://www.kaggle.com/datasets/msambare/fer2013>.

Author contributions

SA: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. AAlq: Conceptualization, Formal analysis, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. AAla: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Resources, Software, Visualization, Writing – original draft, Writing – review & editing. MS: Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. AG: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. The authors extend their appreciation to Prince Sattam Bin Abdulaziz University for funding this study through the project number (PSAU/2024/01/29802).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Adegun, I. P., and Vadapalli, H. B. (2020). Facial micro-expression recognition: a machine learning approach. *Sci. African* 8:e00465. doi: 10.1016/j.sciaf.2020.e00465
- Akhand, M., Roy, S., Siddique, N., Kamal, M. A. S., and Shimamura, T. (2021). Facial emotion recognition using transfer learning in the deep CNN. *Electronics* 10:1036. doi: 10.3390/electronics10091036
- Alreshidi, A., and Ullah, M. (2020). "Facial emotion recognition using hybrid features" in *Informatics*, vol. 7 (MDPI), 6.
- Bodavarapu, P., and Srinivas, P. (2021). Facial expression recognition for low resolution images using convolutional neural networks and denoising techniques. *Indian J. Sci. Technol.* 14, 971–983. doi: 10.17485/IJST/v14i12.14
- Borgalli, R. A., and Surve, S. (2022). Review on learning framework for facial expression recognition. *Imaging Sci. J.* 70, 483–521. doi: 10.1080/13682199.2023.2172526
- Bursic, S., Boccignone, G., Ferrara, A., D'Amelio, A., and Lanzarotti, R. (2020). Improving the accuracy of automatic facial expression recognition in speaking subjects with deep learning. *Appl. Sci.* 10:4002. doi: 10.3390/app10114002
- Chen, G., Chen, Q., Long, S., Zhu, W., Yuan, Z., and Wu, Y. (2022). Quantum convolutional neural network for image classification. *Pattern. Anal. Applic.* 26, 655–667. doi: 10.1007/s10044-022-01113-z
- Chen, X., Liu, L., Deng, Y., and Kong, X. (2019). Vehicle detection based on visual attention mechanism and adaboost cascade classifier in intelligent transportation systems. *Opt. Quant. Electron.* 51, 1–18. doi: 10.1007/s11082-019-1977-7
- Cheng, E.-J., Chou, K. P., Rajora, S., Jin, B. H., Tanveer, M., Lin, C. T., et al. (2019). Deep sparse representation classifier for facial recognition and detection system. *Pattern Recogn. Lett.* 125, 71–77. doi: 10.1016/j.patrec.2019.03.006
- Ciylan, F., and Ciylan, B. (2021). Fake human face recognition with classical-quantum hybrid transfer learning. *Comput. Inform.* 1, 46–55.
- Deepan, P., and Sudha, L. (2020). "Object classification of remote sensing image using deep convolutional neural network" in *The cognitive approach in cloud computing and internet of things technologies for surveillance tracking systems* (United States, Cambridge, Massachusetts: Elsevier), 107–120.
- Dong, N., Kampffmeyer, M., Voiculescu, I., and Xing, E. (2022). Negational symmetry of quantum neural networks for binary pattern classification. *Pattern Recogn.* 129:108750. doi: 10.1016/j.patcog.2022.108750
- El Dahshan, K. A., Elsayed, E. K., Aboshoha, A., and Ebeid, E. A. (2020). Recognition of facial emotions relying on deep belief networks and quantum particle swarm optimization. *Int. J. Intellig. Eng. Syst.* 13, 90–101. doi: 10.22266/ijies2020.0831.09
- Gavade, P. A., Bhat, V. S., and Pujari, J. (2022). Improved deep generative adversarial network with illuminant invariant local binary pattern features for facial expression recognition. *Comput. Methods Biomech. Biomed. Eng. Imaging Vis.* 11, 678–695. doi: 10.1080/21681163.2022.2103450
- Ghasemi, M., Kelarestaghi, M., Eshghi, F., and Sharifi, A. (2020). FDSR: a new fuzzy discriminative sparse representation method for medical image classification. *Artif. Intell. Med.* 106:101876. doi: 10.1016/j.artmed.2020.101876
- Gour, M., Jain, S., and Sunil Kumar, T. (2020). Residual learning based CNN for breast cancer histopathological image classification. *Int. J. Imaging Syst. Technol.* 30, 621–635. doi: 10.1002/ima.22403
- Hsu, C.-C., Zhuang, Y.-X., and Lee, C.-Y. (2020). Deep fake image detection based on pairwise learning. *Appl. Sci.* 10:370. doi: 10.3390/app10010370
- Hu, M., Wang, H., Wang, X., Yang, J., and Wang, R. (2019). Video facial emotion recognition based on local enhanced motion history image and CNN-CTSLSTM networks. *J. Vis. Commun. Image Represent.* 59, 176–185. doi: 10.1016/j.jvcir.2018.12.039
- Jain, D. K., Shamsolmoali, P., and Sehdev, P. (2019). Extended deep neural network for facial emotion recognition. *Pattern Recogn. Lett.* 120, 69–74. doi: 10.1016/j.patrec.2019.01.008
- Jiang, X., Liu, W., Zhang, Y., Liu, J., Li, S., and Lin, J. (2020). Spectral-spatial hyperspectral image classification using dual-channel capsule networks. *IEEE Geosci. Remote Sens. Lett.* 18, 1094–1098. doi: 10.1109/LGRS.2020.2991405
- Jiang, P., Wan, B., Wang, Q., and Wu, J. (2020). Fast and efficient facial expression recognition using a gabor convolutional network. *IEEE Signal Process. Lett.* 27, 1954–1958. doi: 10.1109/LSP.2020.3031504
- Jing, Y., Li, X., Yang, Y., Wu, C., Fu, W., Hu, W., et al. (2022). RGB image classification with quantum convolutional ansatz. *Quantum Inf. Process* 21, 1–19. doi: 10.1007/s11128-022-03442-8
- Kadam, S. S., Adamuthe, A. C., and Patil, A. B. (2020). CNN model for image classification on MNIST and fashion-MNIST dataset. *J. Sci. Res.* 64, 374–384. doi: 10.37398/JSR.2020.640251
- Kaplan, K., Kaya, Y., Kuncan, M., and Ertunc, H. M. (2020). Brain tumor classification using modified local binary patterns (LBP) feature extraction methods. *Med. Hypotheses* 139:109696. doi: 10.1016/j.mehy.2020.109696
- Karnati, M., Seal, A., Bhattacharjee, D., Yazidi, A., and Krejcar, O. (2023). Understanding deep learning techniques for recognition of human emotions using facial expressions: a comprehensive survey. *IEEE Trans. Instrum. Meas.* 72, 1–31. doi: 10.1109/TIM.2023.3243661
- Karnati, M., Seal, A., Yazidi, A., and Krejcar, O. (2022). FLEPNet: feature level ensemble parallel network for facial expression recognition. *IEEE Trans. Affect. Comput.* 13, 2058–2070. doi: 10.1109/TAFFC.2022.3208309
- Kaur, R., and Singh, S. (2022). A comprehensive review of object detection with deep learning. *Digit. Signal Process.* 103812. doi: 10.1016/j.dsp.2022.103812
- Kavitha, M. S., Gangadaran, P., Jackson, A., Venmathi Maran, B. A., Kurita, T., and Ahn, B.-C. (2022). Deep neural network models for Colon Cancer screening. *Cancers* 14:3707. doi: 10.3390/cancers14153707
- Khairuddin, Y., and Chen, Z. (2021). "Facial emotion recognition: State of the art performance on FER2013." arXiv preprint arXiv:2105.03588. doi: 10.48550/arXiv.2105.03588
- Khan, A. R. (2022). Facial emotion recognition using conventional machine learning and deep learning methods: current achievements, analysis and remaining challenges. *Information* 13:268. doi: 10.3390/info13060268
- Kim, J. H., Poulouse, A., and Han, D. S. (2021). The extensive usage of the facial image thresholding machine for facial emotion recognition performance. *Sensors* 21:2026. doi: 10.3390/s21062026
- Kumar, Y., Verma, S. K., and Sharma, S. (2021). An ensemble approach of improved quantum inspired gravitational search algorithm and hybrid deep neural networks for computational optimization. *Int. J. Modern Phys. C* 32:2150100. doi: 10.1142/S012918312150100X
- Kumar, Y., Verma, S. K., and Sharma, S. (2022). Multi-pose facial expression recognition using hybrid deep learning model with improved variant of gravitational search algorithm. *Int. Arab J. Inf. Technol.* 19, 281–287. doi: 10.34028/iajit/19/2/15
- Lazzarin, M., Galli, D. E., and Prati, E. (2022). Multi-class quantum classifiers with tensor network circuits for quantum phase recognition. *Phys. Lett. A* 434:128056. doi: 10.1016/j.physleta.2022.128056
- Li, W., Lu, Z.-d., and Deng, D.-L. (2022). Quantum neural network classifiers: a tutorial. *SciPost Phys. Lecture Notes* 061. doi: 10.21468/SciPostPhysLectNotes.61
- Li, Y., Xiao, J., Chen, Y., and Jiao, L. (2019). Evolving deep convolutional neural networks by quantum behaved particle swarm optimization with binary encoding for image classification. *Neurocomputing* 362, 156–165. doi: 10.1016/j.neucom.2019.07.026
- Li, Y., Zhou, R.-G., Xu, R., Luo, J., and Hu, W. (2020). A quantum deep convolutional neural network for image recognition. *Quantum Sci. Technol.* 5:044003. doi: 10.1088/2058-9565/ab9f93
- Liu, S. (2020). Image classification of static facial expressions in the wild based on bidirectional neural networks (based on pytorch). Available at: <https://users.cecs.anu.edu.au/~Tom.Gedeon/conf/ABCs2020/paper/>
- Liu, C., Hirota, K., Wang, B., Dai, Y., and Jia, Z. (2020). Two-Channel feature extraction convolutional neural network for facial expression recognition. *J. Advan. Comput. Intellig. Intelligent Inform.* 24, 792–801. doi: 10.20965/jaciii.2020.p0792
- Liu, G., Ma, W.-P., Cao, H., and Lyu, L.-D. (2020). A quantum Hopfield neural network model and image recognition. *Laser Phys. Lett.* 17:045201. doi: 10.1088/1612-202X/ab7347
- Ma, Y., Xie, Q., Liu, Y., and Xiong, S. (2020). A weighted KNN-based automatic image annotation method. *Neural Comput. Applic.* 32, 6559–6570. doi: 10.1007/s00521-019-04114-y
- Makhija, Y., and Sharma, R. S. (2019). Face recognition: novel comparison of various feature extraction techniques. *Harmony Search Nat. Inspired Optim. Algorith.* 741, 1189–1198. doi: 10.1007/978-981-13-0761-4_110
- Mehta, D., Siddiqui, M. F. H., and Javaid, A. Y. (2019). Recognition of emotion intensities using machine learning algorithms: a comparative study. *Sensors* 19:1897. doi: 10.3390/s19081897
- Mei, X., Pan, E., Ma, Y., Dai, X., Huang, J., Fan, F., et al. (2019). Spectral-spatial attention networks for hyperspectral image classification. *Remote Sens.* 11:963. doi: 10.3390/rs11080963
- Mohan, K., Seal, A., Krejcar, O., and Yazidi, A. (2020). Facial expression recognition using local gravitational force descriptor-based deep convolution neural networks. *IEEE Trans. Instrum. Meas.* 70, 1–12. doi: 10.1109/TIM.2020.3031835
- Mohan, K., Seal, A., Krejcar, O., and Yazidi, A. (2021). FER-net: facial expression recognition using deep neural net. *Neural Comput. Applic.* 33, 9125–9136. doi: 10.1007/s00521-020-05676-y
- Munawar, H. S., Aggarwal, R., Qadir, Z., Khan, S. I., Kouzani, A. Z., and Mahmud, M. P. (2021). A gabor filter-based protocol for automated image-based building detection. *Buildings* 11:302. doi: 10.3390/buildings11070302
- Mungra, D., Agrawal, A., Sharma, P., Tanwar, S., and Obaidat, M. S. (2020). PRATIT: a CNN-based emotion recognition system using histogram equalization and data augmentation. *Multimed. Tools Appl.* 79, 2285–2307. doi: 10.1007/s11042-019-08397-0
- Okwuashi, O., and Ndehedehe, C. E. (2020). Deep support vector machine for hyperspectral image classification. *Pattern Recogn.* 103:107298. doi: 10.1016/j.patcog.2020.107298

- Oloyede, M. O., Hancke, G. P., and Myburgh, H. C. (2020). A review on face recognition systems: recent approaches and challenges. *Multimed. Tools Appl.* 79, 27891–27922. doi: 10.1007/s11042-020-09261-2
- Qin, S., Zhu, Z., Zou, Y., and Wang, X. (2020). Facial expression recognition based on Gabor wavelet transform and 2-channel CNN. *Int. J. Wavelets Multiresolution Inf. Process.* 18:2050003. doi: 10.1142/S0219691320500034
- Rosen, D. M., Doherty, K. J., Terán Espinoza, A., and Leonard, J. J. (2021). Advances in inference and representation for simultaneous localization and mapping. *Ann. Rev. Control Robotics Autonomous Syst.* 4, 215–242. doi: 10.1146/annurev-control-072720-082553
- Sachadev, J. S., and Bhatnagar, R. (2022). A comprehensive review on brain disease mapping—the underlying technologies and AI-based techniques for feature extraction and classification using EEG signals. *Med. Inform. Bioimaging Using Arti. Intellig.* 1005, 73–91. doi: 10.1007/978-3-030-91103-4_5
- Saurav, S., Saini, R., and Singh, S. (2021). EmNet: a deep integrated convolutional neural network for facial emotion recognition in the wild. *Appl. Intell.* 51, 5543–5570. doi: 10.1007/s10489-020-02125-0
- Shanthi, P., and Nickolas, S. (2021). An efficient automatic facial expression recognition using local neighborhood feature fusion. *Multimed. Tools Appl.* 80, 10187–10212. doi: 10.1007/s11042-020-10105-2
- Singhal, N., Ganganwar, V., Yadav, M., Chauhan, A., Jakhar, M., and Sharma, K. (2021). Comparative study of machine learning and deep learning algorithm for face recognition. *Jordanian J. Comput. Inform. Technol.* 7:1. doi: 10.5455/jjcit.71-1624859356
- Song, J. M., Kim, W., and Park, K. R. (2019). Finger-vein recognition based on deep DenseNet using composite image. *IEEE Access* 7, 66845–66863. doi: 10.1109/ACCESS.2019.2918503
- Speiser, J. L., Miller, M. E., Tooze, J., and Ip, E. (2019). A comparison of random forest variable selection methods for classification prediction modeling. *Expert Syst. Appl.* 134, 93–101. doi: 10.1016/j.eswa.2019.05.028
- Tacchino, F., Barkoutsos, P., Macchiavello, C., Tavernelli, I., Gerace, D., and Bajoni, D. (2020). Quantum implementation of an artificial feed-forward neural network. *Quantum Sci. Technol.* 5:044010. doi: 10.1088/2058-9565/abb8e4
- Tammima, S. (2019). Transfer learning using vgg-16 with deep convolutional neural network for classifying images. *Int. J. Sci. Res. Public.* 9, 143–150. doi: 10.29322/IJSRP9.10.2019.p9420
- Tayba, M. N., Maruf, A. A., Rivas, P., Baker, E., and Orduz, J. (2022). “Using quantum circuits with convolutional neural network for pneumonia detection,” in *Proceedings of the Southwest Data Science Conference*. Waco, TX, USA, 1–12.
- Tsuneki, M. (2022). Deep learning models in medical image analysis. *J. Oral Biosci.* 64, 312–320. doi: 10.1016/j.job.2022.03.003
- Wang, Y., Li, Y., Song, Y., and Rong, X. (2020). The influence of the activation function in a convolution neural network model of facial expression recognition. *Appl. Sci.* 10:1897. doi: 10.3390/app10051897
- Wang, Z., Xu, M., and Zhang, Y. (2022). Quantum pulse coupled neural network. *Neural Netw.* 152, 105–117. doi: 10.1016/j.neunet.2022.04.007
- Yang, Y., Wang, X., Sun, B., and Zhao, Q. (2020). Channel expansion convolutional network for image classification. *IEEE Access* 8, 178414–178424. doi: 10.1109/ACCESS.2020.3027879
- Zahara, L., Musa, P., Wibowo, E. P., Karim, I., and Musa, S. B. (2020). “The facial emotion recognition (FER-2013) dataset for prediction system of micro-expressions face using the convolutional neural network (CNN) algorithm based raspberry Pi” in 2020 fifth international conference on informatics and computing (ICIC) (Gorontalo City, Sulawesi, Indonesia: IEEE), 1–9.
- Zhang, R., Zhu, Y., Ge, Z., Mu, H., Qi, D., and Ni, H. (2022). Transfer learning for leaf small dataset using improved ResNet50 network with mixed activation functions. *Forests* 13:2072. doi: 10.3390/f13122072
- Zhao, H., Zhan, Z. H., Lin, Y., Chen, X., Luo, X. N., Zhang, J., et al. (2019). Local binary pattern-based adaptive differential evolution for multimodal optimization problems. *IEEE Trans. Cybernetics* 50, 3343–3357. doi: 10.1109/TCYB.2019.2927780
- Zheng, Q., Tian, X., Jiang, N., and Yang, M. (2019). Layer-wise learning based stochastic gradient descent method for the optimization of deep convolutional neural network. *J. Intelligent Fuzzy Syst.* 37, 5641–5654. doi: 10.3233/JIFS-190861
- Zhou, W., Gao, S., Zhang, L., and Lou, X. (2020). Histogram of oriented gradients feature extraction from raw Bayer pattern images. *IEEE Trans. Circ. Syst. II Express Briefs* 67, 946–950. doi: 10.1109/TCSII.2020.2980557



OPEN ACCESS

EDITED BY

Deepika Koundal,
University of Petroleum and Energy Studies,
India

REVIEWED BY

Nuryani Nuryani,
Sebelas Maret University, Indonesia
Atef Zaguia,
Taif University, Saudi Arabia

*CORRESPONDENCE

Nisreen Innab
✉ Ninnab@um.edu.sa
Imran Ashraf
✉ imranashraf@ynu.ac.kr

RECEIVED 23 August 2024

ACCEPTED 18 November 2024

PUBLISHED 11 December 2024

CITATION

Na I-s, Aldrees A, Hakeem A, Mohaisen L,
Umer M, AlHammadi DA, Alsubai S, Innab N
and Ashraf I (2024) FacialNet: facial emotion
recognition for mental health analysis using
UNet segmentation with transfer learning
model. *Front. Comput. Neurosci.* 18:1485121.
doi: 10.3389/fncom.2024.1485121

COPYRIGHT

© 2024 Na, Aldrees, Hakeem, Mohaisen,
Umer, AlHammadi, Alsubai, Innab and Ashraf.
This is an open-access article distributed
under the terms of the [Creative Commons
Attribution License \(CC BY\)](#). The use,
distribution or reproduction in other forums is
permitted, provided the original author(s) and
the copyright owner(s) are credited and that
the original publication in this journal is cited,
in accordance with accepted academic
practice. No use, distribution or reproduction
is permitted which does not comply with
these terms.

FacialNet: facial emotion recognition for mental health analysis using UNet segmentation with transfer learning model

In-seop Na¹, Asma Aldrees², Abeer Hakeem³, Linda Mohaisen³,
Muhammad Umer⁴, Dina Abdulaziz AlHammadi⁵,
Shtwai Alsubai⁶, Nisreen Innab^{7*} and Imran Ashraf^{8*}

¹Division of Culture Contents, Chonnam National University, Yeosu, Republic of Korea, ²Department of Informatics and Computer Systems, College of Computer Science, King Khalid University, Abha, Saudi Arabia, ³Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia, ⁴Department of Computer Science & Information Technology, The Islamia University of Bahawalpur, Bahawalpur, Pakistan, ⁵Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia, ⁶Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia, ⁷Department of Computer Science and Information Systems, College of Applied Sciences, AlMaarefa University, Diriyah, Saudi Arabia, ⁸Department of Information and Communication Engineering, Yeungnam University, Gyeongsan, Republic of Korea

Facial emotion recognition (FER) can serve as a valuable tool for assessing emotional states, which are often linked to mental health. However, mental health encompasses a broad range of factors that go beyond facial expressions. While FER provides insights into certain aspects of emotional well-being, it can be used in conjunction with other assessments to form a more comprehensive understanding of an individual's mental health. This research work proposes a framework for human FER using UNet image segmentation and transfer learning with the EfficientNetB4 model (called FacialNet). The proposed model demonstrates promising results, achieving an accuracy of 90% for six emotion classes (happy, sad, fear, pain, anger, and disgust) and 96.39% for binary classification (happy and sad). The significance of FacialNet is judged by extensive experiments conducted against various machine learning and deep learning models, as well as state-of-the-art previous research works in FER. The significance of FacialNet is further validated using a cross-validation technique, ensuring reliable performance across different data splits. The findings highlight the effectiveness of leveraging UNet image segmentation and EfficientNetB4 transfer learning for accurate and efficient human facial emotion recognition, offering promising avenues for real-world applications in emotion-aware systems and effective computing platforms. Experimental findings reveal that the proposed approach performs substantially better than existing works with an improved accuracy of 96.39% compared to existing 94.26%.

KEYWORDS

facial emotion recognition, UNET, EfficientNet, transfer learning, image processing

1 Introduction

Human facial expressions, primarily manifested through facial features, hold considerable emotional significance (Huang et al., 2019). People instinctively understand these expressions as they represent an individual's emotions and demeanor during interactions. With technological progress, there's an increasing interest in equipping machines with cognitive skills, leading to research and debate in areas like human-computer interaction and computer vision. A particular focus is on emotion recognition via facial expressions, with applications in human-computer collaborative systems, responsive animation, and human-robot interaction (Oguine et al., 2022). The challenge of identifying and classifying human emotions has been a topic of interest in psychology, anthropology, and computer science. Some researchers propose universal emotional categories, while others emphasize the cultural impact on emotional perception and expression. Cross-cultural studies reveal both similarities and differences in emotional categorization (Lindquist et al., 2022).

Recognizing human facial emotions (FER) is crucial for digital applications, human-computer interfaces, and behavioral sciences (Gupta and Jain, 2021). Understanding the movements of facial muscles and their link to emotions is essential for creating effective classification algorithms. Classifying facial emotions involves feature extraction (Song et al., 2024) and classification methods (Zhu, 2023). Despite advancements, accurately identifying facial expression subtleties remains challenging, particularly in online settings (Zhang et al., 2020). Using image classification to detect and categorize emotions is an intriguing frontier in emotion recognition. Deep learning models, specifically convolutional neural networks (CNNs), stand out for image classification and can reliably classify emotions in images when trained on datasets of facial expressions (Canal et al., 2022).

Human facial emotion recognition has several medical applications. For example, Vignesh et al. (2023) uses a novel CNN-based model for emotion recognition for psychological profiling. The model incorporates U-Net segmentation layers within VGG layers to extract critical features leading to better performance compared to existing approaches on the FER-2013 dataset. Similarly, other studies also report the use of CNN for enhanced accuracy in facial emotion recognition (Sarvakar et al., 2023; Huang et al., 2023). The study (Sarvakar et al., 2023) introduces a neural networks convolutionary (FERC) approach based on the CNN model. An expressional vector is leveraged in the proposed approach to identify five facial emotions. Contrary to the single-level approach used in traditional CNN, FERC follows a two-level approach.

Along the same course, the authors (Huang et al., 2023) utilize a transfer learning approach where CNN and residual neural network for facial emotion recognition. Features are extracted using the residual network which is later used with the CNN model. The authors found important features to provide better performance. Features around the nose and mouth are reported to be critical features to obtain enhanced accuracy. Results report an 83.37% accuracy with the AffectNet model using RAF-DB dataset which contains real-world expressions.

The study (Talaat, 2023) presents a real-time facial emotion detection approach for children suffering from autism spectrum

disorder (ASD). ASD is a difficult-to-diagnose disorder in the early stage and facial emotions offer an alternative in this regard. Normal and ASD children are reported to show different facial emotions. The study proposes an enhanced deep learning technique based on the CNN model. Robust and improved results are reported from the proposed approach.

1.1 Challenges in existing approaches

Despite the improved accuracy and enhanced performance reported in the existing literature, several challenges require further efforts. To achieve accurate predictions when training neural networks, a significant amount of data is required. However, gathering datasets for subjective emotions poses a big challenge. Many databases are sourced from platforms like Amazon Mechanical Turk (AMT) or utilize hashtags of social media to label image sentiment. These methods demand substantial human effort and time, leading to increased costs. To address these issues, integrating the training dataset with synthetic images is suggested to assess whether it enhances accuracy and reduces the need for real-face images (Huang et al., 2019). Handling the variations in the human face including the color, posture, expression, etc. is challenging for a FER system. Similarly, facial muscular motions vary, as do the skin deformation from one person to another making it difficult to make a FER system capable of recognizing emotions in all scenarios. Consequently, existing FER systems suffer from low accuracy.

1.2 Contributions of this study

In view of the challenges pointed out earlier, this study aims to provide robust and precise results for FER. This research work creates an efficient technique for categorizing human mood from images. This study fulfills the following tasks.

1. The study proposes the use of the UNet model for image segmentation with the EfficientNetB4 transfer learning (TL) model to identify emotions including happy, sad, fear, pain, anger, and disgust.
2. Multiple experiments are performed to identify emotions. The first experiment does not involve UNet-based image segmentation. In the second experiment, UNet segmentation is performed to identify six emotions. In the third experiment, UNet segmentation is performed for binary classification involving happy and sad emotions.
3. Additionally, various ML and DL methods along with TL approaches are adopted for performance comparison. Based on the overall results with all classes (without UNet segmentation), all classes (with UNet segmentation), and binary class (with UNet segmentation), the effectiveness of the various models is assessed.

The remaining sections of the paper are arranged as follows: A summary of is given in Section 2, previous work related to human emotions for image classification. Section 3 details the dataset, including preprocessing steps and data visualization techniques

employed to uncover underlying patterns within the data. This section also outlines the various algorithms utilized in the study. In Section 4, the results are discussed and analyzed. In conclusion, Section 5 summarizes the study's findings and suggests directions for further investigation.

2 Related works

As mentioned, major research development has been conducted on facial emotion recognition systems in the past few years. Several approaches have been developed to solve this problem. There have been approaches using features-based recognition to DL approaches (Vignesh et al., 2023; Sarvakar et al., 2023; Huang et al., 2023; Talaat, 2023). However, the CNN models are widely used for this task and reported good results concerning emotion detection from facial expressions. Qu et al. (2023) proposed a CNN-based system for FER. They used the benchmark dataset "FER 2013." They used two different optimizers for the optimization of the CNN such as stochastic gradient descent (SGD) and Adam with different epochs. The study's results indicate that the CNN achieved a 60.20% accuracy using SGD optimizer on 00 epochs. Meena and Mohbey (2023) proposed a TL-based system for the automatic sentiment classification of images. They used the three TL models such as VGG16, Inception-v3, and XceptionNet. The authors compared the TL model's performance on three different datasets. Results of the study show that the Inception-v3 achieved the highest accuracy on the CK+ dataset which is 99.57%, VGG-19 performed well on the JAFFE dataset and attained an accuracy score of 94%, and on the FER 2013 dataset, XceptionNet achieved the accuracy score of 77.92%.

The research described in Boughida et al. (2022) introduced an innovative technique for recognizing facial expressions by utilizing evolutionary algorithms alongside Gabor filters. Facial landmarks serve to identify crucial facial areas for the extraction of Gabor features. Additionally, a genetic algorithm was employed to concurrently select optimal features and fine-tune support vector machine (SVM) hyperparameters. Regarding the JAFFE, CK, and CK+ datasets, the test results reveal the method's exceptional performance with recognition rates of 96.30%, 94.20%, and 94.26%, respectively. Gubbala et al. (2023) suggested a random forest (RF) model enhanced with AdaBoost for the classification of facial emotions from images. Their model aims to transform features from social media image posts for emotional analysis. The Adaboost-based RF model for emotion classification (ARFEC) model's core stages include class labeling, feature selection, and feature extraction. The study demonstrates that the ARFEC model achieved a peak accuracy rate of 92.57% on the CK+ dataset. Similarly, the research outlined in Oguine et al. (2022) advocated for a more efficient and accurate approach to classifying both digital and real-time facial images into one of the seven emotional categories. Enhanced training efficiency and classification accuracy are achieved through preprocessing and data augmentation methods. The proposed CNN+Haar Cascade model attained the top accuracy of 70.04% on the FER2013 dataset. Gupta and Jain (2021) developed a deep learning-based system for emotion recognition via facial expressions. By utilizing a CNN-based system inspired by the LeNet architecture, the recognition of emotions

through facial features is achieved. Their study employed publicly accessible datasets featuring seven distinct categories. The proposed CNN model recorded a maximum accuracy of 60.37%.

Haider et al. (2023) proposed an innovative method for emotion classification through facial imagery. Their method features a customized ResNet18 model augmented with a triplet loss function (TLF) combined with a TL, along with an SVM model for classification purposes. This approach utilizes a facial vector and classifier to identify facial expressions by exploiting deep features from a modified ResNet trained with triplet loss. During preprocessing, facial areas are identified within the source images via Retina Face, and the features are extracted by training the ResNet18 model on cropped facial images using the triplet loss. The SVM classifier then categorizes facial expressions based on these deep features. Their results indicated that the tailored ResNet18 achieved a maximum accuracy of 99.02% on the MMI dataset. An additional study (Lucey et al., 2010) compiled the extended Cohn-Kanade dataset, which provides annotations for both emotions and action units. The dataset's performance was assessed using a combination of SVM and active appearance models (AAMs) for categorization. AAMs produce a mesh that tracks facial movements across images, yielding two feature vectors. Initially, the mesh vertices undergo translation, scaling, and rotation, followed by conversion of images to grayscale using the input photos and mesh. Through a leave-one-subject-out cross-validation process, they reported over 80% accuracy. Huang et al. (2019) proposed an advanced CNN to enhance the extraction of sentiment from images based on visual content. They significantly enhanced the training set by adding artificial face photos. The set exclusively included synthetic and genuine face images, as well as combinations of both. The result of the study shows that the AlexNet TL model achieved the highest accuracy of 87.79%. Similarly, Anilkumar et al. (2024) propose a deep CNN with hyperparameter optimization (DCNN-HPO) for correctly predicting sentiment analysis by optimizing the DCNN parameters. They used three different publicly available datasets for experiments. The VGG-16 network is used to extract features from each preprocessed image. Next, the DCNN is updated using the retrieved features, and the DCNN's weight parameters are modified via Krill Herd Optimization (KHO). Classification result shows that the proposed DCNN-HPO achieved the highest accuracy of 83.4% DCNN-HPO using the TumEmo dataset. The summary of the discussed literature is presented in Table 1.

3 Methodology

This study proposes a UNet segmentation-based TL approach employing the EfficientNetB4 model for emotion recognition. Figure 1 shows the methodological architecture of the approach.

3.1 Training phase

The training process for the proposed model consists of two key stages:

1. Segmentation with UNet

The UNet architecture is trained on segmented facial regions to isolate critical areas such as eyes, mouth, and facial contours.

TABLE 1 Summary of the related work.

References	Classifiers	Dataset	Performance
Qu et al. (2023)	CNN	FER 2013	60.20%
Meena and Mohbey (2023)	VGG16, Inception-v3, and XceptionNet	FER2013, JAFFE, Cohn-Kanade Dataset (CK+)	77.92% XceptionNet on FER2013, 99.57% Inception-v3 on CK+,94% VGG-19 on JAFFE
Boughida et al. (2022)	SVM kernel (Linear, RBF), Gabor filters	CK+	94.26% Gabor filter
Gubbala et al. (2023)	KNN, SVM, ARFEC	FFHQ, CK+ and FER2013	ARFEC on FFHQ = 89.5 %, CK+ = 92.5 %, FER2013= 89.5%
Oguine et al. (2022)	CNN + Haar Cascade	FER2013	70.04%
Gupta and Jain (2021)	CNN with LeNet	FER 2013	60.37%
Haider et al. (2023)	Customized ResNet18,SVM, LDA, and Softmax	JAFFE, FER2013, AFFECNET, and MMI	99.02% customized ResNet18 on MMI Dataset
Lucey et al. (2010)	AAMs, SVM	CK+	80% SVM with CAPP features
Huang et al. (2019)	AlexNet	Synthetic face dataset collected using FaceGen software	87.79%
Anilkumar et al. (2024)	ConvLSTM, MAN, AHR, DCNN-HPO	GSO-2016, MVSA-Single, TumEmo	83.4% DCNN-HPO using TumEmo

The segmentation network was optimized using the categorical cross-entropy loss function (Equation 1) with the Adam optimizer, achieving rapid convergence. The segmentation step enhances the quality of the input features for emotion classification.

2. Feature Extraction and Classification with EfficientNetB4

The segmented images are input into the EfficientNetB4 model pre-trained on the ImageNet dataset. The transfer learning approach ensures efficient feature extraction with minimal computational overhead. Fine-tuning was performed on the top layers, including a fully connected layer for emotion classification. The training parameters were optimized using the following configuration:

- Learning rate: 1×10^{-4}
- Batch size: 32
- Epochs: 50
- Optimizer: Adam

A combination of data augmentation techniques (rotation, flipping, and scaling) was applied to improve model generalization.

3.2 Testing phase

The testing phase involved evaluating the model on unseen data to measure its performance on both multi-class and binary classification tasks. The metrics used for evaluation included accuracy, precision, recall, and F1-score. The testing process was carried out as follows:

- Multi-class classification: the model classified six emotions: happy, sad, fear, pain, anger, and disgust. It achieved an accuracy of 90%, outperforming prior works with 94.26%.
- Binary classification: for the classification of happy and sad emotions, the model achieved an accuracy of

96.39%, demonstrating superior performance over existing benchmarks.

The results were further validated using a five-fold cross-validation approach to ensure consistent performance across varying data splits.

3.3 Dataset

The Kaggle repository dataset “6 Human Emotions for image classification” contains facial images indicating people’s sentiments (Mohamed, 2024). The images are in 224×224 dimensions, however, the position of the face varies slightly. Because of the automatic registration of the faces, every image has a face that is about in the middle and occupies a similar area. The dataset consists of six different classes such as “Happy,” “Sad,” “Fear,” “Pain,” “Anger,” and “Disgust.” There are a total of 1,200 instances that belong to six different categories. Class-wise instances of the dataset are shown in Table 2.

3.4 Dataset preprocessing

Preprocessing includes image segmentation since it allows for precise extraction of information about distinct image regions and structures. This precision in segmentation proves indispensable for a range of image classification purposes. Moreover, it paves the way for cutting-edge medical research and facilitates various clinical applications.

3.5 UNet for image segmentation

UNet is a convolutional neural network architecture specifically designed for image segmentation tasks. In 2015, Olaf Ronneberger,



TABLE 2 Dataset statistics.

Category	Number of instances
Happy	230
Sad	224
Fear	163
Pain	168
Anger	214
Disgust	201

Philipp Fischer, and Thomas Brox presented it (Krithika Alias AnbuDevi and Suganthi, 2022). The name “UNet” comes from its U-shaped architecture, which is the hallmark of this network. The main objective of image segmentation is to partition an input image into multiple regions and assign each pixel to a specific class or category. The UNet architecture consists of an encoder and a decoder part. Here’s a brief explanation of each of its components.

3.5.1 Encoder

On the left side of the U-shaped network is the encoder, which consists of a series of convolutional and max-pooling layers. Its main function is to extract high-level features and spatial details from the input image. With increased network depth, the receptive field expands, allowing the model to recognize more intricate patterns within the image.

3.5.2 Decoder

The decoder is on the right side of the U-shaped network. It consists of upsampling and convolutional layers. The decoder’s role is to take the learned features from the encoder and gradually upsample the spatial resolution to produce a segmentation map. The upsampling process helps recover the spatial information lost during the downsampling in the encoder.

UNet has proven to be very effective for image-segmentation tasks because it captures both local and global contexts through the combination of encoder-decoder architecture and skip connections. It has been widely adopted and adapted for various segmentation challenges across different domains.

3.6 Mathematical working of UNet architecture for FER

The UNet architecture utilized in this study is composed of a contracting path (encoder) and an expansive path (decoder). Mathematically, the UNet segmentation model can be expressed as:

$$\mathbf{Y} = f_{\text{UNet}}(\mathbf{X}; \theta), \quad (1)$$

where \mathbf{X} is the input image, \mathbf{Y} is the segmented output, and θ represents the trainable parameters of the network. The contracting

path applies convolutional layers followed by max-pooling to extract feature maps:

$$\mathbf{F}_i = \sigma(\mathbf{W}_i * \mathbf{F}_{i-1} + \mathbf{b}_i), \quad i \in [1, N], \quad (2)$$

where \mathbf{F}_i is the feature map at layer i , \mathbf{W}_i and \mathbf{b}_i are the weights and biases, $*$ denotes the convolution operation, and σ is the activation function (ReLU).

The expansive path performs upsampling and concatenation of feature maps to recover spatial information:

$$\mathbf{F}_i^{\text{up}} = \text{Up}(\mathbf{F}_{i+1}) \oplus \mathbf{F}_i^{\text{enc}}, \quad (3)$$

where $\text{Up}(\cdot)$ represents the upsampling operation, and \oplus denotes channel-wise concatenation with feature maps from the encoder ($\mathbf{F}_i^{\text{enc}}$).

3.7 Loss function

To optimize the UNet model for segmentation, we employed the categorical cross-entropy loss, defined as:

$$\mathcal{L}_{\text{CCE}} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C \mathbf{Y}_{i,c} \log(\hat{\mathbf{Y}}_{i,c}), \quad (4)$$

where N is the total number of pixels, C is the number of classes, $\mathbf{Y}_{i,c}$ is the true label, and $\hat{\mathbf{Y}}_{i,c}$ is the predicted probability for class c at pixel i . This loss function ensures pixel-wise classification accuracy for multi-class segmentation.

3.8 ML models for emotion classification

In order to classify human emotion classification using images Classification algorithms based on ML are applied. This paper provides a brief discussion of some prominent classification methods and their theoretical foundation. The models were optimized by fine-tuning various hyperparameters for ML models.

3.8.1 Random forest

For supervised learning, decision trees are improved and employed in random forests. The largest number of votes serves as the basis for this prediction criterion (Manzoor et al., 2021). Because there are improper connections between the trees in the random forest, it has a low error rate when compared to other classifiers. One way to conceptualize RF is as an ensemble model composed of several trees. The ultimate forecast of this classifier is decided by a majority vote after it generates several choice trees. Compared to decision trees, this is more efficient because decision trees collaborate and correct one another’s errors. In a random forest, every tree is trained with distinct data points and has bagging. The trees are therefore unconnected to one another.

3.8.2 Logistic regression

LR is a prevalent statistical technique employed for binary and multiclass classification tasks, even though its name might suggest a regression application (Rymarczyk et al., 2019). LR primarily focuses on classification. It models the connection between independent variables and the likelihood of a specific outcome occurring. By utilizing the logistic function, LR confines predictions within a range of 0–1. During the training process, LR calculates the coefficients for each independent variable through maximum likelihood estimation. These coefficients signify the influence of each variable on the probability of the outcome. LR is easily interpretable, simple to implement, and effective with data that is linearly separable. It stands as a fundamental model in the realm of machine learning.

3.8.3 Extra tree classifier

The extra tree classifier (ETC) functions similarly to the RF classifier, with the exception that a random process of splitting is used in place of the top-down technique, which reduces variance by making the tree more biased (Umer et al., 2022). This is because a significant portion of the generated tree's variance is caused by the selection of the ideal cut-point. In contrast to RF bootstrapping, the ETC does not support this. For instance, the number of split points is equal to k if k attributes are chosen out of the entire N attributes in our training class. Let S represent these split sites, i.e., S_1, S_2, S_3, S_k . These divisions are selected at random. Every split results in a decision tree being generated. Every split yields a score representing the likelihood of choosing every class. Therefore, for class A , PA (i.e., $PA_1, PA_2, PA_3, \dots, PA_k$) provides the probability. The class with the highest probability is selected to determine the prediction, which is determined by averaging the probabilities of each class. Another name for this is majority voting. This reduction in complexity lessens the computational load on the Extra Tree Classifier and enables it to get better results in several high-dimensional complicated challenges.

3.8.4 Support vector machine

SVM is a powerful supervised learning model used for challenges involving both regression and classification (Hearst et al., 1998). They work by locating the ideal hyperplane in the feature space that optimizes the distance between various classes. Support vectors or the data points nearest to the decision boundary, are what decide this hyperplane. SVMs may handle both linearly and non-linearly separable data by using kernel functions to transform the data into a higher-dimensional space. This makes it possible to create more complex decision boundaries.

3.9 DL models for emotion classification

Artificial intelligence methods that imitate human knowledge acquisition and DL are connected to ML approaches. DL is an essential part of data science, which encompasses statistics and predictive modeling. One kind of deep neural network utilized in deep understanding is the CNN, which analyzes visual data. CNN is a DL method that employs weights to recognize various objects

in an input image so it can make distinctions between them. CNN is used to categorize and identify photos due to its high degree of accuracy.

DL architectures ResNet, MobileNet, VGG19, EfficientNetB4, and InceptionV3- are employed to categorize the data. These models are trained by TL. A total of 50 training epochs have been used to train each model. Below is a thorough explanation.

3.9.1 Convolutional neural network

CNNs are based on the visual system of the human brain. CNNs therefore aim to make computers able to see the world as humans do (Lu et al., 2021; Ding et al., 2023). CNNs can thus be applied to NLP, image classification, and diagnosis. CNN is a subset of DNN that has nonlinear activation layers, max pooling, and convolutional layers. The convolutional layer, which is thought to be responsible for the "convolution" operation that gives CNN its name, a CNN's primary layer. Layer inputs are subjected to the application of convolutional layer kernels. A feature map is created by convolving each of the convolutional layers' outputs. Since the input images in this study are inherently nonlinear, the ReLU activation function together with maxpooling layers, helps to augment the non-linearity in the image. Therefore, in the current scenario, CNN with ReLU is straightforward and faster. The ReLU can be defined as follows because it is 0 for all negative inputs:

$$z = \max(0, i) \quad (5)$$

where z shows the output and \max calculates the maximum value from 0 and input value i . In this case, the function suggests that the positive value stays constant and the output z is zero for all negative values.

3.9.2 VGG19

The 19-layer VGG19 model is a deep CNN. For tasks involving picture classification, it is trained using the ImageNet dataset (Rajinikanth et al., 2020; Cao et al., 2024). A 2×2 max pooling layer and a ReLU activation function come after each repeating 3×3 convolutional layer in the architecture. VGG19 is frequently utilized in computer vision research due to its high accuracy on a variety of picture classification benchmarks. Nevertheless, because of its many characteristics, it is computationally costly and challenging to implement on devices with limited resources.

3.9.3 ResNET

Residual Network (ResNet) is a type of CNN that is commonly used for TL, especially in the context of DL for image processing tasks such as image recognition and classification (Yaqoob et al., 2021). ResNet revolutionized DL by enabling the training of extremely deep neural networks with 152 layers or more. Before the introduction of ResNet, such deep networks were hard to train due to the vanishing gradient problem, where the gradient signal gets smaller and smaller as it backpropagates through each layer, eventually becoming too tiny to make any significant changes in the weights in the lower layers. ResNet addresses this by using skip connections, or shortcuts to jump over some layers. The outputs

of these connections, which carry out identity mapping, are added to the stacked layer outputs, effectively allowing the training signal to be directly propagated back through the network. This design makes it possible to train very deep networks, and thus ResNet models can learn richer and more complex feature representations. A ResNet model that has been pre-trained on a large and general dataset like ImageNet is often used as a starting point for a new task. Because the initial layers of a CNN tend to learn features that are generally useful for analyzing images, such as edges and textures, they can be effectively applied to new tasks with little alteration. The later layers of the network, which learn more specific patterns, may be fine-tuned with a smaller dataset specific to the new task, ensuring adaptability and relevance.

3.9.4 EfficientNetB4

CNN architecture EfficientNetB4 was created to minimize the amount of computing power needed for training and deployment while achieving good accuracy on image recognition (Park et al., 2022). After being trained on the ImageNet dataset, EfficientNetB4 performs well on several image recognition benchmarks. Multiple pooling layers with activation functions are included in the model design. With fewer parameters and increased training efficiency, it additionally combines depthwise and pointwise convolutions. Because of its strong transferability to different tasks and datasets, EfficientNetB4 is a valuable tool for TL. However, for effective deployment and training, specific hardware could be needed.

3.9.5 MobileNet

MobileNet uses depth-wise separable convolutions to significantly reduce the number of parameters compared to standard convolutions of the same depth (Srinivasu et al., 2021; Zhu, 2024). Consequently, lightweight deep neural networks are generated. Mobile networks are built using depth-wise separable convolution layers. A pointwise convolution layer plus a depth-wise convolution layer make up each depth-wise detachable convolution layer. MobileNet has 28 layers. Through the manipulation of the width multiplier hyperparameter, a conventional MobileNet can contain as few as 4.2 million parameters. The input image measures 224 by 224 pixels.

3.9.6 InceptionV3

The Inception-V3 model optimizes the network using multiple methods for increased model adaptability (Mujahid et al., 2022). As compared to the V1 and V2 inception models, V3 has a larger network. The DNN model Inception-V3 is trained directly on lesser parameters.

3.10 Evaluation parameters

The evaluation phase, which includes evaluating learning models' performance, is critical to performance analysis. Standard assessment measures such as F1 score, recall, accuracy, and precision are used to evaluate FacialNet model performance (Umer et al., 2022).

Accuracy is the most commonly used performance metric. It is just the ratio of observations that were successfully predicted to all observations. It works well with problems involving binary and multi-class classification.

$$Accuracy = \frac{(\text{Number of correct predictions})}{(\text{Total number of predictions})} \quad (6)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

where *TP* and *TN* show true positives and true negatives, respectively while *FP* and *FN* indicate false positives and false negatives, respectively.

Precision is known as positive predictive value, and it is crucial in situations where the expenses associated with false positives are substantial.

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

The percentage of accurately predicted positive observations is known as a recall to all actual class yes observations. When the expense of false negatives is substantial, it matters.

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

The F1 Score is a harmonic mean of precision and recall.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (10)$$

4 Experiments and results

For human emotions classification extensive experiments are carried out. ML and DL models are applied using the six different classes without UNet segmentation, six different classes with UNet segmentation, as well as, the two classes with UNet segmentation. Results and detailed discussion are analyzed in this section.

4.1 Experimental setup and system specifications

The Python 3.9 programming environment is used to conduct the research. The study's experimental setting includes the computer language Python 3.8, (Scikit learn version Version 1.5.0 and TensorFlow version r2.15), RAM capacity available (8GB DDR4), operating system type (64-bit Windows 11), CPU specifications are Intel Core i7 with a processor frequency at about 2.8 GHz which belongs to the 7th generation and an Nvidia GTX1060 GPU. This information is relevant for comprehending the technical characteristics of the research setting and the computational resources employed in this study. The ML classifiers' performances were evaluated using various performance evaluation metrics.

TABLE 3 Without UNet segmentation six classes (happy, sad, fear, pain, anger and disgust) classification.

Model	Accuracy	Precision	Recall	F1-score
LR	72.40	73.24	73.39	73.31
RF	70.22	71.37	72.42	72.40
ETC	70.42	71.44	72.14	71.89
SVM	72.49	71.93	70.64	71.09
CNN	74.57	75.74	75.89	75.81
VGG19	69.89	72.17	73.76	72.99
ResNET	73.24	72.52	73.34	72.86
EfficientNetB4	81.56	80.96	81.23	81.19
MobileNet	80.15	81.67	80.54	81.09
InceptionV3	79.95	80.25	80.25	80.25

4.2 Results of multiclass without UNet segmentation

The first stage of the experiments involved six classes including happy, sad, fear, pain, anger, and disgust classes without UNet segmentation and ML and TL models. Table 3 provides a summary of the models' performance.

The study showed that TL models EfficientNetB4, MobileNet, and InceptionV3 had the highest accuracy rates. EfficientNetB4 outperformed all other models in accuracy. Overall, the performance of all models using six emotion classes without UNet segmentation data was unsatisfactory.

4.3 Results of models on multiclass with UNet segmentation

Regarding the subsequent series of tests, the six-class classification with UNet segmentation in the dataset is used. Many ML and DL models were trained and tested using the resultant dataset. Table 4 illustrates the performance of various models with the UNet segmentation. It shows that ML models have shown the lowest results as compared to the DL and TL models respectively.

Table 4 presents the performance metrics of various models, ranging from traditional machine learning models like LR and RF to advanced deep learning models such as CNN, VGG19, ResNet, and EfficientNetB4 for emotion recognition tasks. Traditional models like LR and RF achieved relatively low accuracies of 79.56 and 78.16%, respectively, as these models have limited capacity to capture complex patterns in high-dimensional image data. The ETC performed slightly better with an accuracy of 81.17%, and SVM further improved with 80.94%, which can be attributed to its strong ability to handle high-dimensional spaces.

In contrast, DL models showed a marked improvement, with CNN achieving 84.74% accuracy, demonstrating its effectiveness in capturing spatial hierarchies in images. ResNet and InceptionV3 continued this trend, with ResNet scoring 85.67% and InceptionV3 reaching 90.07%, thanks to their deeper architectures and

TABLE 4 With UNet segmentation six classes (happy, sad, fear, pain, anger and disgust) classification.

Models	Accuracy	Precision	Recall	F1-score
LR	79.56	80.37	80.93	80.77
RF	78.16	79.73	80.24	79.97
ETC	80.46	81.17	81.14	81.15
SVM	80.94	81.48	80.17	81.11
CNN	84.74	85.24	85.30	85.27
VGG19	79.25	79.71	80.24	80.15
ResNET	83.36	85.17	85.67	85.42
EfficientNetB4	90.11	90.34	91.27	91.05
MobileNet	88.53	89.86	89.94	89.90
InceptionV3	89.66	90.11	90.07	90.10

additional design elements like residual connections and inception modules. EfficientNetB4, however, stood out with the highest accuracy of 91.27%, due to its compound scaling approach that optimally balances network depth, width, and resolution. MobileNet, while designed for mobile devices, still performed well with 89.94% accuracy, though it trailed behind EfficientNet. VGG19, on the other hand, performed similarly to traditional models with an accuracy of 79.71%, suggesting that depth alone is not enough for superior performance, and additional architectural innovations are essential. Overall, deep learning models, especially EfficientNetB4, significantly outperformed traditional methods in the emotion recognition task.

The integration of UNet segmentation in the proposed model significantly enhances the overall performance by providing a refined preprocessing step that isolates relevant facial features, enabling the classifier to focus on key regions of interest related to emotions. The UNet architecture's encoder-decoder design excels in capturing detailed spatial features, enabling precise segmentation of facial regions such as eyes, mouth, and forehead, which are essential for emotion detection. This enhances the input quality for subsequent models, like EfficientNet, leading to improved accuracy and generalization. However, segmentation does come with its challenges. The final classification accuracy is significantly affected by segmentation quality; poor outcomes can cause errors or omit critical emotional cues, especially in situations with occlusions, varying lighting, or unusual facial expressions. Additionally, segmentation increases computational demands, potentially impacting real-time performance in practical applications. Yet, when used with advanced models like EfficientNet, the precise feature extraction benefits generally surpass the computational costs, making it advantageous for detailed facial analysis tasks like Facial Emotion Recognition (FER). The evaluation showed that the transfer learning models EfficientNetB4, MobileNet, and InceptionV3 classifiers attained accuracy rates of 90.11%, 88.53%, and 89.66%, respectively. The results demonstrated significant enhancements in learning model performance using UNet segmentation across the six classes, with noticeable improvements in machine learning models on the segmented dataset compared to data without UNet segmentation. The EfficientNetB4 model

TABLE 5 Binary class (“positive,” and “negative”) classification with UNet segmentation.

Models	Accuracy	Precision	Recall	F1 score
RF	87.55	88.64	88.19	88.42
LR	89.24	90.54	90.48	90.52
ETC	90.22	91.29	91.41	91.34
SVM	90.57	91.14	90.79	91.04
CNN	92.24	93.25	93.90	93.52
VGG19	89.64	90.56	90.42	90.49
ResNET	94.44	95.42	94.19	94.88
EfficientNetB4	96.39	96.88	97.39	97.27
MobileNet	94.28	93.10	94.04	93.97
InceptionV3	95.24	94.87	95.17	95.11

achieved the highest accuracy. Furthermore, its precision, recall, and F1 scores were 90.34%, 91.27%, and 91.05% respectively. The RF linear model showed the lowest accuracy at 78.16%. The development of the FacialNet model using UNet coupled with EfficientNetB4 leverages the distinct advantages of each framework for emotion recognition and mental health evaluation. UNet is adept at segmenting images to pinpoint crucial facial features indicative of emotional states. Its segmentation accuracy allows for a focused examination of important facial areas. Conversely, EfficientNetB4, with its effective feature extraction and pre-trained weights, captures intricate patterns in high-dimensional facial datasets efficiently. This hybrid model, fusing UNet’s segmentation skills with EfficientNetB4’s feature extraction capabilities, achieves a commendable balance between accuracy and computational cost-ideal for emotion recognition tasks in mental health contexts.

4.4 Binary classification results with UNet segmentation

The experiments involved binary classification on datasets segmented with UNet. Given the dataset contains six classes, the outcomes for multi-class classification were unsatisfactory. Consequently, So, in this set of experiments we treated two classes such as happy and sad (Sad, Fear, Pain, Anger, and Disgust) classes, and performed the binary classification on this type of dataset with UNet segmentation results of the learning models are shown in [Table 5](#).

Results of the experiments show that the proposed EfficientNetB4 with the UNet segmentation features outperformed the other learning models and achieved an accuracy of 96.39%. Followed by InceptionV3 achieved an accuracy of 95.24%. The proposed TL model EfficientNetB4 achieved the highest value for the other evaluation parameters, 96.88% precision, 97.39% recall, and an F1 score of 97.27%. In this part of the experiment, a notable improvement in the performance of ResNet is noted. In this part of the experiment ML model, RF is the least performer and achieved an accuracy of 87.55%. Overall, there is a significant improvement

TABLE 6 Paired *t*-test results between the models.

Model Comparison	<i>t</i> -statistic	<i>p</i> -value
EfficientNetB4 vs. MobileNet	8.75	0.0031
EfficientNetB4 vs. InceptionV3	7.57	0.0048

TABLE 7 UNet segmentation cross-validation results.

Models	Accuracy	Precision	Recall	F1-score
First-fold	96.33	96.54	97.34	97.62
Second-fold	97.84	97.67	98.58	98.59
Third-fold	97.77	97.19	97.49	97.49
Fourth-fold	96.68	97.49	97.59	97.38
Fifth-fold	96.71	97.88	97.89	97.49
Average	96.68	97.45	97.52	97.50

in the performance of the learning classifiers used for the emotions classification is noted.

4.5 Results for statistical significance test

To further evaluate the differences in performance among the models, a paired *t*-test was conducted between EfficientNetB4, MobileNet, and InceptionV3. The results demonstrated in [Table 6](#) show that the difference in performance is statistically significant. EfficientNetB4 significantly outperforms both MobileNet and InceptionV3 across accuracy, precision, recall, and F1-score. The *p*-values for both comparisons are below 0.05, confirming that EfficientNetB4 provides a substantial performance improvement over the other models.

4.6 Cross-validation technique results

One method for assessing how well ML algorithms work is cross-validation. Although there are other cross-validation techniques, *k*-fold cross-validation is preferred because it is well-liked, simple to comprehend, and typically produces less bias than the other techniques.

The choice of *k* = 5 strikes a balance between computational efficiency and reliable estimation. Larger *k*-values (such as 10) would provide even more precise performance estimates but at the cost of increased training time. On the other hand, smaller *k*-values (such as 2 or 3) might not provide enough diversity in training and validation sets, leading to less reliable generalization performance. By choosing *k* = 5, the method ensures a comprehensive evaluation while keeping computational demands manageable, enhancing the robustness and reliability of the reported results.

The data set is split into *k* equal-sized portions for *k*-fold cross-validation. The first *k* groups are used to train the classifiers, while the remaining portion is utilized to evaluate outperformance at each stage. There are *k* repetitions of the validation process. Based on *k* outcomes, the classifier performance is calculated. Various

TABLE 8 Performance comparison with previous approaches.

References	Classifiers	Accuracy	Limitations
Qu et al. (2023)	CNN	60.20%	No preprocessing, No multi-class classification, No segmentation, No cross-validation
Boughida et al. (2022)	Gabor filter	94.26%	No preprocessing, No multi-class classification, No segmentation, No cross-validation
Gubbala et al. (2023)	ARFEC	92.5 %	No preprocessing, No segmentation, No cross-validation
Oguine et al. (2022)	CNN + haar cascade	70.04%	No preprocessing, No multi-class classification, No segmentation, No cross-validation
Gupta and Jain (2021)	CNN with LeNet	60.37%	No preprocessing, No segmentation, No cross-validation
Lucey et al. (2010)	SVM with CAPP features	80.00%	No preprocessing, No segmentation, No cross-validation
Huang et al. (2019)	AlexNet	87.79%	No preprocessing, No segmentation, No cross-validation
Anilkumar et al. (2024)	DCNN-HPO	83.40%	No multi-class classification, No cross-validation
Proposed	EfficientNetB4	96.39%	Computational complexity

Bold values indicating the results of the proposed approach which are better than all previously published research works.

values of k are chosen for cross-validation. Since $k=5$ performs well, we employed it in the experiments. 90% of the data in the five-fold cross-validation procedure were used for training, while 10% were used for testing. All occurrences in the training and test groups were randomly distributed throughout the whole dataset before selecting, training, and testing fresh sets for the following cycle. This process was performed five times for each fold of the process. Finally, averages of all performance metrics are calculated after the five-fold process. The suggested TL model EfficientNetB4 yields a mean accuracy score of 96.68, and average scores of 97.45, 97.52, and 97.50 for precision, recall, and F1, in that order, according to the cross-validation results displayed in Table 7.

4.7 Performance comparison with previous approaches

In Table 8, we performed a comparison with other studies that have previously worked on facial emotion classification to highlight the importance of the suggested method. The accuracy attained in those earlier tests showed a significant gap, indicating that there is a great deal of space for accuracy improvement. The majority of earlier research leveraged TL approaches and concentrated on using the facial expression image dataset directly. This study, on the other hand, made use of UNet segmentation characteristics that were taken from an image dataset. The suggested EfficientNetB4 model produced highly significant results by utilizing this feature set, as seen in the Table 8. This indicates that the suggested method is effective in outperforming the results of earlier research and points to the possibility of more developments in this area.

Facial emotion recognition using CNN in Qu et al. (2023) utilizes a traditional CNN architecture, achieving moderate performance. In contrast, FacialNet’s combination of UNet segmentation and EfficientNetB4 enhances feature extraction accuracy, leading to superior results, particularly for complex emotion datasets. Similarly, Meena and Mohbey (2023) explore transfer learning models for sentiment analysis, but their lack of specialized segmentation, like UNet, limits their performance. FacialNet’s segmentation provides more refined facial feature representation, achieving 96% accuracy for binary classification.

The study by Boughida et al. (2022) uses Gabor filters and a genetic algorithm for feature selection, which, although effective, falls short in real-time applications where FacialNet’s deep learning approach offers faster and more accurate predictions. AdaBoost-based RF models in Gubbala et al. (2023) exhibit inferior performance when compared to the cutting-edge deep learning architecture of FacialNet, which employs the EfficientNetB4 backbone combined with UNet segmentation to achieve notably superior outcomes. In addition, although Oguine et al. (2022) presents a hybrid FER model, it lacks the use of advanced segmentation strategies, resulting in inferior emotion classification accuracy compared to FacialNet, which outperforms due to its precise segmentation and broad emotion classification capabilities.

4.8 Practical implications of proposed approach

This research significantly enhances the field of facial emotion recognition, with notable applications in mental health assessments and emotion-aware technologies. The integration of UNet segmentation with EfficientNet boosts accuracy by capturing detailed facial nuances, rendering the model highly suitable for practical applications such as telemedicine and adaptive educational settings. Future advancements could involve testing with varied datasets, inclusion of multi-modal data, and real-time optimization for use on mobile or embedded platforms. Additional future research directions might focus on further optimizing the model design, investigating supplementary features, and assessing larger, more varied datasets to improve the model’s resilience and ability to generalize effectively.

5 Conclusion

This research proposes FacialNet for human FER using UNet image segmentation in conjunction with TL utilizing the EfficientNetB4 model. The proposed model has demonstrated impressive performance, achieving an accuracy score of 90% for six emotion classes including happy, sad, fear, pain, anger,

and disgust, and 96% for binary classification including positive and negative classes. Through extensive experimentation and comparison with other ML and DL models, as well as state-of-the-art previous research works, we have validated the effectiveness and superiority of our proposed approach. Furthermore, the robustness and generalization capability of the proposed model have been thoroughly evaluated using a five-fold cross-validation technique. This validation methodology ensures the reliability and consistency of our results across different data splits, highlighting the significance and reliability of the proposed approach. The findings indicate that leveraging UNet image segmentation and EfficientNetB4 TL yields promising outcomes in the domain of FER, paving the way for the development of more accurate and efficient emotion recognition systems in various real-world applications.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

I-SN: Writing – review & editing. AA: Writing – review & editing. AH: Writing – review & editing, Writing – original draft, Visualization, Validation, Project administration, Methodology, Data curation, Conceptualization. LM: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Methodology, Funding acquisition, Data curation, Conceptualization. MU: Writing – original draft, Validation, Software, Methodology, Conceptualization. DA: Writing – original draft, Validation, Software, Resources, Methodology, Funding acquisition, Formal analysis, Data curation. SA: Writing – original draft, Visualization, Validation, Software, Resources, Project administration, Methodology, Conceptualization. NI: Writing – review & editing, Validation, Supervision, Investigation, Funding acquisition, Data curation, Conceptualization. IA: Writing – review & editing, Visualization,

Validation, Supervision, Software, Project administration, Methodology, Funding acquisition, Conceptualization.

Funding

The authors extend their appreciation to the Deanship of Research and Graduate Studies at King Khalid University for funding this work through small group research under grant number RGP1/296/45. This work was supported through Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R508), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Nisreen Innab would like to express sincere gratitude to AlMaarefa University, Riyadh, Saudi Arabia, for supporting this research.

Acknowledgments

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This research was supported by the Korea Institute of Marine Science & Technology Promotion(KIMST), by the Ministry of Oceans and Fisheries (RS-2022-KS221676).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Anilkumar, B., Lakshmi Devi, N., Kotagiri, S., and Mary Sowjanya, A. (2024). Design an image-based sentiment analysis system using a deep convolutional neural network and hyperparameter optimization. *Multimed. Tools Appl.* 83, 1–20. doi: 10.1007/s11042-024-18206-y
- Boughida, A., Kouahla, M. N., and Lafifi, Y. (2022). A novel approach for facial expression recognition based on gabor filters and genetic algorithm. *Evol. Syst.* 13, 331–345. doi: 10.1007/s12530-021-09393-2
- Canal, F. Z., Müller, T. R., Matias, J. C., Scotton, G. G., de Sa Junior, A. R., Pozzebon, E., et al. (2022). A survey on facial emotion recognition techniques: a state-of-the-art literature review. *Inf. Sci.* 582, 593–617. doi: 10.1016/j.ins.2021.10.005
- Cao, X., Wang, Z., Chen, Y., and Zhu, J. (2024). Childhood maltreatment and resting-state network connectivity: the risk-buffering role of positive parenting. *Dev. Psychopathol.* 1–12. doi: 10.1017/S0954579424000725
- Ding, J., Chen, X., Lu, P., Yang, Z., Li, X., Du, Y., et al. (2023). Dialogueinab: an interaction neural network based on attitudes and behaviors of interlocutors for dialogue emotion recognition. *J. Supercomput.* 79, 20481–20514. doi: 10.1007/s11227-023-05439-1
- Gubbala, K., Kumar, M. N., and Sowjanya, A. M. (2023). Adaboost based random forest model for emotion classification of facial images. *MethodsX* 11:102422. doi: 10.1016/j.mex.2023.102422
- Gupta, S., and Jain, S. (2021). Feeling recognition by facial expression using deep learning. *J. Phys. Conf. Ser.* 1717:012053. doi: 10.1088/1742-6596/1717/1/012053
- Haider, I., Yang, H. J., Lee, G. S., and Kim, S. H. (2023). Robust human face emotion classification using triplet-loss-based deep cnn features and SVM. *Sensors* 23:4770. doi: 10.3390/s23104770
- Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., and Scholkopf, B. (1998). Support vector machines. *IEEE Intell. Syst. Appl.* 13, 18–28. doi: 10.1109/5254.708428

- Huang, C.-C., Wu, Y.-L., and Tang, C.-Y. (2019). "Human face sentiment classification using synthetic sentiment images with deep convolutional neural networks," in *2019 International Conference on Machine Learning and Cybernetics (ICMLC)* (Kobe: IEEE), 1–5. doi: 10.1109/ICMLC48188.2019.8949240
- Huang, Z.-Y., Chiang, C.-C., Chen, J.-H., Chen, Y.-C., Chung, H.-L., Cai, Y.-P., et al. (2023). A study on computer vision for facial emotion recognition. *Sci. Rep.* 13:8425. doi: 10.1038/s41598-023-35446-4
- Krithika Alias AnbuDevi, M., Suganthi, K. (2022). Review of semantic segmentation of medical images using modified architectures of unet. *Diagnostics* 12:3064. doi: 10.3390/diagnostics12123064
- Lindquist, K. A., Jackson, J. C., Leshin, J., Satpute, A. B., and Gendron, M. (2022). The cultural evolution of emotion. *Nat. Rev. Psychol.* 1, 669–681. doi: 10.1038/s44159-022-00105-4
- Lu, J., Tan, L., and Jiang, H. (2021). Review on convolutional neural network (cnn) applied to plant leaf disease classification. *Agriculture* 11:707. doi: 10.3390/agriculture11080707
- Lucey, P., Cohn, J., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I., et al. (2010). "The extended cohn-kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops* (San Francisco, CA: IEEE). doi: 10.1109/CVPRW.2010.5543262
- Manzoor, M., Umer, M., Sadiq, S., Ishaq, A., Ullah, S., Madni, H. A., et al. (2021). Rfcnn: traffic accident severity prediction based on decision level fusion of machine and deep learning model. *IEEE Access* 9, 128359–128371. doi: 10.1109/ACCESS.2021.3112546
- Meena, G., and Mohbey, K. K. (2023). Sentiment analysis on images using different transfer learning models. *Procedia Comput. Sci.* 218, 1640–1649. doi: 10.1016/j.procs.2023.01.142
- Mohamed, Y. (2024). *6 Human Emotions for image classification*. Kaggle. Available at: <https://www.kaggle.com/datasets/yousefmohamed20/sentiment-images-classifier> (accessed April 04, 2024).
- Mujahid, M., Rustam, F., Álvarez, R., Luis Vidal Mazón, J., Díez, I. T., and Ashraf, I. (2022). Pneumonia classification from X-ray images with inception-v3 and convolutional neural network. *Diagnostics* 12:1280. doi: 10.3390/diagnostics12051280
- Oguine, O. C., Oguine, K. J., Bisallah, H. I., and Ofuani, D. (2022). Hybrid facial expression recognition (fer2013) model for real-time emotion classification and prediction. *arXiv [Preprint]*. arXiv:2206.09509. doi: 10.4850/arXiv.2206.09509
- Park, S.-J., Ko, T., Park, C.-K., Kim, Y.-C., and Choi, I.-Y. (2022). Deep learning model based on 3d optical coherence tomography images for the automated detection of pathologic myopia. *Diagnostics* 12:742. doi: 10.3390/diagnostics12030742
- Qu, D., Dhakal, S., and Carrillo, D. (2023). Facial emotion recognition using cnn in pytorch. *arXiv [Preprint]*. arXiv:2312.10818. doi: 10.48550/arXiv.2312.10818
- Rajinikanth, V., Joseph Raj, A. N., Thanaraj, K. P., and Naik, G. R. (2020). A customized vgg19 network with concatenation of deep and handcrafted features for brain tumor detection. *Appl. Sci.* 10:3429. doi: 10.3390/app10103429
- Rymarczyk, T., Kozłowski, E., Kłosowski, G., and Niderla, K. (2019). Logistic regression for machine learning in process tomography. *Sensors* 19:3400. doi: 10.3390/s19153400
- Sarvakar, K., Senkamalavalli, R., Raghavendra, S., Kumar, J. S., Manjunath, R., Jaiswal, S., et al. (2023). Facial emotion recognition using convolutional neural networks. *Mater. Today Proc.* 80, 3560–3564. doi: 10.1016/j.matpr.2021.07.297
- Song, W., Wang, X., Jiang, Y., Li, S., Hao, A., Hou, X., et al. (2024). Expressive 3d facial animation generation based on local-to-global latent diffusion. *IEEE Trans. Vis. Comput. Graph.* 30, 7397–7407. doi: 10.1109/TVCG.2024.3456213
- Srinivasu, P. N. SivaSai, J. G., Ijaz, M. F., Bhoi, A. K., Kim, W., Kang, J. J. (2021). Classification of skin disease using deep learning neural networks with mobilenet v2 and lstm. *Sensors* 21:2852. doi: 10.3390/s21082852
- Talaat, F. M. (2023). Real-time facial emotion recognition system among children with autism based on deep learning and iot. *Neural Comput. Appl.* 35, 12717–12728. doi: 10.1007/s00521-023-08372-9
- Umer, M., Sadiq, S., Nappi, M., Sana, M. U., Ashraf, I., et al. (2022). ETCNN: extra tree and convolutional neural network-based ensemble model for covid-19 tweets sentiment classification. *Pattern Recognit. Lett.* 164, 224–231. doi: 10.1016/j.patrec.2022.11.012
- Vignesh, S., Savithadevi, M., Sridevi, M., and Sridhar, R. (2023). A novel facial emotion recognition model using segmentation vgg-19 architecture. *Int. J. Inf. Technol.* 15, 1777–1787. doi: 10.1007/s41870-023-01184-z
- Yaqoob, M. K., Ali, S. F., Bilal, M., Hanif, M. S., and Al-Saggaf, U. M. (2021). Resnet based deep features and random forest classifier for diabetic retinopathy detection. *Sensors* 21:3883. doi: 10.3390/s21113883
- Zhang, J., Yin, Z., Chen, P., and Nichele, S. (2020). Emotion recognition using multi-modal data and machine learning techniques: a tutorial and review. *Inf. Fusion* 59, 103–126. doi: 10.1016/j.inffus.2020.01.011
- Zhu, C. (2023). Research on emotion recognition-based smart assistant system: Emotional intelligence and personalized services. *J. Syst. Manag. Sci.* 13, 227–242. doi: 10.33168/JSMS.2023.0515
- Zhu, C. (2024). Computational intelligence-based classification system for the diagnosis of memory impairment in psychoactive substance users. *J. Cloud Comput.* 13:119. doi: 10.1186/s13677-024-00675-z



OPEN ACCESS

EDITED BY

Jussi Tohka,
University of Eastern Finland, Finland

REVIEWED BY

Inam Ullah,
Gachon University, Republic of Korea
Chen Li,
King's College London, United Kingdom

*CORRESPONDENCE

Gang Wang
✉ gang.wang@uestc.edu.cn

[†]These authors have contributed equally to this work

RECEIVED 20 June 2024

ACCEPTED 06 January 2025

PUBLISHED 22 January 2025

CITATION

Zheng J, Wan Y, Yang X, Zhong H, Du M and Wang G (2025) Motion feature extraction using magnocellular-inspired spiking neural networks for drone detection.
Front. Comput. Neurosci. 19:1452203.
doi: 10.3389/fncom.2025.1452203

COPYRIGHT

© 2025 Zheng, Wan, Yang, Zhong, Du and Wang. This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY\)](#). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Motion feature extraction using magnocellular-inspired spiking neural networks for drone detection

Jiayi Zheng^{1,2†}, Yaping Wan^{1†}, Xin Yang², Hua Zhong¹, Minghua Du³ and Gang Wang^{2,4*}

¹Department of Computer, University of South China, Hengyang, China, ²Center of Brain Sciences, Beijing Institute of Basic Medical Sciences, Beijing, China, ³Department of Emergency, The First Medical Center, Chinese PLA General Hospital, Beijing, China, ⁴Chinese Institute for Brain Research, Beijing, China

Traditional object detection methods usually underperform when locating tiny or small drones against complex backgrounds, since the appearance features of the targets and the backgrounds are highly similar. To address this, inspired by the magnocellular motion processing mechanisms, we proposed to utilize the spatial-temporal characteristics of the flying drones based on spiking neural networks, thereby developing the Magno-Spiking Neural Network (MG-SNN) for drone detection. The MG-SNN can learn to identify potential regions of moving targets through motion saliency estimation and subsequently integrates the information into the popular object detection algorithms to design the retinal-inspired spiking neural network module for drone motion extraction and object detection architecture, which integrates motion and spatial features before object detection to enhance detection accuracy. To design and train the MG-SNN, we propose a new backpropagation method called Dynamic Threshold Multi-frame Spike Time Sequence (DT-MSTS), and establish a dataset for the training and validation of MG-SNN, effectively extracting and updating visual motion features. Experimental results in terms of drone detection performance indicate that the incorporation of MG-SNN significantly improves the accuracy of low-altitude drone detection tasks compared to popular small object detection algorithms, acting as a cheap plug-and-play module in detecting small flying targets against complex backgrounds.

KEYWORDS

bio-inspired vision computation, spiking neural networks, motion detection, drone target recognition, motion saliency estimation, visual motion features

1 Introduction

The rapid development of unmanned aerial vehicle technology has led to the wide use of small civilian drones for various tasks such as security patrols, agricultural monitoring, and disaster relief. However, there is also misuse of drones for illegal activities such as smuggling contraband, espionage mapping, and close-range reconnaissance, posing a significant threat to public safety (AL-Dosari et al., 2023). Therefore, it is crucial to develop an early warning detection system for low-altitude, short-range small drones. Traditional radar detection methodologies encounter challenges in identifying small drones due to their limited radar cross-section, low operational altitude, slow velocity, and inclination to conceal within intricate backgrounds, rendering them susceptible to ground clutter interference (Abro et al., 2022).

Recent studies have shown the potential of advanced communication and machine learning approaches in improving UAV detection capabilities and reducing interference from complex backgrounds (Khalil et al., 2022). Conversely, optoelectronic sensors, encompassing the infrared and visible light spectra, prove more adept at detecting short-range, low-altitude drone targets in complex settings, and the image and video data it captures need to be further processed using object detection to output results.

Previous research on drone detection has employed various techniques, primarily developed based on deep neural networks. These techniques are classified into two-stage and single-stage algorithms, depending on whether candidate regions are explicitly generated. Two-stage methods, such as the Faster R-CNN (Ren et al., 2015), have achieved success, although they require substantial computational resources and have certain limitations in real-time processing. In contrast, single-stage algorithms, represented by methods like YOLO (Redmon et al., 2016) and SSD (Liu et al., 2016), offer faster detection speeds but lower accuracy. These models perform effectively on static images and general large-scale datasets but often struggle to identify small targets in cluttered and dynamic environments, particularly due to the information loss associated with small targets. Issues such as motion blur, object occlusion, lighting variations, angle changes, and device defocusing in video object detection highlight the necessity for more efficient and accurate methods for detecting small drones in complex environments (Jiao et al., 2021). To address the above challenges, gaining an understanding of the operational mechanisms of the biological retina (Yücel et al., 2003) offers valuable insights. Serving as the initial stage in visual information processing, the biological retina is responsible for converting optical stimuli into electrical signals. These signals undergo preliminary processing by the retinal neuron network before being transmitted to the output neurons of the retina—ganglion cells. Ultimately, they are transformed into action potentials and conveyed to the visual center via the optic nerve. The biological retina is endowed with highly specialized functions, encompassing high-resolution color perception, swift response to dynamic images, and effective processing of intricate scenes. These attributes equip the retina to manage a wide range of visual information, facilitating complex visual tasks such as motion detection, depth perception, and image segmentation (Neuroscience, 2020).

Despite drawing inspiration from the biological visual system for feature extraction and hierarchical processing, traditional visual perception algorithms struggle to adapt to swift-moving targets or intricate backgrounds, particularly within dynamic environments where erroneous detections are prevalent. Unlike conventional artificial neural networks, the biological retina possesses the ability to directly process dynamic temporal information and adjust to complex environments through mechanisms such as neural plasticity, a trait that proves challenging to completely replicate (Wohrer and Kornprobst, 2009; Hagsins, 1972; Field and Chichilnisky, 2007; Beaudot et al., 1993). Bio-inspired models that emulate the workings of the biological retina offer improved capabilities in extracting motion features, thereby elevating the precision and dependability of object detection.

Spiking Neural Networks (SNNs), recognized as the third generation of neural networks (Maass, 1997), are computational models that closely emulate biological neural networks by processing information through the spiking activity of neurons. Unlike conventional Artificial Neural Networks (ANNs), SNN neurons

communicate using binary events rather than continuous activation values. This approach not only mirrors the structure and function of the biological retina but also encodes and transmits information through the processing of temporal spike sequences, displaying spatiotemporal dynamic characteristics. This intricate activity pattern enables the system to maintain overall stability while adapting to environmental changes and acquiring new motion information through plasticity mechanisms, mirroring the visual filtering observed in biological systems. Due to their event-driven nature, SNNs can more accurately utilize energy when processing sensor data similar to the retina (Jang et al., 2019), which is particularly useful for drone applications (Dupeyroux et al., 2021; Sanyal et al., 2024). They can be applied to tasks requiring real-time or edge computing and can integrate with neuromorphic processors (Calimera et al., 2013) to achieve rapid response in challenging scenarios. Recent years have witnessed the versatility and efficiency of SNNs across diverse domains (Mehonic et al., 2020; Kim et al., 2020), notably excelling in speech recognition (Wu et al., 2020; Wu et al., 2018), image classification (Kim et al., 2022; Vaila, 2021; Zhu et al., 2024), sensory fusion (Glatz et al., 2019), motion control (Glatz et al., 2019), and optical flow computation (Gehrig et al., 2020; Ponghiran et al., 2022). Compared to earlier methods such as Convolutional Neural Networks (CNNs) and optical flow techniques, SNNs provide a more biologically plausible and energy-efficient solution, particularly well-suited for feature extraction of small drones in scenarios where rapid adaptation to environmental changes is crucial, and facilitates a synergistic balance between the efficient encoding and processing of visual information and biological authenticity.

In time-sensitive scenarios, the incorporation of motion features proves advantageous for visual perception tasks, particularly in the context of processing temporal information and its implications for learning mechanisms. Currently, there is no research utilizing SNNs to model the dynamic visual information processing mechanisms of the retina and apply the motion information extracted by SNNs to drone object detection tasks. To address this issue and achieve both biological realism and efficiency in handling complex dynamic visual tasks, we introduce dynamic temporal information into the retinal output model. We have devised a primary motion saliency estimation algorithm, exclusively comprising an SNN architecture, serving as a visual motion perception model to emulate the processing and output of dynamic information by the biological retina in visual perception tasks. The acquired motion information is subsequently amalgamated with spatial information for utilization in drone object detection tasks.

Our research encompasses several key aspects: First, we develop a magnocellular pathway dataset based on the biological characteristics of the retinal magnocellular pathway computational model. Second, we investigate how SNNs encode and transmit temporal information through spike sequences, emulating the biological retina extraction of dynamic visual information. Third, we propose a biologically inspired visual motion perception model, referred to as the Magnospiking Neural Network (MG-SNN). This model comprises a computational framework solely using spiking neural networks to process visual information, acting as a primary motion saliency estimation model aligned with the retinal magnocellular pathway. We validate the accuracy of the SNN model in extracting motion features. Finally, the MG-SNN is used as a motion feature extraction module, which is combined with the object detection model to form a target detection framework, and the experimental results indicate

that the framework can accurately identify and detect low-altitude drone targets.

Specifically, the main contributions of this paper are summarized as follows:

- This research is the first attempt (to our knowledge) to effectively simulate the magnocellular function of extracting motion features of objects using a two-layer spiking neural network framework, as a motion detection plug-in geared towards object detection;
- Experimental validation shows that the MG-SNN model closely matches biological retina processing and enhances object detection accuracy and reliability, demonstrating the potential of biologically inspired SNN models in drone detection;
- In conjunction with the magnocellular pathway computational model, we design the Visual-Magnocellular Dynamics Dataset (VMD) for supervised learning of motion features. The MG-SNN, combined with popular traditional object detection models, improves small drone detection performance in complex backgrounds.

The remainder of this paper is organized as follows. In Section 2, we introduce the related work on motion saliency computation and motion object detection, biologically inspired retinal models, and spiking neural networks for visual tasks. In Section 3, we present the retinal-inspired spiking neural network module for drone motion extraction and object detection architecture. This includes introducing the magnocellular pathway dataset inspired by the retinal magnocellular pathway computational model, explaining the proposed spike temporal encoding method for processing input video frames, and discussing in detail the primary motion saliency estimation model MG-SNN based on the SNN architecture, along with theoretical derivations and feasibility explanations of the proposed method. In Section 4, we describe the comparative experimental conditions and evaluation methods for motion feature extraction and object detection, followed by a thorough discussion and analysis of the experimental results. Finally, in Section 5, we provide a summary of the entire paper.

2 Related work

2.1 Motion saliency computation and motion object detection

The initial research into motion saliency calculation initially emphasized single visual cues, such as motion speed or direction. However, these methods often lacked adaptability to rapidly changing scenes. Researchers utilized techniques like Support Vector Machines (SVM) to improve the prediction of salient motion areas, but these approaches incurred substantial computational loads when handling large-scale video data. In recent years, composite models (Wang et al., 2017; Bi et al., 2021) have gained traction by integrating multiple visual cues to enhance overall system performance. Notably, models (Maczyta et al., 2019) have been employed to extract motion saliency over video segments, leveraging their exceptional feature learning capabilities for dynamic scene analysis. Similarly, Guo et al. (2019) calculated motion saliency between adjacent frames by analyzing

optical flow fields to obtain foreground priors. They utilized a multi-cue framework to integrate various saliency cues and achieve temporal consistency.

In the early research on moving object detection, traditional algorithms focused on simple techniques such as background subtraction and threshold processing. For instance, reference (Yang et al., 2012) utilized dynamic thresholds to compensate for the shortcomings of fixed-threshold background subtraction, enabling timely background updates and overcoming the limitations of traditional background update methods. While these techniques are straightforward to implement, their performance in dynamic backgrounds is suboptimal and offers limited potential for improvement. With advancements in computational power, methods integrating multiple sensory information (such as motion, color, and geometry) (Bhaskar, 2012; Minaeian et al., 2015) began to be employed to enhance the accuracy and robustness of object detection. Compared to traditional algorithms that detect small target locations through inter-frame target association, deep learning-based methods operate directly on keyframes by generating bounding boxes around targets to detect and track moving objects more effectively in complex environments. For example, methods from the YOLO series (Redmon et al., 2016) and the SSD series (Liu et al., 2016) regress directly on the input image to obtain localization and classification information for motion objects.

The algorithms exhibit certain limitations in the task of motion object detection, potentially resulting in the oversight of smaller or infrequently appearing objects, the loss of temporal information, and insufficient accuracy in dense scenes. In specific situations, they require more computational resources, which may reduce their suitability for highly real-time applications. Our approach departs significantly from previous methodologies by directly integrating biological principles into object motion sensitivity, as opposed to relying on arbitrary network architectures or parametric models. Opting for spiking neural networks over artificial neural networks holds promise in providing a more organic approach to processing visual information, thereby enabling the attainment of detection outcomes that more closely mirror human visual perception.

2.2 Biologically-inspired retinal models

The initial research into motion saliency calculation initially emphasized single visual cues, such as motion speed or direction. However, these methods often lacked adaptability to rapidly changing scenes. In the realm of bio-inspired retinal models, initial research centered on emulating the photoreception and primary processing mechanisms characteristic of the human retina. For instance, Melanitis and Nikita, 2019 explored the simulation of photoreceptors and initial signal processing in computational models of the retina, to replicate the early stages of visual processing observed in biological systems. Recently, researchers have been investigating the potential use of these models in more complex visual tasks for feature extraction and decision support. For example, Aboudib et al. (2016) proposed a bio-inspired framework for visual information processing that specifically focuses on modeling bottom-up visual attention, utilizing the retinal model for testing and theoretical validation.

Given the characteristics of various types of neurons and neural circuits in the retina, researchers have developed a range of models

tailored to distinct task types within the retina-inspired visual motion perception domain. Most of these models are multi-scale CNNs and Recurrent neural network (RNNs) models, constructed to mimic biological visual perception mechanisms. For instance, [Zheng et al., 2021](#) designed a model comprising feed-forward convolutional filters and recurrent units to represent temporal dynamics displayed in continuous natural videos and neural responses within the retina; [McIntosh et al. \(2016\)](#) developed a deep learning model based on CNNs to capture responses to motion stimuli; [Parameshwara \(2022\)](#) designed a retinal-inspired visual sensor model and framework, integrating CNNs and LSTMs to execute motion perception tasks. Moreover, [Lehnert et al. \(2019\)](#) introduced a retinal-inspired visual module encompassing CNN and LSTM layers for navigation perception tasks in complex settings. These models extract motion features from time-series images to identify and analyze motion stimuli and thus might lose temporal information.

Furthermore, attention mechanisms, inspired by the biological visual system, are used to enhance the recognition accuracy of significant motion targets by emulating the retinal focus mechanism on crucial visual features to detect prominent moving objects within videos. [Lukanov et al. \(2021\)](#) proposed an end-to-end model grounded in feature saliency, influenced by the retinal sampling mechanisms observed in primates; the BIT model ([Sokhandan and Monadjemi, 2024](#)) employs a bio-inspired mechanism with an attention mechanism to effectively track targets in video sequences, and [Malowany and Guterman \(2020\)](#) utilize deep feedforward CNNs combined with top-down attention mechanisms from the human visual system for object recognition tasks.

While drawing inspiration from the multilayered visual systems and yielding outputs consistent with visual mechanisms, these models fall short of replicating the information-processing pathways of the human brain. By simplifying complex biological structures and functions to achieve specific capabilities, they do not truly reflect the transmission and processing of information in the temporal dimension.

2.3 Spiking neural networks for visual tasks

Furthermore, attention mechanisms, inspired by the biological visual system, are used to enhance the recognition accuracy of significant motion targets by emulating the retinal focus mechanism on crucial visual features to detect prominent moving objects within videos. Initially, the application of SNNs in visual tasks was primarily focused on basic image and video processing tasks, such as image reconstruction and simple object recognition. These tasks utilized the temporal dynamics of SNNs to mimic the primary stages of visual perception. Despite the tremendous success of CNNs in visual tasks, research into SNNs aims to leverage their event-driven characteristics for encoding information, with the expectation of achieving greater efficiency in power consumption and algorithmic complexity. Most relevant to processing visual motion information is the SpikeMS model ([Parameshwara et al., 2021](#)), which accurately segments and tracks dynamic moving targets in video sequences. This model uses an architecture that combines multilayer CNNs with SNNs to extract spatial features from video sequences and ultimately produces segmentation results for dynamic targets. Similarly, the SpikeFlowNet model ([Lee et al., 2020](#)) utilizes a deep SNN encoder and

an ANN decoder architecture for self-supervised optical flow estimation. Additionally, the U-Net-like SNN model ([Cuadrado et al., 2023](#)) integrates the U-Net architecture with SNN neuron models to extract motion and optical flow information in the temporal dimension, by combining event-based camera data with SNNs for optical flow and depth prediction. Another architecture ([Hagenaars et al., 2021](#)) designed for optical flow estimation processes event data using an ANN-SNN hybrid approach. It is evident that most visual motion perception models related to SNNs are based on hybrid ANN-SNN architectures. Although SNN neurons are introduced to handle the temporal dimension, fully simulating the dynamic behavior of neurons remains challenging, and there is a performance and accuracy loss during the conversion process.

Frame-based images and feature vectors need to be encoded as spike trains to be processed within SNNs. These spike events are non-differentiable, making traditional backpropagation methods challenging to employ. Early attempts at training SNNs focused on biologically inspired Hebbian mechanisms ([Sejnowski and Tesauro, 1989](#)). Spike Time Dependent Plasticity (STDP) ([Mozafari et al., 2018](#)) strengthens synapses that may aid in neuron firing, thus avoiding the gradient issue. In ANN-SNN methods, input representations are formed by binning events within time intervals and converting them into image-based frame structures, referred to as “event frames.” Most dynamic information processed in SNNs also originates from event data generated by Dynamic Vision Sensors (DVS) ([Haessig et al., 2019](#)). However, our method is distinct in that it directly feeds video frames into the network through spike encoding, capitalizing on the temporal properties of SNNs combined with the temporal properties of spike trains. By encoding pixel intensity as spike timing, this approach naturally reduces the processing of redundant information while preserving all significant information, as only notable visual changes trigger spikes.

3 Materials and methods

This section will provide a detailed overview of the learning and inference process of the algorithm developed using SNNs, which simulates the retinal channel process of handling motion information and extracting accurate motion feature information. The extracted visual dynamic features are then used as a motion guidance module applied to drone object detection. First, we will introduce the overall architecture of the retinal-inspired spiking neural network for drone motion extraction and object detection. Then, we will describe the two main components: extracting visual motion information with MG-SNN and applying it to drone object detection. In the first part, the MG-SNN (Magno-Spiking Neural Network) model for motion saliency estimation includes the design of the Visual Magnocellular Dynamics Dataset (VMD), inspired by the computational model of the retinal magnocellular pathway. We will discuss the process of handling multiple frames through a spike temporal encoding strategy. Subsequently, we propose a Dynamic Threshold Multi-frame Spike Time Sequence backpropagation method (DT-MSTS) based on dynamic thresholds and the STDP rule to guide the learning of the SNN network. In the second part, concerning the application to object detection, we will primarily discuss combining MG-SNN with the YOLO model to achieve the task of detecting small drone targets.

3.1 Overall architecture

The overall architecture of the drone motion extraction and object detection system based on a retina-inspired spiking neural network is illustrated in **Figure 1**. In the motion feature extraction module, MG-SNN is constructed by modeling the structure of the biological retina. The input video stream is converted into a temporal spike

sequence using a spike temporal encoding strategy in the photoreceptor simulation layer. These sequences then undergo processing in the inner plexiform layer (IPL). During forward propagation in the IPL, the integrate-and-fire (IF) neurons in each layer integrate the presynaptic spike sequences. By employing a dynamic threshold mechanism, thresholds are dynamically calculated based on the time steps of the input spikes, enabling IF neurons to

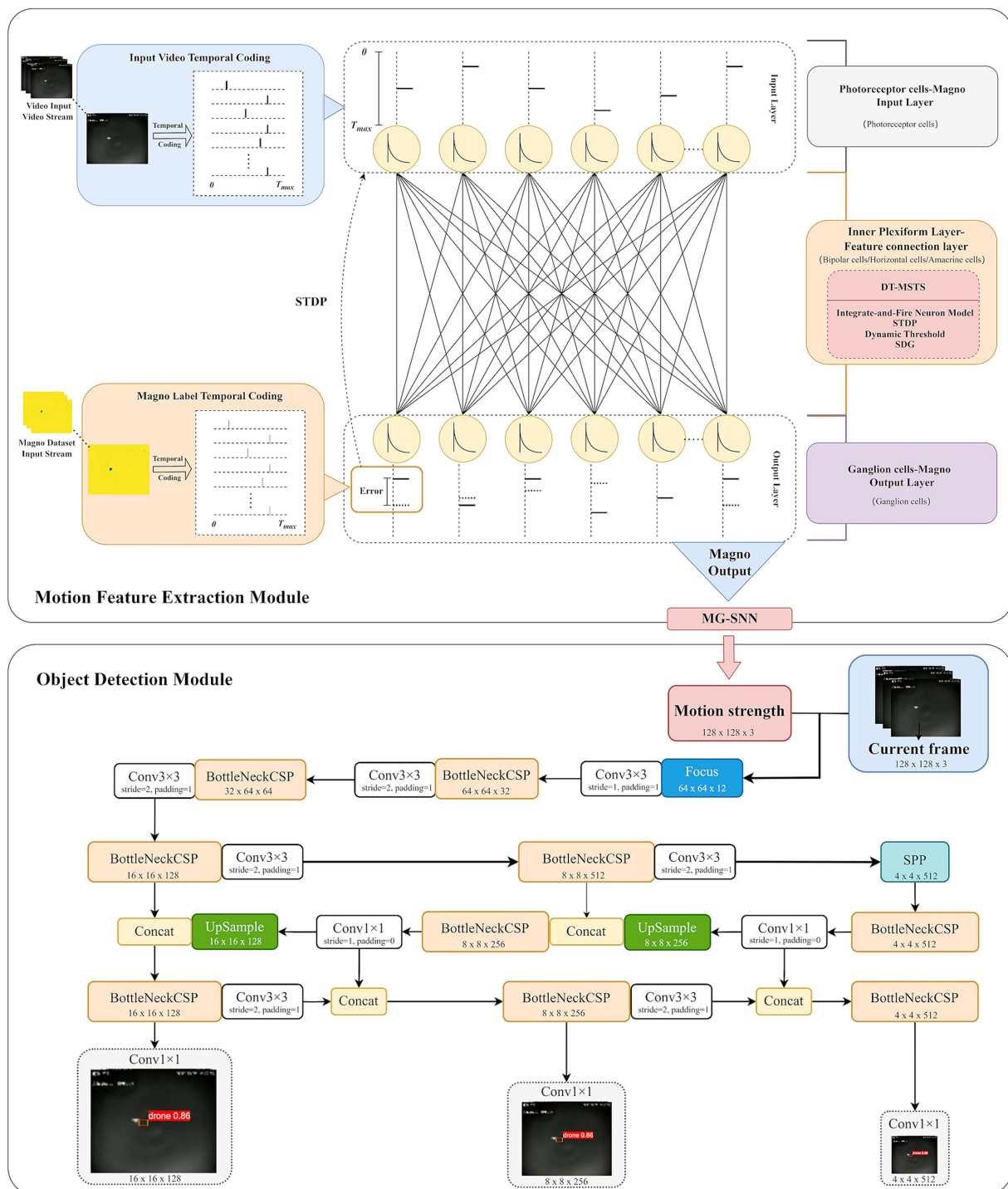


FIGURE 1

This is the overall architecture of the drone motion extraction and object detection system based on a retina-inspired spiking neural network.

determine whether to fire at specific moments. This allows for synchronous processing of each video frame and the generation of corresponding postsynaptic spike sequences. Simultaneously, the label information of the magnocellular pathway dataset is also converted into spike time sequences using the spike temporal encoding strategy, guiding the subsequent backpropagation process. Ensuring that each neuron can fire at least once, synaptic weights are adjusted according to the spike times of the neurons. This process accurately measures the differences between the output spike sequences and the target spike sequences for supervised learning. It not only simulates the learning process of bipolar cells, horizontal cells, and amacrine cells in the biological retina but also preserves temporal precision.

In the ganglion cell output layer, the output corresponds with motion saliency estimation consistent with the magnocellular pathway. Information is transmitted through discrete-time sequences in the network layers, aligning with the dynamic processing characteristics of the biological retina. After processing the forward and backward propagation of the input spike time sequences within a time step, the membrane potentials of all IF neurons are reset to zero, ensuring stability and continuity when the network processes multiple frames continuously.

In the object detection module, the visual motion features extracted by MG-SNN are combined with the YOLO model (Redmon et al., 2016) to perform drone object detection tasks. This combination enhances the detection capability of small targets in dynamic scenes, achieving accurate detection and rapid response. YOLOv5 is primarily utilized in this experiment. All input channels are first sliced and sent to the convolutional layer, to create a visual object detection model based on SNN motion guidance.

In this implementation, the focus is placed on achieving high detection accuracy in challenging scenarios. MG-SNN serves as a motion-guidance module that extracts dynamic features from video frames and generates a motion intensity map, converting spiking activity into a single-channel grayscale image where dynamic regions are assigned higher intensity values (e.g., 1) and static backgrounds are assigned lower intensity values (e.g., 0). Following the YOLOv5 framework, all input channels are sliced and sent to the convolutional layers. The convolutional responses of the motion intensity map and preprocessed video frames are concatenated and passed into the detection pipeline. This design ensures that regions with higher motion intensity responses are more likely to be activated during subsequent processing, thereby enhancing the detection of dynamic objects within the scene. The synchronized processing of MG-SNN outputs with the original video frames ensures that the entire object detection framework operates in real time without introducing latency. This architecture highlights the flexibility and utility of MG-SNN as a plug-and-play module that enhances object detection tasks. It effectively balances computational efficiency with detection accuracy, addressing the challenges of detecting small and dynamic objects in complex environments.

3.2 Drone motion feature extraction based on retinal-inspired spiking neural networks

3.2.1 Magnocellular pathway dataset inspired by the retinal magnocellular pathway computational model

The structure and function of the retina (Yücel et al., 2003) are the cornerstone of the visual system, with its layered structure facilitating

the efficient transmission and processing of visual signals. As illustrated in Figure 2, these layers consist of the Outer Nuclear Layer (ONL), Outer Plexiform Layer (OPL), Inner Nuclear Layer (INL), Inner Plexiform Layer (IPL), and Ganglion Cell Layer (GCL). The ONL houses photoreceptor cell bodies, while the OPL and IPL serve as synaptic connection layers. The INL includes horizontal, bipolar, and amacrine cells (Stacy and Lun Wong, 2003). Horizontal cells regulate the electrical signals of photoreceptor cells through lateral inhibition, while amacrine cells are responsible for signal processing within the retina by forming synapses with ganglion and bipolar cells. Ultimately, the processed visual information is conveyed to the primary visual cortex in the form of action potentials (Arendt, 2003; Benoit et al., 2010).

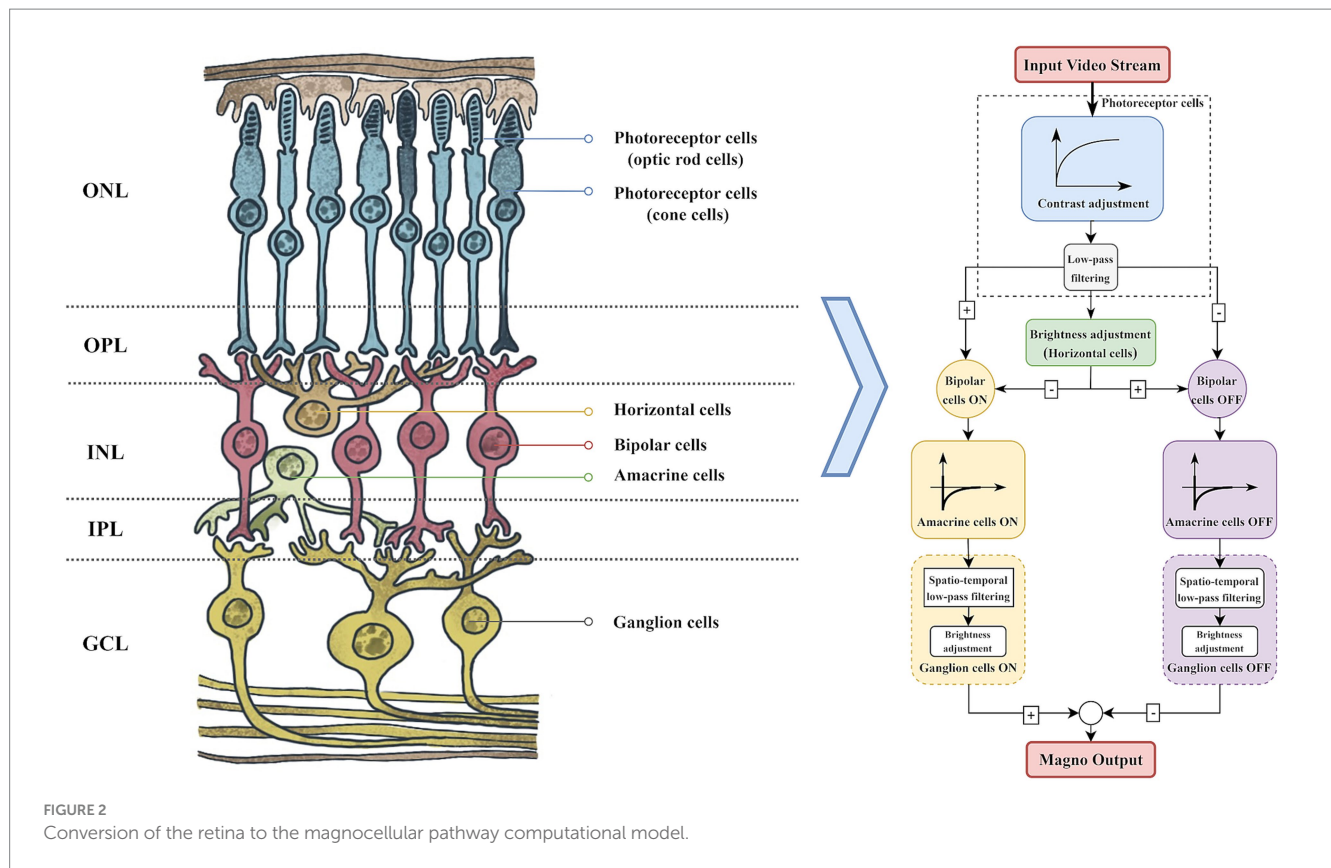
The biological visual system processes visual information through two parallel pathways, one dedicated to motion information and the other to static appearance information (Bock and Goode, 2008). These pathways are commonly referred to as the magnocellular pathway (Magno) and the parvocellular pathway (Parvo). The magnocellular pathway plays a crucial role in the processing of visual motion information.

We referred to the magnocellular pathway computational model proposed by Benoit et al. (2010). According to this model, video streams are processed through photoreceptor cells to acquire visual data and normalize light intensity (Beaudot, 1994), thereby enhancing the contrast in dark areas of the video frames while maintaining the visibility of bright areas. The processed frames undergo low-pass filtering and pass through the ON/OFF channels of horizontal and bipolar cells, forming synaptic triads. In the magnocellular pathway, amacrine cells establish connections with bipolar cells and ganglion cells, thereby providing high-pass filter functionality that enhances sensitivity to temporal and spatial changes in visual information. When processing spatial information, ganglion cells act as spatial low-pass filters and compress the contrast of video frames, thereby enhancing low-frequency spatial motion contour information. This dual functionality allows ganglion cells to play a crucial role in integrating visual information, particularly in visual tracking and target recognition in dynamic environments.

By processing multi-frame historical information, it demonstrates exceptional sensitivity to moving objects, effectively filtering out noise and static backgrounds, as illustrated in Figure 3. This capability is crucial for extracting motion intensity information from visual scenes, facilitating attention guidance and target search. Leveraging this motion processing response, the magnocellular pathway improves focus on potential target areas while adeptly suppressing static backgrounds, which is particularly valuable in visual information processing.

In this research, we utilized the output from the magnocellular pathway as the label information for our neural network model and developed the Visual Magnocellular Dynamics Dataset (VMD) as illustrated in Figure 4. This dataset is constructed based on the Anti-UAV-2021 Challenge dataset¹ and the Anti-UAV-2023 Challenge dataset (Zhao et al., 2023). The videos showcase natural and man-made elements in the backgrounds, such as clouds, buildings, trees, and mountains, realistically simulating scenarios encountered in drone surveillance tasks, the dataset includes target objects of

¹ <https://anti-uav.github.io/dataset/>



varying sizes, from large to extremely small, intensifying the difficulty of object detection. Furthermore, the Anti-UAV-2023 Challenge dataset to enrich the VMD dataset, aimed specifically at small target recognition tasks, which include more challenging video sequences featuring dynamic backgrounds, complex rapid movements, and small targets, thereby encompassing a wider range of small target drone scenarios.

The VMD dataset comprises a total of 650 video samples, divided into 500 training samples and 150 test samples, each showcasing various natural and man-made diverse scenes with target objects of small sizes, scenes such as open skies, urban environments, forested areas, and mountainous regions. Motion complexity is introduced with sequences containing both static and dynamic backgrounds, and targets moving at different speeds and directions, challenging the motion detection capabilities of the model. The VMD dataset is created based on the magnocellular pathway computational model and is developed using the bioinspired library in OpenCV. Several preprocessing steps are applied to ensure the quality and consistency of the dataset: normalization of pixel values, setting the video frame resolution to 120×100 pixels to ensure computational efficiency, and adjusting each video segment to a frame rate of 20 frames per second with durations ranging from approximately 5 to 10 s, and only the content within the salient bounding boxes is retained to ensure precise labeling. The choice of a 120×100 resolution is a practical balance that provides sufficient detail for detecting and identifying small drone targets in complex scenarios. Compared to the simpler tasks often addressed by existing models, such as MNIST handwritten digit classification (32×32 resolution), our approach processes more complex inputs while maintaining an efficient computational profile.

This resolution ensures that the detection framework operates effectively without compromising the precision required for small drone detection.

3.2.2 Video frame processing based on spike temporal encoding

To replicate visual processing akin to that of the human brain and extract motion features, it is crucial to effectively retain and transform the wealth of information present in external stimuli into sequences of neuronal action potentials. The selection of an appropriate encoding strategy is vital for connecting neuronal action potential sequences with behavioral information and for closely integrating the mechanisms of processing in the primary visual cortex with spiking neural networks (Field and Chichilnisky, 2007). Currently, two main types of spike encoding are employed in SNNs (Brette, 2015) rate coding and temporal coding.

In most sensory systems, neurons adjust their firing frequency according to the frequency and intensity of stimuli. However, rate coding (Field and Chichilnisky, 2007; Salinas et al., 2000) does not fully account for the rapid response capability of the visual system. Furthermore, accurately representing complex values with single neuron spikes is challenging and results in a loss of temporal information. Visual information transmission involves multiple synaptic transmissions, with each processing stage being extremely brief. Consequently, the firing frequency of neurons in the primary visual cortex is relatively low in these rapid response processes, a single neuron may only fire 0 or 1 action potential, making it impossible to estimate instantaneous firing rates based on the interval between two action potentials (Thorpe et al., 2001; Salinas et al., 2000),

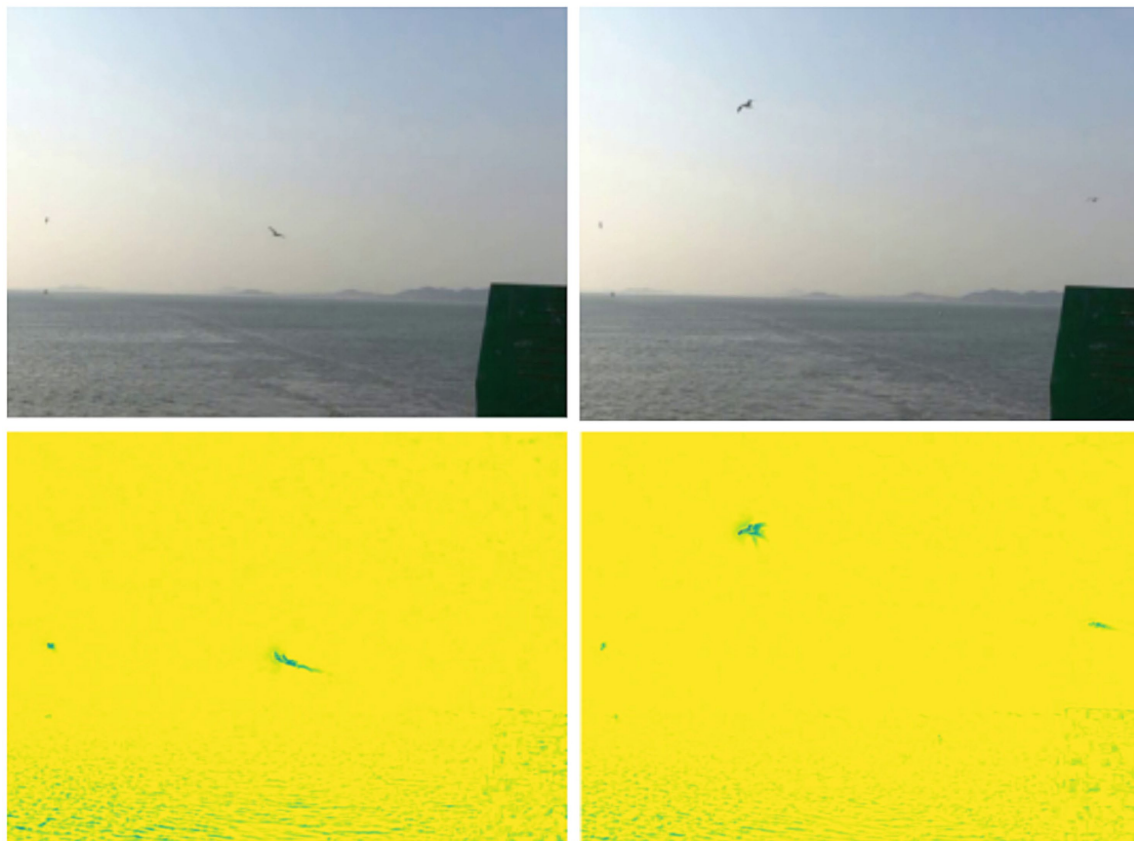


FIGURE 3

Input video images of flying birds with magnocellular pathway outputs, respectively, are the results of the two frames before and after. In order to obtain better viewing results, we performed min-max normalization of the dynamic motion.

the use of changes in firing frequency to encode specific features of complex stimuli is inadequate.

To simulate the flexibility and adaptability demonstrated by the primary visual cortex in processing images or video data, this study adopts a temporal encoding strategy for spike encoding of input information. By representing specific values at precise moments of a single spike, the temporal structure of action potential sequences can encode information related to temporal changes in stimuli (Bair and Koch, 1996), such as the temporal variations in stimulus intensity. This allows for a more accurate representation of input grayscale value information in the temporal dimension.

$$T_i = (1.0 - I_i) \times T_{\max} \quad (1)$$

In subsequent experiments, we determine the activation time T_i of each input neuron based on the normalized intensity value of the i -th pixel as shown in Equation 1. For this purpose, we employ a spike temporal encoding method to process the input video frames. The specific encoding formula is described as follows:

T_{\max} represents the maximum time step of the input spike sequence, while I_i is the normalized intensity value of the i -th pixel. Under this encoding mechanism, each pixel in the input layer generates a single spike at a specific moment T_i , forming a temporal spike sequence. The higher the intensity value of the input, the earlier

the corresponding spike firing time. Figure 5 illustrates the visualization result after encoding a video frame.

Temporal encoding utilizes earlier spike firing times to represent pixels with higher grayscale values, while higher thresholds cause neuronal discharge to delay, indicating that the pixel has a lower grayscale value. Our experiments make use of this time encoding mechanism to accurately map the temporal dimension of visual information, enabling efficient and sensitive processing of visual stimuli within the spiking neural network.

3.2.3 Spiking neurons

In this study, we utilized Integrate-and-Fire (IF) neurons to develop a motion saliency estimation model using a pure SNN architecture. IF neurons operate by accumulating incoming signals until a certain threshold is reached, after which an action potential or “spike” is generated, and then resets its state, mimicking the basic behavior of biological neurons (Smith et al., 2000).

As Figure 6 illustrates, input video frames are encoded into a time spike sequence of length $[0, t_{\max}]$, then the presynaptic spike sequence enters the network. Through IF neurons, when the membrane potential of an IF neuron exceeds the threshold potential V_{th} , it generates a postsynaptic spike sequence. To ensure that non-firing neurons also transmit video frame information, it is defined that non-firing neurons emit a single spike containing minimal

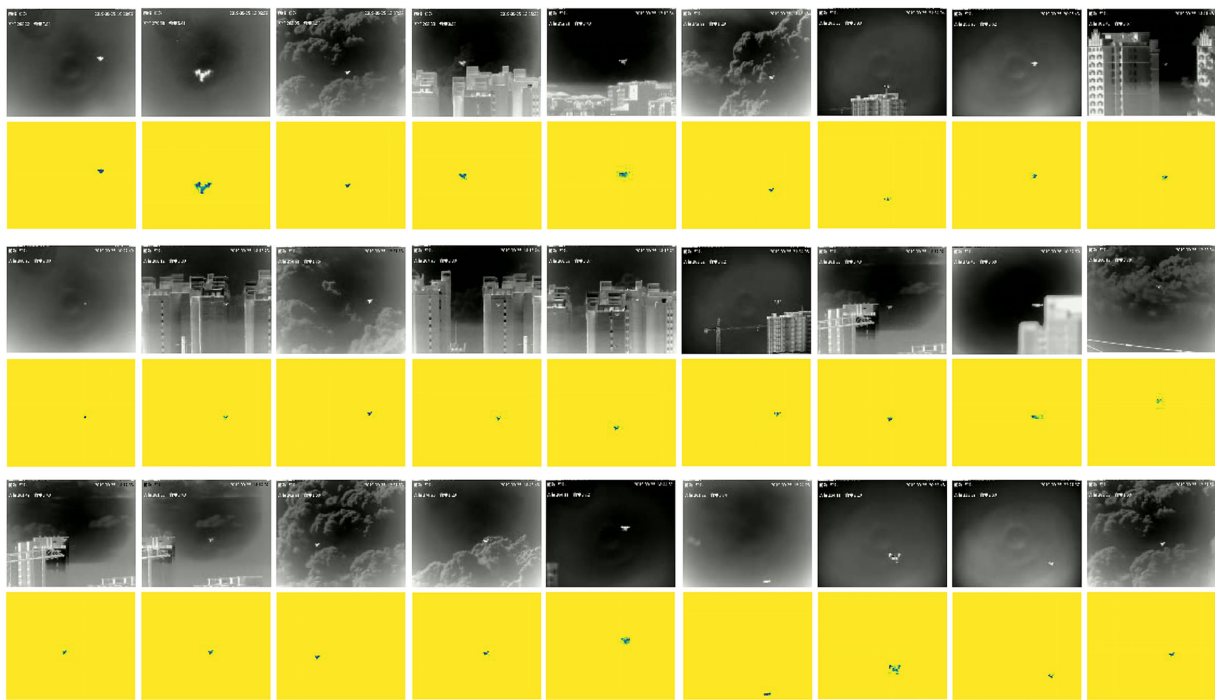


FIGURE 4
The Anti-UAV2021 challenge dataset and VMD dataset contain large and small objects on clear backgrounds, as well as complex backgrounds (clouds and cities).

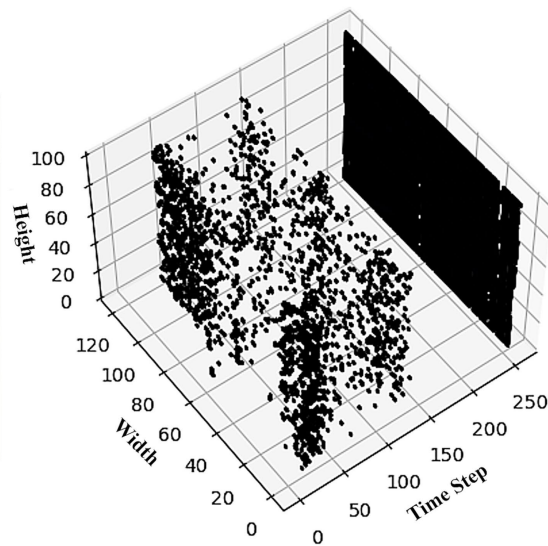
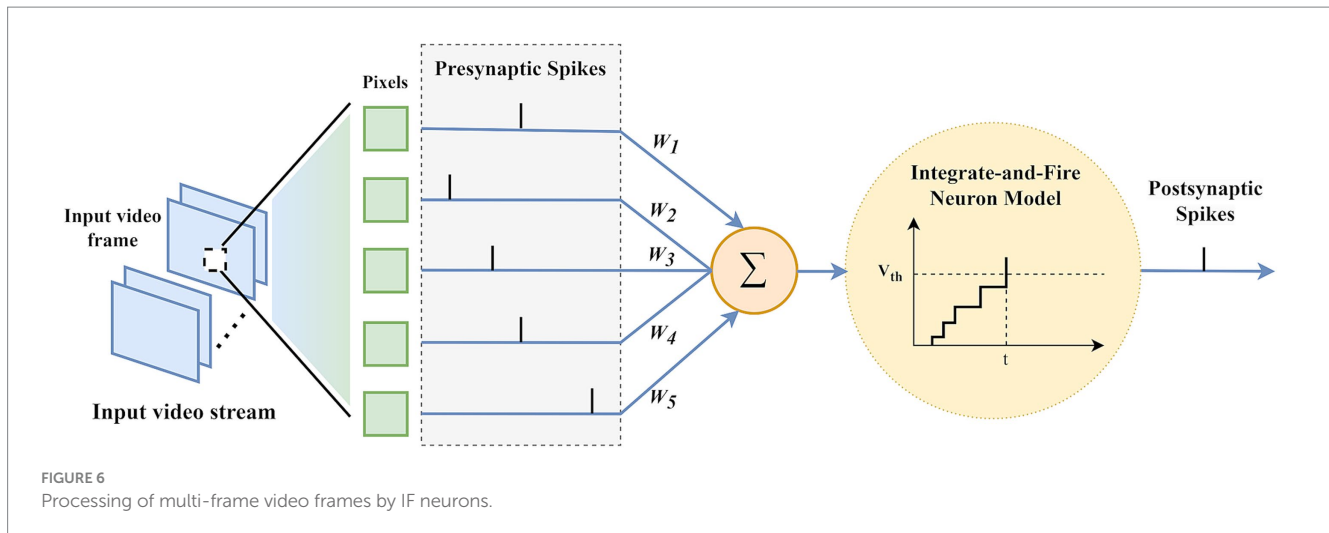


FIGURE 5
The visualization depicts a single drone video frame along with its spike temporal coding sequence spanning 256-time steps.

information at time t_{\max} . The simplified formula for IF neurons is shown in [Equation 2](#):

$$\begin{aligned} (t) &= V(t-T) + \sum_l w^l(t) S^{l-1}(t) \\ \text{if } V(t) \geq V_{th} \text{ then } r(t) &= 1, V(t) = V_{reset} \end{aligned} \tag{2}$$

where $V(t)$ represents the membrane potential of the neuron, w^l is the synaptic weight between layers, and $S^{l-1}(t)$ is the incoming spike sequence from the previous layer. The spike firing rate $r(t)$ is 1 when a spike is emitted and 0 when no spike is emitted. When the membrane potential exceeds the threshold V_{th} , the membrane potential is immediately reset to the resting potential V_{reset} , which is typically set to 0.



During the processing of video frames after spike temporal coding by IF neurons, the threshold of the neuron affects its moment of discharge and maintains the system homeostasis (Abbott, 1999). Based on the leaky adaptive threshold (LAT) mechanism (Falez, 2019), this study introduces a dynamic threshold mechanism that linearly increases with the input time steps. This mechanism is the first attempt to combine dynamic threshold adjustment with a video frame time encoding strategy, aiming to emphasize the importance of spike information in early time steps, which are considered to contain more distinct features compared to later information. This strategy, by enhancing the sensitivity of neurons to high-intensity inputs in early time steps, optimizes the efficiency of information processing.

The dynamic threshold mechanism contains a baseline threshold V_{th} , which is linearly with the increase in input time steps to preserve the unique response characteristics of each IF neuron as in Equation 3. The threshold adjustment D_{th} occurs when the neuron is excited and upon receiving inhibitory spikes, reducing the discrepancy between the actual firing time T_i and the expected firing time T_{label} . The introduction of a dynamic threshold allows the neuron threshold to adjust automatically, encouraging the firing time T_i to approach the target time T_{label} , while maintaining the system equilibrium. The threshold learning rate θ allows for adjustment of the rate of threshold change based on specific circumstances. The specific adjustment rule is as follows:

$$D_{th} = V_{th} + \theta T_i \quad (3)$$

This rule is designed to correct the timing error between the actual firing timestamp T_i and the target timestamp T_{label} each time the neuron fires. The specific value of the threshold learning rate θ depends on the dataset and characteristics of the input video frames and requires an exhaustive search within the range $[0, t_{max}]$ to determine the optimal value. Since the membrane potential is determined by synaptic weights and the input spike sequence, designing an appropriate dynamic threshold rule can effectively enhance the influence of the input spike sequence on the membrane potential, thereby significantly improving the overall performance of the network.

3.2.4 Backpropagation method

The Spike-Timing Dependent Plasticity (STDP) rule adjusts synaptic strengths based on the precise timing of neuronal spikes. This rule leverages the temporal relationships between neuron firing times (Diehl and Cook, 2015), not only effectively encoding temporal information within neural circuits but also facilitating the update of specific synaptic weights. By adjusting synaptic weights based on the relative timing differences between input and output spikes, a biologically plausible learning mechanism is achieved.

$$V_{total}(t) = \sum_{t' \in \{r(t)=1\}} V(t') \quad (4)$$

This study introduces a method that combines the STDP rule (Bi and Poo, 1998; Caporale and Dan, 2008) and a time error function—Dynamic Threshold Multi-frame Spike-Time Sequence Backpropagation Method (DT-MSTS)—to perform backpropagation computations after temporal encoding of video frames. Our approach made improvements based on the BP-STDP method described in the literature (Sjöström and Gerstner, 2010). As shown in Equation 4, an IF neuron with no leak characteristics accumulates membrane potential over time with the output spike sequence and fires when its membrane potential $V(t)$ reaches the neuron threshold V_{th} .

Considering that network decisions rely on the first spike signal from the output layer, earlier spikes thus contain more information. Under the same input and synaptic weights, the membrane potential of an IF neuron approximates the activation value of the Rectified Linear Unit (ReLU) neuron Tavanaei and Maida, 2019 as in Equation 5. We can assume there is an approximate relationship between the output of the ReLU neuron and the firing time t^l of the corresponding IF neuron:

$$y^l \sim t_{max} - t^l \quad (5)$$

In the forward propagation process we constructed, y_j^l represents the activation value of the j -th neuron in layer l , and z_j^l is the weighted input of that neuron. As IF neurons approximate ReLU

neurons, in the ReLU function, $\partial y_j^l / \partial z_j^l$ acts as the derivative at that point as shown in Equation 6:

$$\frac{\partial y_j^l}{\partial z_j^l} = \begin{cases} 1 & \text{if } y_j^l > 0 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

However, in IF neurons, since t_j^l is not a function of w_j^l , we cannot compute $\partial t_j^l / \partial w_j^l$. For each neuron j in layer l , if its threshold time t_j^l is less than t_{\max} , the derivative of its membrane potential V_j^l with respect to input weight w_{ji}^l can be calculated through the spike activity of the preceding layer neuron i . If $t_j^l < t_{\max}$, then assume $\partial t_j^l / \partial V_j^l = -1$, where t_{\max} is the maximum spike firing time.

$$e_{ij} = \mu (T_i^l - T_o^l)^2 / t_{\max}^2 \quad (7)$$

If IF neurons fire within the maximum time window, the time error gradient related to the neuron firing time can be calculated. During the learning process, we initially utilize the Stochastic Gradient Descent (SGD) algorithm in conjunction with the backpropagation algorithm to minimize the mean squared time error function. For each training sample, the mean squared time error function e_{ij} is defined as follows in Equation 7:

where T_i^l represents the target spike firing time, and T_o^l is the spike firing time output for each layer, μ used for error updating. As the equation illustrates, we introduce the STDP factor $\varepsilon_i(t)$ to guide the backpropagation update process of the time error function. This means that if the firing time of the label information in the magnocellular pathway is earlier than the output spike firing time, synaptic connections will be weakened through a negative feedback STDP factor ($\varepsilon_i(t) = -1$); conversely, if the firing time is later, connections will be strengthened through a positive feedback STDP factor ($\varepsilon_i(t) = 1$).

$$\varepsilon_i(t) = \begin{cases} 1 & T_i^l > T_o^l, T_i^l \neq t_{\max}, T_o^l \neq t_{\max} \\ -1 & T_i^l < T_o^l, T_i^l \neq t_{\max}, T_o^l \neq t_{\max} \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

Thus, combining the mean squared time error function e_{ij} , the total loss function L is defined as:

$$L = \frac{1}{2} \sum_{j=1}^O \frac{\mu (T_j^l - T_j^a)^2}{t_{\max}^2} \quad (9)$$

where O is the number of output layer neurons, where μ is time error update parameter. For the output layer ($l = o$), the error term is calculated as:

$$\delta_j^w = -\frac{\mu (T_j^w - T_j^a)}{t_{\max}^2} \quad (10)$$

where δ_j^w directly reflects the prediction error of each output neuron. Finally, the error term obtained can be used to adjust the network weights:

$$w_{ji}^l = w_{ji}^l + \beta \delta_j^w S_i^{l-1} \varepsilon_i(t) \quad (11)$$

where β is the learning rate, which controls the step size of weight updates, and S_i^{l-1} is the output of the previous layer of neurons. This ensures that the network can learn to reduce output errors, thereby improving the accuracy of outputs corresponding to the magnocellular pathway according to the calculations in Equations 8–11. Ultimately, the output layer of ganglion cells will produce a time spike sequence that corresponds to that of the magnocellular pathway in the primary visual.

3.3 Drone object detection based on retinal-inspired spiking neural networks

The YOLOv5 structure comprises several crucial components that ensure its efficiency and accuracy in object detection tasks. Firstly, YOLOv5 employs the Cross-Stage Partial Network (CSPNet) (Wang et al., 2020) as part of its backbone network, enhancing the model learning capability and generalization ability. CSPNet reduces computational cost while preserving spatial feature information by dividing the feature map into two parts: one that passes directly through dense blocks and another that merges with the backbone network. Additionally, YOLOv5 incorporates the Path Aggregation Network (PANet) (Liu et al., 2018) and the Spatial Pyramid Pooling (SPP) (He et al., 2015) module. PANet enhances feature fusion by combining high-level and low-level features, thereby improving object detection performance. The SPP module acts as a spatial pyramid pooling module, integrating information at different scales through pooling operations of various sizes, effectively expanding the receptive field and capturing more contextual information, which enhances the accuracy of drone detection.

The YOLOv5 structure incorporates multiple convolutional layers, pooling layers, and activation function layers, which collectively enable the model to extract crucial features from images and map these features to specific detection results through the final output layer. The Feature Pyramid Network (FPN) (Lin et al., 2017) connects up sampled mappings with corresponding feature mappings in the down sampling branch.

By integrating the dynamic visual features extracted by MG-SNN as a motion-guidance module with the spatial information present in drone video frames into the YOLOv5 model, the primary motion saliency estimation features output by MG-SNN are linked with the convolutional responses of preprocessed video frames. This connection ensures that regions with higher motion intensity response values are more likely to be activated during subsequent processing. Consequently, YOLOv5 is effectively guided to focus on key dynamic areas during detection, which leads to a reduction in false positives and an improvement in recognition accuracy. This innovative approach realizes the integration of SNNs as a visual motion information guidance module with the spatial appearance information

represented by deep neural networks for object detection and recognition.

4 Results

In this section, we conduct experiments on the VMD datasets to validate the performance of our model and evaluate its performance across various scenarios. Additionally, we introduced new comparative methods for experimentation and examined the superiority of our model compared to traditional methods based on the magnocellular pathway.

The experiments were conducted on an Ubuntu operating system. The experimental setup was executed on a PC equipped with an AMD EPYC 7502 32-core processor and an A100-PCI-E-40GB GPU. We set the number of training epochs to 20 and employed a learning rate strategy, while the input size for the network was fixed at 120×100 . The other parameter settings are shown in Table 1.

4.1 Quantitative results of motion feature extraction

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |T_i - \hat{T}_i| \quad (12)$$

To quantify and evaluate the performance of MG-SNN in simulating the output of the magnocellular pathway, we employed a statistical error measurement method: Mean Absolute Error (MAE). As shown in Equation 12, for each neuron i in the video frame, where T_i is the firing time of the spike sequence output by the i -th neuron in the magnocellular pathway, and \hat{T}_i is the firing time of the spike

sequence output by the i -th neuron in the MG-SNN, with n being the total number of input neurons.

When the MAE value is smaller, it indicates better model performance. By synthesizing this metric, we can comprehensively assess the model performance on the magnocellular pathway dataset, ensuring the model not only achieves outputs consistent with the visual pathway but also possesses robustness against outliers.

As shown in Figure 7, the performance of MG-SNN is evaluated by analyzing the training and testing MAE loss over 20 epochs on the VMD dataset. Initially, both training and testing losses exhibit a sharp decline, indicating that the model learns and fits the data quickly during the early epochs, the losses converge rapidly, stabilizing after approximately 10 epochs, which illustrates that MG-SNN avoids overfitting, demonstrates a certain level of generalization, and maintains stable learning of spatiotemporal information present in dynamic data.

Visual representations observed in the early visual areas of the primate brain show similarities to those in CNN frameworks trained on real images (Arulkumaran et al., 2017). This indicates that CNN frameworks also possess a degree of brain inspiration, capable of mimicking the hierarchical structure of simple and complex cells, thereby simulating the function of the retina in object perception to provide stable object representations. To enrich the comparative experiments based on this theoretical foundation, we designed a convolutional neural network model with 3×3 two-dimensional convolution kernels (referred to as RetinaCNN) to simulate the output of the magnocellular pathway. The structure is 1C16-3C32-3C1. RetinaCNN directly processes grayscale intensity information in the video stream, sequentially through convolution and activation functions in each layer, ultimately generating an output consistent with the magnocellular pathway. Additionally, based on the RetinaCNN model, a spike-time encoding-based CNN-SNN motion saliency estimation model, named RetinaSNN, was developed by replacing the original activation functions with IF neurons. The structure of RetinaSNN is 1C16-IFNode-32C3-IFNode-1C3.

We conducted tests on the VMD dataset, where each network input consists of three video frames. This comparative experiment includes the output results of the magnocellular pathway computational model, the CNN model (RetinaCNN), and the CNN-SNN hybrid model (RetinaSNN). Furthermore, ordinary frame difference (OFD) and multi-frame difference (MFD) methods were added to enrich the comparative experiments. To achieve a processing mechanism consistent with MG-SNN, the multi-frame difference method accumulates data from three frames in the channel dimension for learning.

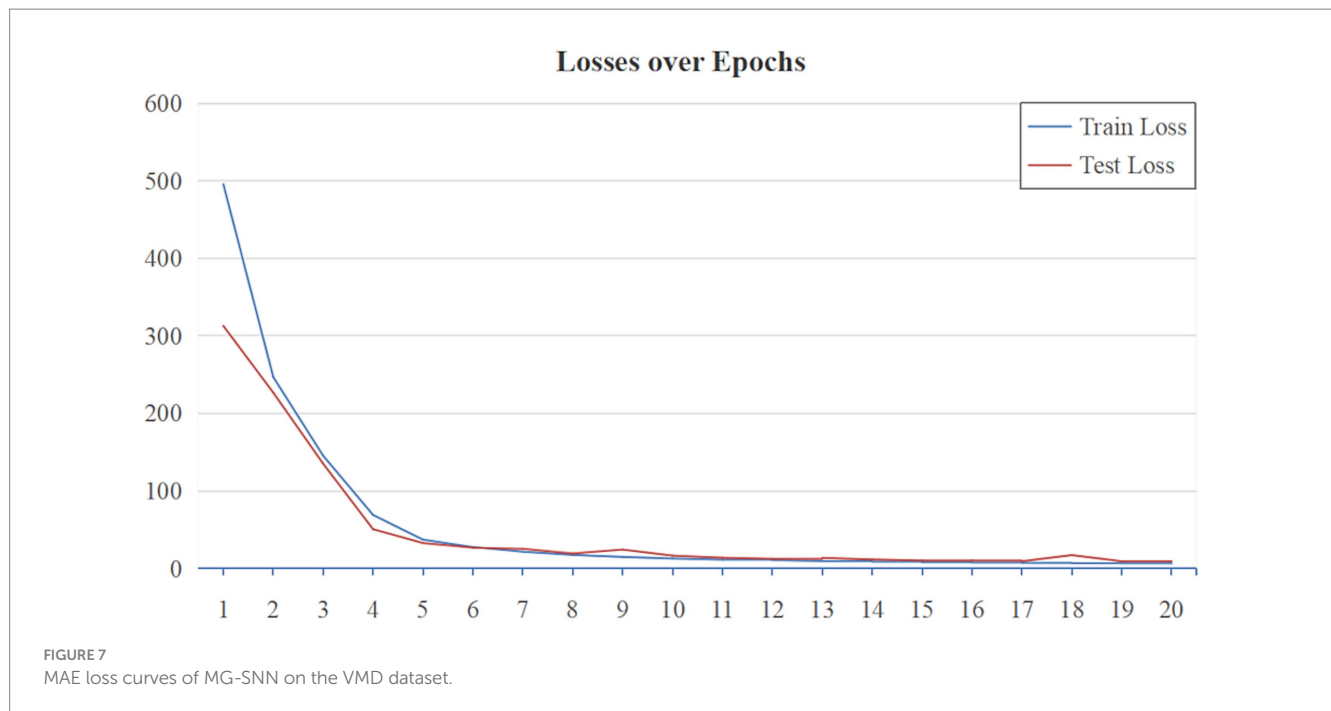
4.2 Qualitative results of motion feature extraction

Since the output of MG-SNN is a temporal spike sequence, for a more stable and accurate analysis of experimental results at the beginning of the test set, it is necessary to transcode the output before performing visual analysis. The visualization results are shown in Figure 8.

The results indicate that most “edge glow” and “video subtitles” phenomena caused by the camera are effectively filtered out regardless of the target size, but MG-SNN does not eliminate all noise

TABLE 1 This is a table of the parameter settings for MG-SNN.

Parameters	Description	Value
t_{\max}	Maximum time step of the input spike sequence	256
V_{th}	Baseline threshold for dynamic threshold	0 mV
V_{reset}	Resting potential	0 mV
\dot{e}	Dynamic threshold learning rate	0.5
i	Time error update parameter	0.02
I	Number of input neurons in the photoreceptor layer	12,000
O	Number of output neurons in the ganglion cell layer	12,000
\hat{a}	Learning rate	10^{-6}
Gray level	Maximum gray value of temporal coding video frame	255
w_{\max}^o	Initialize maximum synaptic weights	1
w_{\min}^o	Initialize minimum synaptic weights	-1



interference, as some irrelevant neurons fire prematurely. This does not affect the identification of the main dynamic targets. In complex background test scenarios, compared to metrics such as OFD and MFD, MG-SNN can better focus on filtering and identifying dynamic information, effectively filtering out most of the interference caused by camera shake and moving cloud backgrounds. Its performance falls short in urban backgrounds, possibly due to inadequate filtering of the complex background and the generation of leading spikes by buildings after spike temporal encoding, making it difficult to identify objects clearly. Nonetheless, the neurons corresponding to tiny targets in complex mixed backgrounds can still produce leading spikes, ensuring effective recognition of moving targets.

Table 2 presents the experimental results, showcasing the performance and effectiveness of different methods on the VMD dataset in handling visual perception tasks. In terms of Mean Absolute Error (MAE) performance, MG-SNN demonstrates an ability to achieve an MAE of 6.4733 within a relatively short training period (20 epochs), showcasing its rapid adaptation to initial training data and its quick attainment of optimal performance in the short term. Notably, MG-SNN outperforms traditional 2D convolutional neural networks (RetinaCNN) and hybrid CNN-SNN architectures (RetinaSNN) in terms of accuracy. This superior performance indicates it effectively captures and processes spatiotemporal information. RetinaCNN struggles to process complex dynamic scenes in comparison due to their inadequate capture of deep spatiotemporal features. Furthermore, the lower MAE observed in the CNN-SNN hybrid architecture compared to traditional CNNs indicates that spike-time encoding-based methods can better extract spatiotemporal information to some extent.

4.3 Quantitative results of object detection

In this section, we leverage the motion features generated by MG-SNN for drone object detection. We use the Average Precision

(AP) value as a quantitative measure, reflecting the model detection accuracy at varying thresholds. As shown in Equations 13, 14, Precision represents the proportion of correctly detected results, while Recall represents the proportion of all objects that are correctly detected.

$$Precision = \frac{TP}{TP + FP} \quad (13)$$

$$Recall = \frac{TP}{TP + FN} \quad (14)$$

TP denotes the number of correctly detected objects, FP denotes the number of non-object targets detected as objects, and FN denotes the number of missed object targets.

In this comparative experiment, MG-SNN utilizes the VMD dataset, consistent with previous experiments, while other models use the Anti-UAV-2021 Challenge dataset and the Anti-UAV-2023 Challenge dataset. All models are provided with identical inputs, which include complex backgrounds such as clouds and buildings, reflecting real-world scenarios in drone surveillance. Due to the limited computational capacity of MG-SNN, the input size is restricted to 100×120. Therefore, the input features were adjusted to 100×120 before feature fusion to obtain the corresponding quantitative results. The Intersection-over-Union (IoU) threshold greater than 0.25 method was employed. Since the input size is small and the images are of low resolution with fewer pixels per target, a higher IoU threshold might cause valid detections to be overlooked. Using an IoU of 0.25, the model achieves a better balance on the 120×100 input images, striking an optimal balance between precision and recall.

4.3.1 Cooperate with different object detection models

We demonstrate the compatibility of MG-SNN with various object detection models by integrating the motion features extracted by MG-SNN with YOLOv6-l (Li et al., 2022), YOLOv5-s, and

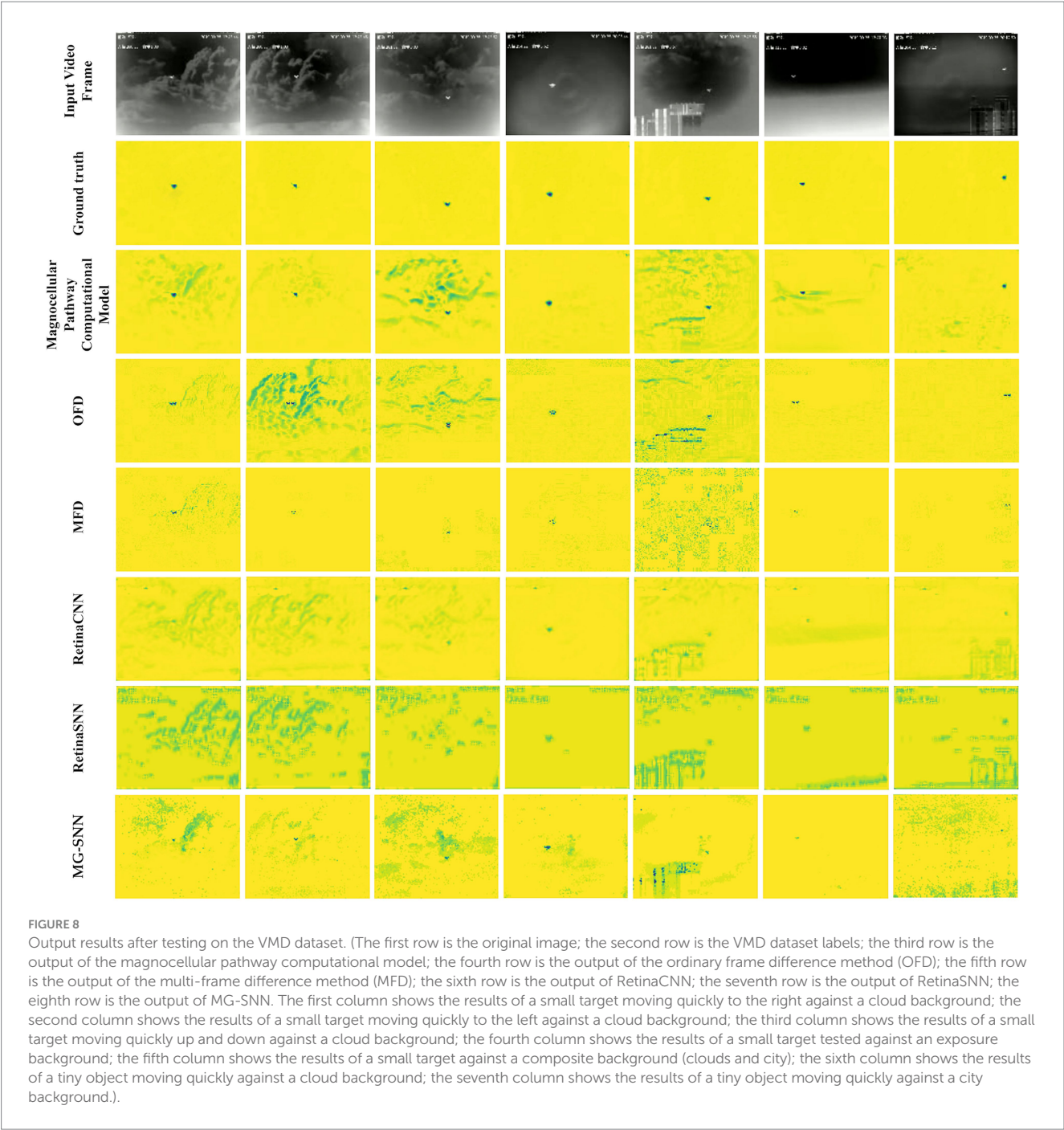


TABLE 2 Comparison of experimental results.

Method	Structure	Network structure	Learning	Minimum MAE during the first 20 Epochs	MAE	Epoch
MG-SNN	SNN	12000FC-IFNode-12000FC-IFNode	DT-MSTS (Dynamic thresholds +STDP)	6.4733	6.4733	20
RetinaCNN	CNN	1C16-3C32-3C1	Backpropagation	35.7711	13.7086	200
RetinaSNN	Spiking CNN	1C16-IFNode-32C3-IFNode-1C3	ANN-SNN Conversion	8.8214	5.1386	200

TABLE 3 Ablation study on the generalization of MG-SNN when applying to popular object detection methods.

Methods	Precision (%)	Recall (%)	AP (%)
YOLOv6-l	90.6	78.0	82.4
MG-SNN+ YOLOv6-l	95.6	80.6	85.0
YOLOv5-s	93.5	79.2	81.9
MG-SNN+ YOLOv5-s	95.5	79.4	84.3
YOLOv5-x	95.4	77.4	82.8
MG-SNN+ YOLOv5-x	98.3	81.1	86.1

Bold values indicate the best performance for each metric within the respective method comparison.

TABLE 4 Computational complexity and performance study of MG-SNN when applied to common object detection methods.

Methods	Prrams (M)	GFLOPs	FPS
YOLOv6-l	59.54	150.5	47.9
MG-SNN+ YOLOv6-l	59.54	152.0	45.2
YOLOv5-s	7.01	15.8	128.2
MG-SNN+ YOLOv5-s	7.01	16.0	126.0
YOLOv5-x	86.17	203.8	64.5
MG-SNN+ YOLOv5-x	86.18	204.5	69.9

YOLOv5-x. We also compare the performance of these combined models with the original YOLOv6 and YOLOv5 structures to highlight the superiority of adding motion information. YOLOv6-l decouples the detection head and redesigns it with an efficient decoupled head, enhancing the model's detection accuracy and convergence speed.

The evaluation results are shown in Table 3. The combination of MG-SNN and YOLO models consistently outperforms standalone YOLO models in terms of detection AP, precision, and recall. Notably, the combination of MG-SNN and YOLOv5-x achieves a precision of 98.3%. Our method improves precision, indicating that it can detect more true objects in complex backgrounds. Further analysis of recall and AP shows that MG-SNN + YOLOv5-x achieves a recall of 81.1% and an AP of 86.1%, both of which are the highest values in Table 3. This demonstrates that the combination not only effectively reduces false positives but also accurately identifies all true targets. The YOLO methods are limited to handling single-frame information, neglecting the processing of motion information in multi-frame inputs. Adding MG-SNN enhances the capability to capture deep spatiotemporal features, resulting in a significant 2.0 to 5.0 AP improvement in the performance of popular object detection algorithms. This improvement indicates that input from MG-SNN effectively compensates for the lack of motion information when dealing with complex dynamic scenes.

Table 4 shows that integrating MG-SNN into existing object detection models introduces minimal computational overhead while maintaining real-time performance. For example, after adding MG-SNN, the GFLOPs of the model slightly increase from 203.8 to 204.5, while the FPS improves from 64.5 to 69.9. This demonstrates that the enhanced

TABLE 5 This is the result of a comparison experiment.

Methods	Precision (%)	Recall (%)	AP (%)	GFLOPs	FPS
YOLOv7-x	95.9	80.5	85.6	188.0	65.4
CFINet	93.4	62.0	71.4	111.57	38.6
DyHead	85.8	72.0	73.4	43.52	15.4
YOLOv5-x	95.4	77.4	82.8	203.8	64.5
MG-SNN+ YOLOv5-x	98.3	81.1	86.1	204.5	69.9

Bold values indicate the best performance for each metric within the respective method comparison.

motion-guidance functionality provided by MG-SNN does not compromise the efficiency of the framework. Specifically, our method is capable of processing up to 69.9 frames per second, making it highly suitable for real-time small drone detection tasks, which effectively balances computational complexity and performance, meeting the demands of dynamic and time-sensitive environments.

4.3.2 Comparison to the advanced competing methods

We compared our method with popular object detection algorithms, including YOLOv7 (Wang et al., 2023), CFINet (Yuan et al., 2023), and DyHead (Dai et al., 2021). Compared to YOLOv5 and YOLOv6, the YOLOv7 model introduces a new set of trainable Bag-of-Freebies strategies to enhance detection performance in small targets and complex scenes by better leveraging cross-layer feature fusion. CFINet is a network architecture that improves small object detection performance through coarse-to-fine region proposal networks (RPN) and imitation learning (Yuan et al., 2023). DyHead employs an attention mechanism to unify different detection heads into a dynamic head framework (Dai et al., 2021).

The evaluation results are shown in Table 5. Although YOLOv7x achieves a high precision of 95.9%, its recall and AP rates of 80.5 and 85.6%, respectively, still fall short of the performance of MG-SNN combined with YOLOv5-x (98.3%). By capturing motion information, MG-SNN can more accurately identify and locate targets in dynamic scenes, effectively enhancing the contrast between targets and backgrounds. This enables the detection algorithm to more precisely separate and identify small targets. The CFINet and DyHead models, which are designed for small object detection, achieve AP values of 96.4 and 91.2%, respectively. However, the recall of CFINet is only 62.0%, lower than the performance of MG-SNN combined with YOLOv5-x (81.1%). Compared with other models, MG-SNN + YOLOv5-x achieves a competitive balance between computational complexity and real-time performance, with GFLOPs increasing slightly to 204.5 while maintaining a high FPS of 69.9, which demonstrates the integration's ability to enhance detection capabilities without significant additional computational cost, making it suitable for dynamic, real-time applications. MG-SNN has proven to outperform other methods in detecting small-target drones within complex backgrounds. This is because it can extract motion information and integrate spatiotemporal features from historical data. Combining motion saliency extraction networks with advanced object detection networks



significantly enhances overall object detection performance. This approach effectively utilizes spatiotemporal features from historical data to improve the detection of small drones in complex backgrounds.

4.4 Qualitative results of object detection

The experimental results are shown in Figure 9. In the figure, green rectangles represent ground truth annotations, and red rectangles represent the detection result bounding boxes. The test data covers urban and cloud backgrounds, where drone targets are difficult to identify. In complex backgrounds, YOLOv5 can detect drone targets in most scenarios, but some bounding boxes do not fully overlap with the actual annotations, resulting in false detections. The CFINet and DyHead models fail to generate accurate detection results, missing the targets and producing erroneous detections. YOLOv7 fails to detect drone targets in several scenarios, indicating a tendency to miss small targets. By utilizing the motion saliency features extracted by the SNN and combining them with the response maps generated by YOLOv5, the target areas are significantly enhanced. Post-processing the spatiotemporal depth information of the video frames improves

target recognition accuracy. This demonstrates that MG-SNN can be combined with other models for tasks such as object detection. By effectively integrating spatiotemporal information, the reliability and accuracy of detection are enhanced, providing stronger technical support for various practical applications.

5 Discussion

The MG-SNN has demonstrated outstanding performance in complex dynamic visual tasks, producing outputs that align with the processing of dynamic data in the primary visual cortex. It shows lower MAE in traditional performance evaluation metrics, validating its accuracy in extracting motion information. Compared to traditional convolutional neural networks and hybrid architectures, MG-SNN has demonstrated stable responses to dynamic targets with reduced iterations. The visual dynamic features extracted by MG-SNN have served as a motion guidance module, enhancing the object detection capabilities of small drones in complex backgrounds and enabling the deployment of drone feature extraction on neuromorphic hardware. Experimental results have indicated that the fused model outperforms the original model in terms of recognition accuracy

and reliability. It can be flexibly integrated into existing object detection frameworks, effectively addressing the adaptability issues of traditional visual perception algorithms when handling fast-moving targets and complex backgrounds.

The two-layer MG-SNN model involves video frames passing through the photoreceptor input layer before the extracted features are transmitted to the ganglion cell output layer. During this process, spikes are fired when a neuron membrane voltage reaches a certain threshold, influenced by the input signals and synaptic weights, using neuron populations for information encoding helps mitigate noise. Even if individual neurons transmit erroneous information, the network as a whole can correct this deviation, reflecting the collective intelligence of biological neural systems.

The YOLO method is limited to processing single-frame information and neglects the handling of motion information in multi-frame inputs. Traditional artificial neural networks process only spatial information, while SNNs propagate spike times from presynaptic to postsynaptic neurons, thereby conveying temporal information. Other potential information in presynaptic neurons, which could provide valuable insights for the network, is discarded. Experimental results demonstrate that MG-SNN + YOLO achieves significant performance improvements over the baseline YOLO model, with an accuracy increase of 2.0–5.0%, a recall improvement of 0.2–3.7%, and an AP enhancement of 2.4–3.3%. Adding MG-SNN enhances the ability to capture deep spatiotemporal features, making the model more robust in distinguishing targets from complex backgrounds, leading to higher precision and recall, and stronger generalization capabilities. MG-SNN effectively compensates for the lack of motion information in handling complex dynamic scenes. Through advanced temporal processing, bio-inspired feature extraction, and spatiotemporal information computation, the combined architecture processes these scenes efficiently and enhances object detection accuracy. By integrating the MG-SNN motion guidance module with the YOLO framework, the system maintains real-time performance while improving detection capabilities, particularly for small targets in dynamic scenarios.

Future work will prioritize optimizing the current implementation of MG-SNN to enable seamless real-time integration for dynamic environments. Deploying MG-SNN on neuromorphic hardware optimized for event-driven and energy-efficient processing, such as Loihi or SpiNNaker, designed for event-driven and energy-efficient processing, will reduce computational overhead and latency, addressing current challenges in resource-constrained systems. Simplifying the MG-SNN architecture through model pruning and approximation will further enhance scalability, making the MG-SNN + YOLO framework more suitable for real-time detection tasks while maintaining accuracy and robustness in complex dynamic scenes, including improved resolution handling. In addition to improving real-time scalability, future research will explore the application of MG-SNN in swarms of small drones, transitioning from single-drone operations to collaborative multi-drone systems. This will involve integrating multi-source information, including pose estimation and data from lidar, RGB-D cameras, and inertial sensors, to enhance motion feature extraction and target detection in dynamic environments. Transitioning the framework

towards online algorithms, incorporating event-based processing and real-time learning techniques will reduce memory consumption, computational overhead, and latency by optimizing the spiking neuron calculations within the current model. These improvements will also enhance system responsiveness and adaptability. With its advanced temporal processing, bio-inspired feature extraction, and combined spatio-temporal information computation, the MG-SNN framework has the potential to provide robust, scalable, and energy-efficient solutions for complex real-world scenarios, especially in resource-constrained systems and multi-drone platforms.

6 Conclusion

Achieving motion feature extraction and object detection for objects in terms of complex dynamic backgrounds and neuromorphic hardware deployment remains a challenging task. This study has delved into the potential of integrating the processing mechanisms of the biological retina with spiking neural networks (SNNs) for the first time. A two-layer pure SNN model, the Magno-Spiking Neural Network (MG-SNN), has been proposed to simulate the visual information transmission process and achieve motion feature outputs consistent with biological visual pathways as a motion feature extraction module for object detection tasks. A Visual-Magnocellular Dynamics Dataset (VMD) has been developed and a multi-frame spike temporal encoding strategy has been adopted to effectively extract and process dynamic visual information. By combining dynamic thresholds and the STDP rule, a Dynamic Threshold Multi-frame Spike Time Sequence (DT-MSTS) backpropagation method has been proposed to facilitate the extraction of motion features within the SNN architecture. Additionally, MG-SNN has been integrated with the YOLO model to design a retinal-inspired spiking neural network architecture for drone motion extraction and object detection. This study has demonstrated the synergistic advantages of retinal mechanisms and SNNs in visual information processing, highlighting the potential for advancing drone visual detection technology, explores the possibility of deploying neuromorphic chips in the form of software, and points towards future directions for managing complex spatiotemporal data in real-world object detection tasks. Future research will focus on expanding the applicability of MG-SNN to broader contexts, including collaborative multi-drone systems and dynamic, resource-constrained environments. Advancements such as the deployment of neuromorphic hardware, the development of efficient real-time algorithms, and the integration of multi-source information will further enhance the system scalability, robustness, and energy efficiency, and are expected to extend the MG-SNN to semantic segmentation or video tracking. These efforts aim to bridge the gap between experimental research and practical deployment, enabling applications in areas such as multi-drone coordination, large-scale surveillance, and disaster response.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession

number(s) can be found below: <https://github.com/cecilia-zz-jy/MG-SNN>.

Author contributions

JZ: Methodology, Software, Writing – original draft. YW: Conceptualization, Supervision, Writing – original draft. XY: Resources, Writing – review & editing. HZ: Data curation, Visualization, Writing – review & editing. MD: Resources, Writing – review & editing. GW: Conceptualization, Supervision, Writing – original draft.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This research is supported by the Beijing Nova Program (2022038), the National Natural Science Foundation of China (62102443), and Hunan Provincial Natural Science Foundation Key Joint Project Between Province and City (2024JJ7428).

References

- Abbott, L. F. (1999). Lapicque's introduction of the integrate-and-fire model neuron (1907). *Brain Res. Bull.* 50, 303–304. doi: 10.1016/S0361-9230(99)00161-6
- Aboudib, A., Gripon, V., and Coppin, G. (2016). A biologically inspired framework for visual information processing and an application on modeling bottom-up visual attention. *Cogn. Comput.* 8, 1007–1026. doi: 10.1007/s12559-016-9430-8
- Abro, G. E. M., Zulkifli, S. A. B. M., Masood, R. J., Asirvadam, V. S., and Laouti, A. (2022). Comprehensive review of UAV detection, security, and communication advancements to prevent threats. *Drones* 6:284. doi: 10.3390/drones6100284
- AL-Dosari, K., Hunaiti, Z., and Balachandran, W. (2023). Systematic review on civilian drones in safety and security applications. *Drones* 7:210. doi: 10.3390/drones7030210
- Arendt, D. (2003). Evolution of eyes and photoreceptor cell types. *Int. J. Dev. Biol.* 47, 563–571. doi: 10.1387/ijdb.14868881
- Arulkumaran, K., Deisenroth, M. P., Brundage, M., and Bharath, A. A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Process. Mag.* 34, 26–38. doi: 10.1109/MSP.2017.273465
- Bair, W., and Koch, C. (1996). Temporal precision of spike trains in extrastriate cortex of the behaving macaque monkey. *Neural Comput.* 8, 1185–1202. doi: 10.1162/neco.1996.8.6.1185
- Beaudot, W. (1994). The neural information processing in the vertebrate retina: A melting pot of ideas for artificial vision. Grenoble: INPG.
- Beaudot, W., Palagi, P., and Héroult, J. (1993). Realistic simulation tool for early visual processing including space, time and colour data. *New Trends Neural Comput.* 686, 370–375. doi: 10.1007/3-540-56798-4_175
- Benoit, A., Caplier, A., Durette, B., and Héroult, J. (2010). Using human visual system modeling for bio-inspired low level image processing. *Comput. Vis. Image Underst.* 114, 758–773. doi: 10.1016/j.cviu.2010.01.011
- Bhaskar, H. (2012). Integrated human target detection, identification and tracking for surveillance applications, in 2012 6th IEEE international conference intelligent systems.
- Bi, G., and Poo, M. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neurosci.* 18, 10464–10472. doi: 10.1523/JNEUROSCI.18-24-10464.1998
- Bi, H., Yang, L., Zhu, H., Lu, D., and Jiang, J. (2021). Steg-net: spatiotemporal edge guidance network for video salient object detection. *IEEE Trans. Cogn. Dev. Syst.* 14, 902–915. doi: 10.1109/TCDS.2021.3086196
- Bock, G. R., and Goode, J. A. (2008). “Higher-Order Processing in the Visual System” in Novartis Foundation Symposia (Chichester, UK: John Wiley & Sons), 256.
- Brette, R. (2015). Philosophy of the spike: rate-based vs. spike-based theories of the brain. *Front. Syst. Neurosci.* 9:151. doi: 10.3389/fnsys.2015.00151
- Calimera, A., Macii, E., and Poncino, M. (2013). The human brain project and neuromorphic computing. *Funct. Neurol.* 28:191. doi: 10.11138/Fneur/2013.28.3.191
- Caporale, N., and Dan, Y. (2008). Spike timing--dependent plasticity: a Hebbian learning rule. *Annu. Rev. Neurosci.* 31, 25–46. doi: 10.1146/annurev.neuro.31.060407.125639
- Cuadrado, J., Rançon, U., Cottureau, B. R., Barranco, F., and Masquelier, T. (2023). Optical flow estimation from event-based cameras and spiking neural networks. *Front. Neurosci.* 17:1160034. doi: 10.3389/fnins.2023.1160034
- Dai, X., Chen, Y., Xiao, B., Chen, D., Liu, M., Yuan, L., et al. (2021). Dynamic head: unifying object detection heads with attentions. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.
- Diehl, P. U., and Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. *Front. Comput. Neurosci.* 9:99. doi: 10.3389/fncom.2015.00099
- Dupeyron, J., Hagenaars, J. J., Paredes-Vallés, F., and de Croon, G. C. H. E. (2021). Neuromorphic control for optic-flow-based landing of MAVs using the Loihi processor, in 2021 IEEE international conference on robotics and automation (ICRA).
- Falez, P. (2019). Improving Spiking Neural Networks Trained with Spike Timing Dependent Plasticity for Image Recognition. [Ph.D. thesis]. [Lille (France)]: Université de Lille doi: 10.13140/RG.2.2.28194.86729
- Field, G. D., and Chichilnisky, E. J. (2007). Information processing in the primate retina: circuitry and coding. *Annu. Rev. Neurosci.* 30, 1–30. doi: 10.1146/annurev.neuro.30.051606.094252
- Gehrig, M., Shrestha, S. B., Mouritzen, D., and Scaramuzza, D. (2020). Event-based angular velocity regression with spiking networks, in 2020 IEEE international conference on robotics and automation (ICRA).
- Glatz, S., Martel, J., Kreiser, R., Qiao, N., and Sandamirskaya, Y. (2019). Adaptive motor control and learning in a spiking neural network realised on a mixed-signal neuromorphic processor, in 2019 international conference on robotics and automation (ICRA).
- Guo, F., Wang, W., Shen, Z., Shen, J., Shao, L., and Tao, D. (2019). Motion-aware rapid video saliency detection. *IEEE Trans. Circuits Syst. Video Technol.* 30, 4887–4898. doi: 10.1109/TCSVT.2019.2929560
- Haessig, G., Berthelon, X., Ieng, S.-H., and Benosman, R. (2019). A spiking neural network model of depth from defocus for event-based neuromorphic vision. *Sci. Rep.* 9:3744. doi: 10.1038/s41598-019-40064-0
- Hagenaars, J., Paredes-Vallés, F., and De Croon, G. (2021). Self-supervised learning of event-based optical flow with spiking neural networks. *Adv. Neural Inf. Process. Syst.* 34, 7167–7179. doi: 10.48550/arXiv.2106.10584
- Hagins, W. A. (1972). The visual process: excitatory mechanisms in the primary receptor cells. *Annu. Rev. Biophys. Bioeng.* 1, 131–158. doi: 10.1146/annurev.bb.01.060172.001023
- He, K., Zhang, X., Ren, S., and Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 37, 1904–1916. doi: 10.1109/TPAMI.2015.2389824

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fncom.2025.1452203/full#supplementary-material>

- Jang, H., Simeone, O., Gardner, B., and Gruning, A. (2019). An introduction to probabilistic spiking neural networks: probabilistic models, learning rules, and applications. *IEEE Signal Process. Mag.* 36, 64–77. doi: 10.1109/MSP.2019.2935234
- Jiao, L., Zhang, R., Liu, F., Yang, S., Hou, B., Li, L., et al. (2021). New generation deep learning for video object detection: a survey. *IEEE Trans. Neural Net. Learn. Syst.* 33, 3195–3215. doi: 10.1109/TNNLS.2021.3053249
- Khalil, H., Rahman, S. U., Ullah, I., Khan, I., Alghadhbani, A. J., Al-Adhaileh, M. H., et al. (2022). A UAV-swarm-communication model using a machine-learning approach for search-and-rescue applications. *Drones* 6:372. doi: 10.3390/drones6120372
- Kim, Y., Chough, J., and Panda, P. (2022). Beyond classification: directly training spiking neural networks for semantic segmentation. *Neuromorph. Comput. Eng.* 2:44015. doi: 10.1088/2634-4386/ac9b86
- Kim, S., Park, S., Na, B., Kim, J., and Yoon, S. (2020). Towards fast and accurate object detection in bio-inspired spiking neural networks through Bayesian optimization. *IEEE Access* 9, 2633–2643. doi: 10.1109/ACCESS.2020.3048444
- Lee, C., Kosta, A. K., Zhu, A. Z., Chaney, K., Daniilidis, K., and Roy, K. (2020). Spike-flownet: event-based optical flow estimation with energy-efficient hybrid neural networks. *Eur. Conf. Comput. Vis.* 12370, 366–382. doi: 10.1007/978-3-030-58595-2_22
- Lehnert, H., Escobar, M.-J., and Araya, M. (2019). Retina-inspired visual module for robot navigation in complex environments, in 2019 international joint conference on neural networks (IJCNN).
- Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., et al. (2022). YOLOv6: a single-stage object detection framework for industrial applications. *Arxiv*. doi: 10.48550/arXiv.2209.02976
- Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). “Feature pyramid networks for object detection” in Proceedings of the 2017 IEEE conference on computer vision and pattern recognition (CVPR) (Honolulu, HI), 2117–2125. doi: 10.1109/CVPR.2017.106
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). SSD: single shot MultiBox detector. *Comput. Vis.* 9905, 21–37. doi: 10.1007/978-3-319-46448-0_2
- Liu, S., Qi, L., Qin, H., Shi, J., and Jia, J. (2018). Path aggregation network for instance segmentation, in Proceeding of the IEEE Conference Computer Vision Pattern Recognition 8759–8768.
- Lukanov, H., König, P., and Pipa, G. (2021). Biologically inspired deep learning model for efficient foveal-peripheral vision. *Front. Comput. Neurosci.* 15:746204. doi: 10.3389/fncom.2021.746204
- Maass, W. (1997). Networks of spiking neurons: the third generation of neural network models. *Neural Netw.* 10, 1659–1671. doi: 10.1016/S0893-6080(97)00011-7
- Maczyta, L., Bouthemy, P., and Le Meur, O. (2019). CNN-based temporal detection of motion saliency in videos. *Pattern Recogn. Lett.* 128, 298–305. doi: 10.1016/j.patrec.2019.09.016
- Malowany, D., and Guterman, H. (2020). Biologically inspired visual system architecture for object recognition in autonomous systems. *Algorithms* 13:167. doi: 10.3390/a13070167
- McIntosh, L., Maheswaranathan, N., Nayeibi, A., Ganguli, S., and Baccus, S. (2016). Deep learning models of the retinal response to natural scenes. *Adv. Neural Inf. Proces. Syst.* 29, 1369–1377. doi: 10.5555/3157096.3157249
- Mehonic, A., Sebastian, A., Rajendran, B., Simeone, O., Vasilaki, E., and Kenyon, A. J. (2020). Memristors—from in-memory computing, deep learning acceleration, and spiking neural networks to the future of neuromorphic and bio-inspired computing. *Adv. Intell. Syst.* 2:2000085. doi: 10.1002/aisy.202000085
- Melanitis, N., and Nikita, K. S. (2019). Biologically-inspired image processing in computational retina models. *Comput. Biol. Med.* 113:103399. doi: 10.1016/j.combiomed.2019.103399
- Minaeian, S., Liu, J., and Son, Y.-J. (2015). Vision-based target detection and localization via a team of cooperative UAV and UGVs. *IEEE Trans Syst Man Cybern Syst* 46, 1005–1016. doi: 10.1109/TSMC.2015.2496099
- Mozafari, M., Kheradpisheh, S. R., Masquelier, T., Nowzari-Dalini, A., and Ganjtabesh, M. (2018). First-spike-based visual categorization using reward-modulated STDP. *IEEE Trans. Neural Netw. Learn. Syst.* 29, 6178–6190. doi: 10.1109/TNNLS.2018.2826721
- Neuroscience, B. (2020). 29th annual computational Neuroscience meeting: CNS* 2020. *BMC Neurosci.* 21:54. doi: 10.1186/s12868-020-00591-1
- Parameshwara, C. M. (2022). Bio-inspired motion perception: From ganglion cells to autonomous vehicles. Maryland: University of Maryland, College Park.
- Parameshwara, C. M., Li, S., Fermüller, C., Sanket, N. J., Evanusa, M. S., and Aloimonos, Y. (2021). Deep spiking neural network for motion segmentation, in 2021 IEEE/RSJ international conference on intelligent robots and systems (IROS), 3414–3420.
- Ponghiran, W., Liyanagedera, C. M., and Roy, K. (2022). Event-based temporally dense optical flow estimation with sequential learning. *Arxiv*. doi: 10.48550/arXiv.2210.01244
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). “You only look once: unified, real-time object detection” in Proceedings of the 2016 IEEE conference on computer vision and pattern recognition (CVPR) (Las Vegas, NV), 779–788. doi: 10.1109/CVPR.2016.91
- Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster R-CNN: towards real-time object detection with region proposal networks. *Adv. Neural Inf. Proces. Syst.* 39, 1137–1149. doi: 10.1109/TPAMI.2016.2577031
- Salinas, E., Hernandez, A., Zainos, A., and Romo, R. (2000). Periodicity and firing rate as candidate neural codes for the frequency of vibrotactile stimuli. *J. Neurosci.* 20, 5503–5515. doi: 10.1523/JNEUROSCI.20-14-05503.2000
- Sanyal, S., Manna, R. K., and Roy, K. (2024). EV-planner: energy-efficient robot navigation via event-based physics-guided neuromorphic planner. *IEEE Robot. Automat. Lett.* 9, 2080–2087. doi: 10.1109/LRA.2024.3350982
- Sejnowski, T. J., and Tesauro, G. (1989). “The Hebb rule for synaptic plasticity: algorithms and implementations” in Neural Models of Plasticity. eds. J. H. Byrne and W. O. Berry (San Diego, FL: Academic Press), 94–103.
- Sjöström, P., and Gerstner, W. (2010). Spike-timing dependent plasticity. *Scholarpedia* 5:1362. doi: 10.4249/scholarpedia.1362
- Smith, G. D., Cox, C. L., Sherman, S. M., and Rinzel, J. (2000). Fourier analysis of sinusoidally driven thalamocortical relay neurons and a minimal integrate-and-fire-or-burst model. *J. Neurophysiol.* 83, 588–610. doi: 10.1152/jn.2000.83.1.588
- Sokhandan, A., and Monadjemi, A. (2024). Visual tracking in video sequences based on biologically inspired mechanisms. *Comput. Vis. Image Underst.* 239:102724. doi: 10.1016/j.cviu.2018.10.002
- Stacy, R. C., and Lun Wong, O. (2003). Developmental relationship between cholinergic amacrine cell processes and ganglion cell dendrites of the mouse retina. *J. Comp. Neurol.* 456, 154–166. doi: 10.1002/cne.10509
- Tavanaei, A., and Maida, A. (2019). BP-STDP: approximating backpropagation using spike timing dependent plasticity. *Neurocomputing* 330, 39–47. doi: 10.1016/j.neucom.2018.11.014
- Thorpe, S., Delorme, A., and Van Rullen, R. (2001). Spike-based strategies for rapid processing. *Neural Netw.* 14, 715–725. doi: 10.1016/S0893-6080(01)00083-1
- Vaila, R. (2021). Deep convolutional spiking neural networks for image classification: Boise State University.
- Wang, C. Y., Bochkovskiy, A., and Liao, H. Y. M. (2023). YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 7464–7475.
- Wang, C. Y., Liao, H. Y. M., Wu, Y. H., Chen, P. Y., Hsieh, J. W., and Yeh, I. H. (2020). CSPNet: a new backbone that can enhance learning capability of CNN, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 390–391.
- Wang, W., Shen, J., Yang, R., and Porikli, F. (2017). Saliency-aware video object segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 40, 20–33. doi: 10.1109/TPAMI.2017.2655501
- Wohrer, A., and Kornprobst, P. (2009). Virtual retina: a biological retina model and simulator, with contrast gain control. *J. Comput. Neurosci.* 26, 219–249. doi: 10.1007/s10827-008-0108-4
- Wu, J., Chua, Y., and Li, H. (2018). A biologically plausible speech recognition framework based on spiking neural networks, in 2018 international joint conference on neural networks (IJCNN), 1–8.
- Wu, J., Yilmaz, E., Zhang, M., Li, H., and Tan, K. C. (2020). Deep spiking neural networks for large vocabulary automatic speech recognition. *Front. Neurosci.* 14:513257. doi: 10.3389/fnins.2020.513257
- Yang, J.-B., Shi, M., and Yi, Q.-M. (2012). A new method for motion target detection by background subtraction and update. *Phys. Procedia* 33, 1768–1775. doi: 10.1016/j.phpro.2012.05.283
- Yuan, X., Cheng, G., Yan, K., Zeng, Q., and Han, J. (2023). Small object detection via coarse-to-fine proposal generation and imitation learning. Proceedings of the IEEE/CVF international conference on computer vision.
- Yücel, Y. H., Zhang, Q., Weinreb, R. N., Kaufman, P. L., and Gupta, N. (2003). Effects of retinal ganglion cell loss on magno-, parvo-, koniocellular pathways in the lateral geniculate nucleus and visual cortex in glaucoma. *Prog. Retin. Eye Res.* 22, 465–481. doi: 10.1016/S1350-9462(03)00026-0
- Zhao, J., Li, J., Jin, L., Chu, J., Zhang, Z., Wang, J., et al. (2023). The 3rd anti-UAV workshop and challenge: methods and results. *Arxiv*. doi: 10.48550/arXiv.2305.07290
- Zheng, Y., Jia, S., Yu, Z., Liu, J. K., and Huang, T. (2021). Unraveling neural coding of dynamic natural visual scenes via convolutional recurrent neural networks. *Patterns* 2:100350. doi: 10.1016/j.patter.2021.100350
- Zhu, R.-J., Zhang, M., Zhao, Q., Deng, H., Duan, Y., and Deng, L.-J. (2024). Tcja-snn: Temporal-channel joint attention for spiking neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* 1–14. doi: 10.1109/TNNLS.2024.3377717

Frontiers in Computational Neuroscience

Fosters interaction between theoretical and experimental neuroscience

Part of the world's most cited neuroscience series, this journal promotes theoretical modeling of brain function, building key communication between theoretical and experimental neuroscience.

Discover the latest Research Topics

[See more →](#)

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact

