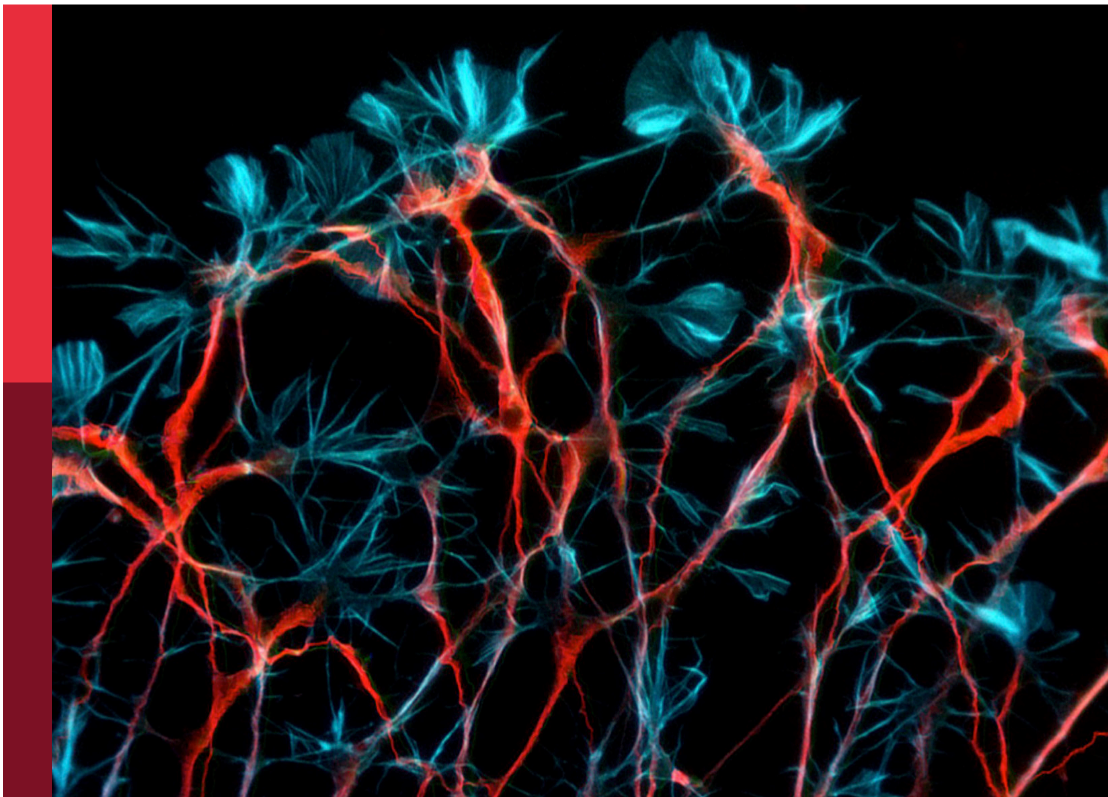
frontiers | Research Topics

Computational intelligence for signal and image processing

Edited by
Baiyuan Ding and Deepika Koundal

Published in
Frontiers in Computational Neuroscience



ISSN 1664-8714
ISBN 978-2-8325-3546-2
DOI 10.3389/978-2-8325-3546-2

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

Topic editors

Baiyuan Ding — National University of Defense Technology, China
Deepika Koundal — University of Petroleum and Energy Studies, India

Citation

Ding, B., Koundal, D., eds. (2023). *Computational intelligence for signal and image processing*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-8325-3546-2

Table of contents

TYPE Editorial
PUBLISHED 27 September 2023
DOI 10.3389/fncom.2023.1284600

OPEN ACCESS

EDITED AND REVIEWED BY
Si Wu,
Peking University, China

*CORRESPONDENCE
Deepika Koundal
✉ dkoundal@ddn.upes.ac.in

RECEIVED 28 August 2023
ACCEPTED 07 September 2023
PUBLISHED 27 September 2023

CITATION
Koundal D and Ding B (2023) Editorial: Computational intelligence for signal and image processing.
Front. Comput. Neurosci. 17:1284600.
doi: 10.3389/fncom.2023.1284600

Editorial: Computational intelligence for signal and image processing

Deepika Koundal[1]* and Baiyuan Ding[2]

[1]AI Cluster, School of Computer Science, University of Petroleum and Energy Studies, Dehradun, India, [2]Science and Technology on Automatic Target Recognition Laboratory, National University of Defence Technology, Chansha, China

KEYWORDS

image processing, signal processing, machine learning, deep learning, computational techniques

Editorial on the Research Topic
Computational intelligence for signal and image processing

1. Introduction

The contemporary world features an array of sensors, each with distinct functions. Data from these sensors primarily come in the form of signals, images, videos, and similar formats (Cheng D. et al., 2022). Effectively deciphering this data holds the key to enhancing daily life and industrial efficiency (Wang et al., 2023). Initially, humans were responsible for processing and interpreting signal and image data, a process with limited accuracy and efficiency (Liu F. et al., 2023). However, the evolution of computational intelligence, including machine learning and deep learning, has enabled the automated handling of sensor measurements, reducing the need for human involvement (Jiang et al., 2023). Consequently, vast amounts of signal and image data can be efficiently processed for diverse applications (Cheng L. et al., 2022; Wang et al., 2022; Fu et al., 2023), given their varied and abundant nature, which encompasses radar signals, biomedical signals, optical images, and distinctive medical images (Zhuang et al., 2022a). To this end, distinct computational intelligence algorithms are necessary for various signal and image types (Zhuang et al., 2022b; Dang et al., 2023; Lu et al., 2023). Recent strides in machine learning and deep learning have introduced a suite of tools for signal and image processing like convolutional neural networks, deep belief networks, and deep generative models (Liu et al., 2021). Integrating these pioneering computational intelligence techniques into the realm of signal and image processing holds the promise of delivering accurate and rapid interpretations (Cong et al., 2023; Liu H. et al., 2023).

2. Contributions

Within this research domain, a total of 10 articles have been published. Pan et al. introduced a stepped image semantic segmentation network structure that incorporated a multi-scale feature fusion scheme and boundary optimization. It enhanced the model accuracy by optimizing the spatial pooling pyramid module in the Deeplab V3+ network by employing the Funnel ReLU activation function for accuracy improvement. Experimental results have shown that the enhanced networks achieved a 96.35% accuracy. Furthermore, Zhijian et al. explored a method for simulating the infrared data, fusing simulated 3D infrared targets with real infrared images. Real infrared images were fused into panoramic backgrounds, simulating infrared characteristics on aircraft components like the tail nozzle, skin, and tail flame. This approach, driven by Unity3D, allowed flexible aircraft trajectory and attitude editing, generating diverse multi-target infrared data. The experimental results have shown that the simulated image closely resembled the real infrared images and aligned with real data's target detection algorithm performance. Another study by Prabhakar et al. focused on EEG signal modeling and classification. With a sparse representation model and sparseness measurement analysis for EEG signals, Swarm Intelligence (SI) techniques were harnessed for Hidden Markov Model (HMM)-based classification. Additionally, a Convolutional Neural Network (CNN)-powered deep learning methodology achieved a remarkable 98.94% classification accuracy.

Additionally, Fan et al. have given insights to elucidate the association between Tic disorder and gut microbiota. A total of 78 stool samples were examined from Tic disorder cases and 62 from healthy controls, utilizing a case-control design for all studies. The results have shown variations in gut microbiota taxonomy between Tic disorder cases and controls, albeit with inconsistencies across studies. In another study, Saikumar et al. integrated the Internet of Things sensor data into a deep learning-based application for diagnosing heart conditions. The Internet of Things sensor data related to heart disease was utilized to train the deep graph convolutional network (DG_ConvoNet). The K-means technique was employed to reduce sensor data noise, aiding the clustering of unstructured data. Extracted features were then used in Linear Quadratic Discriminant Analysis. DG_ConvoNet, a deep learning approach, exhibited 96% accuracy, 80% sensitivity, 73% specificity, 90% precision, 79% F-Score, and a 75% area under the ROC curve, proficiently classifying and predicting heart ailments. Furthermore, Yan et al. have discussed urban street color analysis schemes by merging the color cards with efficient software recognition by addressing the challenges in quantifying urban color research. Using the China Building Color Card and Python's HSV color segmentation, Avenida de Almeida Ribeiro's colors from various angles have been assessed. This approach combined color card colorimetry and computer recognition by capturing both building and environmental influences. The method comprehensively quantified, compiled, summarized, and compared the architectural and environmental colors, offering practical universality. The findings aided Macao's color planning and urban renewal, presenting a novel urban color study approach. Gezawa et al. introduced a fused feature network that handled the shape classification and segmentation tasks by a dual-branch approach and feature learning. A feature encoding network was devised for network simplification by integrating two distinct building blocks with interposed batch normalization and rectified linear unit layers. It accelerated learning, mitigating gradient vanishing due to the limited number of layers for propagation. The framework also introduced a grid feature extraction module using convolution blocks and max-pooling to hierarchically represent input grid features. The max-pooling reduced the overfitting risk by gradually diminishing spatial dimensions, network parameters, and processing load. The grid size limitations were handled by locally sampling a constant point number from each grid region via a basic K-nearest neighbor by enhancing approximation functions for detailed feature characterization. It has shown superior performance with state-of-the-art techniques.

In another study, Ming et al. introduced deep CNN using CT scans for the diagnosis of severe pneumonia with pulmonary infection. An EC-U-net model has been employed on 120 patients to find accuracy in comparison to the traditional CNN. The learning rate of the model has decreased in over 40 training cycles by yielding results nearer to mask images. The given EC-U-net has outperformed the CNN with a higher Dice coefficient and lower loss. The method has increased diagnostic accuracy by reducing false rates and improving the recognition of infection-related features in CT scans by showing potential for clinical applications. Zhang et al. discussed a neural learning approach for the prediction of the best grasp configuration for each detected object from the image. A 3D-plane-based approach was used to filter the cluttered background and then the objects and grasp candidates by two separate branches were detected by an additional alignment module. A series of experiments are conducted on two public datasets to evaluate the performance of the proposed model in predicting reasonable grasp configurations "from a cluttered scene." A deep learning-based method was proposed by Liu et al. to classify the data, screen out double-peak data, and realize the segmentation of the integral regions through the given U-Net segmentation model. The presented classification model exhibited an accuracy of 99.59%, while the segmentation model achieved an intersection over a union value of 0.9680 by using the combined loss function.

3. Conclusion

This editorial presented 10 research articles focused on the applications of Computational Intelligence for Signal and Image Processing. The aim was to gather related articles in the Signal and Image Processing industry, such as education, healthcare, and security. The findings presented in this Research Topic showcased more active development and research within the field of Computational Intelligence methods in the times ahead. To facilitate this progression, future approaches might encompass harnessing Computational Intelligence techniques to improve prediction precision and enhance the reliability of prediction models.

Author contributions

DK: Data curation, Formal analysis, Investigation, Methodology, Project administration, Resources, Validation, Visualization, Writing—original draft, Writing—review and editing. BD: Conceptualization, Supervision, Visualization, Writing—review and editing.

Acknowledgments

We would like to convey our thanks to the authors for their invaluable contributions to this Research Topic. Our deep appreciation goes to the diligent reviewers for their meticulous and punctual assessments, significantly elevating the caliber of this publication. Lastly, we wish to recognize the steadfast support provided by the editorial team of the Frontiers in Computational Neuroscience journal, whose efforts have been instrumental in bringing this Research Topic to fruition.

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

Cheng, D., Chen, L., Lv, C., Guo, L., and Kou, Q. (2022). Light-guided and cross-fusion U-net for anti-illumination image super-resolution. *IEEE Trans. Circ. Syst. Video Technol.* 32, 8436–8449. doi: 10.1109/TCSVT.2022.3194169

Cheng, L., Yin, F., Theodoridis, S., Chatzis, S., and Chang, T. (2022). Rethinking Bayesian learning for data analysis: the art of prior and inference in sparsity-aware modeling. *IEEE Sig. Proc. Mag.* 39, 18–52. doi: 10.1109/MSP.2022.3198201

Cong, R., Sheng, H., Yang, D., Cui, Z., and Chen, R. (2023). Exploiting spatial and angular correlations with deep efficient transformers for light field image super-resolution. *IEEE Trans. Multimed.* doi: 10.1109/TMM.2023.3282465. [Epub ahead of print].

Dang, W., Xiang, L., Liu, S., Yang, B., Liu, M., Yin, Z., et al. (2023). (2023). A feature matching method based on the convolutional neural network. *J. Imag. Sci. Technol.* 67, 3. doi: 10.2352/J.ImagingSci.Technol.2023.67.3.030402

Fu, C., Yuan, H., Xu, H., Zhang, H., and Shen, L. (2023). TMSO-Net: Texture adaptive multi-scale observation for light field image depth estimation. *J. Vis. Commun. Image Rep.* 90, 103731. doi: 10.1016/j.jvcir.2022.103731

Jiang, H., Chen, S., Xiao, Z., Hu, J., Liu, J., Dustdar, S., et al. (2023). Pa-count: passenger counting in vehicles using wi-fi signals. *IEEE Trans. Mob. Comput.* doi: 10.1109/TMC.2023.3263229. [Epub ahead of print].

Liu, F., Zhao, X., Zhu, Z., Zhai, Z., and Liu, Y. (2023). Dual-microphone active noise cancellation paved with Doppler assimilation for TADS. *Mech. Syst. Sig. Proc.* 184, 109727. doi: 10.1016/j.ymssp.2022.109727

Liu, H., Xu, Y., and Chen, F. (2023). Sketch2Photo: Synthesizing photo-realistic images from sketches via global contexts. *Eng. Appl. Artif. Int.* 117, 105608. doi: 10.1016/j.engappai.2022.105608

Liu, R., Wang, X., Lu, H., Wu, Z., Fan, Q., Li, S., et al. (2021). SCCGAN: style and characters inpainting based on CGAN. *Mobile Netw. Appl.* 26, 3–12. doi: 10.1007/s11036-020-01717-x

Lu, S., Liu, S., Hou, P., Yang, B., Liu, M., Yin, L., et al. (2023). Soft tissue feature tracking based on deep matching network. *Comput. Model. Eng. Sci.* 136, 363–379. doi: 10.32604/cmes.2023.025217

Wang, S., Hu, X., Sun, J., and Liu, J. (2023). Hyperspectral anomaly detection using ensemble and robust collaborative representation. *Inf. Sci.* 624, 748–760. doi: 10.1016/j.ins.2022.12.096

Wang, W., Chen, Z., and Yuan, X. (2022). Simple low-light image enhancement based on Weber–Fechner law in logarithmic space. *Signal Proc. Image Commun.* 106, 742. doi: 10.1016/j.image.2022.116742

Zhuang, Y., Chen, S., Jiang, N., and Hu, H. (2022a). An effective WSSENet-based similarity retrieval method of large lung ct image databases. *KSII Trans. Int. Inf. Syst.* 16, 13. doi: 10.3837/tiis.2022.07.013

Zhuang, Y., Jiang, N., Xu, Y., Xiangjie, K., and Kong, X. (2022b). Progressive distributed and parallel similarity retrieval of large ct image sequences in mobile telemedicine networks. *Wireless Commun. Mobile Comput.* 2022, 1–13. doi: 10.1155/2022/6458350

frontiers | Frontiers in Computational Neuroscience

ORIGINAL RESEARCH
published: 02 June 2022
doi: 10.3389/fncom.2022.895680

A Study of English Learning Vocabulary Detection Based on Image Semantic Segmentation Fusion Network

*Leying Pan**

School of International Studies, Zhejiang Business College, Hangzhou, China

College students learn words always under both teachers' and school administrators' control. Based on multi-modal discourse analysis theory, the analysis of English words under the synergy of different modalities, students improve the motivation and effectiveness of word learning, but there are still some problems, such as the lack of visual modal memory of pictures, incomplete word meanings, little interaction between users, and lack of resource expansion function. To this end, this paper proposes a stepped image semantic segmentation network structure based on multi-scale feature fusion and boundary optimization. The network aims at improving the accuracy of the network model, optimizing the spatial pooling pyramid module in Deeplab V3+ network, using a new activation function Funnel ReLU (FReLU) for vision tasks to replace the original non-linear activation function to obtain accuracy compensation, improving the overall image segmentation accuracy through accurate prediction of the boundaries of each class, reducing the intra-class error in the prediction results. The accuracy compensation is obtained by replacing the original linear activation function with FReLU. Experimental results on the Englishhnd dataset demonstrate that the improved network can achieve 96.35% accuracy for English characters with the same network parameters, training data and test data.

Keywords: multi-modal discourse analysis, learning, image semantic, feature fusion, Deeplab V3+ network

Edited by:
Baiyuan Ding,
National University of Defense Technology, China

Reviewed by:
Nian Yang,
Anhui University of Science and Technology, China
Zhijian Diao,
Nanchang JiaoTong Institute, China

***Correspondence:**
Leying Pan
lindaply123@sina.com

Received: *14 March 2022*
Accepted: *21 April 2022*
Published: *02 June 2022*

Citation:
Pan L (2022) A Study of English Learning Vocabulary Detection Based on Image Semantic Segmentation Fusion Network.
Front. Comput. Neurosci. 16:895680.
doi: 10.3389/fncom.2022.895680

INTRODUCTION

English Word Memory provides solutions for teaching English vocabulary in college. Some competitions have been held in Jiangsu alone with over 90 undergraduate and higher education institutions and over 200,000 students participating. Unlike the traditional way of learning by reading word books and memorizing words, the corpus-based English word platform brings together a variety of learning contents such as pronunciation, spelling and example sentences, and collaboratively uses media forms such as sound, image, text and color to generate dynamic vocabulary exercises, allowing students to learn word collocations and usage in the exercises, improving the efficiency of learning. In addition, the platform provides teachers with management and supervisory functions for vocabulary teaching (Liu et al., 2007; Liu, 2021).

Both the design and the use of English words contain a five-level system of multimodal discourse analysis theoretical framework. At the cultural and contextual levels, English Words users are divided into a teacher side and a student side (Dai et al., 2018; Yin, 2021). Because both teachers and students are in the same cultural context, the ideology and the structure of the subject matter are potentially the same, and the scope of the discourse is the same (Huang et al., 2019). The word content of English Words is designed from the textbook in which it is taught, and reflects the conceptual meaning and schematic meaning of words through word interpretation, usage, and example sentences in both English and Chinese. Through "check-in" and "ranking", students and teachers can interact with each other and realize interpersonal meaning (Wu and Chen, 2020).

At the formal level, the different formal systems for achieving meaning include the "lexico-grammatical system of language" (Sung et al., 2016). The lexicon refers to the items that are already given meaning in their own right, while the grammar is a more complex system of structural rules for combining these items. Since one modality cannot fully express the meaning of communication. Other modal forms need to be used to enhance and complement the meaning. Chen Hsieh et al. (2017) classifies the relationship of multimodal discourse forms into two types-"complementary" and "non-complementary". English words are dynamically generated based on a large-scale corpus of vocabulary exercises for students, and the non-reinforcing relationship between visual and auditory modalities is used to complement each other through word pronunciation identification, interpretation, and detailed example sentences, so that students can repeatedly compare and contrast in different contexts to promote learning through practice (Huang et al., 2011; Duman et al., 2015).

Image semantic segmentation, as a cornerstone technique in computer vision tasks, is different from target detection and image classification in that each pixel in an image is assigned a predefined label indicating its semantic class to achieve the task of pixel-level classification (Saalbach et al., 2009; Chen et al., 2021). Specifically, image semantic segmentation is the process of distinguishing at the pixel level exactly what and where the target object is in an image, i.e., first detecting the target in the image, then depicting the outline between each individual and the scene, and finally classifying them by assigning a color to things that belong to the same class (Lyu et al., 2018; Zhang et al., 2020).

In recent years, with the development of deep learning technology in computer vision, image semantic segmentation has been widely used in autonomous driving, intelligent medical treatment, etc. (Sanonguthai, 2011). The intrinsic invariance of DCNN (Di Wu et al., 2021) can learn dense abstract features, which is much better than the performance of traditional systems designed based on sample features. However, existing semantic segmentation algorithms still suffer from intra-class semantic misidentification, small-scale object loss, and blurred segmentation boundaries. Therefore, capturing more feature information and optimizing for the target boundary are important research elements to improve the segmentation accuracy.

In 2016, Xue et al. (2018) proposed Deeplab V2 model based on Deeplab V1 network (Zhang et al., 2018), using inflated convolution instead of partial pooling operation for down sampling filter for feature extraction, and using spatial deterministic pyramid pooling (ASPP) module (Xie et al., 2018) for multi-scale feature extraction, In 2017, Deeplab V3 (Laufer, 2006) improved the ASPP module on the basis of V2 network to form an end-to-end network structure, and eliminated the CRF boundary optimization module. In the field of semantic segmentation, the network structure usually adopts the codec-decoder structure; except for Deeplab V3+, almost all of the above mentioned algorithms do not consider using the effective decoder module, or only use the codec-symmetric structure with a single structure, which fails to effectively fuse the high-level semantic information and the low-level spatial information across layers in the up sampling process, and loses the important pixel information of the feature map.

Related Work

Through literature combing, we found that the current research on adaptation of English learning supported by artificial intelligence is mainly about the design and development of adaptive, wisdom-adapted related learning systems for students, and there is no literature on learning adaptation from the students' perspective. Since, learning adaptability is related to learning performance, learning quality, etc. Therefore, the literature is extended to study "learning performance," "learning effectiveness," "learning quality," "teaching effectiveness," and "teaching quality" related to artificial intelligence-supported learning. "Teaching quality," etc. (Duman et al., 2015; Xue et al., 2018; Zhou et al., 2020). A review of the literature shows that the research focuses on two aspects of speaking and composition, and specific teaching practices of AI English learning tools are mainly educational APPs and intelligent online systems. In speaking training and assessment, Gorman's "English Fun Dubbing" has stimulated students' interest and confidence in speaking learning, and thus improved students' English learning petrifaction (Hessamy and Ghaderi, 2014). Liu et al. (2021) study came to a similar conclusion that although a small number of students were not very active, most of them were able to accept the learning mode of using English Fun Voiceover to learn speaking, and there was a significant difference between English majors and non-English majors in their willingness to use English APPs for listening and speaking. In terms of smart writing, the smart writing system criterion significantly improved the quality of students' writing in Attali, which found that the number of student essay revisions was positively correlated with improved scores, but 70% of the students in that study lacked confidence or interest in the system (Cameron, 2002), and if teachers do not approve of the smart writing system, then students also If teachers do not approve of the intelligent writing system, then students will also lack motivation to use it consistently. Gu and Zhang (2020) pointed out that the intelligent composition review system can help students develop the habit of repeated revision, but it should not be It is still necessary to have teacher guidance. Zhang and Liu (2021) study found that course assessment mechanisms, students' vocabulary levels, their perceptions of the feedback from the intelligent writing system, and the quality of the intelligent system itself affect students' use, and that students' motivation to learn English affects their learning outcomes in the automatic evaluation feedback system. Zhao et al. (2017) concluded that at the individual level, students' familiarity with computers, online learning experience, existing language ability and writing level, and learning autonomy bring about different usage effects in the collaborative artificial intelligence system, but there are no related research topics.

PROPOSED ALGORITHM

Deeplab V3+ Network Architecture

The Deeplab V3+ network architecture is the latest generation of semantic segmentation network framework in the Deeplab series proposed by Google Labs, with superior performance on multiple datasets. Or Xception as the backbone network, using a data normalization (BN) layer to prevent training overfitting (Liu et al., 2020), and adding a decoder network component to build an end-to-end coder-decoder network model.

The structure of DeeplabV3+ network is shown in **Figure 1**. The input image is passed through a neural network with an inflated convolution to reduce the number of down sampling while ensuring a large perceptual field, and the high-level semantic information and low-level spatial information are

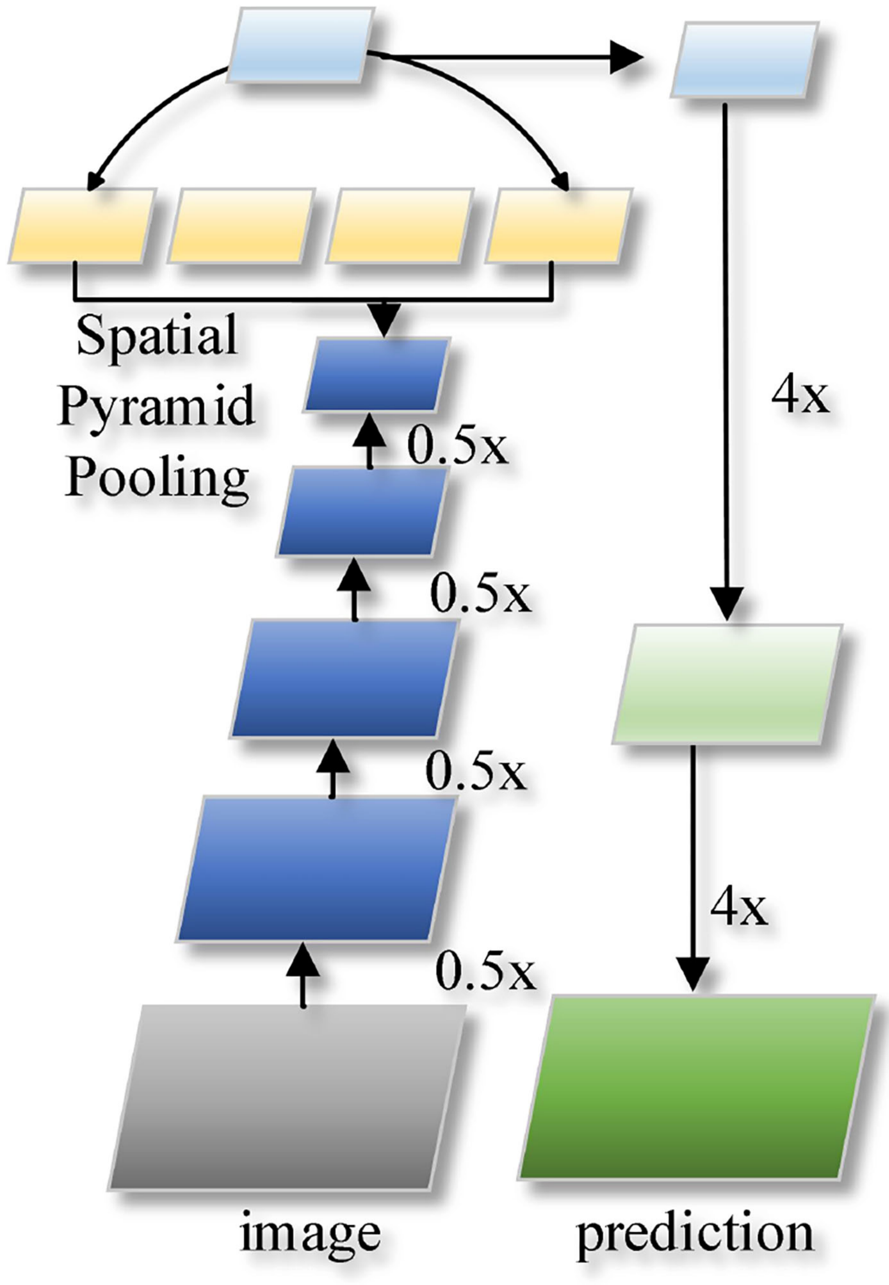


FIGURE 1 | DeeplabV3+ network structure.

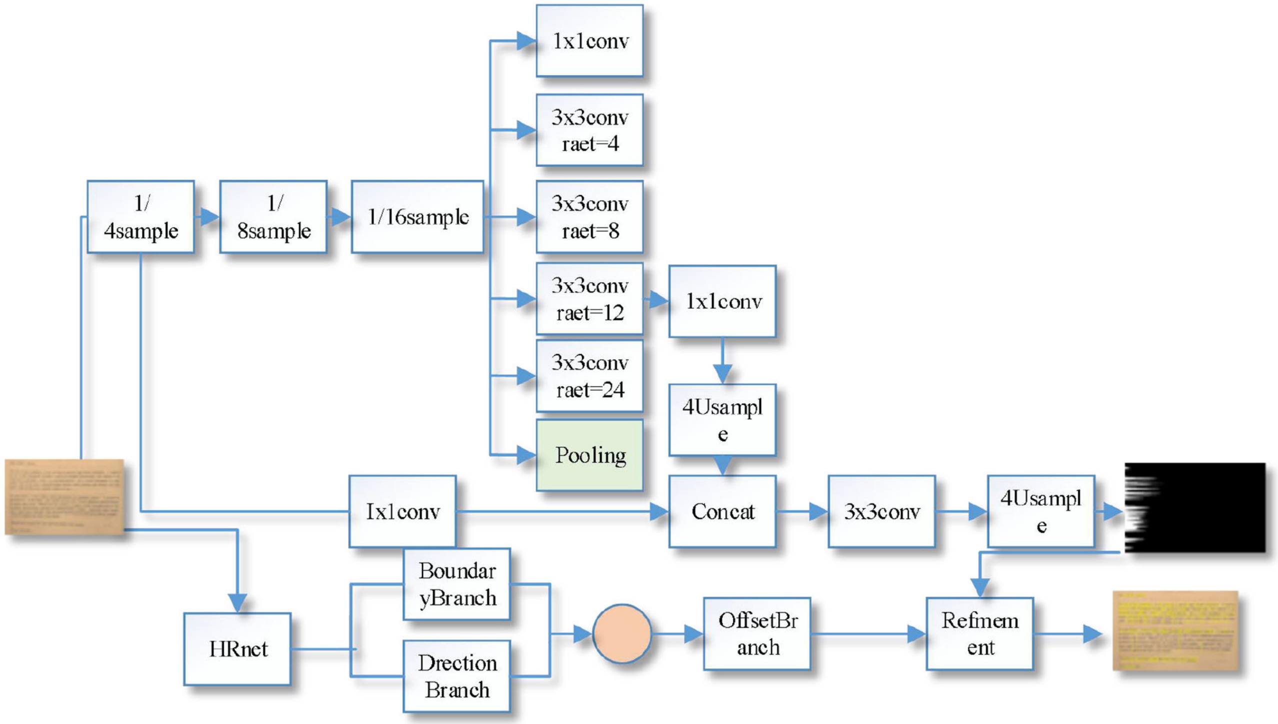


FIGURE 2 | Improved ladder-type DeeplabV3+ network structure.

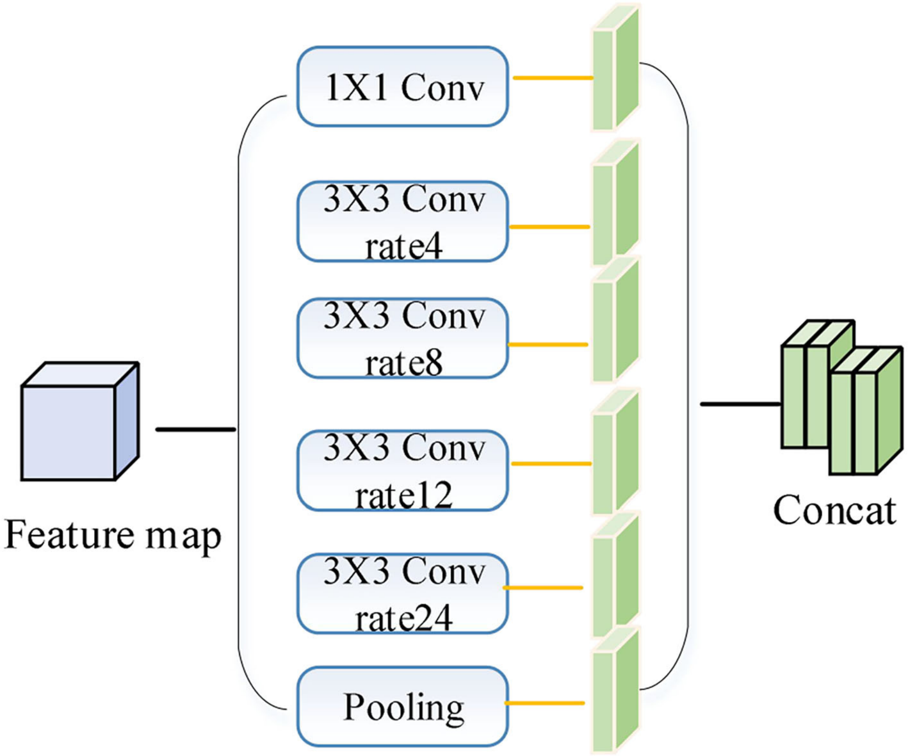


FIGURE 3 | Improved ASPP module.

extracted separately. The number of channels is adjusted using the convolution operation, and the bilinear FOE (Tian et al., 2019) quadruple up sampling is used to fuse the low-level spatial information with the adjusted number of channels across the layers, and the quadruple up sampling restores the original image resolution and spatial details.

Improved Stepwise Deeplab V3+ Network

Compared with the Deeplab V3+ network, the large scale target prediction is more prone to the problem of missing small scale targets and rough category boundaries. The improved Deeplab V3+ network is shown in **Figure 2**, which is based on ResNet-101 (Xie et al., 2019) as the backbone network, including encoder, decoder, and optimizer.

DeeplabV3+ network uses V3 model as encoder, and continues to use the original expansion convolution of V3 model ASPP module with expansion rate of 6, 12, and 18, while the feature map resolution decreases as CNN extracts the image feature information. Considering that when extracting low-resolution features, the expansion convolution of 4 and 8 can better capture the details of small-scale targets than the expansion convolution of 6, and when segmenting large-scale targets, it is necessary to obtain a larger sensory field, and the expansion convolution of 24 has a larger sensory field than the expansion convolution of 18, which is more favorable when segmenting large-scale targets. The ASPP parameters proposed in this paper are compared with the ASPP modules (6, 12, 18) provided by the V3+ model, and the proposed parameters are better than the original ones.

The original Deeplab V3+ model only designs a simple decoder, and the decoder mainly handles high and low-level feature map fusion operations; when performing feature map cross-layer fusion, considering that the 1/4 times downsampled feature map of the ResNet101 network contains rich low-level spatial information, while the 1/16 feature map generated by the encoder ASPP module contains rich high-level Therefore, in the fusion of feature maps, it is necessary to resize the high-level feature map generated by the ASPP module to the low-level feature map generated by the backbone network, so the 1/16th feature map generated by the encoder ASPP module

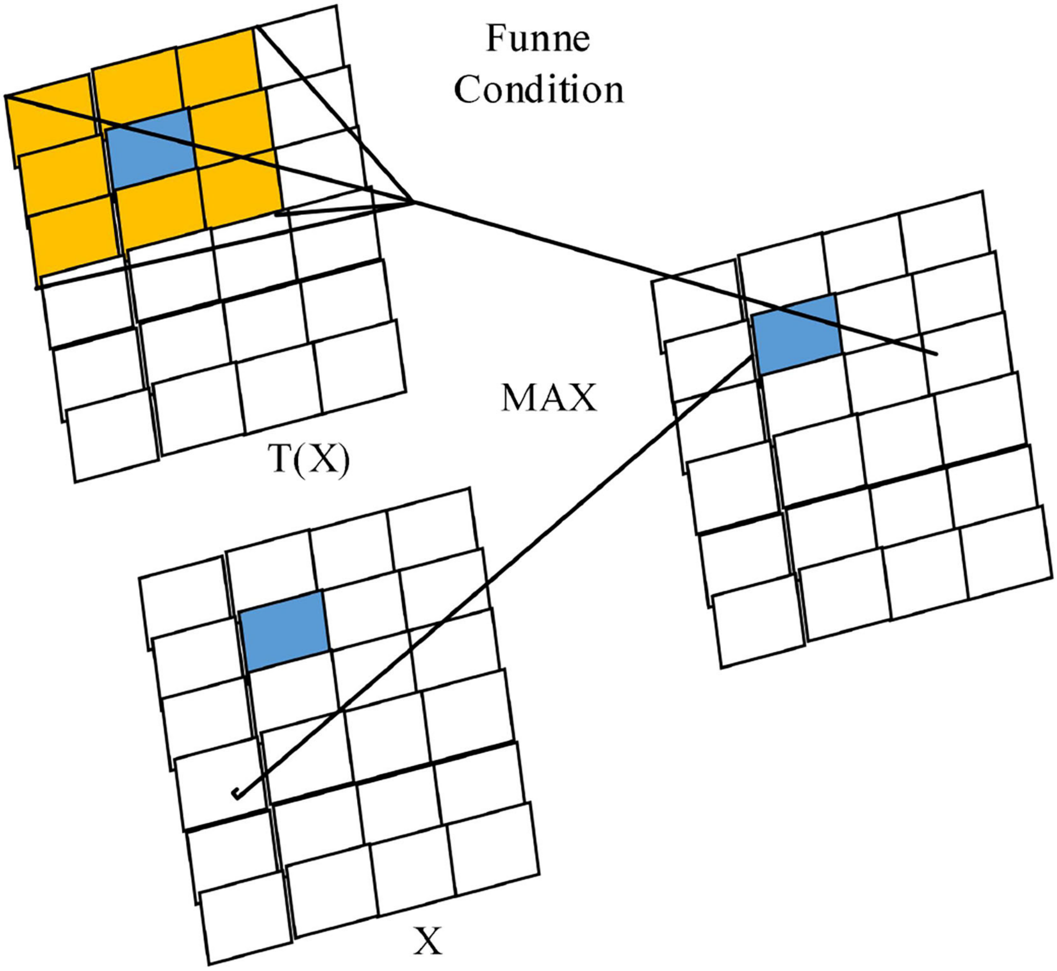


FIGURE 4 | Two-dimensional FReLU activation function with funnel condition.

is upsampled 4 times and then fused with the 1/4th feature map generated by the backbone network. Then convolution and upsampling operations are performed to generate the prediction result map; the ReLU activation function is used in the original codec network for non-linear activation, the reliability of the ReLU activation function has been recognized in the field of deep learning, but it lacks pixel-level modeling capability in computer vision tasks, so this paper uses the two-dimensional visual activation function FReLU to replace the ReLU activation function in the codec to obtain accuracy compensation (Shin et al., 2016).

Encoder Optimization

The ASPP module passes the input feature map evenly through different expansion rates of expansion convolution and global average pooling layers. The smaller expansion rate is more effective in segmenting small-scale targets; the larger expansion rate is more effective in segmenting large targets. The ASPP module in the encoder is improved as shown in **Figure 3**. The 1/16 feature maps generated by the backbone network are put into the 1 × 1 convolution, the expanded convolution with 4, 8, 12, and 24 expansion rates, and the global average pooling layer to generate 6 1/16-size feature maps with 256 channels, and the 6 feature maps are stitched together in the channel dimension to generate the ASPP module feature maps. The ASPP module feature maps are stitched together in the channel dimension to generate ASPP module feature maps, which can better extract multi-scale image features and improve the segmentation capability of the network for different scales of objects.

Code-and-Decoder Modeling Capability Optimization

In deep learning, CNN have good performance superiority in processing visual tasks. Non-linear activation function is a necessary component of CNN to provide good nonlinear modeling capability (Shin et al., 2016). Nowadays, the main common activation functions are ReLU and its evolved PReLU.

$$ReLU(x) = \begin{cases} x & \text{if} \quad x > 0 \\ 0 & \text{if} \quad x \leq 0 \end{cases}$$
$$PReLU(x) = \{ \begin{matrix} x_i, & \text{if} \quad x_i > 0 \\ a_i x_i & \text{if} \quad x_i \leq 0 \end{matrix}) \tag{1}$$

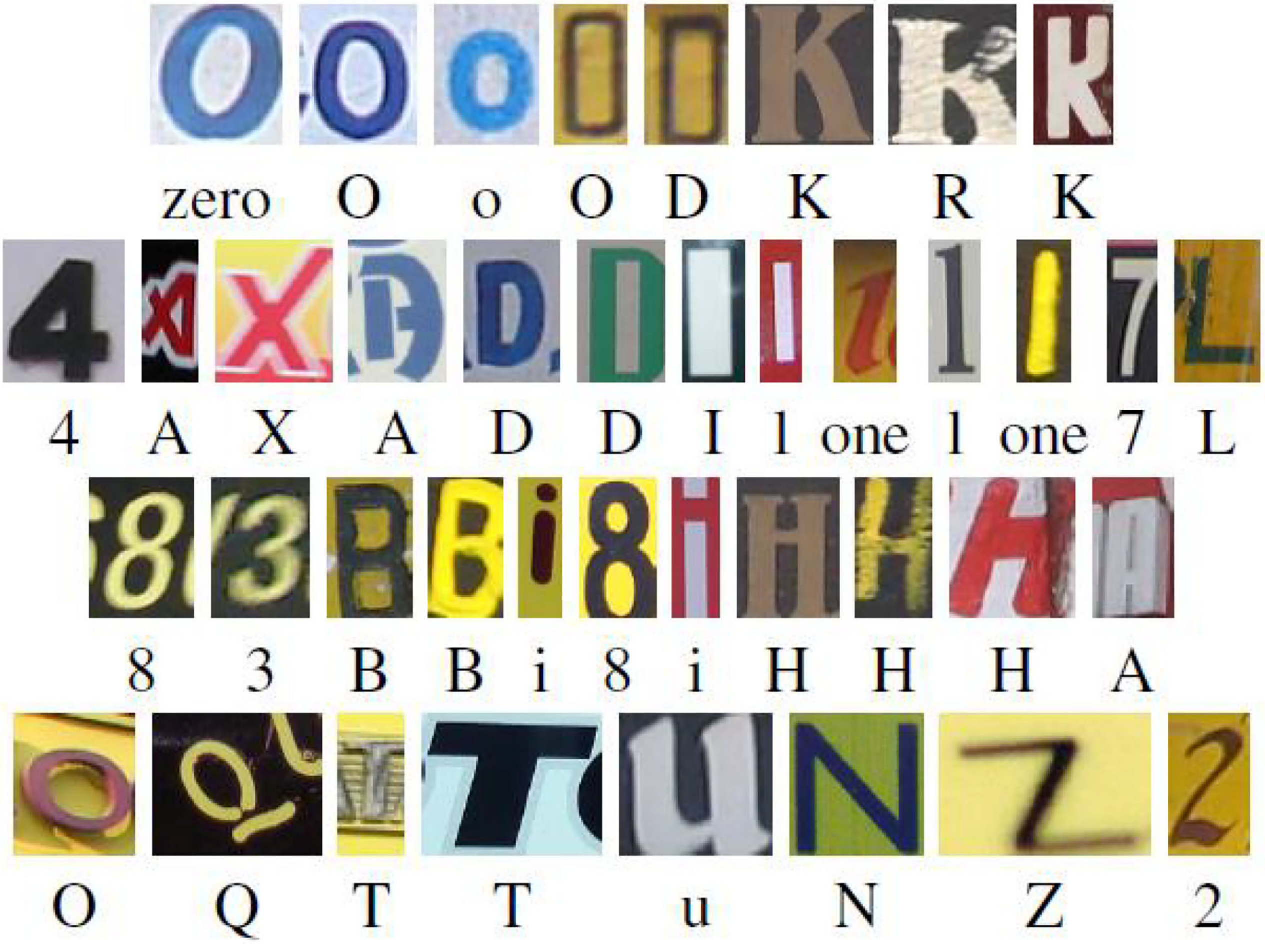
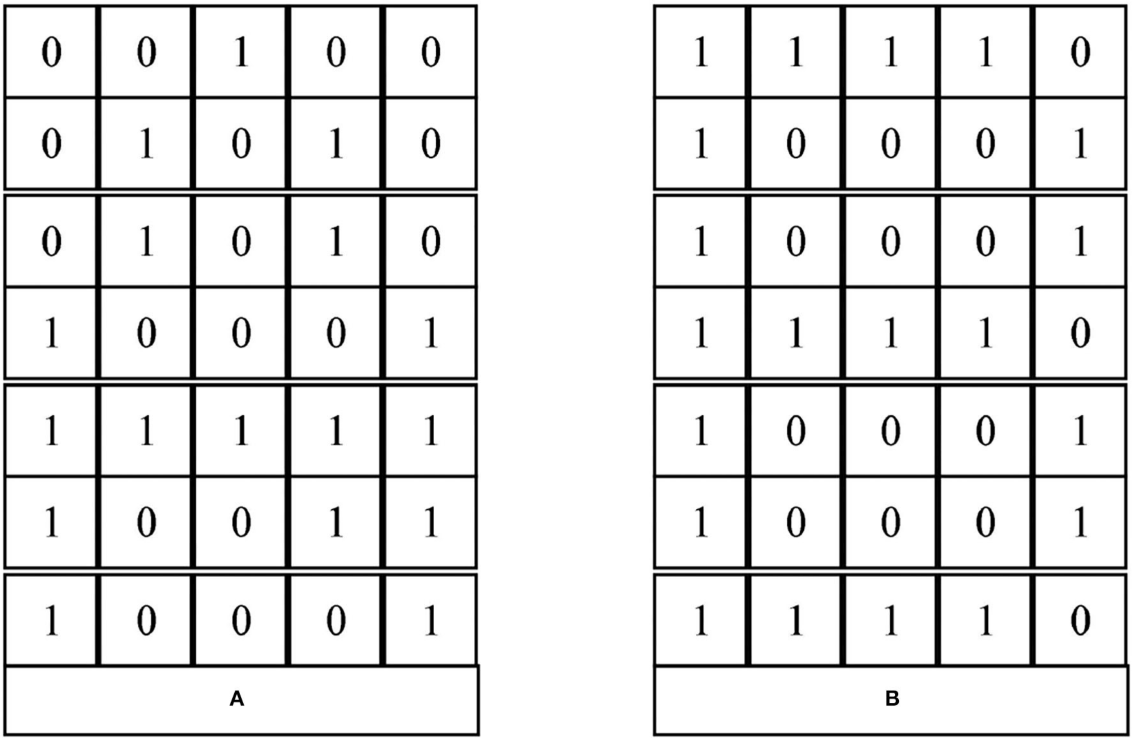


FIGURE 5 | Englishhnd dataset.



0	0	1	0	0
0	1	0	1	0
0	1	0	1	0
1	0	0	0	1
1	1	1	1	1
1	0	0	1	1
1	0	0	0	1
A				

1	1	1	1	0
1	0	0	0	1
1	0	0	0	1
1	1	1	1	0
1	0	0	0	1
1	0	0	0	1
1	1	1	1	0
B				

FIGURE 6 | Representation of the letters A and B.

ReLU as the most commonly used activation function, when the input is greater than zero, for the linear part of the function. However, when the input is less than zero, the function is adjusted by artificially setting the zero value. Therefore, there is a dead zone of activation, which leads to the poor robustness of the activation function during training, and the problem of "necrosis" of neurons when facing large gradient input. The gradient value is zero.

PReLU adds a linear activation part to the input less than zero by introducing a random parameter a that varies with the data computation. The above activation functions have been applied in various fields of deep learning with proven reliability. However, in the field of computer vision, these activation functions are unable to extract finer pixel-level spatial modeling capabilities, so the semantic segmentation network FReLU, a visual task activation function proposed by Shin et al. (2016) and Zhang et al. (2019), is used to compensate for the accuracy and obtain richer spatial contextual semantic information.

FReLU is a two-dimensional funnel-like activation function proposed specifically for computer vision tasks, which is expanded to two dimensions by adding the funnel condition T(X) to the one-dimensional ReLU activation function (as shown in **Figure 4**), introducing only a small amount of computation and overfitting risk to improve the vision task with spatially insensitive information in the activation network, with the expression:

$$\begin{aligned} f\left(x_{c,i,j}\right) &= max\left(x_{c,i,j},\ \mathrm{T}\left(x_{c,i,j}\right)\right) \\ \mathrm{T}\left(x_{c,i,j}\right) &=_{c,i,j}^{\omega} \cdot p_c^{\omega} \end{aligned} \tag{2}$$

where $x_{c,i,j}$ is the two-dimensional spatial location of the cth channel non-linear activation function $f\,(.)$ and function $T\,(.)$ is the functor condition; $x_{c,i,j}^{\omega}$ is the parametric pooling window on $x_{c,i,j}$; p_c^{ω} is the shared coefficient on the common channel; (.) is the dot product operation.

Its funnel condition is a square sliding window with preset parameters, which is realized by deep separable convolution and data normalization (BN), which can enhance the spatial dependence between pixel and pixel kweek, activate spatially insensitive information still while obtaining rich spatial context information, and improve the pixel-level spatial modeling capability. The graphical depiction of the funnel condition pixel-level modeling capability is shown in **Figure 4**; only a small number of parameters are introduced, introducing very little complexity. Considering the fact that in natural objects, besides vertical and horizontal directions, diagonal and circular arcs are also common, the pixel spatial information extracted by different activation layers is represented by squares of different sizes, and the diagonal and circular arc activation domains are formed by extreme approximation thinking to avoid the lack of modeling capability caused by using only the usual horizontal and vertical activation domains (Radwan et al., 2016).

METHOD IMPLEMENTATION

Data Pre-processing

The effect of this paper's model in English semantic analysis is verified. English is composed of letters, and compared with other languages, there are only 26 letters in English, and the form changes are relatively simple, so English characters can be recognized directly by neural networks.

The dataset used in this paper is Englishhnd (Saghezchi et al., 2013), in which the symbols used in English and Kannada are included. The English language includes: Latin characters (excluding accent marks) and Arabic numerals, and the dataset includes 64 classes (0–9, a–z, A–Z) of characters. Among them, there are 7,705 characters from natural images, 3,410 characters input by computer handwriting board, and 62,992 characters merged by computer font. Some of their characters, as shown in **Figure 5**.

Since this paper only recognizes English characters, firstly, the characters corresponding to "0∼9" in the data set are screened out. Second, each English letter is digitally represented as a 7×5 squares, as shown in **Figure 6**.

Figure 6 gives the digital representation of the capital letters A and B. The part of the letter with data is represented by 1 and the part without data is represented by 0. For different 52 letters (including case) there are 52 different representations. Then, according to the order from rows to columns, we can get 3 vectors of dimension 35 for different letters. "A" and "B" is represented as follows:

A = [0 0 1 0 0 0 1 0 1 0 0 1 0 1 0 1 0 0 0 1 1 1 1 1
1 1 0 0 0 1 1 0 0 0 1]
B = [1 1 1 1 0 1 0 0 0 1 1 0 0 0 1 1 1 1 1 0 1 0 0 0
1 1 0 0 0 1 1 1 1 1 0]

After digitizing the characters, the captured images are often disturbed by noise due to the actual English character recognition. Therefore, in order to simulate the actual application scenario, this paper superimposes noisy data on Englishhnd. The operation of adding noise can be implemented by the rand function in python software.

Simulation Results

In order to better evaluate the performance of the RBF network, a BP neural network is used in the paper for comparative simulation tests. In order to ensure the consistency of time and

TABLE 1 | Network parameters.

Parameter name	Parameter value
Number of hidden layers	1
Number of hidden units	500
Enter the number of nodes	35
Number of output nodes	52
Target error	0.0001
Maximum training times	40

TABLE 2 | Dataset parameters.

	BP network	Our model
Recognition accuracy	88.56%	96.35%
AUC	0.72	0.89

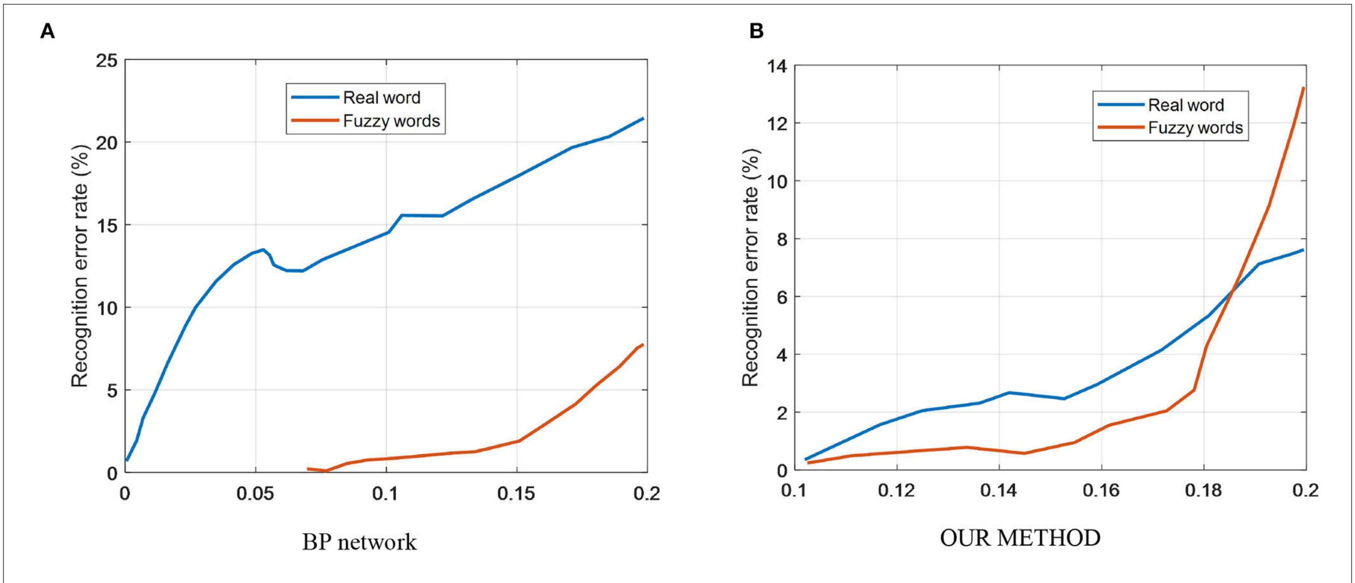


FIGURE 7 | Relationship between word vector dimensionality and F1, training time. **(A)** BP network. **(B)** Our method.

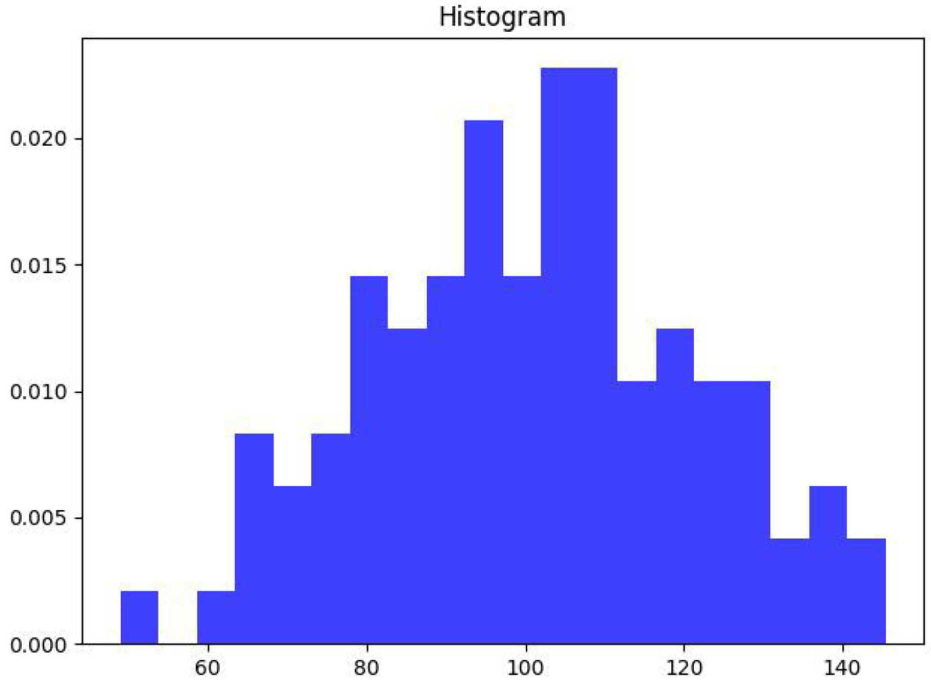
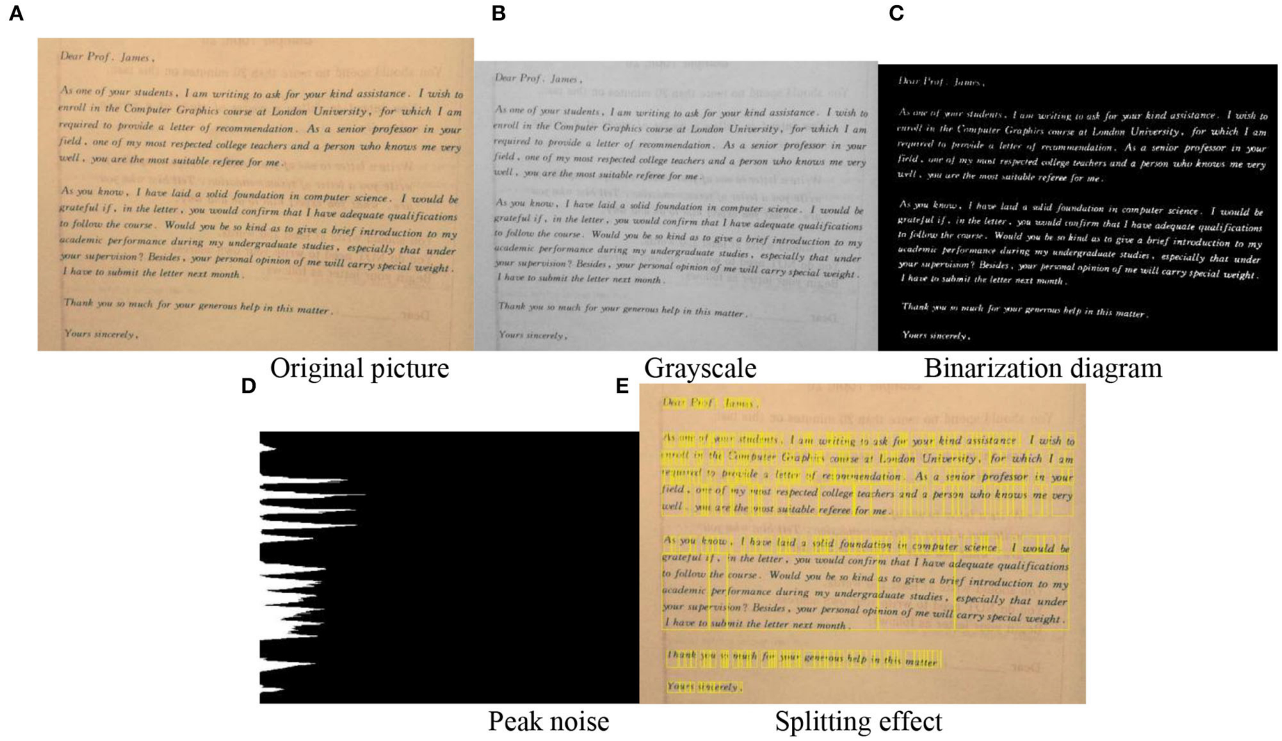


FIGURE 8 | Histogram of word vectors.



Dear Prof. James,

As one of your students, I am writing to ask for your kind assistance. I wish to enroll in the Computer Graphics course at London University, for which I am required to provide a letter of recommendation. As a senior professor in your field, one of my most respected college teachers and a person who knows me very well, you are the most suitable referee for me.

As you know, I have laid a solid foundation in computer science. I would be grateful if, in the letter, you would confirm that I have adequate qualifications to follow the course. Would you be so kind as to give a brief introduction to my academic performance during my undergraduate studies, especially that under your supervision? Besides, your personal opinion of me will carry special weight. I have to submit the letter next month.

Thank you so much for your generous help in this matter.

Yours sincerely,

FIGURE 9 | The semantic segmentation process of English units in this model. **(A)** Original picture. **(B)** Grayscale. **(C)** Binarization diagram. **(D)** Peak noise. **(E)** Splitting effect.

space complexity during the training of the two networks, the parameters of the two networks, as shown in **Table 1**.

In **Figure 7**, the solid line shows the change in the error rate at test with the test dataset after adding noise after training with the ideal signal; the dashed line shows the change in the error rate after using the noise-added signal.

As can be seen from the solid line in **Figure 7A**, the BP network is trained using the ideal signal without noise, and the error rate of the network recognition increases more when the test dataset is noise-added. The dashed line in **Figure 7A** shows that the network is less affected by noise in the test when trained using characters with noise-added signals. Therefore, the BP network is more disturbed by image noise, and this network has better recognition accuracy only when the test set is not noise-added.

The solid line in **Figure 7B** shows that when the our model is trained with noiseless data, the recognition error rate of the network only changes significantly after the mean value of noise exceeds 0.1 for the test data.

As can be seen from the dashed line in **Figure 7B**, when training with noisy data, the performance of the network also deteriorates after the mean value of noise exceeds 0.1 for the test data. Since the dashed line follows basically the same trend as the solid line, the our model is less disturbed by noise when performing character recognition and has stronger noise immunity compared to the BP neural network.

Table 2 gives the test results of BP and our models with a noise level of 0.1 for the test set after adding noise to the training data.

As can be seen from **Table 2**, with the same network parameters, training data and test data, the recognition accuracy of our model for English characters can reach 96.35%, which is 7.79% higher than that of BP network (88.56%); the AUC of our model reaches 0.89, which is closer to 1 than that of BP network (0.72).

As can be seen from **Figure 8**, the word vector histogram of this paper's scheme, an exact graphical representation of the distribution of the value data. The range of values is segmented, i.e., the entire range of values is divided into a series of intervals, and then how many values are counted in each interval. The values are usually specified as consecutive, non-overlapping intervals of variables. Intervals must be adjacent and usually (but not necessarily) of equal size.

Recognition of Segmentation Effects

The semantic segmentation process of English units in this model is shown in **Figure 9**, where **Figure 9A** shows the original image and **Figure 9E** shows the segmentation effect on English words. It is thought that the pixels in the image with gray scale values in the same class belong to the same object. Since it is a direct application of the gray scale characteristics of the image, the calculation is convenient and concise, and the applicability is strong. Obviously, the key and difficulty of the threshold segmentation approach is how to obtain a suitable threshold value. The threshold setting in **Figure 9B** is vulnerable to noise and luminance. The approaches in recent years are: the approach of selecting the threshold value with the maximum correlation criterion, the approach based on the image topology stable state, the Yager measure minimization approach, the gray scale co-generation matrix approach, the variance method, the entropy method, the peak and valley analysis method, etc. **Figure 9C** shows several algorithms that are more successful in improving the traditional shareholding method. In more cases, the selection of thresholds will be a combination of 2 or more approaches, which are also a trend in the development of image segmentation.

CONCLUSIONS

Analyzing English words under the synergistic effect of different modalities, students improve the motivation and effectiveness of word learning, but there are still some problems. In this paper, we construct a stepped network framework based on the Deeplab V3+ network, retain the inflated convolution and code-decoder structures in the original network, and replace the original non-linear activation function ReLU with a more effective visual activation function FReLU by improving the spatial pooling determinant module. Experimental results on the Englishhnd dataset show that the improved network results on the Englishhnd dataset show that the improved network has high recognition accuracy for English characters.

DATA AVAILABILITY STATEMENT

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

AUTHOR CONTRIBUTIONS

LP was responsible for designing the framework of the entire manuscript, from topic selection to solution to experimental verification.

REFERENCES

Cameron, L. (2002). Measuring vocabulary size in English as an additional language. *Lang. Teach. Res.* 6, 145–173. doi: 10.1191/1362168802lr103oa

Chen Hsieh, J. S., Wu, W. C. V., and Marek, M. W. (2017). Using the flipped classroom to enhance EFL learning. *Comput. Assist. Lang. Learn.* 30, 1–21. doi: 10.1080/09588221.2015.1111910

Chen, M., Wu, J., Liu, L., Zhao, W., Tian, F., Shen, Q., et al. (2021). DR-Net: An improved network for building extraction from high resolution remote sensing image. *Remote Sens.* 13:294. doi: 10.3390/rs13020294

Dai, Y., Huang, Z., Gao, Y., Xu, Y., Chen, K., Guo, J., et al. (2018). "Fused text segmentation networks for multi-oriented scene text detection," in *2018 24th International Conference on Pattern Recognition (ICPR)* (Beijing: IEEE), 3604–3609. doi: 10.1109/ICPR.2018.8546066

Di Wu, C. Z., Ji, L., Ran, R., Wu, H., and Xu, Y. (2021). Forest fire recognition based on feature extraction from multi-view images. *Traitement du Signal* 38, 775–783. doi: 10.18280/ts.380324

Duman, G., Orhon, G., and Gedik, N. (2015). Research trends in mobile assisted language learning from 2000 to 2012. *Recall* 27, 197–216. doi: 10.1017/S0958344014000287

Gu, S., and Zhang, F. (2020). Applicable scene text detection based on semantic segmentation. *J. Phys. Conf. Ser.* 1631, 012080. doi: 10.1088/1742-6596/1631/1/012080

Hessamy, G., and Ghaderi, E. (2014). The role of dynamic assessment in the vocabulary learning of Iranian EFL learners. *Proc. Soc. Behav. Sci.* 98, 645–652. doi: 10.1016/j.sbspro.2014.03.463

Huang, Y. M., Chiu, P. S., Liu, T. C., and Chen, T. S. (2011). The design and implementation of a meaningful learning-based evaluation method for ubiquitous learning. *Comput. Educ.* 57, 2291–2302. doi: 10.1016/j.compedu.2011.05.023

Huang, Z., Zhong, Z., Sun, L., and Huo, Q. (2019). "Mask R-CNN with pyramid attention network for scene text detection," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)* (Waikoloa, HI: IEEE), 764–772. doi: 10.1109/WACV.2019.00086

Laufer, B. (2006). Comparing focus on form and focus on forms in second-language vocabulary learning. *Can. Modern Lang. Rev.* 63, 149–166. doi: 10.3138/cmlr.63.1.149

Liu, J., Geng, Y., Zhao, J., Zhang, K., and Li, W. (2021). Image semantic segmentation use multiple-threshold probabilistic R-CNN with feature fusion. *Symmetry* 13, 207. doi: 10.3390/sym13020207

Liu, R., Mi, L., and Chen, Z. (2020). AFNet: adaptive fusion network for remote sensing image semantic segmentation. *IEEE Trans. Geosci. Remote Sens.* 60, 1–16. doi: 10.1109/TGRS.2020.3035561

Liu, W. (2021). Real-time obstacle detection based on image semantic segmentation and fusion network. *Traitement du Signal* 38, 443–449. doi: 10.18280/ts.380223

Liu, Y., Zhang, D., Lu, G., and Ma, W. Y. (2007). A survey of content-based image retrieval with high-level semantics. *Pattern Recogn.* 40, 262–282. doi: 10.1016/j.patcog.2006.04.045

Lyu, P., Liao, M., Yao, C., Wu, W., and Bai, X. (2018). "Mask textspotter: an end-to-end trainable neural network for spotting text with arbitrary shapes," in *Proceedings of the European Conference on Computer Vision (ECCV)* (Munich), 67–83. doi: 10.1007/978-3-030-01264-9_5

Radwan, A., Huq, K. M. S., Mumtaz, S., Tsang, K. F., and Rodriguez, J. (2016). Low-cost on-demand C-RAN based mobile small-cells. *IEEE Access* 4, 2331–2339. doi: 10.1109/ACCESS.2016.2563518

Saalbach, A., Lange, O., Nattkemper, T., and Meyer-Baese, A. (2009). On the application of (topographic) independent and tree-dependent component analysis for the examination of DCE-MRI data. *Biomed. Signal Process. Control* 4, 247–253. doi: 10.1016/j.bspc.2009.03.010

Saghezchi, F. B., Radwan, A., Rodriguez, J., and Dagiuklas, T. (2013). Coalition formation game toward green mobile terminals in heterogeneous wireless networks. *IEEE Wireless Commun.* 20, 85–91. doi: 10.1109/MWC.2013.6664478

Sanonguthai, S. (2011). Teaching IELTS writing module through English debate: a case study in Thailand. *Lang. Test. Asia* 1, 1–61. doi: 10.1186/2229-0443-1-4-39

Shin, H., Ahn, B., and Bae, D. (2016). English vocabulary learning through metacognitive memory strategy and vocabulary testing. *J. Modern Brit. Am. Lang. Lit.* 34, 121–149. doi: 10.21084/jmball.2016.02.34.1.121

Sung, Y. T., Chang, K. E., and Liu, T. C. (2016). The effects of integrating mobile devices with teaching and learning on students' learning performance: a meta-analysis and research synthesis. *Comput. Educ.* 94, 252–275. doi: 10.1016/j.compedu.2015.11.008

Tian, Z., Shu, M., Lyu, P., Li, R., Zhou, C., Shen, X., et al. (2019). "Learning shape-aware embedding for scene text detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Long Beach, CA), 4234–4243. doi: 10.1109/CVPR.2019.00436

Wu, J., and Chen, B. (2020). English vocabulary online teaching based on machine learning recognition and target visual detection. *J. Intell. Fuzzy Syst.* 39, 1745–1756. doi: 10.3233/JIFS-179948

Xie, E., Zang, Y., Shao, S., Yu, G., Yao, C., and Li, G. (2019). "Scene text detection with supervised pyramid context network," in *Proceedings of the AAAI Conference on Artificial Intelligence* (Hawaii), 9038–9045. doi: 10.1609/aaai.v33i01.33019038

Xie, T., Zhang, C., Zhang, Z., and Yang, K. (2018). Utilizing active sensor nodes in smart environments for optimal communication coverage. *IEEE Access* 7, 11338–11348. doi: 10.1109/ACCESS.2018.2889717

Xue, Y., Geng, H., Zhang, H., Xue, Z., and Xu, G. (2018). Semantic segmentation based on fusion of features and classifiers. *Multimedia Tools Appl.* 77, 22199–22211. doi: 10.1007/s11042-018-5858-z

Yin, M. (2021). Research on English vocabulary classification based on computer deep learning. *J. Phys. Conf. Ser.* 1992, 022074. doi: 10.1088/1742-6596/1992/2/022074

Zhang, C., and Liu, X. (2021). Feature extraction of ancient Chinese characters based on deep convolution neural network and big data analysis. *Comput. Intell. Neurosci.* 2021, 1–10. doi: 10.1155/2021/2491116

Zhang, C., Xie, T., Yang, K., Ma, H., Xie, Y., Xu, Y., et al. (2019). Positioning optimisation based on particle quality prediction in wireless sensor networks. *IET Netw.* 8, 107–113. doi: 10.1049/iet-net.2018.5072

Zhang, R., Li, G., Li, M., and Wang, L. (2018). Fusion of images and point clouds for the semantic segmentation of large-scale 3D scenes based on deep learning. *ISPRS J. Photogramm. Remote Sens.* 143, 85–96. doi: 10.1016/j.isprsjprs.2018.04.022

Zhang, Z., Zhang, C., Li, M., and Xie, T. (2020). Target positioning based on particle centroid drift in large-scale WSNs. *IEEE Access* 8, 127709–127719. doi: 10.1109/ACCESS.2020.3008373

Zhao, W., Wang, B., Coniam, D., and Xie, B. (2017). Calibrating the CEFR against the China Standards of English for College English vocabulary education in China. *Lang. Test. Asia* 7, 1–18. doi: 10.1186/s40468-017-0036-1

Zhou, H., Fang, Z., Gao, Y., Huang, B., Zhong, C., and Shang, R. (2020). Feature fusion network based on attention mechanism for 3D semantic segmentation of point clouds. *Pattern Recogn. Lett.* 133, 327–333. doi: 10.1016/j.patrec.2020.03.021

Conflict of Interest: The 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.

ORIGINAL RESEARCH
published: 15 July 2022
doi: 10.3389/fncom.2022.930827

An Infrared Sequence Image Generating Method for Target Detection and Tracking

Huang Zhijian[1,2], *Hui Bingwei*[3]* and *Sun Shujin*[2]

[1] School of Computer Engineering and Applied Mathematics, Changsha University, Changsha, China, [2] Hunan Province Key Laboratory of Industrial Internet Technology and Security, Changsha, China, [3] Automatic Target Recognition (ATR) Key Laboratory, School of Electronic Science, National University of Defense Technology, Changsha, China

Training infrared target detection and tracking models based on deep learning requires a large number of infrared sequence images. The cost of acquisition real infrared target sequence images is high, while conventional simulation methods lack authenticity. This paper proposes a novel infrared data simulation method that combines real infrared images and simulated 3D infrared targets. Firstly, it stitches real infrared images into a panoramic image which is used as background. Then, the infrared characteristics of 3D aircraft are simulated on the tail nozzle, skin, and tail flame, which are used as targets. Finally, the background and targets are fused based on Unity3D, where the aircraft trajectory and attitude can be edited freely to generate rich multi-target infrared data. The experimental results show that the simulated image is not only visually similar to the real infrared image but also consistent with the real infrared image in terms of the performance of target detection algorithms. The method can provide training and testing samples for deep learning models for infrared target detection and tracking.

Keywords: infrared image simulation, infrared target simulation, infrared radiation, deep learning, Unity3D

OPEN ACCESS

Edited by:
Deepika Koundal,
University of Petroleum and Energy Studies, India

Reviewed by:
Arvind Dhaka,
Manipal University Jaipur, India
Amita Nandal,
Manipal University Jaipur, India

****Correspondence:***
Hui Bingwei
huibingwei07@163.com

Received: *28 April 2022*
Accepted: *10 June 2022*
Published: *15 July 2022*

Citation:
Zhijian H, Bingwei H and Shujin S (2022) An Infrared Sequence Image Generating Method for Target Detection and Tracking.
Front. Comput. Neurosci. 16:930827.
doi: 10.3389/fncom.2022.930827

INTRODUCTION

With the rapid development of deep-learning technology, data-driven models and algorithms have become a hot topic in infrared target detection and tracking (Dai et al., 2021; Hou et al., 2022). Unlike conventional methods, data-driven methods require a large amount of infrared data for model training and testing (Yi et al., 2019; Junhong et al., 2020).

However, the current infrared image datasets used for object detection and tracking are of poor quality (Hui et al., 2020). The cost of measured data is high, and it is difficult to obtain infrared images in various scenarios (Zhang et al., 2018). For example, the target type in real data is single, and it is difficult to obtain infrared images of important types of aircraft. The authenticity of the simulation data is insufficient (Xia et al., 2015). The battlefield in modern warfare involves a wide range of complex environments. It is difficult for knowledge-based models to simulate a complex infrared battlefield. These problems significantly limit research progress in infrared target detection and tracking.

Currently, infrared target simulation can be performed using two approaches: methods based on infrared characteristic modeling (Shuwei and Bo, 2018; Guanfeng et al., 2019; Yongjie et al., 2020) and methods based on deep neural networks (Mirza and Osindero, 2014; Alec et al., 2016; Junyan et al., 2017; Chenyang, 2019; Yi, 2020). The former is typically based on infrared radiation theory. Physical models of various parts of an aircraft (such as engines, tail nozzles, tail flames, and casings) are established, atmospheric radiation is modeled, and infrared simulation data under various conditions are obtained. These methods start with a physical model and have strong interpretability. If sufficient parameters are added, high-fidelity infrared images can be produced (Yunjey et al., 2020). With a large number of parameters and calculations, they are suitable for simple target simulations. However, these are unsuitable for real-environment simulations with complex types of ground objects (Chenyang, 2019; Rani et al., 2022). Methods based on deep learning, typically using a generative adversarial network (GAN), learn the style of the infrared image from a large number of real infrared images and then transfer visible light images to infrared images (Alec et al., 2016; Junyan et al., 2017; Chenyang, 2019; Yi, 2020). These methods do not require complex physical modeling processes and are fast, but lack authenticity and reliability (Shi et al., 2021; Bhalla et al., 2022). More importantly, the method is based on deep learning and cannot add infrared targets as needed, nor can it edit the flight trajectory and attitude, which is exactly what the infrared target dataset needs most.

Therefore, it is meaningful and valuable to study an infrared data generation method that conforms to the real infrared radiation characteristics, and can add multiple types and multiple aircraft targets arbitrarily. This paper proposed a new method, and its main contributions are as follows:

(1) A method combining the real infrared data of background with the simulated infrared data of target is proposed, which can easily generate multi-target infrared simulation data with high authenticity. It uses the panorama of the real infrared data mosaic as the background, rather than the direct 3D infrared simulation of the ground objects. It can avoid the complex problem of infrared modeling of ground objects. Compared with the 3D infrared simulation of the whole scene, it is much easier, and the generated data are more authentic.

(2) The method is based on the Unity3D to fuse the target model with the infrared scene. It can freely add the type and number of aircrafts, edit the aircraft trajectory, and attitude. So it can generate rich multi-target infrared simulation data.

(3) Starting from the infrared radiation characteristics, our method simulates the physical characteristics of the key parts of the 3D target (the tail nozzle, skin, and tail flame), which can generate high authenticity infrared target data.

METHODS

Overall Framework

Figure 1 shows the overall framework of this study, divided into three branches: infrared background stitching, infrared radiation modeling, and flight trajectory editing. The infrared radiation modeling branch first establishes a 3D model on the basis of the size of the aircraft and then establishes an infrared radiation model of the aircraft according to the infrared radiation theory (such as the engine nozzle, skin, and tail flame). The infrared background stitching branch performs panoramic stitching based on real infrared dataset, and after uniform light processing, a uniform infrared panoramic image is obtained. We used the infrared panorama as background for the 3D scene. The flight-trajectory editing branch provides trajectory-editing tools. Users can call editing tools to create flight trajectories based on the aircraft performance parameters. The trajectory included the time, position, and attitude of each node. The observation window can track and record targets in a field of view of a specified size. Because multiple and various types of aircrafts can be selected and various trajectories can be edited, a rich variety of infrared simulation data can be obtained.

Infrared Target Modeling

As an infrared radiation source, the radiation characteristics of different parts of an aircraft show evident differences owing to different degrees of heat generation. The main components with the strongest infrared radiation include the engine nozzle, aircraft skin, and tail flame (Haixing et al., 1997). This study starts with the basic theory of infrared radiation, grasps the main infrared radiation characteristics of each component, and establishes its infrared radiation intensity model.

Assuming that the infrared detector can perceive light of wavelengths ranging from λ_1 to λ_2 (only mid-wave infrared is considered in this study, that is, the wavelength range is 3–5 μm), according to the Planck's law (Yu, 2012), the infrared radiation intensity of a gray body can be expressed as:

$$M_{\lambda_1 \sim \lambda_2} = \int_{\lambda_2}^{\lambda_1} \frac{c_1}{\lambda^5} \frac{1}{e^{c_2/\lambda T}-1} d\lambda = \frac{c_1 T^4}{c_2^4} \int_{c_2/\lambda_2 T}^{c_1/\lambda_1 T} \frac{(c_2/\lambda T)^3}{e^{c_2/\lambda T}-1} d\left(\frac{c_2}{\lambda T}\right) \quad (1)$$

where T is the gray body surface temperature, c_1 is the first radiation constant, typically $(3.741774 \pm 0.0000022) \times 10^{-16}\mathrm{W} \cdot \mathrm{m}^2$, and c_2 is the second radiation constant, typically $(1.4387869 \pm 0.00000012) \times 10^{-2}\mathrm{m} \cdot \mathrm{K}$. Assuming $x = c_2/\lambda T$, the above equation can be simplified as follows:

$$M_{\lambda_1 - \lambda_2} = \frac{c_1 T^4}{c_2^4} \int_{c_2/\lambda_2 T}^{c_1/\lambda_2 T} \frac{x^3}{e^x - 1} dx \quad (2)$$

Nozzle Radiation Model

When the fuel in an engine burns, it emits high-temperature radiation, which is the main heat source when the aircraft is flying (Chuanyu, 2013). As an extension of the engine outside the fuselage, the tail nozzle also exhibits relatively strong infrared radiation. The tail nozzle is a typical gray body, and the surface emissivity is approximately in the range of 0.8–0.9. According to Equation (2), the relationship between the infrared radiation intensity of the tail nozzle I_W and temperature T_W is as follows:

$$I_W = \frac{\varepsilon_W}{\pi} \int_{\lambda_1}^{\lambda_2} \frac{c_1}{\lambda^5} \frac{1}{e^{c_2/\lambda T_\mathrm{w}}-1} d\lambda \cdot S_W \cdot \cos\theta_W \quad (3)$$

where ε_M is the radiation rate of the nozzle surface, which is determined by the aircraft surface material. $\mathrm{S_M}$ is the cross-sectional area of the skin facing the probe. θ_M is the angle

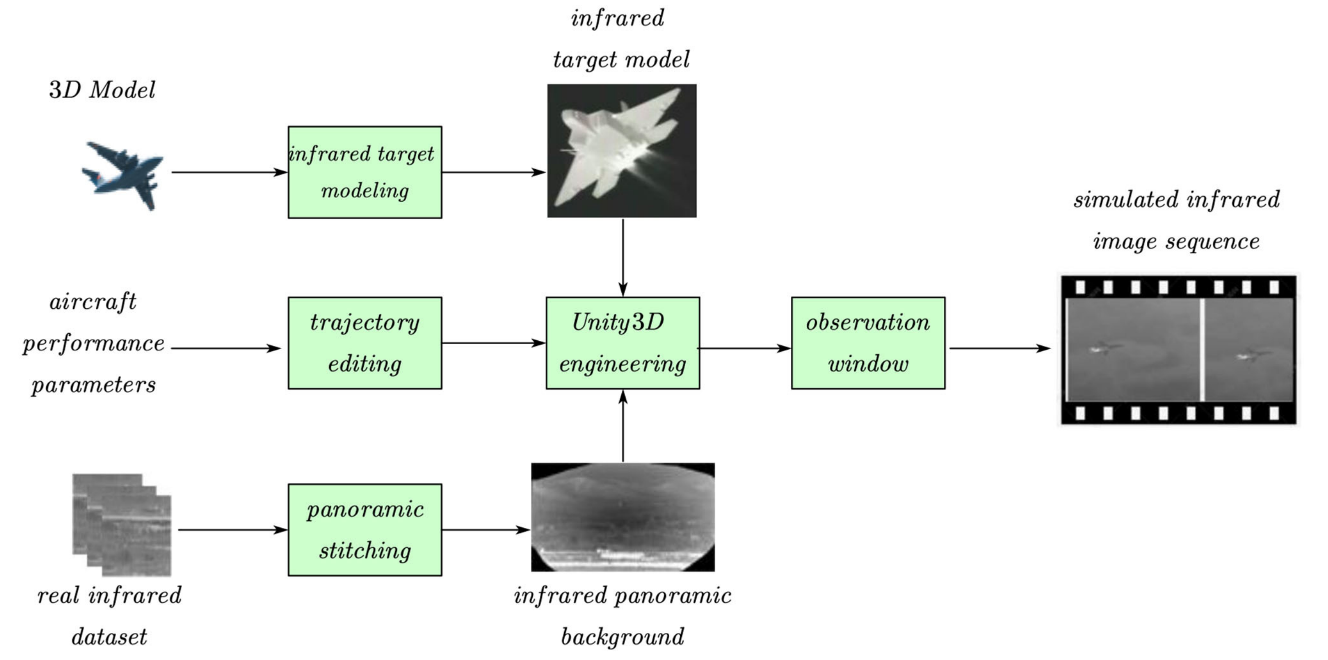


FIGURE 1 | Overall framework of this study.

between the orientation of the probe and the orientation of the infrared radiation.

Aircraft Skin Radiation Model

Aircraft skin temperature is mainly affected by two factors: the ambient temperature of the atmosphere and the temperature generated by the friction between the aircraft and the atmosphere during the high-speed motion. Because this study only considers aircraft flying at medium and low altitudes, the linear relationship between the atmospheric ambient temperature T_0 and altitude H satisfies T0=(288.2-0.0065 H) K, and T0=280 K for simplicity. The temperature T_{M} generated by friction and flight speed follow the following functional relationship: $T_{\mathrm{M}} = T_0\left(1+0.16M^2\right)$, where M is the Mach number of the aircraft.

Furthermore, according to Equation (2), the functional relationship between the aircraft skin radiation intensity I_{M} and temperature T_{M} is as follows:

$$I_M = \frac{\varepsilon_M}{\pi}\int_{\lambda_1}^{\lambda_2}\frac{c_1}{\lambda^5}\frac{1}{e^{c_1/2T_M}-1}d\lambda\cdot S_M\cdot\cos\theta_M \tag{4}$$

where ε_M is the skin surface emissivity, which is determined by the surface material of the aircraft skin. S_{M} is the cross-sectional area of the aircraft skin facing the probe, and θ_M is the angle between the probe and infrared radiation orientation.

Tail Flame Radiation Model

The high-temperature flame and high-temperature gas injected by the engine form the tail flame of the aircraft. We assume that the gas temperature in the tail nozzle is T_{F}, the tail flame temperature is T_{P}, and the gas pressures inside and outside the tail nozzle are P_{P} and P_{F}, respectively; then, we have:

$$T_p = T_F\left(P_p/P_F\right)^{(\gamma-1)/\gamma} \tag{5}$$

where γ is the specific heat of the gas; its value for turbofan aeroengines is 1.3. According to Equation (2), the functional relationship between the radiation intensity I_{P} of the tail nozzle and temperature T_{P} can be established as follows:

$$I_p = \frac{\varepsilon_p}{\pi}\int_{2}^{2}\frac{c_1}{\lambda^5}\frac{1}{e^{\sigma_2/2T_p}-1}d\lambda\cdot S_p\cdot\cos\theta_p \tag{6}$$

where ε_ρ is the surface emissivity of the aircraft tail flame, S_{P} is the cross-sectional area of the aircraft tail flame facing the probe, and θ_P is the angle between the probe and infrared radiation orientation. To improve the intuitive effect, the tail flame is typically simulated by particle flow. Based on the above-infrared radiation model, a 3D target with infrared radiation characteristics was obtained. The infrared radiation intensity of an aircraft dynamically changes with the speed and attitude of the target. **Figure 2** shows the simulation effect of F-35 aircraft at different attitudes. **Figure 3** shows the simulation effect of Su-35 aircraft at different speeds.

Panoramic Stitching of Infrared Images

We expect the targets to fly in a wide infrared scene to obtain a simulated image sequence of moving targets. However, the field of view of infrared sensors is typically narrow. For example, the field of view in the public infrared dataset (Hui et al., 2020)

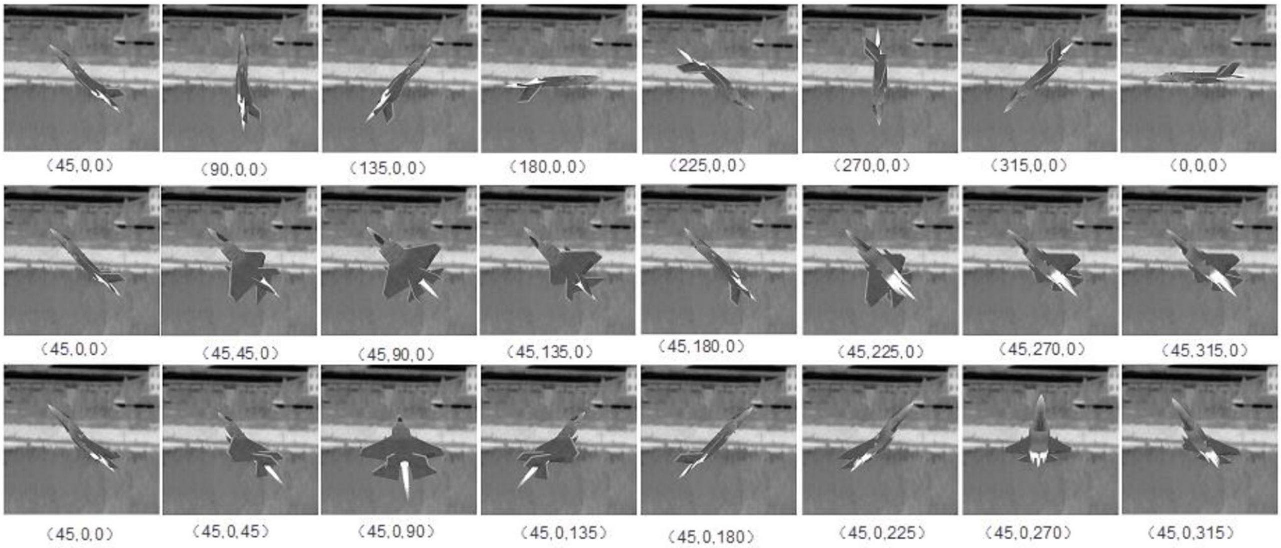


FIGURE 2 | Simulation effect of F-35 aircraft at different attitudes. The speed is Mach 1, and the background is a real infrared image. The coordinates are roll, yaw, and pitch.

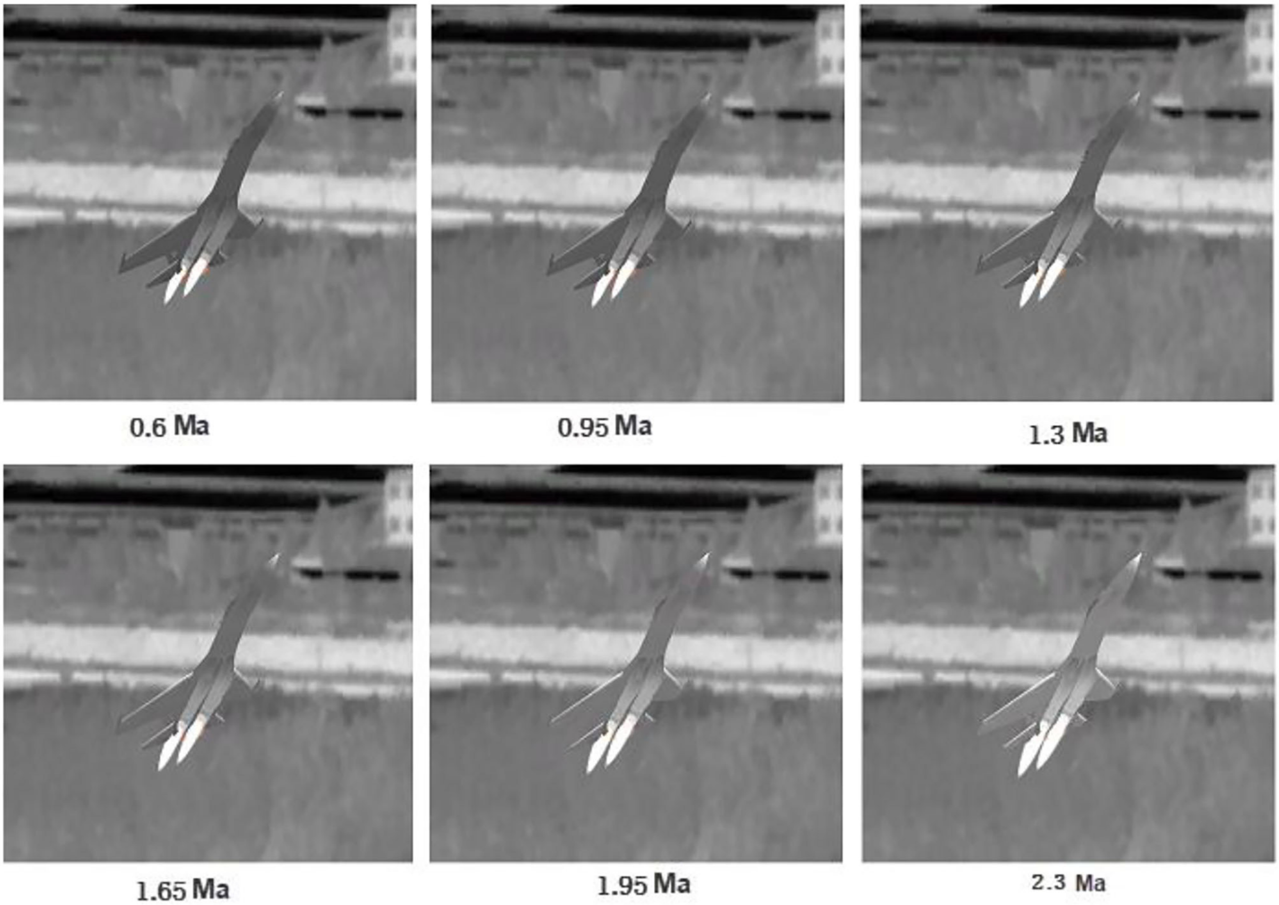


FIGURE 3 | Infrared characteristics of Su-35 aircraft at different speeds. The speed varies from 0.6 to 2.3 Ma.

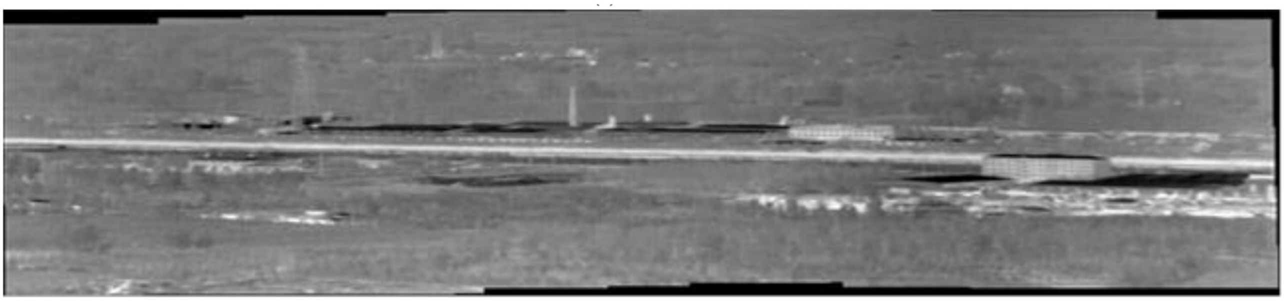


FIGURE 4 | Panoramic stitching results of real infrared images.

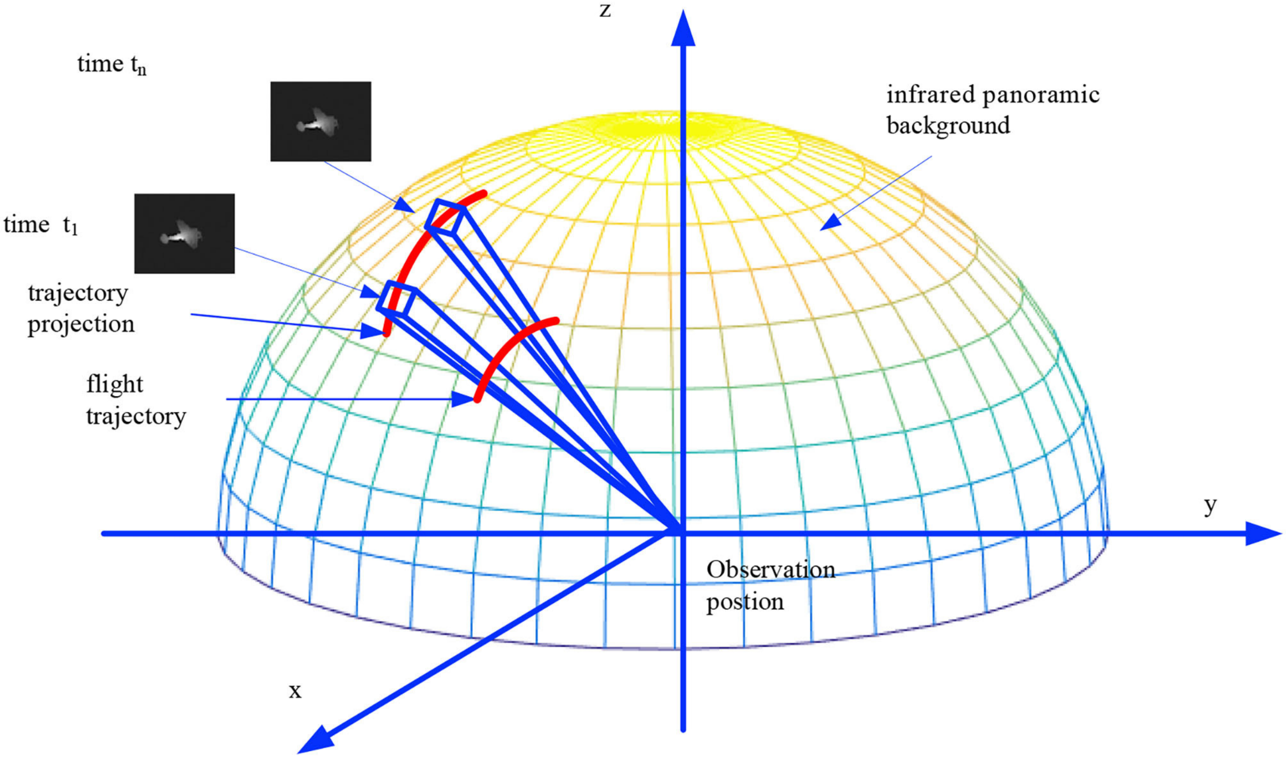


FIGURE 5 | Fusion of simulation targets and real infrared scene.

(dataset used for infrared detection and tracking of dim-small aircraft targets under a ground/air background, http://www.csdata.org/p/387/) is only 1° × 1°.

To obtain a continuous projection of the moving target in a real infrared scene, it is necessary to stitch infrared images of a narrow field of view into a panoramic image. In view of the small texture and low contrast of infrared images, a stitching and fusion method must be adopted specifically for infrared images, as detailed in our previous paper (Zhijian et al., 2021), which describes how to stitch a panoramic image from infrared sequence images. **Figure 4** shows only a part of the stitching results.

Fusion of Simulated Targets and Real Infrared Scene

This study realized the fusion of a static real infrared scene and dynamic simulated targets based on the Unity3D engine. The main steps were as follows: (1) Constructing a hemisphere with the camera position as the center and the real farthest observation distance as the radius. The panoramic image obtained by splicing real infrared images was used as the epidermis to cover the hemisphere to obtain a pseudo 3D scene, as shown in **Figure 5**. (2) Based on the flight trajectory (information, such as the position, attitude, and speed of the aircraft at each moment, is set), the 3D infrared simulation target flies in a 3D space. (3)

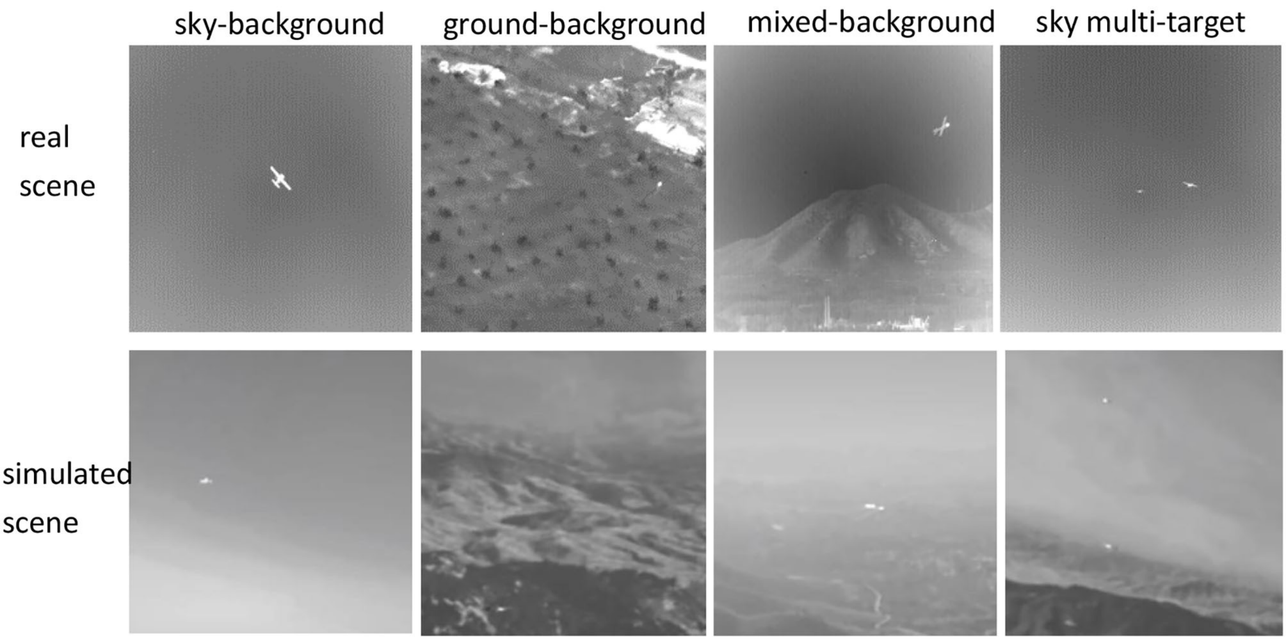


FIGURE 6 | Real infrared scene and simulated infrared scene.

Through human–computer interaction, the observation position and viewing angle were dynamically adjusted to track and observe the targets. (4) Each frame of the observation projects the target onto the infrared background and obtains the target infrared data with the real infrared background. With continuous observation, dynamic simulation image sequences of the targets can be obtained.

EXPERIMENT AND ANALYSIS

Dataset and Experiment Setting

The real infrared data used in this experiment comes from the public infrared dataset (Hui et al., 2020) (dataset used for infrared detection and tracking of dim-small aircraft targets under a ground/air background, http://www.csdata.org/p/387/). The dataset covers a variety of scenes such as sky and ground, with a total of 22 data segments, 30 tracks, 16,177 images, and 16,944 targets. Each frame is a gray image with a resolution of 256 × 256 pixels, BMP format, 1° × 1° field of view. Each target corresponds to a label position, and each data segment corresponds to a label file. This data set is usually used in the basic research of dim-small target detection, precision guidance, and infrared target characteristics.

The hardware environment of this experiment is: Dual Core CPU above 2.0 GHz and body memory above 4G. Software environment: system software above Windows 7. The experiment is based on the development of 2021.2.6f1 version of Unity3D. The development language is c#, and the development platform is visual studio 2017.

Subjective Analysis

We selected four scenes from real infrared data introduced in (Hui et al., 2020): sky background, ground background, mixed background, and sky multi-target, which are from data 1, data 7, data 3, and data 2, respectively, in the public dataset. Correspondingly, we also intercepted the above four scenarios from the simulation data, and the comparative results are shown in **Figure 6**. Visually and intuitively, both the real and simulated data have the following characteristics: (1) The images are gray overall, which conforms to the characteristics of infrared images. (2) The images have low contrast and relatively few textural features. (3) The target appears as bright spots and diffuses into the surroundings. Therefore, the simulated and real infrared data are intuitively similar.

Objective Analysis

The purpose of this study was to provide simulation data for the training and testing of infrared target detection and tracking models. Therefore, determining whether the performance of an algorithm on simulated data is consistent with that of the algorithm on real data is the most effective evaluation method (Deng et al., 2022). We used two algorithms (Zhijian et al., 2021; Deng et al., 2022) employed in the 2nd Sky Cup National Innovation and Creativity Competition in 2019 for testing. We compared their performance both on real infrared data and simulated data generated by our method.

In the experiment, the data shown in **Figure 6** were used; the real infrared data came from data 1, data 7, data 3, and data 2 in the public dataset (Hui et al., 2020). The simulation data also included the sky background, ground background, mixed background, and multiple targets. The resolution was 256 × 256. The targets were all small, that is, <10 pixels.

As in (Zhijian et al., 2021; Deng et al., 2022), four indicators, namely the accurate detection rate, correct detection rate, missed detection rate, and false alarm rate, were used to evaluate the performance of the algorithm. An accurate detection (Acc) is when the detection result is within the 3 × 3 pixel range of the ground truth. Correct detection (Corr) is when the detection result is within the 9 × 9 pixel range of the ground truth. Missing detection (Miss) is when the detection result is outside the 9 × 9 pixel range of the ground truth. A false alarm (FA) refers to a detected non-real target. **Tables 1**, **2** present the detection results without changing any parameters of the original algorithm.

TABLE 1 | Infrared target detection results on real and simulated data with algorithm (Tianjun et al., 2019).

	Sky		Ground		Mixed		Multi-targets	
	Real	**Simu**	**Real**	**Simu**	**Real**	**Simu**	**Real**	**Simu**
Acc (%)	100	99.5	91.5	84.7	94.7	90.2	99.0	98.4
Corr (%)	100	100	93.0	90.1	96.0	93.4	99.5	99.5
Miss (%)	0.0	0.0	4.0	9.9	4.0	6.6	0.5	0.5
FA (%)	0.0	0.0	1.8	3.0	1.2	0.5	0.0	0.0

TABLE 2 | Infrared target detection results on real and simulated data with algorithm (Xianbu et al., 2019).

	Sky		Ground		Mixed		Multi-targets	
	Real	**Simu**	**Real**	**Simu**	**Real**	**Simu**	**Real**	**Simu**
Acc (%)	100	99.2	92.7	88.3	65.0	70.0	98.7	92.4
Corr (%)	100	100	97.2	92.1	79.0	83.3	99.2	95.3
Miss (%)	0.0	0.0	2.8	17.9	21.0	16.7	0.8	4.7
FA (%)	0.0	0.0	1.5	2.3	0.0	1.3	0.0	0.0

As shown in **Table 1**, the algorithm reported in (Tianjun et al., 2019) performed well on the above four types of scenes, particularly in terms of the Acc and Corr indicators on sky background and multi-target scenes, which reached more than 99%. The performance on the ground background and mixed background is slightly worse; nevertheless, the accurate detection rate is above 90%. On the simulation data, the algorithm also performed well on sky background and multi-target scenes and is similar to the detection results on real data. On the ground and mixed backgrounds, the detection results of the simulated data are slightly worse than those of the real data; nevertheless, the maximum difference in the accurate detection rates is no more than 7% (on the ground background, the difference between the accurate detection rates of the real and simulated data was 6.8).

The performance of the simulation data generated by our method and the real data in the algorithm (Tianjun et al., 2019) is compared as shown in **Figure 7**. When it performs well on the real dataset, the simulation data generated by our method also perform well, such as in sky and multi-targets scenarios. When its performance of real datasets is poor, the simulation data generated by our method is also poor, such as in ground and mixed scenarios. This consistency is both reflected in the ACC and Corr indicators. Therefore, the simulation data generated by our method are consistent with the real data on the performance of algorithm (Tianjun et al., 2019).

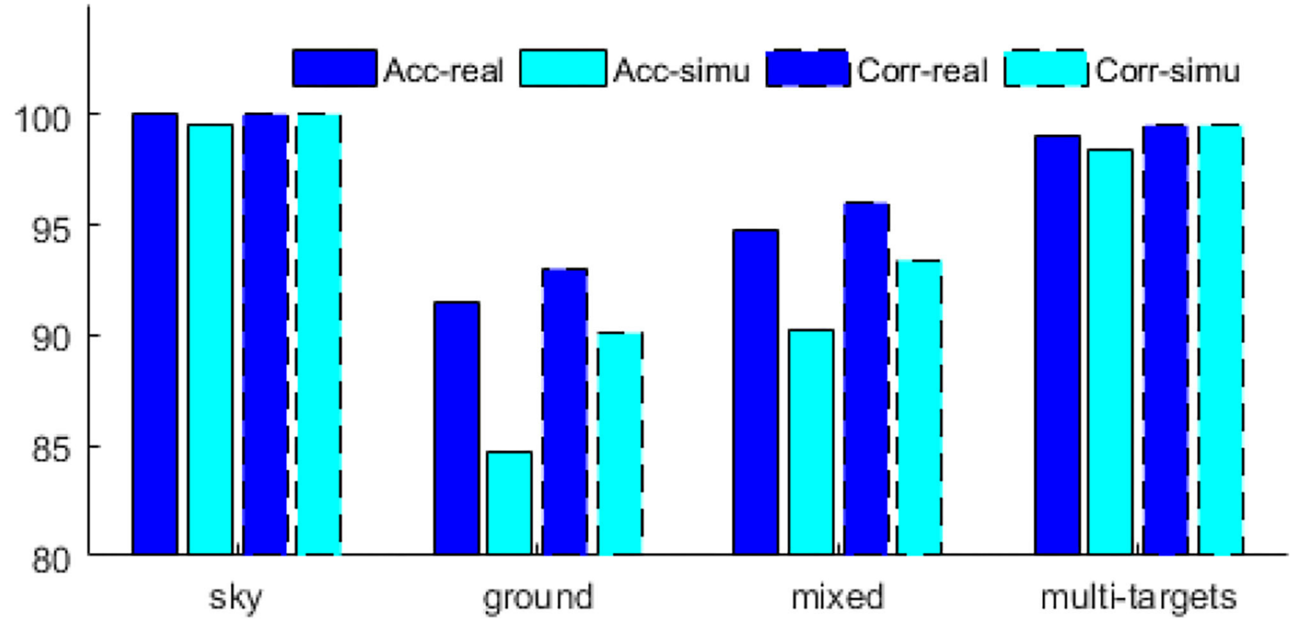


FIGURE 7 | Performance of simulation data and real data on algorithm (Tianjun et al., 2019).

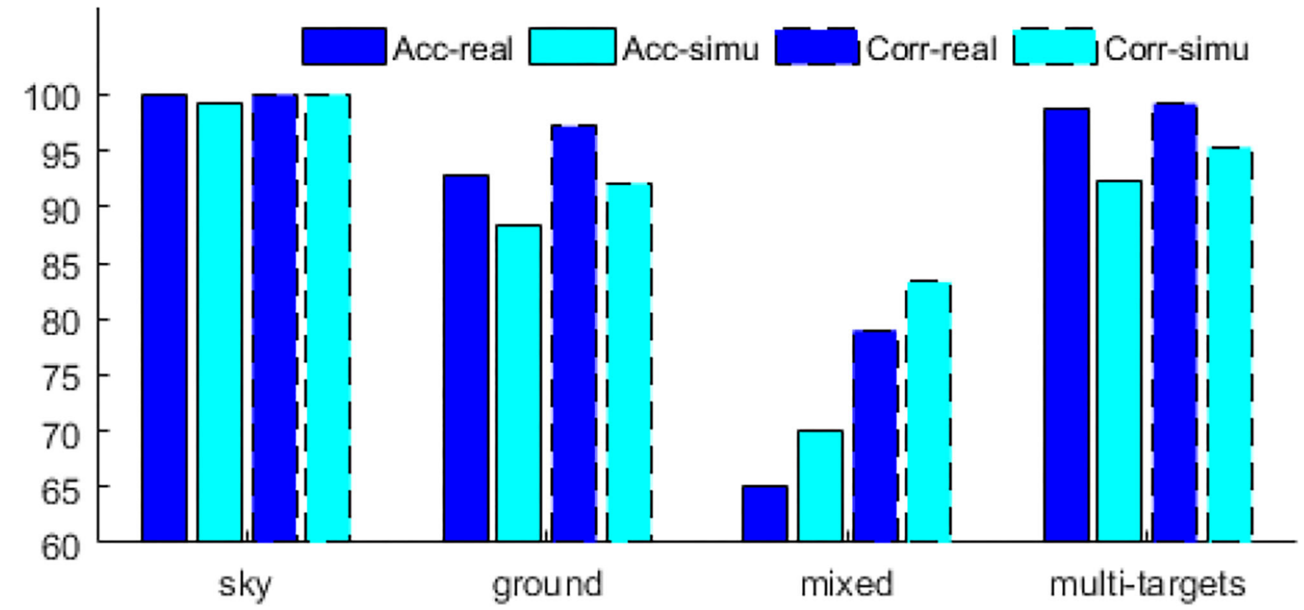


FIGURE 8 | Performance of simulation data and real data on algorithm (Xianbu et al., 2019).

As shown in **Table 2**, the performance of the algorithm (Xianbu et al., 2019) is similar to that of the algorithm (Tianjun et al., 2019) on sky background, ground background, and multi-target scenes; however, the Acc drops to 65% on the mixed background. This may be related to the applicability of the algorithm in different scenarios. Interestingly, the detection results on the simulated data also drop to 70%. Both simulation data and real data show the low performance of the algorithm (Tianjun et al., 2019) in mixed scenarios. Regardless of the scenario, the maximum difference between the accurate detection rates of the simulated and real data is still <7% (in a multi-target scenario, the difference between the accurate detection rates of the real and simulated data is 6.3).

Similarly, the performance of the simulation data generated by our method and the real data in the algorithm (Xianbu et al., 2019) is compared as shown in **Figure 8**. When it performs well on real datasets, the simulation data generated by our method performs also well, such as in sky, ground, and multi-targets scenarios. When its performance on the real dataset is poor, the simulation data generated by our method are also poor, such as in the mixed scene. Therefore, the simulation data generated by our method are consistent with the real data on the performance of algorithm (Xianbu et al., 2019).

CONCLUSION AND FUTURE WORK

Training infrared target detection and tracking models based on deep learning requires a large number of infrared sequence images. The cost of acquisition real infrared target sequence images is high, while conventional simulation methods lack authenticity. This paper proposes a novel infrared data simulation method that combines real infrared images and simulated 3D infrared targets. Firstly, it stitches real infrared images into a panoramic image which is used as background. Then, the infrared characteristics of 3D aircraft are simulated on the tail nozzle, skin, and tail flame, which are used as targets. Finally, the background and targets are fused based on Unity3D, where the aircraft trajectory and attitude can be edited freely to generate rich multi-target infrared data. The experimental results show that the simulated image is not only visually similar to the real infrared image but also consistent with the real infrared image in terms of the performance of target detection algorithms. The method can provide training and testing samples for deep learning models for infrared target detection and tracking.

The infrared simulation of the target in this method has not considered the environmental factors (such as weather, temperature, illumination, etc.) and the sensor error. It is necessary to further improve the precision of target infrared simulation to meet some special application scenarios. This is also the direction of our future work.

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

HZ contributed the main ideas and designed the algorithm. HB contributed the main ideas. SS contribution on experiments and result analysis. All authors contributed to the article and approved the submitted version.

FUNDING

This work was supported by the Key Laboratory Fund of Basic Strengthening Program (JKWATR-210503), Changsha Municipal Natural Science Foundation (kq2202067), and the Basic Science and Technology Research Project of the National Key Laboratory of Science and Technology on Automatic Target Recognition of Scientific Research under Grant (WDZC20205500209).

ACKNOWLEDGMENTS

Professor Fan Hongqi from the ATR Key Laboratory, National University of Defense Technology, provided real infrared data. We thank Shi Tianjun of the Harbin Institute of Technology and Dong Xiaohu of the National University of Defense Technology for their outstanding contributions in the second Sky-Cup National Innovation and Creativity Competition, which provided the test algorithms.

REFERENCES

Alec, R., Luke, M., and Soumith, C. (2016). *Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks*. San Juan: ICLR.

Bhalla, K., Koundal, D., Bhatia, S., Rahmani, M. K., and Tahir, M. (2022). Fusion of infrared and visible images using fuzzy based siamese convolutional network. *Comput. Mater. Continua*. 3, 2022. doi: 10.32604/cmc.2022.021125

Chenyang, L. (2019). *The Infrared Imaging Simulation System Based on Three-Dimensional Scene and its Implementation*. Beijing: The University of Chinese Academy of Sciences.

Chuanyu, Z. (2013). *Infrared Image Fromation for Multiple Targets*. Harbin: Harbin Institute of Technology.

Dai, Y., Wu, Y., Zhou, F., and Barnard, K. (2021). Attentional local contrast networks for infrared small target detection. *IEEE Trans. Geosci. Remote Sens*. 59, 9813–9824. doi: 10.1109/TGRS.2020.3044958

Deng, L., Xu, D., Xu, G., and Zhu, H. (2022). A generalized low-rank double-tensor nuclear norm completion framework for infrared small target detection. *IEEE Trans. Aerosp. Electr. Syst*. 1. doi: 10.1109./TAES.2022.3147437

Guanfeng, Y., Changhao, Z., and Yue, C. (2019). Research on infrared imaging simulantion for enhanced synthetic vision system. *Aeronaut. Comput. Tech*. 49, 100–103.

Haixing, Z., Jianqi, Z., and Wei, Y. (1997). Theoretical calculation of the IR radiation of an aeroplane. *J. Xidian Univ*. 24, 78–82.

Hou, Q., Wang, Z., Tan, F., Zhao, Y., Zheng, H., Zhang, W., et al. (2022). RISTDnet: robust infrared small target detection network. *IEEE Geosci. Remote Sens. Lett*. 19, 1–5. doi: 10.1109/LGRS.2021.3050828

Hui, B., Song, Z., Fan, H., Zhong, P., Hu, W., Zhang, X., et al. (2020). A dataset for infrared detection and tracking of dim-small aircraft targets under ground/air background. *Chinese Sci. Data*. 5, 291–302. doi: 10.11922/csdata.2019.0074.zh

Junhong, L., Ping, Z., Xiaowei, W., and Shize, H. (2020). Infrared small-target detection algorithms: a survey. *J. Image Graph*. 25, 1739–1753. doi: 10.11834/jig.190574

Junyan, Z., Taesung, P., and Isola, P. (2017). *Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks*. Venice: ICCV.

Mirza, M., and Osindero, S. (2014). Conditional generative adversarial nets. *Comput. Sci*. 2672–80. Avaialble online at: https://arxiv.org/abs/1411.1784

Rani, S., Singh, B. K., Koundal, D., and Athavale, V. A. (2022). Localization of stroke lesion in MRI images using object detection techniques: a comprehensive review. *Neurosci Inform*. 2, 100070. doi: 10.1016/j.neuri.2022.100070

Shi, Q., Gao, Y., Zhang, X., Li, Z., Du, J., Shi, R., et al. (2021). Cryogenic background infrared scene generation method based on a light-driven blackbody micro cavity array. *Infrared Phys. Technol*. 117, 103841. doi: 10.1016/j.infrared.2021.103841

Shuwei, T., and Bo, X. (2018). Research on infrared scene built by computer. *Electro-Optic. Technol. Appl*. 33, 58–61.

Tianjun, S., Guangzhen, B., Fuhai, W., Chaofei, L., and Jinnan, G. (2019). An infrared small target detection and tracking algorithm applying for multiple scenarios. *Aero Weaponry*. 26, 35–42. doi: 10.12132/ISSN.1673-5048.2019.0220

Xia, W., Hao, W., and Chao, X. (2015). Overview on development of infrared scene simulation. *Infrared Technol*. 7, 537–43. doi: 10.11846/j.issn.1001_8891.201507001

Xianbu, D., Ruigang, F., Yinghui, G., and Bio, L. (2019). Detecting and tracking of small infrared targets adaptively in complex background. *Aero Weaponry*. 26, 22–28. doi: 10.12132/ISSN.1673-5048.2019.0233

Yi, H. (2020). *RGB-to-NIR Image Translation Using Generative Adversarial Network*. Wuhan: Central Normal China University.

Yi, Y., Changbin, X., and Yuying, M. (2019). A review of infrared dim small target detection algorithms with low SNR. *Laser Infrared*. 49, 643–649. doi: 10.3969/j.issn.1001-5078.2019.06.001

Yongjie, Z., Zhenya, X., and Jianxun, L. (2020). Study on simulation model of aircraft infrared hyperspectral image. *Aero Weaponry*. 27, 91–96. doi: 10.12132/ISSN.1673-5048.2019.0082

Yu, C. (2012). *Design of Infrared Decoy HIL Simulation System Based on Finite Element Module*. Harbin: Harbin Institute of Technology.

Yunjey, C., Youngjung, U., Jaejun, Y., and Ha, J.-W. (2020). "StarGAN v2: diverse image synthesis for multiple domains," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Seattle).

Zhang, R., Mu, C., Yang, Y., and Xu, L. (2018). Research on simulated infrared image utility evaluation using deep representation. *J. Electr. Imag*. 27, 013012. doi: 10.1117/1.JEI.27.1.013012

Zhijian, H., Bingwei, H., and Shujin, S. (2021). *An Automatic Image Stitching Method for Infrared Image Series*. Xi'an: ICCAIS.

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.

frontiers | Frontiers in Computational Neuroscience

TYPE Original Research
PUBLISHED 25 August 2022
DOI 10.3389/fncom.2022.986591

Check for updates

OPEN ACCESS

EDITED BY
Deepika Koundal,
University of Petroleum and Energy Studies, India

REVIEWED BY
Arvind Dhaka,
Manipal University Jaipur, India
Neelam Goel,
Panjab University, India
Rajit Nair,
VIT Bhopal University, India

*CORRESPONDENCE
Fei Fan
fanfansofia@hotmail.com
Fei Han
hf4383@126.com

†These authors have contributed equally to this work

RECEIVED 05 July 2022
ACCEPTED 29 July 2022
PUBLISHED 25 August 2022

CITATION
Fan F, Bian Z, Zhang X, Wu H, Wang S, Zhang S, Wang Q and Han F (2022) Big data analytics frameworks for the influence of gut microbiota on the development of tic disorder. *Front. Comput. Neurosci.* 16:986591. doi: 10.3389/fncom.2022.986591

Big data analytics frameworks for the influence of gut microbiota on the development of tic disorder

Fei Fan[1]*†, Zhaoxiang Bian[2], Xuan Zhang[2], Hongwei Wu[3], Simeng Wang[1], Si Zhang[1], Qiong Wang[4] and Fei Han[1]*†

[1]Department of Pediatrics, Guang'anmen Hospital, China Academy of Chinese Medical Sciences, Beijing, China, [2]Chinese EQUATOR Centre, Hong Kong Chinese Medicine Clinical Study Centre, Chinese Clinical Trial Registry (Hong Kong), School of Chinese Medicine, Hong Kong Baptist University, Kowloon, Hong Kong SAR, China, [3]Institute of Chinese Materia Medica, China Academy of Chinese Medical Sciences, Beijing, China, [4]Clinical Medical School, Beijing University of Chinese Medicine, Beijing, China

The association between gut microbiota and psychiatric disorders has received increasing research attention. Meanwhile, big data analysis has been utilized in many filed including business, human healthcare analysis, *etc*. The primary objective of this article was to provide insights into Big Data Analytics (BDA) to clarify the association between gut microbiota and TD (Tic disorder). Specifically, we investigated the recent studies related to gut microbiota composition differences in patients with TD compared to health people. We searched on PubMed and Embase (Ovid) databases for relevant published articles until June 15, 2021. A total of 78 TD and 62 health control stool samples were examined. Case-control design was applied in all the studies. No consensus was evident in α-diversity and β-diversity. The abundance of phyla *Bacteroidetes* and *Firmicutes* was predominant at the taxa level. Gut microbiota taxonomic differences were found between TD cases and controls, though inconsistently across studies. Further studies are needed to reveal the underlying pathophysiology of TD and correlation between TD and gut microbiota composition.

KEYWORDS

tic disorder, gut microbiota, data analysis, bacteroidetes, firmicutes

Introduction

Tic disorder (TD) is characterized by sudden, recurrent, non-rhythmic movement, or phonic tic with childhood onset, ongoing throughout adulthood (Plessen, 2013). According to the Diagnostic and Statistical Manual of Mental Disorders (DSM)-5 (American Psychiatric Association., 2013), TD includes Tourette syndrome (TS), chronic motor or vocal tic disorder (CTD), provisional tic disorder (PTD), other specified tonic disorders, and unspecified tic disorders. TD is the most common movement disorder in children, but the reported prevalence of TD varies considerably (Cubo et al., 2011; Yang et al., 2016; Mohammadi et al., 2021) because a significant proportion of patients do not recognize their tics (Ueda and Black, 2021). Children with TD may experience subjective discomfort, sustained social problems, sleep difficulties, and many emotional problems (Conte et al., 2020; Fernández de la Cruz and Mataix-Cols, 2020; Isaacs et al., 2021). TD is commonly associated with obsessive-compulsive disorder (OCD), attention-deficit/hyperactivity disorder (ADHD), and anxiety disorders (Hirschtritt et al., 2015; Eapen et al., 2016). Thus, research to understand the development of TD is receiving increasing attention lately. TD occurs through interactions including but not limited to genetic (Cao et al., 2021), neurobiochemical (Kanaan et al., 2017), inflammation-related (Martino et al., 2021), immunological (Lamothe et al., 2021), and environmental factors (Storch et al., 2017). However, its pathophysiology remains unknown.

Gut microbiota is a variety of microorganisms in the gastrointestinal tract, normally more than 1,000 bacterial species and with more than nine million genes. Gut microbiota is extremely diverse and changeable with the majority of bacteria from the four dominant phyla including Bacteroides, Firmicutes, Proteobacteria, and Actinobacteria, which constitutes more than 98% of all of the human gut microbes. Gut microbiota constitute a very important part in both of the health maintenance and the disease pathogenesis process. It is a known fact that a diverse and stable and gut microbiota is essential to for various normal physiologic functions such as immunology regulation, prevention of bacterial infection, energy harvest and metabolism, and so on. Meanwhile, the gut microbiota is associated with disease is often characterized by a decrease or increase in species richness and proliferation of some specific pathogens. The gut microbiota plays an important role in the extensive reciprocal connections between the gastrointestinal system and human brain, forming the microbiome-gut-brain axis (Cryan et al., 2020). The association between gut microbiota and psychiatric disorders has received increasing research attention (Morais et al., 2021). Over the past decade, many studies have revealed that the gut microbiota is directly involved in the production of various neurotransmitters, such as gamma-aminobutyric acid (GABA), serotonin (5-HT), glutamate, and dopamine (DA) (Bull-Larsen and Mohajeri, 2019; Altaib et al., 2021; Bhatt et al., 2022), which are closely associated with a number of psychiatric disorders, including TD (Kanaan et al., 2017), ADHD (Turna et al., 2020), OCD (Simpson et al., 2021), and anxiety (Ridaura and Belkaid, 2015).

Gastrointestinal symptoms are not common in TD patients (Fernández de la Cruz and Mataix-Cols, 2020). However, studies show that TD patients have a higher risk of metabolic or cardiovascular disease than the general population, which also plays an important role in the pathogenesis and course of TD, suggesting a relationship between TD and microbiota (Brander et al., 2019; Fernández de la Cruz and Mataix-Cols, 2020; Tomasova et al., 2021). Most TD patients have sleep disorder (Hibberd et al., 2020; Isomura et al., 2022) and are sensitive to psychological stress (Tilling and Cavanna, 2020). Meanwhile, gut microbiota can get disrupted under psychological stress (Madison and Kiecolt-Glaser, 2019; McGuinness et al., 2022) and is correlated with the sleep behavior (Qi et al., 2022). Recent studies have shown that the gut microbiota plays an indispensable role in regulating microglial maturation and function (Bairamian et al., 2022). Circulation of microbe-derived neurotransmitters, including acetylcholine, GABA, and 5-HT, can regulate microglial activation (Fung et al., 2017). Interestingly, abnormalities in microglial activation, development, and function in the basal ganglia of TD patients are also widely recognized (Frick and Pittenger, 2016). Some studies have demonstrated that fecal microbiota transplantation (FMT) effectively ameliorates TD symptoms (Zhao et al., 2017, 2020). Animal studies have also shown that microbiota have the potential to improve tic syndromes (Liao et al., 2019). Despite evidence pointing to a connection between gut microbiota and TD, the nature of this relationship remains unclear. Better understanding of which microbiome is associated with TD and its pathophysiological effects will enable researchers to provide new therapeutic and diagnostic avenues of TD in the future.

Thus, the primary objective of this review was to investigate and compare the recent studies relating to gut microbiota composition differences in patients with TD.

Thus, the primary objective of this work is to summarize, investigate and compare recent studies on gut microbiota composition differences in patients with TD.

Materials and methods

This work has been uploaded and accepted into PROSPERO under the identification number CRD42021265088, performed in accordance with PRISMA guidelines (Page et al., 2021).

Information sources

The databases PubMed and Embase (Ovid) were searched for human studies in English up until June 15, 2021,

Abbreviations: TD, Tic disorder; DSM, Diagnostic and Statistical Manual of Mental Disorders; TS, Tourette syndrome; CTD, chronic motor or vocal tic disorder; PTD, provisional tic disorder; OCD, obsessive-compulsive disorder; ADHD, attention-deficit/hyperactivity disorder; GABA, gamma-aminobutyric acid; 5-HT, serotonin; DA, glutamate and dopamine; FMT, fecal microbiota transplantation; HC, healthy control; NOS, Newcastle-Ottawa Scale; GSI, gastrointestinal severity index; DRA, dopamine receptor antagonists; YGTSS, Yale Global Tic Severity Scale; ASD, autism spectrum disorders; SCFA, short-chain fatty acid.

using the following search strategies (for PubMed): ["tic disorder"(Text Word) OR "tic disorders"(Text Word) OR "tourette syndrome"(Text Word) OR "gilles de la tourette"(Text Word) OR "pediatric autoimmune neuropsychiatric disorders associated with streptococcal infections"(Text Word)] AND ["gut microbiota*"(Text Word) OR "gut microbiome*"(Text Word) OR "intestinal microbiota"(Text Word) OR "intestinal microbiome"(Text Word) OR "gastrointestinal microbiota"(Text Word) OR "gastrointestinal microbiome"(Text Word)] (**Supplementary Material 1**). Gray literature was included if fulfill the inclusion criteria.

Inclusion and exclusion criteria

Inclusion criteria:

- Original observational studies performed on TD patients diagnosed according to DSM-5 (or IV) or ICD-11 (or 10).
- Detection of gut microbiota composition through high-throughput sequencing techniques.
- Inclusion of a healthy control (HC) group.
- Published in English.

Exclusion criteria:

- Animal studies.

Study selection

Studies were imported into the Mendeley reference manager[1] to remove duplicates using its automatic function. Files generated from PubMed and Embase were reviewed and selected using the website: http://syrf.org.uk independently by authors FF and SW based on titles and abstracts, and later the included studies were whole-text reviewed manually. Studies inconsistently agreed upon both reviewers were resolved by a third author, FH.

Outcome measures

Data were extracted from the TD and HC groups using a Microsoft Excel file (Supporting Information 2), focusing on the demographics, microbiota analysis methodology, α- and β-diversity, clinical information, and other relevant findings. A meta-analysis was not performed in the present study.

Risk of bias assessment

The Newcastle-Ottawa Scale (NOS) was used to evaluate the risk of bias in case–control studies. The NOS scale contains three categories comprising total of eight items: selection (four items), comparability (one item), and exposure (three items). Quality score with a maximum of ten was obtained using a rating algorithm: 0–5 (poor), 6–7 (moderate), and 8–10 (high).

Results

Study selection

Study selection was conducted using the PRISMA guidelines. Using keywords, we found 41 studies from the literature search. After the automatic removal of duplicates, 35 unique articles were identified. After screening the titles and abstracts of these articles, six were assigned to a full-text assessment, out of which three unqualified articles were removed (one did not focus on TD and two did not have original gut microbiota statistics). Finally, we focused on three articles for further analysis (Lee and Wong, 2018; Zhao et al., 2020; Xi et al., 2021; **Figure 1**).

Assessment of study quality/bias

Estimates of bias were obtained for the three studies that compared patients with TD with HCs using the NOS, as indicated in **Table 1**. One study (Xi et al., 2021) received a score of six (moderate) because the interview was not blinded to the status. The second study received a score of four (low) (Zhao et al., 2020) due to the HC being only one child and thus the resulting potential biases, and the last study received three (low) (Lee and Wong, 2018) due to inadequate description of the study.

Characteristics of studies

Demographic data of the three studies are shown in **Table 2**. Two out of three studies were conducted in Beijing, including a total of 54 patients diagnosed with TD and 51 HCs (Zhao et al., 2020; Xi et al., 2021). The other study was conducted in Taiwan, which included 24 TD patients and 11 HCs (Lee and Wong, 2018). The total sample size of the selected studies ranged from 6 to 99, with the number of cases ranging from 5 to 49, and the number of controls ranging from 1 to 50. With these three studies combined, a total of 78 cases and 62 controls were investigated and included TD patients and

[1] https://www.mendeley.com/

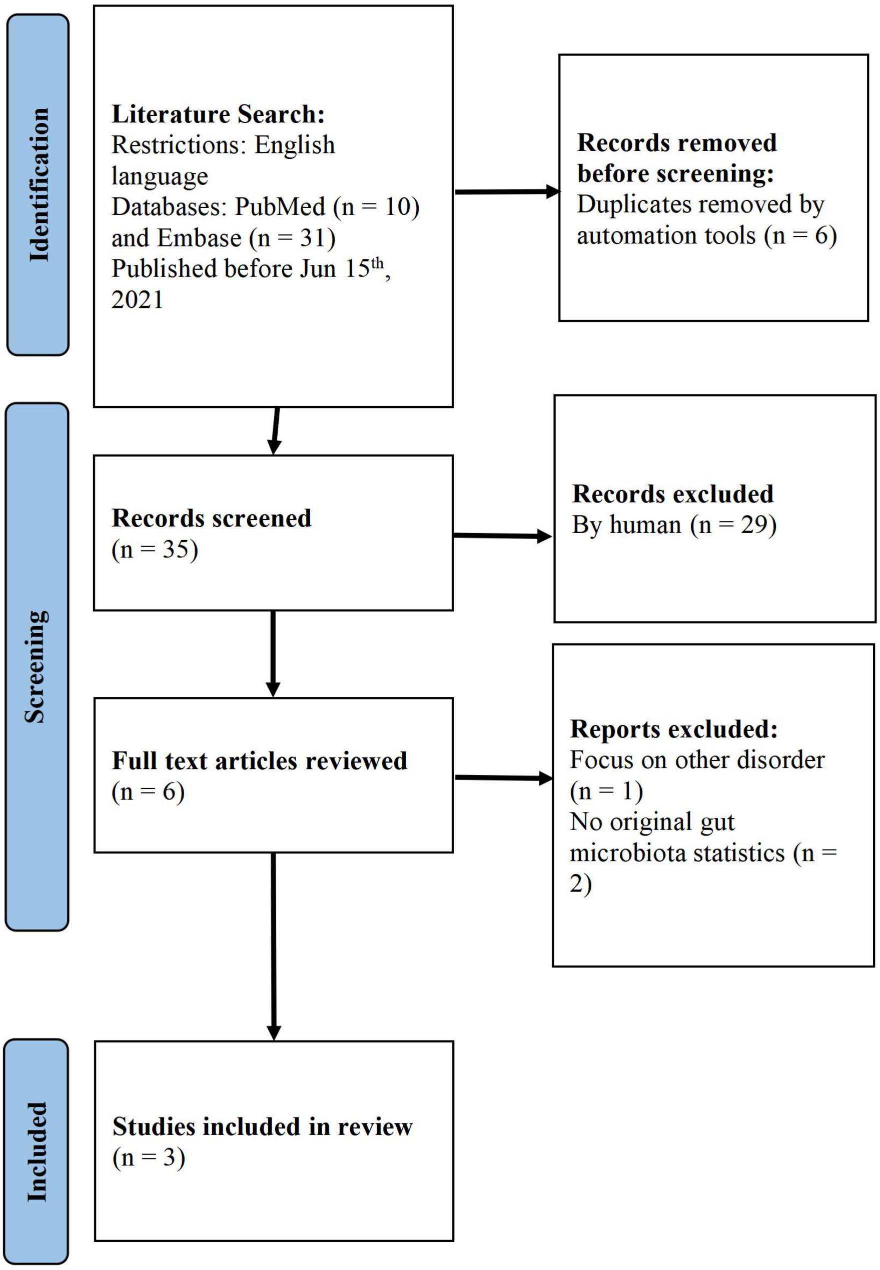

Identification
Literature Search:
Restrictions: English language
Databases: PubMed (n = 10) and Embase (n = 31)
Published before Jun 15th, 2021
Records removed before screening:
Duplicates removed by automation tools (n = 6)
Screening
Records screened
(n = 35)
Records excluded
By human (n = 29)
Full text articles reviewed
(n = 6)
Reports excluded:
Focus on other disorder (n = 1)
No original gut microbiota statistics (n = 2)
Included
Studies included in review
(n = 3)

FIGURE 1
PRISMA flowchart of the screening process.

TABLE 1 Quality assessment of included studies based on Newcastle-Ottawa scale (NOS).

No.	Study	Year	Selection	Comparability	Exposure	Total
1	Lee and Wong (2018)	2018	1	1	1	3
2	Zhao et al. (2020)	2020	2	1	1	4
3	Xi et al. (2021)	2021	3	2	1	6

TABLE 2 Demographic data of the studies.[a]

No.	Study	Year	City	Participants		Age mean (SD)		Male (m/f)		BMI mean (SD)	
				TD	HC	TD	HC	TD	HC	TD	HC
1	Lee and Wong (2018)	2018	Taiwan	*n* = 24	*n* = 11	NA	NA	NA	NA	NA	NA
2	Zhao et al. (2020)	2020	Beijing	*n* = 5	*n* = 1	8	14	5/0	1/0	18.0	NA
3	Xi et al. (2021)	2021	Beijing	*n* = 49	*n* = 50	8.84 (2.35)	8.78 (2.26)	38/11	39/11	18.28 (2.99)	17.22 (2.66)

[a]Data are presented as mean (standard deviation, SD) or number of participants. m, male; f, female; TD, tic disorder; HC, healthy controls; BMI, body mass index; DSM-5, Diagnostic and Statistical Manual of Mental Disorders-5.

TABLE 3 Clinical information of patients with tic disorder (TD) and healthy controls (HCs).[a]

No.	Study	Diagnoses (n)	Diagnostic instrument	Disease duration (SD), year	YGTSS scores (SD)	Comorbidities (n)	GSI (SD)	Gastrointestinal disturbances (%)	Medication (n)
1	Lee and Wong (2018)	TS: severe tics (14); mild tics (10)	N/A	severe tics: 4.5 (2.33) mild tics: 2.25 (2.5)	TTS scores: severe tics, 27.4 (7.5); mild tics, 14.8 (4.1)	N/A	N/A	N/A	N/A
2	Zhao et al. (2020)	TS	DSM-5	1.5–4	YGTSS-TTS > 13	ADHD (3), variant asthma (1)	N/A	N/A	Tiapride (3); aripiprazole (2); trihexyphenidyl (2); risperidone (1)
3	Xi et al. (2021)	TD: TS (23); PTD (17); CTD (9)	DSM-5	2.11 (1.92)	36.71 (16.73)	N/A	2.31 (1.86)	mild constipation, 26.53; abdominal pain, 28.57	DRAs (12); topiramate (1); valproate (1); treatment-naive (35)

[a]SD, standard deviation; YGTSS, Yale Global Tic Severity Scale; YGTSS-TTS, Yale Global Tic Severity Scale Total Tic Scale (combined motor tic and vocal tic score); GSI, Gastrointestinal Severity Index; TD, tic disorder; PTD, provisional tic disorder; CTD, chronic motor or vocal tic disorder; TS, Tourette syndrome; DRA, dopamine receptor antagonist.

TABLE 4 Microbiota analysis methodology and diversity results.[a]

No.	Study	Samples	Stool storage	Genetic quantification	Alpha diversity	Beta diversity
1	Lee and Wong (2018)	Stool	N/A	N/A	N/A	N/A
2	Zhao et al. (2020)	Stool	−80°C	Shotgun metagenomic sequencing	A reduced OTU number	A different cluster in PCoA
3	Xi et al. (2021)	Stool	−80°C	Shotgun metagenomic sequencing	No significant difference[b]	No significant difference[b]

[a]OTU, operational taxonomic unit; TD, tic disorder; HC, healthy controls; PCoA, principal coordinate analysis. [b]Between treatment-naïve TD patients and HCs.

HCs younger than 18 years. Moreover, the study design of two studies was cross-sectional and compared gut microbiota in TD patients with that in a HC group (Lee and Wong, 2018; Xi et al., 2021).

In two studies (Zhao et al., 2020; Xi et al., 2021), patients were assessed according to the DSM-5 criteria. We found that only one study (Xi et al., 2021) mentioned gastrointestinal disturbances (mild constipation and abdominal pain), and provided gastrointestinal severity index (GSI) scores. Two studies (Zhao et al., 2020; Xi et al., 2021) included cases that received dopamine receptor antagonists (DRA) and other medications, while the rest (Zhao et al., 2020; Xi et al., 2021) did not mention these criteria. In addition, only one study (Xi et al., 2021) excluded antibiotics/probiotics taken within 4 weeks prior to sample collection and any infective or other severe disease conditions that may influence the gut microbiota. The ability to compare or interpretation of individual studies is limited by the extensive variability of different aspects of the studies (**Table 3**).

Microbiota analysis

There were some differences in the sample analysis with respect to the diversity of results in the included studies, as shown in **Table 4**. Two out of three studies (Zhao et al., 2020; Xi et al., 2021) used shotgun metagenomic sequencing and analyzed the α-diversity

TABLE 5 Different microbiota findings in tic disorder (TD) patients.[a]

No.	Study	Gut microbiota profiles	Other findings
1	Lee and Wong (2018)	Family: ↓:*Prevotellaceae* Genus: ↑:*Ruminococcus* ↓:*Prevotella*[b] Species: ↓:*Clostridium bartlettii*, *Prevotella copri*, and *Subdoligranulum variabile*	*Prevotella* was negatively correlated with the severity of tics.
2	Zhao et al. (2020)	Genus: ↓:*Bifidobacterium*, *Catenibacterium*, *Collinsella*, and *Dorea* Species: ↑:*Bacteroides vulgatus* ↓:*Allisonella histaminiformans*, *Bacteroides coprocola*, *Catenibacterium mitsuokai*, *Dialister succinatiphilus*, *Holdemanella biformis*, and *Roseburia faecis*	
3	Xi et al. (2021)	Species: ↑:*Bacteroides plebeius*, *Ruminococcus lactaris* ↓:*Prevotella stercorea*, *Streptococcus lutetiensis*	*Bacteroides eggerthii*, *Bacteroides dorei*, and *Bacteroides thetaiotaomicron* positive correlations with the YGTSS scores.

[a]TD, tic disorder; YGTSS, Yale Global Tic Severity Scale. [b]Severe TS samples ($n = 14$).

and β-diversity of their samples without mentioning the exact index.

Microbiota findings

The gut microbiota of TD patients was compared to that of HCs to assess changes in different individuals' bacterial abundances. The findings are presented in **Table 5** and a more comprehensive listing in **Supplementary Material 2**. A study by Lee and Wong (2018) stated that the *Prevotellaceae* family and *Prevotella* genus were decreased and *Ruminococcus* genus was increased in TD patients. In the study by Zhao et al. (2020), *Bifidobacterium*, *Catenibacterium*, *Collinsella*, and *Dorea* genera were decreased in TD patients. In another study by Xi et al. (2021), the species *Bacteroides plebeius*, *Ruminococcus lactaris*, *Prevotella* stercorea, and *Streptococcus lutetiensis* were decreased in TD patients. Moreover, Xi et al. (2021) found that *Bacteroides eggerthii*, *Bacteroides dorei*, and *Bacteroides thetaiotaomicron* species were positively correlated with the Yale Global Tic Severity Scale (YGTSS) scores (as with the severity of tics). Genus *Prevotella* was negatively correlated with the severity of tics in another study (Lee and Wong, 2018).

Discussion

Due to the limited treatment methods for tic disorder at present, and the effectiveness of some treatment methods is not so effective, or the effectiveness is limited, so the exploration of its pathogenesis is particularly important, which will guide the better diagnosis and treatment of tic disorder in the future. In recent 10 years, in addition to finding better drug treatments, there are more and more studies on the influences of both hereditary and environmental factors on the occurrence and development of tic disorders (TD). Understanding the microbiome associated with TD has the potential to further research on TD pathophysiology and provide individual treatment options. Although many microbiome infections appear to be correlated with TD (Müller et al., 2004; Mell et al., 2005; Prasad, 2021), to our knowledge, so far no study has revealed the fine-grain pathophysiology. In this work, we attempt to assess whether individuals with TD had a distinct gut microbiota composition compared to HCs. Notably, all the studies identified that the gut microbiota of individuals with TD were distinguishable from that of HCs, although the results of each study varied. The fine structure of the gut microbiota varies greatly among cases (Caporaso et al., 2011).

Main findings

Overall, no consensus regarding α-diversity and β-diversity was found. Xi et al. (2021) found no significant differences in diversity. However, Zhao et al. (2020) found some possible differences, but this was not described in detail. At the taxa level, the abundance of phyla *Bacteroidetes* and *Firmicutes* was the predominant difference between TD patients and HCs. One family, one genus, and three species of *Bacteroidetes* were found to be decreased, while two species were found to be increased in patients with TD. Two genera and eight species of *Firmicutes* were found to be decreased, while one genus and one species were found to be increased in TD patients. A study by Lee and Wong (2018) found that the proportion of genus *Prevotella* was negatively correlated with the severity of tics. Meanwhile, Xi et al. (2021) found that the species *Bacteroides eggerthii*, *Bacteroides dorei*, and *Bacteroides thetaiotaomicron* were positively correlated with severity. *Bacteroidetes* and *Firmicutes* phyla are also the most dominant gut microbiota in normal people (Jandhyala et al., 2015) and are correlated with inflammatory conditions such as inflammatory bowel disease (Stojanov et al., 2020). The establishment of the gut microbiota has been shown to be a progressive process, and the ratio of *Firmicutes* to *Bacteroidetes* is significantly correlated with human age (Ley et al., 2006). The *Firmicutes*/*Bacteroidetes* ratio increases from birth to adulthood and further changes with age (Mariat et al., 2009). Reports have shown that changes in the ratio of *Firmicutes*/*Bacteroidetes* are significant factors affecting childhood diseases childhood obesity (Indiani CMDSP et al., 2018), autism spectrum disorders (ASD) (Strati et al., 2017), and others (Quagliariello et al., 2016; Valentini et al., 2020). TD typically begins in childhood and often improves in early adulthood, but the reason remains unknown (Hartmann et al., 2020). Current studies link age correlation with TD and the ratio of *Firmicutes* to *Bacteroidetes*, although the result is still not definitive. Further studies should focus on this ratio to reveal more comparable results.

Bacteria with increased abundance were found in the gut microbiota of patients with various inflammatory diseases (Zhang et al., 2015; Mondot et al., 2016), suggesting a potential pro-inflammatory effect. Moreover, other studies suggest that decreased abundance of genus *Bifidobacterium* (Plaza-Díaz et al., 2017) and species *Holdemanella biformis* (Pujo et al., 2021), which also decreased in this study, had an anti-inflammatory effect. Zhao et al. (2020) analyzed a wide range of inflammatory markers associated with the gut microbiota. Several studies have confirmed this mechanism, and reported elevated levels of pro-inflammatory cytokines [including IL-12 and TNF-α (Leckman et al., 2005)] and decreased levels of anti-inflammatory cytokines (including IL-13) in TD patients (Parker-Athill et al., 2015). In addition, the decreased levels of *Prevotella copri*, *Prevotella stercorea*, and *Roseburia faecis* also determine short-chain fatty acid (SCFA) levels (Louis et al., 2010; Liu et al., 2021). SCFAs play an anti-inflammatory and antimicrobial role in various interactions between gut microbiome and host metabolism (Tan et al., 2014; Sanna et al., 2019). Additionally, microbial metabolites can affect central neurotransmitters by activating afferent nerve fibers. SCFAs can stimulate the release of central neurotransmitters (including 5-HT) in the intestine (Yano et al., 2015). *Bifidobacterium* is a key member of the human gut microbiota affecting GABA production (Barrett et al., 2012). High levels of *Ruminococcus lactaris* (Dan et al., 2020) and low levels of the genera *Collinsella* and *Dorea* (Strati et al., 2017) have also been found in ASD patients with constipation symptoms, further explaining the potential role and related symptoms of *Ruminococcus lactaris* in the pathological mechanism of neurodevelopmental disorders.

Treatment and diet

Although there have studies that attempted to utilize FMT (Zhao et al., 2017, 2020) in the treatment of TS (the most severe type of TD), the results have been limited. Zhao et al. (2020) found that FMT might reduce fecal lipopolysaccharide levels in TD patients and increase *Bacteroides coprocola* and *Dialister succinatiphilus* abundance and decrease *Bacteroides vulgatus* abundance. In the study by Xi et al. (2021), DRA-treated patients showed enrichment of *Bacteroides dorei*, *Escherichia coli*, *Bacteroides caccae*, and *Ruminococcus gnavus*. These enterotypes also seem to have some functional relevance to diet. The genus *Bacteroides* is associated with high-fat or high-protein diets and *Prevotella* with high-carbohydrate diets (Wu and Hui, 2011).

Risk of bias

Of the three studies, Xi et al. (2021) displayed age and BMI information as mean and SD, and Zhao et al. (2020) included mean age and BMI. It has been reported that age and BMI are related to the composition of the gut microbiota (Haro et al., 2016; Odamaki et al., 2016). The study by Zhao et al. (2020) was the only study with all-male cases. This actually made the samples more homogeneous because gut microbiota composition has also been shown to differ according to sex (Haro et al., 2016). Lee and Wong (2018) study had scarce demographic data. Although all included studies reported YGTSS scores, there was a lack of consistent diagnostic criteria for the case definition. The reliability and accuracy of microbiome studies depend largely on the molecular biology techniques used, and differences in databases can affect the results of microbiome data (Haro et al., 2016). The studies in this review lack such information, and it is recommended that all studies use uniform classification criteria and databases to obtain more comparable results.

Limitation

However, there are several limitations that should be acknowledged. First, this review included only three studies and a small sample size; thus, more TD patients enrolled from different studies are needed to make our results more reliable and reasonable. Second, *in vitro* and *in vivo* experiments were not conducted in the included studies. Finally, differences in the study population, including age, sex, height, weight, genetics, emotion, stress, and environmental factors, were not analyzed in the included studies.

Conclusion

Emerging scientific data support the significant role of the gut microbiota in the regulation of the central nervous system. The results of the included studies show that the gut microbiota in children with TD is significantly different from healthy children. There is variability in microbial diversity as well as the abundance of taxa in patients with TD, which suggesting the complicity of the phenomenon. Furthermore, pro-inflammatory cytokines and central neurotransmitters may both play an important role in the pathophysiology of the gut microbiota in TD.

Data availability statement

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

Author contributions

FF and SW contributed to the study conception. HW designed the project. ZB and XZ collected the data and performed the formal analysis of finding. QW and SZ organized and integrated the data. FF drafted the manuscript. FF and FH critically reviewed the manuscript. ZB contributed to the visualization. FH acquired the funding source. All authors have read and agreed to the published version of the manuscript.

Funding

This study was funded through a grant from the Fundamental Research Funds for the Central Public Welfare Research Institutes, Beijing, China (grant numbers: ZZ13-024-5 and ZZ15-XY-PT-03).

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.2022.986591/full#supplementary-material

SUPPLEMENTARY MATERIAL 1
Methodology.

SUPPLEMENTARY MATERIAL 2
Microbiota analysis.

SUPPLEMENTARY MATERIAL 3
Study quality of case-control studies.

References

Altaib, H., Nakamura, K., Abe, M., Badr, Y., Yanase, E., Nomura, I., et al. (2021). Differences in the Concentration of the Fecal Neurotransmitters GABA and Glutamate Are Associated with Microbial Composition among Healthy Human Subjects. *Microorganisms* 9:378. doi: 10.3390/microorganisms9020378

American Psychiatric Association. (2013). *Diagnostic and Statistical Manual of Mental Disorders (DSM-5)*, 5th Edn. Washington, DC: American Psychiatric Association. doi: 10.1176/appi.books.9780890425596

Bairamian, D., Sha, S., Rolhion, N., Sokol, H., Dorothée, G., Lemere, C. A., et al. (2022). Microbiota in neuroinflammation and synaptic dysfunction: a focus on Alzheimer's disease. *Mol. Neurodegener.* 17:19. doi: 10.1186/s13024-022-00522-2

Barrett, E., Ross, R. P., O'Toole, P. W., Fitzgerald, G. F., and Stanton, C. (2012). γ-Aminobutyric acid production by culturable bacteria from the human intestine. *J. Appl. Microbiol.* 113, 411–417. doi: 10.1111/j.1365-2672.2012.05344.x

Bhatt, S., Kanoujia, J., Mohanalakshmi, S., Patil, C. R., Gupta, G., Chellappan, D. K., et al. (2022). Role of Brain-Gut-Microbiota Axis in Depression: Emerging Therapeutic Avenues. *CNS Neurol. Disord. Drug Targets* [Epub Online ahead of print]. doi: 10.2174/1871527321666220329140804

Brander, G., Isomura, K., Chang, Z., Kuja-Halkola, R., Almqvist, C., Larsson, H., et al. (2019). Association of Tourette Syndrome and Chronic Tic Disorder With Metabolic and Cardiovascular Disorders. *JAMA Neurol.* 76, 454–461. doi: 10.1001/jamaneurol.2018.4279

Bull-Larsen, S., and Mohajeri, M. H. (2019). The Potential Influence of the Bacterial Microbiome on the Development and Progression of ADHD. *Nutrients* 11:2805. doi: 10.3390/nu11112805

Cao, X., Zhang, Y., Abdulkadir, M., Deng, L., Fernandez, T. V., Garcia-Delgar, B., et al. (2021). Whole-exome sequencing identifies genes associated with Tourette's disorder in multiplex families. *Mol. Psychiatry* 26, 6937–6951. doi: 10.1038/s41380-021-01094-1

Caporaso, J. G., Lauber, C. L., Costello, E. K., BergLyons, D., Gonzalez, A., Stombaugh, J., et al. (2011). Moving pictures of the human microbiome. *Genome Biol.* 12:R50.

doi: 10.1186/gb-2011-12-5-r50

Conte, G., Valente, F., Fioriello, F., and Cardona, F. (2020). Rage attacks in Tourette syndrome and chronic tic disorder: a systematic review. *Neurosci. Biobehav. Rev.* 119, 21–36. doi: 10.1016/j.neubiorev.2020.09.019

Cryan, J. F., O'Riordan, K. J., Sandhu, K., Peterson, V., and Dinan, T. G. (2020). The gut microbiome in neurological disorders. *Lancet Neurol.* 19, 179–194. doi: 10.1016/S1474-4422(19)30356-4

Cubo, E., Gabriel, Y., Galán, J. M., Villaverde, V. A., Velasco, S. S., Benito, V. D., et al. (2011). Prevalence of tics in schoolchildren in central Spain: a population-based study. *Pediatric neurology* 45, 100–108. doi: 10.1016/j.pediatrneurol.2011.03.003

Dan, Z., Mao, X., Liu, Q., Guo, M., Zhuang, Y., Liu, Z., et al. (2020). Altered gut microbial profile is associated with abnormal metabolism activity of Autism Spectrum Disorder. *Gut Microbes* 11, 1246–1267. doi: 10.1080/19490976.2020.1747329

Eapen, V., Cavanna, A. E., and Robertson, M. M. (2016). Comorbidities, Social Impact, and Quality of Life in Tourette Syndrome. *Front. Psychiatry* 7:97. doi: 10.3389/fpsyt.2016.00097

Fernández de la Cruz, L., and Mataix-Cols, D. (2020). General health and mortality in Tourette syndrome and chronic tic disorder: A mini-review.

Neurosci. Biobehav. Rev. 119, 514–520. doi: 10.1016/j.neubiorev.2020.11.005

Frick, L., and Pittenger, C. (2016). Microglial Dysregulation in OCD, Tourette Syndrome, and PANDAS. *J. Immunol. Res.* 2016:8606057. doi: 10.1155/2016/8606057

Fung, T. C., Olson, C. A., and Hsiao, E. Y. (2017). Interactions between the microbiota, immune and nervous systems in health and disease. *Nat. Neurosci.* 20, 145–155. doi: 10.1038/nn.4476

Haro, C., Rangel-Zúñiga, O. A., Alcalá-Díaz, J. F., Gómez-Delgado, F., Pérez-Martínez, P., Delgado-Lista, J., et al. (2016). Intestinal Microbiota Is Influenced by Gender and Body Mass Index. *PloS One* 11:e0154090. doi: 10.1371/journal.pone.0154090

Hartmann, A., Worbe, Y., and Black, K. J. (2020). Tourette syndrome research highlights from 2019. *F1000Res.* 9:1314. doi: 10.12688/f1000research.27374.2

Hibberd, C., Charman, T., Bhatoa, R. S., Tekes, S., Hedderly, T., Gringras, P., et al. (2020). Sleep difficulties in children with Tourette syndrome and chronic tic disorders: a systematic review of characteristics and associated factors. *Sleep* 43:zsz308. doi: 10.1093/sleep/zsz308

Hirschtritt, M. E., Lee, P. C., Pauls, D. L., Dion, Y., Grados, M. A., Illmann, C., et al. (2015). Lifetime prevalence, age of risk, and genetic relationships of comorbid psychiatric disorders in Tourette syndrome. *JAMA Psychiatry* 72, 325–333. doi: 10.1001/jamapsychiatry.2014.2650

Indiani CMDSP, Rizzardi, K. F., Castelo, P. M., Ferraz, L. F. C., Darrieux, M., and Parisotto, T. M. (2018). Childhood Obesity and Firmicutes/Bacteroidetes Ratio in the Gut Microbiota: A Systematic Review. *Child. Obesity* 14, 501–509. doi: 10.1089/chi.2018.0040

Isaacs, D. A., Riordan, H. R., and Claassen, D. O. (2021). Clinical Correlates of Health-Related Quality of Life in Adults With Chronic Tic Disorder. *Front. Psychiatry* 12:619854. doi: 10.3389/fpsyt.2021.619854

Isomura, K., Sidorchuk, A., Sevilla-Cermeño, L., Åkerstedt, T., Silverberg-Morse, M., Larsson, H., et al. (2022). Insomnia in Tourette Syndrome and Chronic Tic Disorder. *Move. Disord.* 37, 392–400. doi: 10.1002/mds.28842

Jandhyala, S. M., Talukdar, R., Subramanyam, C., Vuyyuru, H., Sasikala, M., and Nageshwar Reddy, D. (2015). Role of the normal gut microbiota. *World J. Gastroenterol.* 21, 8787–8803. doi: 10.3748/wjg.v21.i29.8787

Kanaan, A. S., Gerasch, S., García-García, I., Lampe, L., Pampel, A., Anwander, A., et al. (2017). Pathological glutamatergic neurotransmission in Gilles de la Tourette syndrome. *Brain* 140, 218–234. doi: 10.1093/brain/aww285

Lamothe, H., Tamouza, R., Hartmann, A., and Mallet, L. (2021). Immunity and Gilles de la Tourette syndrome: A systematic review and meta-analysis of evidence for immune implications in Tourette syndrome. *Eur. J. Neurol.* 28, 3187–3200. doi: 10.1111/ene.14983

Leckman, J. F., Katsovich, L., Kawikova, I., Lin, H., Zhang, H., Krönig, H., et al. (2005). Increased serum levels of interleukin-12 and tumor necrosis factor-alpha in Tourette's syndrome. *Biol. Psychiatry* 57, 667–673. doi: 10.1016/j.biopsych.2004.12.004

Lee, W. T., and Wong, L. C. (2018). Alterations of the intestinal microbiota were correlated with the severity of Tourette syndrome in children. *Mov. Disord.* 33:S275–S275.

Ley, R. E., Turnbaugh, P. J., Klein, S., and Gordon, J. I. (2006). Microbial ecology: human gut microbes associated with obesity. *Nature* 444, 1022–1023. doi: 10.1038/4441022a

Liao, J. F., Cheng, Y. F., Li, S. W., Lee, W. T., Hsu, C. C., Wu, C. C., et al. (2019). Lactobacillus plantarum PS128 ameliorates 2,5-Dimethoxy-4-iodoamphetamine-induced tic-like behaviors via its influences on the microbiota-gut-brain-axis. *Brain Res. Bull.* 153, 59–73. doi: 10.1016/j.brainresbull.2019.07.027

Liu, P., Jiang, Y., Gu, S., Xue, Y., Yang, H., Li, Y., et al. (2021). Metagenome-wide association study of gut microbiome revealed potential microbial marker set for diagnosis of pediatric myasthenia gravis. *BMC Med.* 19:159. doi: 10.1186/s12916-021-02034-0

Louis, P., Young, P., Holtrop, G., and Flint, H. J. (2010). Diversity of human colonic butyrate-producing bacteria revealed by analysis of the butyryl-CoA:acetate CoA-transferase gene. *Environ. Microbiol.* 12, 304–314. doi: 10.1111/j.1462-2920.2009.02066.x

Madison, A., and Kiecolt-Glaser, J. K. (2019). Stress, depression, diet, and the gut microbiota: human-bacteria interactions at the core of psychoneuroimmunology and nutrition. *Curr. Opin. Behav. Sci.* 28, 105–110. doi: 10.1016/j.cobeha.2019.01.011

Mariat, D., Firmesse, O., Levenez, F., Guimarães, V., Sokol, H., Doré, J., et al. (2009). The Firmicutes/Bacteroidetes ratio of the human microbiota changes with age. *BMC Microbiol.* 9:123. doi: 10.1186/1471-2180-9-123

Martino, D., Schrag, A., Anastasiou, Z., Apter, A., Benaroya-Milstein, N., Buttiglione, M., et al. (2021). Association of Group A Streptococcus Exposure and Exacerbations of Chronic Tic Disorders: A Multinational Prospective Cohort Study. *Neurology* 96:e1680–e1693. doi: 10.1212/WNL.0000000000011610

McGuinness, A. J., Davis, J. A., Dawson, S. L., Loughman, A., Collier, F., O'Hely, M., et al. (2022). A systematic review of gut microbiota composition in observational studies of major depressive disorder, bipolar disorder and schizophrenia. *Mol. Psychiatry* 27, 1920–1935. doi: 10.1038/s41380-022-01456-3

Mell, L. K., Davis, R. L., and Owens, D. (2005). Association between streptococcal infection and obsessive-compulsive disorder, Tourette's syndrome, and tic disorder. *Pediatrics* 116, 56–60. doi: 10.1542/peds.2004-2058

Mohammadi, M. R., Badrfam, R., Khaleghi, A., Ahmadi, N., Hooshyari, Z., and Zandifar, A. (2021). Lifetime Prevalence, Predictors and Comorbidities of Tic Disorders: A Population-Based Survey of Children and Adolescents in Iran. *Child Psychiatry Hum. Dev.* doi: 10.1007/s10578-021-01{\break}186-7

doi: 10.1007/s10578-021-01186-7

Mondot, S., Lepage, P., Seksik, P., Allez, M., Tréton, X., Bouhnik, Y., et al. (2016). Structural robustness of the gut mucosal microbiota is associated with Crohn's disease remission after surgery. *Gut* 65, 954–962. doi: 10.1136/gutjnl-2015-309184

Morais, L. H., Schreiber, H. L. IV, and Mazmanian, S. K. (2021). The gut microbiota-brain axis in behaviour and brain disorders. *Nat.Rev. Microbiol.* 19, 241–255. doi: 10.1038/s41579-020-00460-0

Müller, N., Riedel, M., Blendinger, C., Oberle, K., Jacobs, E., and Abele-Horn, M. (2004). Mycoplasma pneumoniae infection and Tourette's syndrome. *Psychiatry Res.* 129, 119–125. doi: 10.1016/j.psychres.2004.04.009

Odamaki, T., Kato, K., Sugahara, H., Hashikura, N., Takahashi, S., Xiao, J. Z., et al. (2016). Age-related changes in gut microbiota composition from newborn to centenarian: a cross-sectional study. *BMC Microbiol.* 2016:90. doi: 10.1186/s12866-016-0708-5

Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., et al. (2021). The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *BMJ* 372:n71. doi: 10.1136/bmj.n71

Parker-Athill, E. C., Ehrhart, J., Tan, J., and Murphy, T. K. (2015). Cytokine correlations in youth with tic disorders. *J. child Adolescent Psychopharmacol.* 25, 86–92. doi: 10.1089/cap.2014.0103

Plaza-Díaz, J., Ruiz-Ojeda, F. J., Vilchez-Padial, L. M., and Gil, A. (2017). Evidence of the Anti-Inflammatory Effects of Probiotics and Synbiotics in Intestinal Chronic Diseases. *Nutrients* 9:555. doi: 10.3390/nu9060555

Plessen, K. J. (2013). Tic disorders and Tourette's syndrome. *Eur. Child Adolescent Psychiatry* 22:S55–S60. doi: 10.1007/s00787-012-0362-x

Prasad, K. M. (2021). Infectious agents as risk factors for psychosis - A time to reconsider and reinvigorate investigations. *Schizophrenia Res.* 233, 111–113. doi: 10.1016/j.schres.2021.07.007

Pujo, J., Petitfils, C., Le Faouder, P., Eeckhaut, V., Payros, G., Maurel, S., et al. (2021). Bacteria-derived long chain fatty acid exhibits anti-inflammatory properties in colitis. *Gut* 70, 1088–1097. doi: 10.1136/gutjnl-2020-321173

Qi, X., Ye, J., Wen, Y., Liu, L., Cheng, B., Cheng, S., et al. (2022). Evaluating the Effects of Diet-Gut Microbiota Interactions on Sleep Traits Using the UK Biobank Cohort. *Nutrients* 14:1134. doi: 10.3390/nu14061134

Quagliariello, A., Aloisio, I., Bozzi Cionci, N., Luiselli, D., D'Auria, G., Martinez-Priego, L., et al. (2016). Effect of Bifidobacterium breve on the Intestinal Microbiota of Coeliac Children on a Gluten Free Diet: A Pilot Study. *Nutrients* 8:660. doi: 10.3390/nu8100660

Ridaura, V., and Belkaid, Y. (2015). Gut microbiota: the link to your second brain. *Cell* 161, 193–194. doi: 10.1016/j.cell.2015.03.033

Sanna, S., van Zuydam, N. R., Mahajan, A., Kurilshikov, A., Vich Vila, A., Võsa, U., et al. (2019). Causal relationships among the gut microbiome, short-chain fatty acids and metabolic diseases. *Nature genetics* 51, 600–605. doi: 10.1038/s41588-019-0350-x

Simpson, C. A., Diaz-Arteche, C., Eliby, D., Schwartz, O. S., Simmons, J. G., and Cowan, C. S. M. (2021). The gut microbiota in anxiety and depression - A systematic review. *Clin. Psychol. Rev.* 83:101943. doi: 10.1016/j.cpr.2020.101943

Stojanov, S., Berlec, A., and Štrukelj, B. (2020). The Influence of Probiotics on the Firmicutes/Bacteroidetes Ratio in the Treatment of Obesity and Inflammatory Bowel disease. *Microorganisms* 8:1715. doi: 10.3390/microorganisms8111715

Storch, E. A., Johnco, C., McGuire, J. F., Wu, M. S., McBride, N. M., Lewin, A. B., et al. (2017). An initial study of family accommodation in children and adolescents with chronic tic disorders. *Eur. child Adolescent Psychiatry* 26, 99–109. doi: 10.1007/s00787-016-0879-5

Strati, F., Cavalieri, D., Albanese, D., De Felice, C., Donati, C., Hayek, J., et al. (2017). New evidences on the altered gut microbiota in autism spectrum disorders. *Microbiome* 5:24. doi: 10.1186/s40168-017-0242-1

Tan, J., McKenzie, C., Potamitis, M., Thorburn, A. N., Mackay, C. R., and Macia, L. (2014). The role of short-chain fatty acids in health and disease. *Advances Immunol.* 121, 91–119. doi: 10.1016/B978-0-12-800100-4.00003-9

Tilling, F., and Cavanna, A. E. (2020). Relaxation therapy as a treatment for tics in patients with Tourette syndrome: a systematic literature review. *Neurol. Sci.*41, 1011–1017. doi: 10.1007/s10072-019-04207-5

Tomasova, L., Grman, M., Ondrias, K., and Ufnal, M. (2021). The impact of gut microbiota metabolites on cellular bioenergetics and cardiometabolic health. *Nutr. Metabolism* 18:72. doi: 10.1186/s12986-021-00598-5

Turna, J., Grosman Kaplan, K., Anglin, R., Patterson, B., Soreni, N., Bercik, P., et al. (2020). The gut microbiome and inflammation in obsessive-compulsive disorder patients compared to age- and sex-matched controls: a pilot study. *Acta Psychiatrica Scandinavica* 142, 337–347. doi: 10.1111/acps.13175

Ueda, K., and Black, K. J. (2021). A Comprehensive Review of Tic Disorders in Children. *J. Clin. Med.* 10:2479. doi: 10.3390/jcm10112479

Valentini, F., Evangelisti, M., Arpinelli, M., Di Nardo, G., Borro, M., Simmaco, M., et al. (2020). Gut microbiota composition in children with obstructive sleep apnoea syndrome: a pilot study. *Sleep Med.* 76, 140–147. doi: 10.1016/j.sleep.2020.10.017

Wu, S. V., and Hui, H. (2011). Treat your bug right. *Front. Physiol.* 2:9. doi: 10.3389/fphys.2011.00009

Xi, W., Gao, X., Zhao, H., Luo, X., Li, J., Tan, X., et al. (2021). Depicting the composition of gut microbiota in children with tic disorders: an exploratory study. *J. Child Psychol. Psychiatry Allied Disciplines* 62, 1246–1254. doi: 10.1111/jcpp.13409

Yang, C., Zhang, L., Zhu, P., Zhu, C., and Guo, Q. (2016). The prevalence of tic disorders for children in China: A systematic review and meta-analysis. *Medicine* 95:e4354. doi: 10.1097/MD.0000000000004354

Yano, J. M., Yu, K., Donaldson, G. P., Shastri, G. G., Ann, P., Ma, L., et al. (2015). Indigenous bacteria from the gut microbiota regulate host serotonin biosynthesis. *Cell* 161, 264–276. doi: 10.1016/j.cell.2015.02.047

Zhang, X., Zhang, D., Jia, H., Feng, Q., Wang, D., Liang, D., et al. (2015). The oral and gut microbiomes are perturbed in rheumatoid arthritis and partly normalized after treatment. *Nat. Med.* 21, 895–905. doi: 10.1038/nm.3914

Zhao, H., Shi, Y., Luo, X., Peng, L., Yang, Y., and Zou, L. (2017). The Effect of Fecal Microbiota Transplantation on a Child with Tourette Syndrome. *Case Rep. Med.* 2017:6165239. doi: 10.1155/2017/6165239

Zhao, H. J., Luo, X., Shi, Y. C., Li, J. F., Pan, F., Ren, R. R., et al. (2020). The Efficacy of Fecal Microbiota Transplantation for Children With Tourette Syndrome: A Preliminary Study. *Front. Psychiatry* 11:554441. doi: 10.3389/fpsyt.2020.554441

TYPE Original Research
PUBLISHED 16 September 2022
DOI 10.3389/fncom.2022.951718

OPEN ACCESS

EDITED BY
Deepika Koundal,
University of Petroleum and Energy Studies, India

REVIEWED BY
Pranay Goel,
Indian Institute of Science Education and Research, Pune, India
Lingchao Meng,
Northwest University for Nationalities, China

*CORRESPONDENCE
Chun Zhu
czhu@must.edu.mo

RECEIVED 24 May 2022
ACCEPTED 23 August 2022
PUBLISHED 16 September 2022

CITATION
Yan L, Li Q, Zhang Y and Zhu C (2022) Computer vision quantization research on the architectural color of Avenida de Almeida Ribeiro in Macau based on the human eye perspective.
Front. Comput. Neurosci. 16:951718.
doi: 10.3389/fncom.2022.951718

Computer vision quantization research on the architectural color of Avenida de Almeida Ribeiro in Macau based on the human eye perspective

Lina Yan[1], Qian Li[2], Yi Zhang[3] and Chun Zhu[1]*

[1]Faculty of Humanities and Arts, Macau University of Science and Technology, Taipa, China, [2]Department of Architecture, School of Civil Engineering and Mechanics, Yanshan University, Qinhuangdao, China, [3]Department of Architectural Design, Shanghai GOODLINKS International Design Group, Shanghai, China

In this study, a new quantifiable and refined urban street color analysis method was proposed by combining professional color cards and efficient software color recognition, which solved the problems of low efficiency and difficulty in the quantification of urban color research and analysis. The research mainly uses China Building Color Card (CBCC) and Python (use programs for the HSV color segmentation of pictures) and other software to carry out color recognition for a street view. From the aspects of color composition, type, proportion, visual level, and color sequence of the street facade, this article makes a quantitative analysis of the color of Avenida de Almeida Ribeiro in Macao from multiple angles. The method of combining color card colorimetry with computer color recognition, which not only considers the inherent color of the building but also includes the color situation under the influence of the environment, can express the "actual color situation" of the building more completely. This article quantifies, combs, summarizes, and compares architectural color and environmental color completely. This method has good universality and ease of use in practice, and the conclusion of the study can provide a reference for the color planning of Macao, the color selection of urban renewal has reference significance, and provide a new method for the study of urban color.

KEYWORDS

computer vision, architectural color, streets of Macao, color identification, quantitative research

Introduction

Research background

The urban color shows the unique style and temperament of a city, and streets and buildings are the main manifestations of urban color. With the continuous improvement of the demand for urban space quality, the study of urban color emerged in western countries in the mid-twentieth century. In 1978, French scholar Jean-Philippe

Lenclos, established "Atelire 3D Coulour," which designed and studied the urban color environment of residential and industrial environments in many cities (Lenclos, 2019). In 1996, Professor Michael Lancaster of the University of Greenwich put forward "Color Landscape Theory," which emphasized: "to show the relationship between colors and colors and between colors and environment, and proposed to show the color characteristics of cities in the context" (Hsiao et al., 2013). Haroldting, a professor of architecture in the United States, took color in architectural design as the object and wrote "Color Consulting" to discuss architectural color in cities (Tang et al., 2020).

In addition to the development of western color theory, Japan is one of the first countries in Asia to study urban color. In the 1970s, based on the research of French scholars, Japan established the Color Planning Center (Hsiao, 1995), an institution dedicated to the study of urban environmental Color. Subsequently, the country such as China, Korea of Italy, Germany, and Asia brings color into urban landscape environment management successively. At the end of 1990, with the introduction of urban color theory in China, urban color received more and more attention in urban construction, and color has been included in detailed urban control planning as a guiding indicator. Nowadays, many cities in China began to put forward representative urban construction colors, such as Beijing proposed compound gray, Harbin proposed beige and white.

Macao not only combines the diversity of Chinese and Western cultures but also shows its unique charm through its urban color. Just as the famous American scholar Jane Jacobs mentioned in her urban diversity theory: diversity is nature to big city (Wickersham, 2001). Urban color is also an objective existence of urban diversity and plays an important role in the urban spatial image. Streets and buildings, as the main embodiment of urban color, do not exist in isolation. The combination of architecture and environment can form a unique urban personality.

Object of study

Avenida de Almeida Ribeiro in Macao, built-in 1918, is about 580 M long, ending at "Avenida da Praia Grande" (road name) in the east and "Rua das Lorchas" (road name) in the west. The road width is 9–12 M, showing a changing trend of wide in the east and narrow in the west. As a relatively prosperous street in Macao, it has witnessed the development process of Macau's inner port area from the old fishing port wharf to the commercial hotel building. The building near Rua Das Lorchas in the west retains the original overhang structure. In the middle section are the "Instituto Para Os Assuntos Municipais" (Municipal Department) and "Largo Do Senado" (square), a hotel and a bank built-in the twentieth century. The eastern end, near Avenida da Praia Grande, was built after the twentieth century.

Avenida de Almeida Ribeiro is popularly known by residents as "The New Road." Different from other roads in the region, Avenida de Almeida Ribeiro, a major urban renewal project of Macao in the twentieth century, is an important passageway through the entire inner harbor from east to west based on the original inner harbor area of Macao. Before 1918, Avenida de Almeida Ribeiro was the section from Largo do Senado to the inner-Harbor area. In 1918, after the renovation, the government renamed the new road as "Avenida de Almeida Ribeiro."

Macao's government attaches great importance to the protection and management of historical and cultural heritage. Avenida de Almeida Ribeiro is an important part of Macao's World Cultural Heritage, and its street facade has retained its original appearance in the early twentieth century. The former municipal and commercial space forms not only show the unique charm of the coexistence of Chinese and Western cultures but also completely retain the original color of the street facade, which is beneficial to the study of the traditional color of Macao city streets in this article (Figures 1, 2).

Computational vision

Human vision mainly relies on light-sensitive cells in the retina of the eye, and color vision with the cone cells in the retina, the layer of nerve cells that transmits visual signals to the brain. In other words, color perception is based on cells and is a subjective feeling. However, the computer is based on image pixel color data statistics and can be more objective and quantified statistics.

The concept of computer vision was first proposed in 1970 (Szeliski, 2010). It is a method of translating three-dimensional objects into two-dimensional images into pixel numbers, color values, and other information for analysis through a software editing algorithm (Lee et al., 2015). The purpose is to establish a quantitative understanding of spatial images with the help of computers. Compared with the traditional way of using color cards directly or using an electronic color spectrometer to identify the color of buildings, the method of computer vision extraction is more similar and efficient to human visual perception. The traditional color card colorimetric method, through the naked eye judgment, will inevitably produce color perception error. The same color in different environments will be affected by weather, ambient light and other environmental factors and change its original color.

Consequently, this article uses computer vision to quantify color value recognition to make up for the color deviation caused by eye color recognition. Through a picture color segmentation program to obtain a variety of color statistics. This is a more objective and reproducible approach.

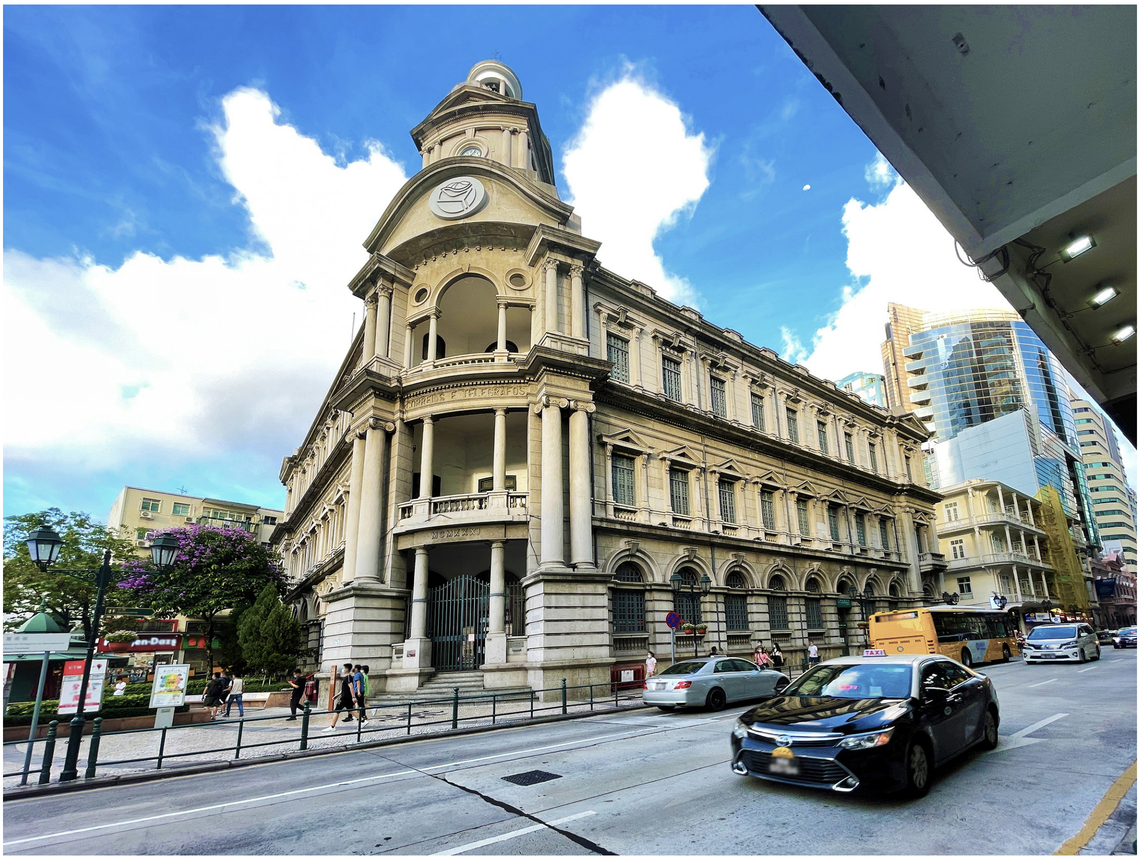


FIGURE 1
The post office of the Macao special administrative.

Color extraction and analysis methods

Foundation of architectural color system

Based on the "Munsell Color System," the Chinese architectural color system has formed The color standard of GB/T 15608-2006 (The Chinese Color System). The standard divides color into hue (H), value (V), and chroma (C) based on the three attributes of color perception. "CBCC China Building Color Card" was compiled under this standard. This study follows China's architectural color standards, based on the Munsell Color System, with the help of the "CBCC Chinese Architectural Color Card" as the reference for building inherent color sampling, combined with computer vision. "HSV color space" (Sural et al., 2002) (hue—H, saturation—S, and value—V) was used as the extraction of environment color and space color.

The computer algorithm can be easily used in HSV color space to present the hue, saturation, value, and shade of the color in the form of data. The description of HSV color space is close to the human perception of color. HSV encapsulates information about color in ways that are more familiar to humans: "What color is this? What about the depth? How about light and shade?" In addition, these color data can be separately and independently processed to facilitate more refined color quantization research.

Street color extraction—A combination of old and new methods

The method of this study combines color cards with computer color recognition.

First, "CBCC China Building Color Card" was used to compare the actual buildings with Color cards on-site. To avoid the color difference caused by weather, light changes and environmental reasons, the on-site color taking time is from 9 a.m. to 11 a.m. or from 3 p.m. to 5 p.m. on cloudy days for color card comparison and shooting (18/05/2021–20/05/2021). The number of recorded photographs shall be at least five for each building. We took 460 photos to provide a field data source for subsequent computer color recognition.

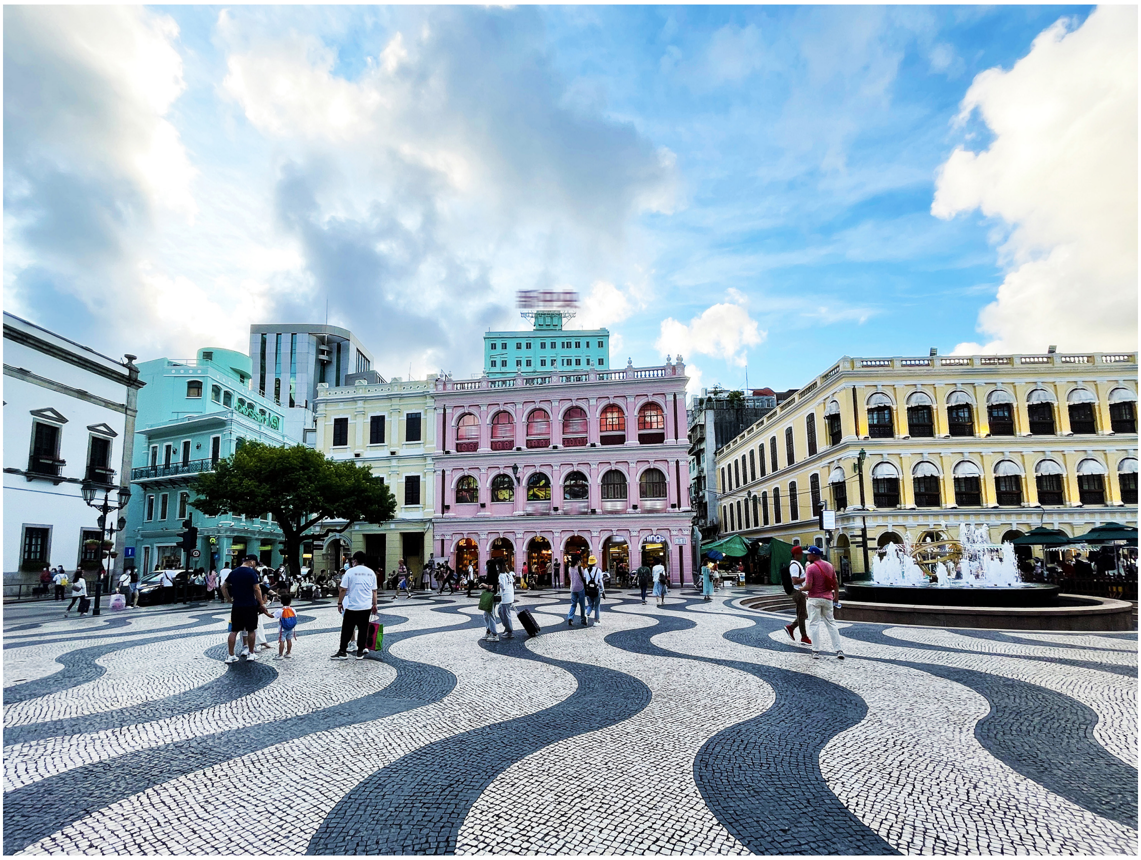


FIGURE 2
Largo do Senado.

Second, color segmentation and recognition of architectural environment color and space color were carried out with the help of Python (a program for the HSV color segmentation of pictures) (Srane96, 2019). This python program (HSV-Color-Range-Calculator) can be used to calculate HSV color ranges for each color and see the result live. We referenced and modified his program for image color processing.

Finally, the obtained color data are the approximate value of the color deviation perceived by human eyes. After sorting out the data, the overall color situation of street building facades can be obtained (Figure 3).

Color analysis method—Quantitative statistics

After extracting the "inherent color" of buildings mentioned above, the inherent color is classified into main color, auxiliary color, and ornament color according to the ratio of architectural color area and its proportion is counted.

At the same time, the computer is used to perceive and recognize the "environment color" of the building. The color sequence of the whole street building is compared and summarized according to the style characteristics and color types of the building facades on both sides of the street.

Then, with the help of Rhino and Grasshopper and other software combined with the current photos, the influence of distance and color on the "space color" of the building was analyzed.

In general, a relatively complete architectural color classification and analysis system is formed in this article from the three perspectives of "color type, color sequence, and color visual level" of street building facades.

Analysis of architectural color

Classification of architectural colors

There are a total of 73 buildings on both sides. Using CBCC color cards to compare the walls, doors, windows, and decoration of the buildings, 186 types of color samples were

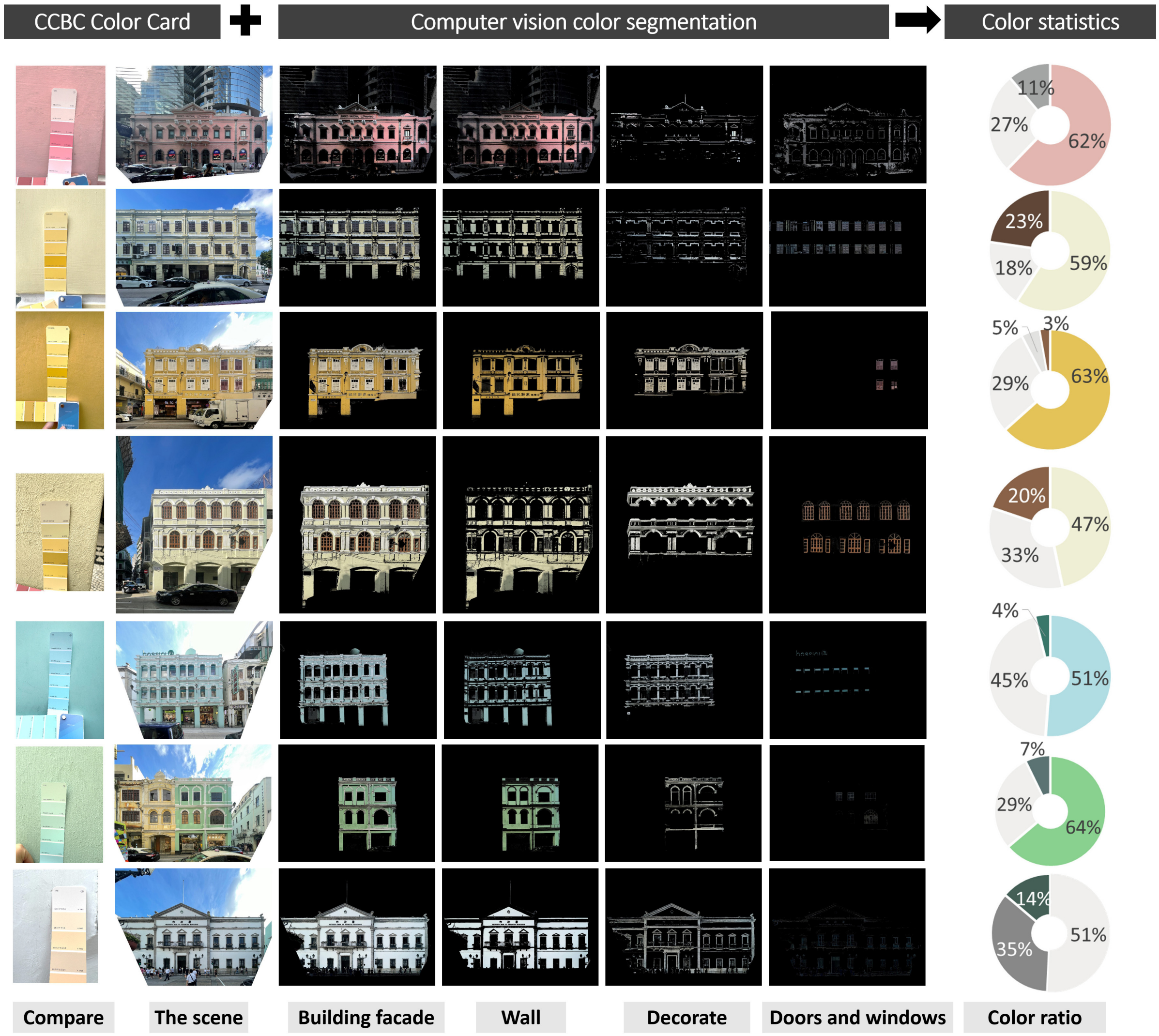


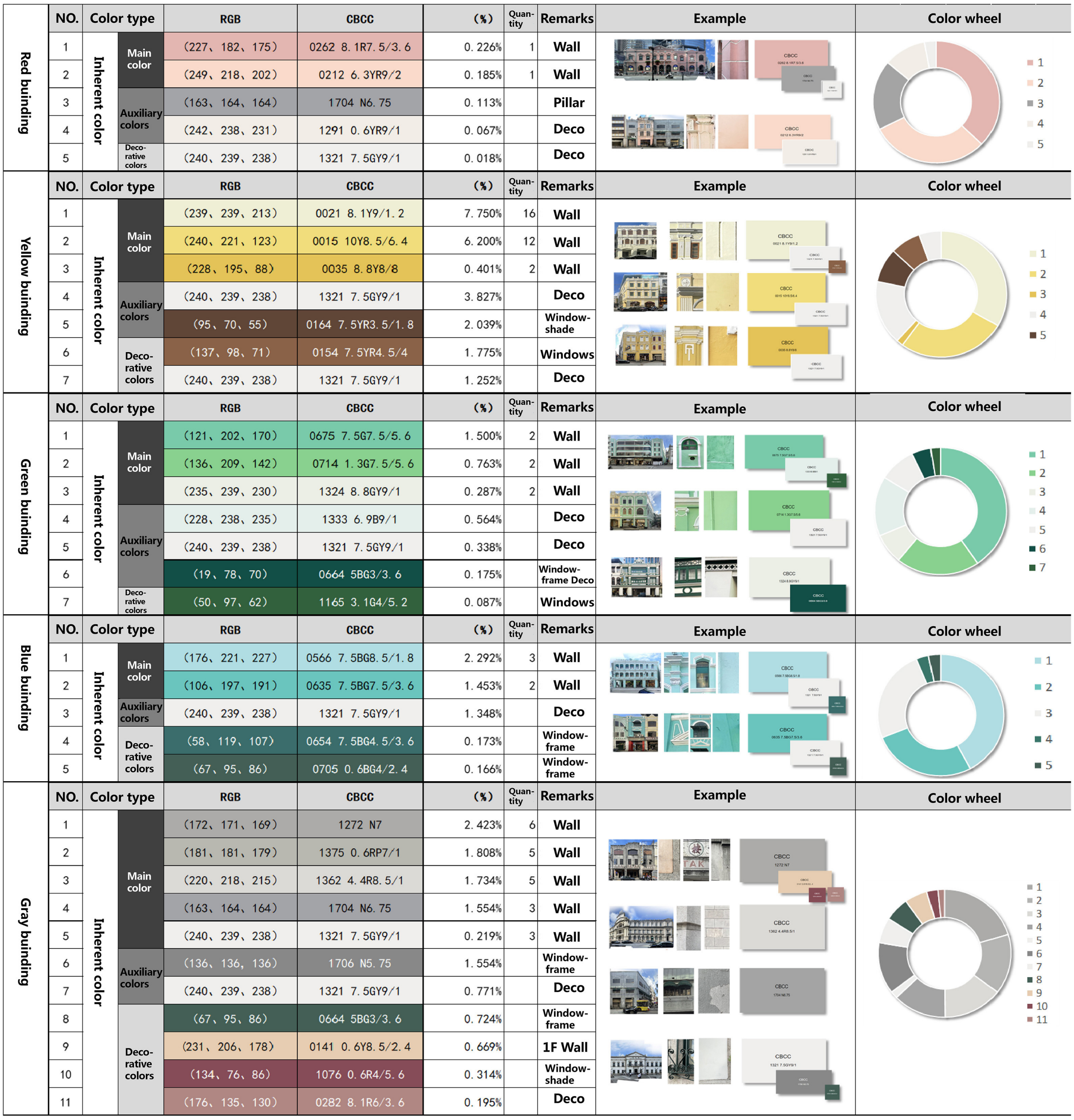
FIGURE 3
Color card selection and computer vision recognition of architectural color analysis example.

obtained. But many of the colors are the same, so we excluded the same color samples and ended up with 35 colors comparable to the color card. Among them, gray is the most abundant architectural color type and has more decorative colors. The rest is composed of a large wall face and white decoration. From the perspective of inherent color, the main color of street architecture is clear, which can be divided into red, yellow, green, blue, and gray. Auxiliary color is mainly located in the building decoration part of the gray color system. Ornament color is mainly located in the blinds, window frames, and shop sign text position (Figure 4).

Next, this study splices a large number of building facade photos taken by field research into a complete street facade map. As shown in the figure, the upper part is the full elevation of the east side of the street, and the lower part is the full elevation of the west side of the street. The color calculation range is only for the building part, and the sky and ground are not included in the calculation. After stitching, the image size is 319.28*71.79 cm, 37,710*8,479 pixels, and 300 dpi.

On this basis, the “environmental color” of street buildings is recognized by Python. Finally, color data obtained by computer color recognition are quantified and integrated with the inherent color obtained above. Complete quantitative results of street color obtained after statistics are as follows (Figure 5).

In terms of the proportion of color types of the whole building, the number of buildings identified as yellow is the largest, accounting for 33.07%. This was followed by 31.43%



	NO.	Color type		RGB	CBCC	(%)	Quantity	Remarks	Example	Color wheel
Red buinding	1	Inherent color	Main color	(227、182、175)	0262 8. 1R7. 5/3. 6	0. 226%	1	Wall		
	2			(249、218、202)	0212 6. 3YR9/2	0. 185%	1	Wall		
	3		Auxiliary colors	(163、164、164)	1704 N6. 75	0. 113%		Pillar		
	4			(242、238、231)	1291 0. 6YR9/1	0. 067%		Deco		
	5		Deco-rative colors	(240、239、238)	1321 7. 5GY9/1	0. 018%		Deco		
	NO.	Color type		RGB	CBCC	(%)	Quantity	Remarks	Example	Color wheel
Yellow buinding	1	Inherent color	Main color	(239、239、213)	0021 8. 1Y9/1. 2	7. 750%	16	Wall		
	2			(240、221、123)	0015 10Y8. 5/6. 4	6. 200%	12	Wall		
	3			(228、195、88)	0035 8. 8Y8/8	0. 401%	2	Wall		
	4		Auxiliary colors	(240、239、238)	1321 7. 5GY9/1	3. 827%		Deco		
	5			(95、70、55)	0164 7. 5YR3. 5/1. 8	2. 039%		Window-shade		
	6		Deco-rative colors	(137、98、71)	0154 7. 5YR4. 5/4	1. 775%		Windows		
	7			(240、239、238)	1321 7. 5GY9/1	1. 252%		Deco		
	NO.	Color type		RGB	CBCC	(%)	Quantity	Remarks	Example	Color wheel
Green buinding	1	Inherent color	Main color	(121、202、170)	0675 7. 5G7. 5/5. 6	1. 500%	2	Wall		
	2			(136、209、142)	0714 1. 3G7. 5/5. 6	0. 763%	2	Wall		
	3			(235、239、230)	1324 8. 8GY9/1	0. 287%	2	Wall		
	4		Auxiliary colors	(228、238、235)	1333 6. 9B9/1	0. 564%		Deco		
	5			(240、239、238)	1321 7. 5GY9/1	0. 338%		Deco		
	6			(19、78、70)	0664 5BG3/3. 6	0. 175%		Window-frame Deco		
	7		Deco-rative colors	(50、97、62)	1165 3. 1G4/5. 2	0. 087%		Windows		
	NO.	Color type		RGB	CBCC	(%)	Quantity	Remarks	Example	Color wheel
Blue buinding	1	Inherent color	Main color	(176、221、227)	0566 7. 5BG8. 5/1. 8	2. 292%	3	Wall		
	2			(106、197、191)	0635 7. 5BG7. 5/3. 6	1. 453%	2	Wall		
	3		Auxiliary colors	(240、239、238)	1321 7. 5GY9/1	1. 348%		Deco		
	4		Deco-rative colors	(58、119、107)	0654 7. 5BG4. 5/3. 6	0. 173%		Window-frame		
	5			(67、95、86)	0705 0. 6BG4/2. 4	0. 166%		Window-frame		
	NO.	Color type		RGB	CBCC	(%)	Quantity	Remarks	Example	Color wheel
Gray buinding	1	Inherent color	Main color	(172、171、169)	1272 N7	2. 423%	6	Wall		
	2			(181、181、179)	1375 0. 6RP7/1	1. 808%	5	Wall		
	3			(220、218、215)	1362 4. 4R8. 5/1	1. 734%	5	Wall		
	4			(163、164、164)	1704 N6. 75	1. 554%	3	Wall		
	5			(240、239、238)	1321 7. 5GY9/1	0. 219%	3	Wall		
	6		Auxiliary colors	(136、136、136)	1706 N5. 75	1. 554%		Window-frame		
	7			(240、239、238)	1321 7. 5GY9/1	0. 771%		Deco		
	8		Deco-rative colors	(67、95、86)	0664 5BG3/3. 6	0. 724%		Window-frame		
	9			(231、206、178)	0141 0. 6Y8. 5/2. 4	0. 669%		1F Wall		
	10			(134、76、86)	1076 0. 6R4/5. 6	0. 314%		Window-shade		
	11			(176、135、130)	0282 8. 1R6/3. 6	0. 195%		Deco		

FIGURE 4
The statistical table of "inherent color" of buildings based on the CCBC color card.

gray, 17.15% green, and 15.89% blue red buildings are less, accounting for 2.79%.

From the distribution of the overall building color (Figure 5: H-information) on both sides of the street, yellow, green, and blue occupy the majority, and red is less. It is worth noting that most of the peripheral green and blue colors in polar coordinates are not inherent colors of the building body. Among them, green (10GY ∼ 7.5 g) is greatly affected by the construction enclosure with higher purity, and blue (2.5B—10B) is greatly affected by the reflection of the sky. Therefore, excluding these two colors with greater interference, yellow (5RY—2.5Y) can be obtained as the main hue of the street building facade. Followed by yellow-green (10Y—10GY), blue-green (7.5G—2.5B), and red (10RP—7.5R), among which red (near 2.5R) with high saturation is the color of the shop's sign.

From the perspective of the overall building color saturation (Figure 6: S-information), the saturation is at a low value, and the overall color saturation of the street is mostly between 0

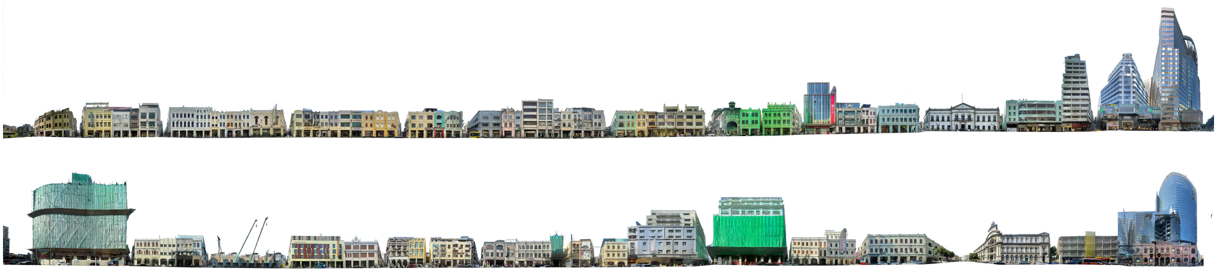


FIGURE 5
Two facades of Avenida de Almeida Ribeiro, Macau.

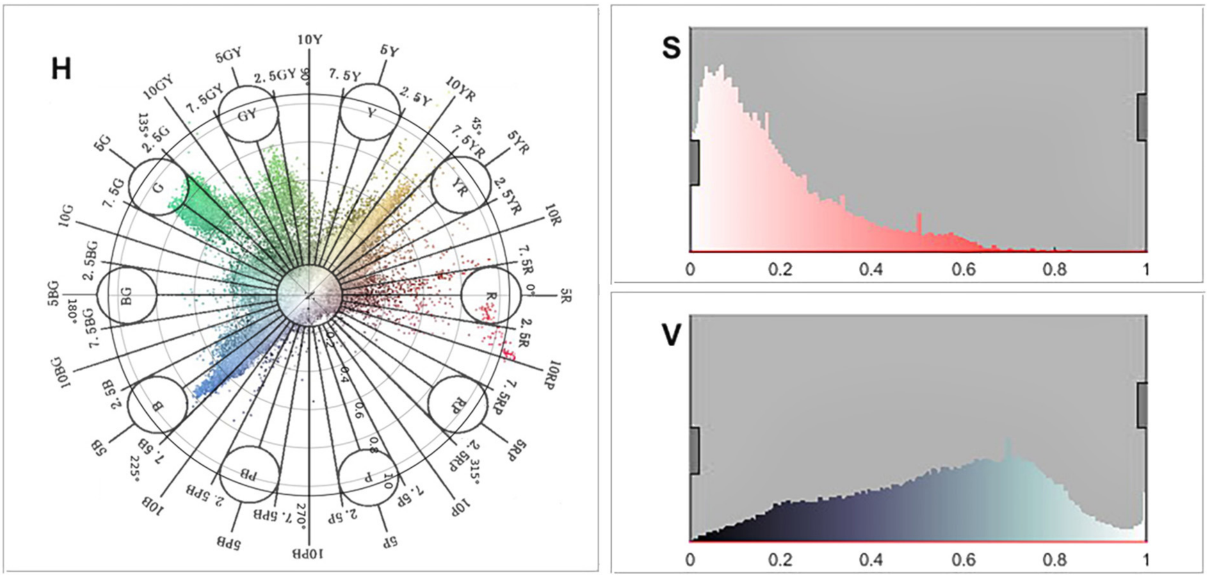


FIGURE 6
HSV color information analysis diagram.

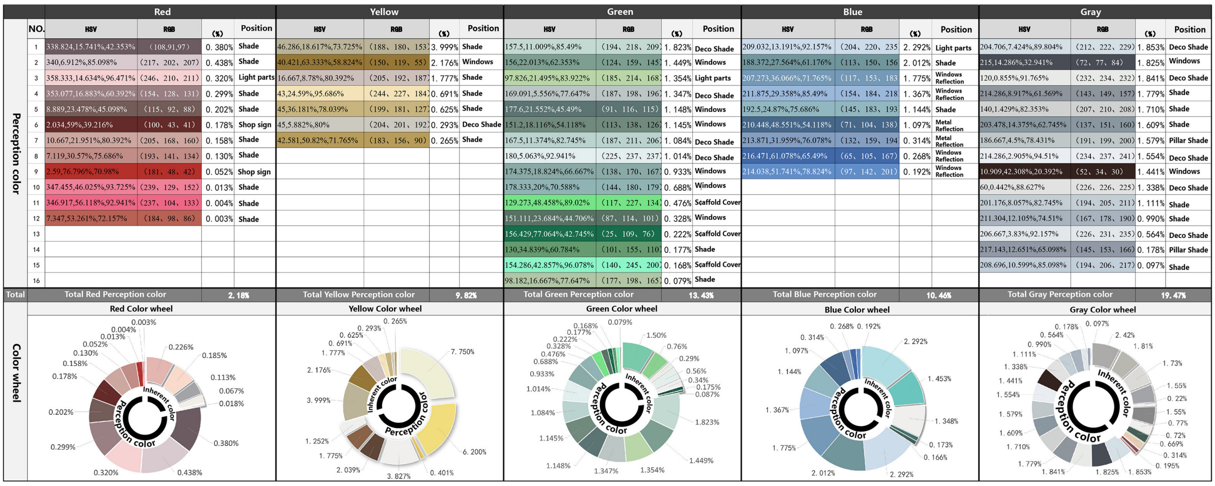
and 0.6. The overall architectural color lightness (Figure 6: V-information) tends to be medium-high, mostly between 0.2 and 0.8.

On the whole, architectural colors are characterized by clear hue types, low overall saturation, and medium-high lightness (Figure 7).

Color sequence of street building facades

As the street with the largest concentration of historical buildings in Macao, Avenida de Almeida Ribeiro's architectural color sequence can better reflect the color characteristics of historical buildings in Macao. According to the location of buildings on both sides of the street, the corresponding lightness, saturation, main color, and auxiliary color of each identified building are arranged and expanded accordingly, that is, the continuous color sequence of the building facade is obtained (Figure 8).

In terms of the value and saturation of the color sequence, the data fluctuation of the west facade is small and the color continuity is good. The lightness and saturation values on both sides of the street are very similar. The average saturation of buildings on the east side is 21.13%, and the average lightness is 62.52%. On the west side, the average saturation is 22.09% and the average lightness is 59.85%. The east side of the building is affected by facade maintenance and construction, the continuity of color is blocked and broken, and there is a high saturation of construction envelope color. Nevertheless, although the continuity of color on the east side is blocked, the intensity



	Red				Yellow				Green				Blue				Gray			
NO.	HSV	RGB	(%)	Position	HSV	RGB	(%)	Position	HSV	RGB	(%)	Position	HSV	RGB	(%)	Position	HSV	RGB	(%)	Position
1	338.824,15.741%,42.353%	（108,91,97）	0. 380%	Shade	46.286,18.617%,73.725%	（188、180、153）	3. 999%	Shade	157.5,11.009%,85.49%	（194、218、209）	1. 823%	Deco Shade	209.032,13.191%,92.157%	（204、220、235	2. 292%	Light parts	204.706,7.424%,89.804%	（212、222、229）	1. 853%	Deco Shade
2	340,6.912%,85.098%	（217、202、207）	0. 438%	Shade	40.421,63.333%,58.824%	（150、119、55）	2. 176%	Windows	156,22.013%,62.353%	（124、159、145）	1. 449%	Windows	188.372,27.564%,61.176%	（113、150、156	2. 012%	Shade	215,14.286%,32.941%	（72、77、84）	1. 825%	Windows
3	358.333,14.634%,96.471%	（246、210、211）	0. 320%	Light parts	16.667,8.78%,80.392%	（205、192、187）	1. 777%	Shade	97.826,21.495%,83.922%	（185、214、168）	1. 354%	Light parts	207.273,36.066%,71.765%	（117、153、183	1. 775%	Windows Reflection	120,0.855%,91.765%	（232、234、232）	1. 841%	Deco Shade
4	353.077,16.883%,60.392%	（154、128、131）	0. 299%	Shade	43,24.59%,95.686%	（244、227、184）	0. 691%	Shade	169.091,5.556%,77.647%	（187、198、196）	1. 347%	Deco Shade	211.875,29.358%,85.49%	（154、184、218	1. 367%	Windows Reflection	214.286,8.917%,61.569%	（143、149、157）	1. 779%	Shade
5	8.889,23.478%,45.098%	（115、92、88）	0. 202%	Shade	45,36.181%,78.039%	（199、181、127）	0. 625%	Shade	177.6,21.552%,45.49%	（91、116、115）	1. 148%	Windows	192.5,24.87%,75.686%	（145、183、193	1. 144%	Shade	140,1.429%,82.353%	（207、210、208）	1. 710%	Shade
6	2.034,59%,39.216%	（100、43、41）	0. 178%	Shop sign	45,5.882%,80%	（204、201、192）	0. 293%	Deco Shade	151.2,18.116%,54.118%	（113、138、126）	1. 145%	Windows	210.448,48.551%,54.118%	（71、104、138）	1. 097%	Metal Reflection	203.478,14.375%,62.745%	（137、151、160）	1. 609%	Shade
7	10.667,21.951%,80.392%	（205、168、160）	0. 158%	Shade	42.581,50.82%,71.765%	（183、156、90）	0. 265%	Shade	167.5,11.374%,82.745%	（187、211、206）	1. 084%	Deco Shade	213.871,31.959%,76.078%	（132、159、194	0. 314%	Metal Reflection	186.667,4.5%,78.431%	（191、199、200）	1. 579%	Pillar Shade
8	7.119,30.57%,75.686%	（193、141、134）	0. 130%	Shade					180,5.063%,92.941%	（225、237、237）	1. 014%	Deco Shade	216.471,61.078%,65.49%	（65、105、167）	0. 268%	Windows Reflection	214.286,2.905%,94.51%	（234、237、241）	1. 554%	Deco Shade
9	2.59,76.796%,70.98%	（181、48、42）	0. 052%	Shop sign					174.375,18.824%,66.667%	（138、170、167）	0. 933%	Windows	214.038,51.741%,78.824%	（97、142、201）	0. 192%	Windows Reflection	10.909,42.308%,20.392%	（52、34、30）	1. 441%	Windows
10	347.455,46.025%,93.725%	（239、129、152）	0. 013%	Shade					178.333,20%,70.588%	（144、180、179）	0. 688%	Windows					60,0.442%,88.627%	（226、226、225）	1. 338%	Deco Shade
11	346.917,56.118%,92.941%	（237、104、133）	0. 004%	Shade					129.273,48.458%,89.02%	（117、227、134）	0. 476%	Scaffold Cover					201.176,8.057%,82.745%	（194、205、211）	1. 111%	Shade
12	7.347,53.261%,72.157%	（184、98、86）	0. 003%	Shade					151.111,23.684%,44.706%	（87、114、101）	0. 328%	Windows					211.304,12.105%,74.51%	（167、178、190）	0. 990%	Shade
13									156.429,77.064%,42.745%	（25、109、76）	0. 222%	Scaffold Cover					206.667,3.83%,92.157%	（226、231、235）	0. 564%	Deco Shade
14									130,34.839%,60.784%	（101、155、110）	0. 177%	Shade					217.143,12.651%,65.098%	（145、153、166）	0. 178%	Pillar Shade
15									154.286,42.857%,96.078%	（140、245、200）	0. 168%	Scaffold Cover					208.696,10.599%,85.098%	（194、206、217）	0. 097%	Shade
16									98.182,16.667%,77.647%	（177、198、165）	0. 079%	Shade								
Total	Total Red Perception color		2. 18%		Total Yellow Perception color		9. 82%		Total Green Perception color		13. 43%		Total Blue Perception color		10. 46%		Total Gray Perception color		19. 47%	

FIGURE 7
Statistical table of building "environmental color" based on computer vision analysis.

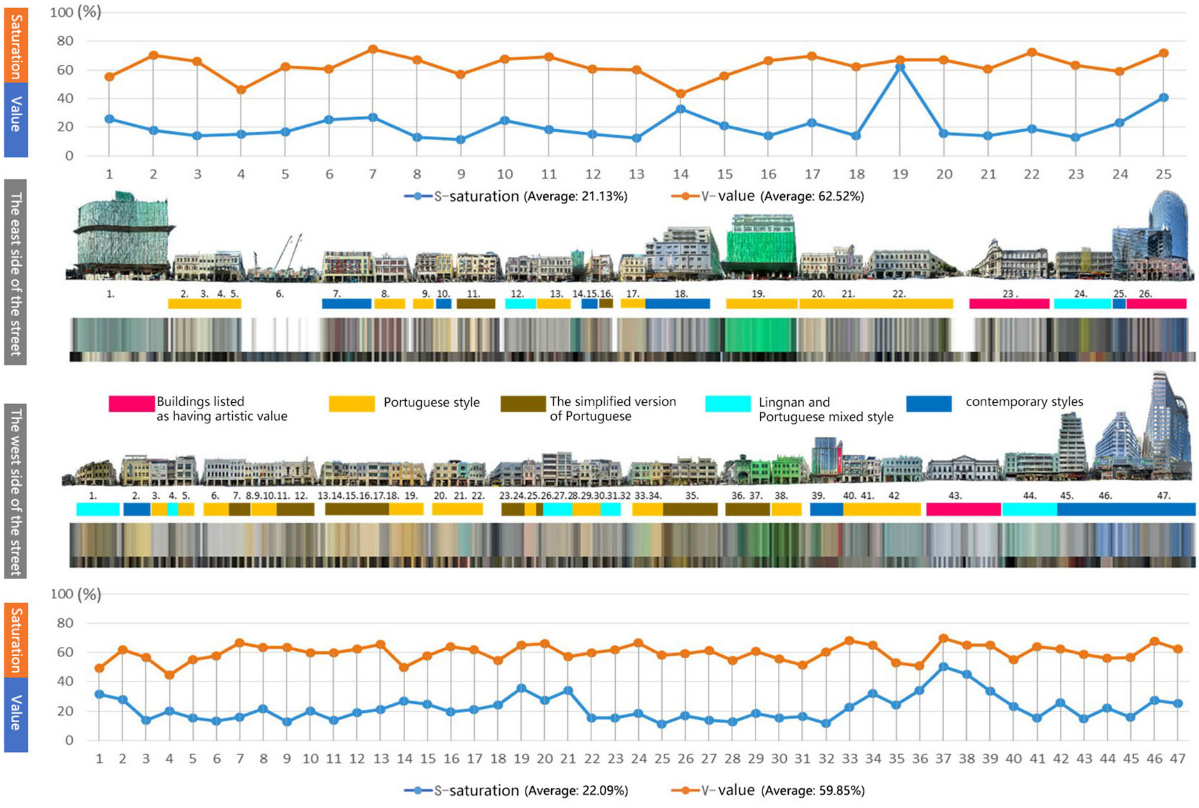


FIGURE 8
Analysis of the changing trend of building type and facade color lightness and saturation.

and area of maintenance also reflect that Macao attaches great importance to the protection and maintenance of historic building facades.

The style of the architectural is corresponding to the color. The relation between architectural style and color can be obtained by calculating pixel values of different colors. The architectural styles on both sides of the street can be roughly divided into (1) buildings listed as having artistic value; (2) Portuguese-style architecture, (3) the simplified version of Portuguese architecture, (4) Lingnan and Portuguese mixed style architecture, and (5) Contemporary architectural styles. In terms of the color pixel value of each type, Portuguese architecture occupies the highest proportion (42.3%). Portuguese architecture accounted for 30.6% of the total pixels, and simplified Portuguese architecture accounted for 11.6% of the total pixels. Contemporary architectural style occupies the second place, accounting for 35.6% due to its higher overall height and larger color area. Again, 8.7% of Macau's buildings were listed as artistic, including the baroque civil affairs office (Instituto Para Os Assuntos Municipais). The post office of the Macao special administrative, and the preserved pink facades on the ground floor of the Banco Nacional Ultramarino (B.N.U Building). Finally, construction and maintenance accounted for 2.2% of the total. In addition, through the perception of cold and warm architectural colors, it is found that Portuguese classical historical buildings are mainly yellow-green warm colors. Buildings in the Chinese Lingnan style are mainly gray; contemporary buildings are a cool shade of blue with glass walls and marble veneers.

In conclusion, the building color value of the whole street is higher, and the saturation is lower. Higher color value makes the streets look brighter, and lower saturation makes the street feel softer. These styles not only blend on the same road but also maintain their color characteristics and coordinate with each other, forming the color gene barcode with the characteristics of Macao City.

Visual hierarchy of street architectural colors

Define the visual hierarchy of colors

Based on human field angle, with the help of Python and GH (Grasshopper parametric analysis software), the spatial color recognition of street buildings affected by distance is studied, that is, the color change analysis of street buildings located in the front, middle, and back of different visual levels.

The computer color perception setting is based on the height of the human eye position. From the average sizes of men and women in China (male: 1.75 m, female: 1.63 m), the total average height is 1.69 m (World Data, 2020). Some studies have demonstrated that the ratio of head height to body height in adults is 1:7.5 and the height range of the eyes is 1/2 of head height (Shi and Huang, 2015). Therefore, according to the calculation, the average eye height of Chinese adults is about 1.58 m.

The upper and lower limits of the visual field color discrimination range are 30° up and 40° down. The left and right boundaries are 30°-60° to the left and 30°-50° to the right (Mollon, 1982) (Figures 9, 10).

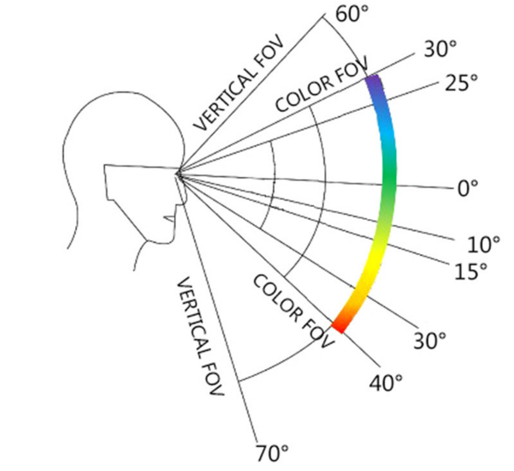


FIGURE 9
The range of color discrimination of human visual field (vertical direction).

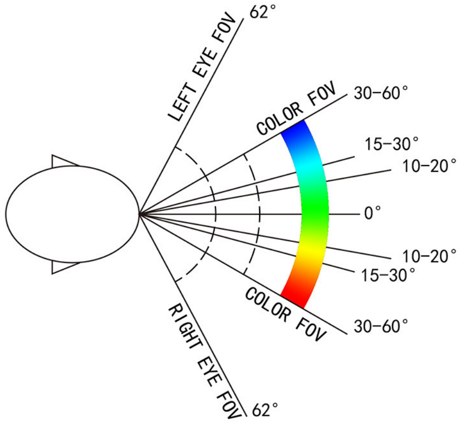


FIGURE 10
The range of color discrimination of human visual field (horizontal direction).

In addition, the perceived depth of the view level is divided into three scales according to "Exterior Modular theory,"(Ashihara, 1981) which is convenient for calculation and statistics. The observation range is set as follows: 0–25 m is the close-up view that can see the details of the building; The mid-range from 25 to 100M of the building outline can be observed; able to see objects with the blurred outline of 100M or more in distance by color or light (Yang et al., 2020).

To have a comprehensive understanding of people's different visual feelings in the two directions of the street, six isometric observation sections were set between the beginning and end

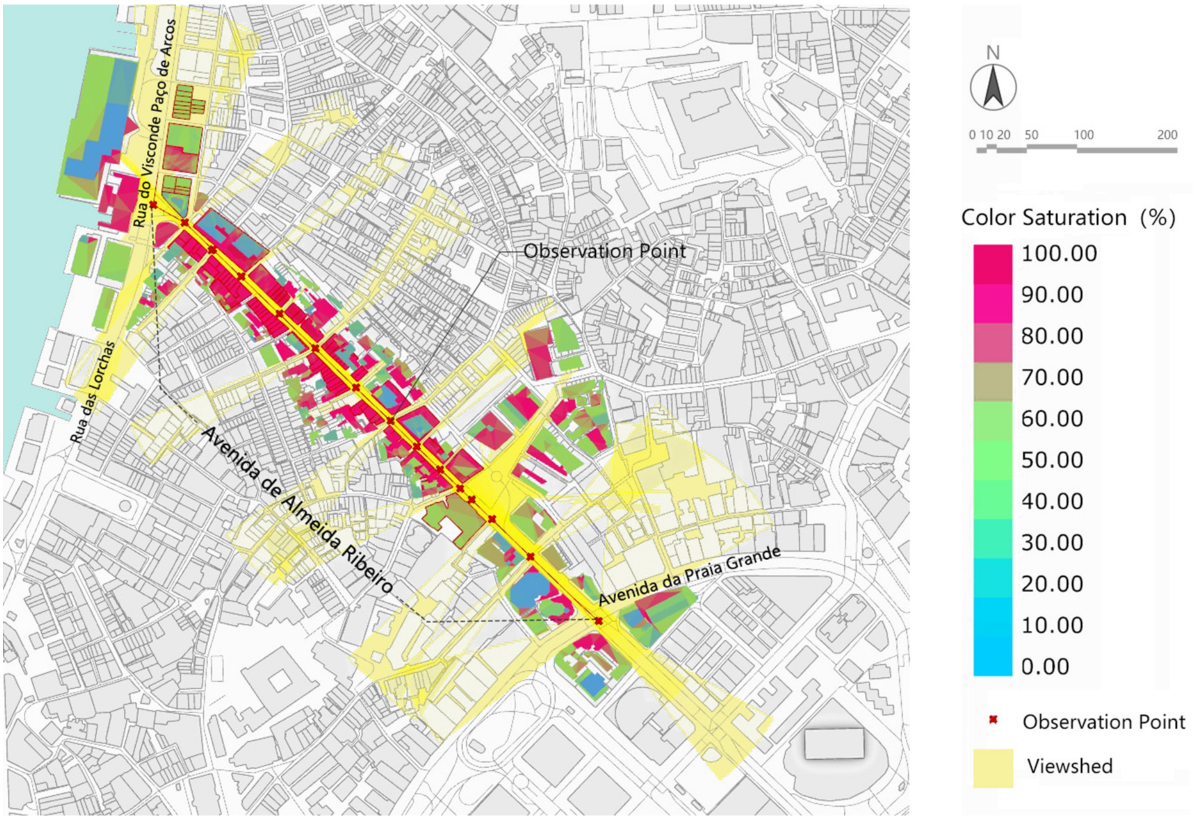


FIGURE 11
Analysis of Architectural Color Hierarchy perception of Avenida de Almeida Ribeiro in Macao.

of the sidewalk on both sides of the street with a total length of about 590 M according to the 100 M boundary of the vision. There were seven observation points including the starting and ending points, and each observation point was a two-way forward and backward observation. Thus, a total of 14 observation angles were used to analyze the spatial color of street building facades.

Visual hierarchy analysis of color

According to the overall color level perception analysis of street building facades, the spatial color of the street building facade is affected by distance, street width, and building height.

Avenida de Almeida Ribeiro is 9–12 M wide. The road is wide on the east and narrow on the west. The horizontal angle of view is set to 0–25 m at close range, which can completely cover the color recognition of the building facades on both sides of the street. For the vertical view, the limits are set to 30° up and 40° down. Avenida da Praia Grande to Largo Do Senado is dominated by high-rise buildings. The section from Largo do Senado to Rua do Visconde Paco de Arcos is dominated by the three stories Macau Varanda building with a height of about 15 M. Therefore, the spatial color perception of the section of high-rise buildings is dominated by the color of low-rise buildings. According to the calculation, the color perception degree of the high-rise building section is 40–70%; the section of the Macau Varanda building can completely cover the building facade, and the color perception is 80–100%. The overall color perception degree is bounded by Largo do Senado, presenting two obvious spatial color perception states (Figure 11).

Thus, it can be seen that the integrity of spatial color perception can be better ensured by controlling the width of the street and building height within the close-range view of 25 M.

At the same time, space color is greatly affected by the distance between the two directions of the street. Avenida de Almeida Ribeiro is 590 M long and has a relatively straight road, which is relatively transparent from one end without large buses. This kind of road state has a great influence on the architectural space color on both sides of the street. As the depth of field changes, the greater the refraction of light by atmospheric dust at greater distances, the less clear the distant buildings become. It also reduces color saturation. In good weather, the dust refracts atmospheric blue light more, and the farther away from the building, the bluer it is. The color and

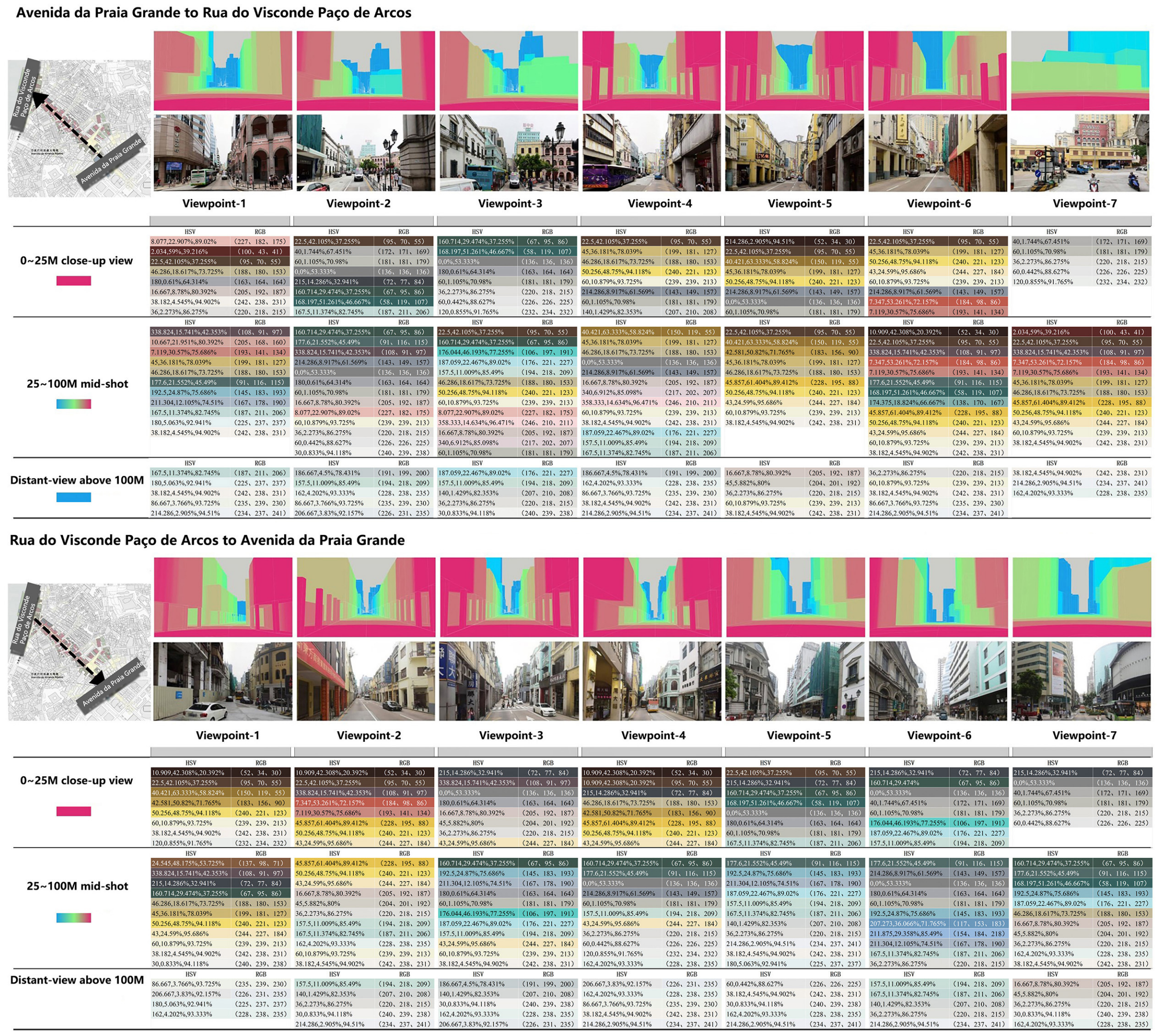


FIGURE 12
Statistical table of spatial color levels and colors of 14 observation points in two directions of the street.

the distance of the building present a subtle hierarchical change (Figure 12).

From the data of the specific spatial color perceptive area, the variation trend of 0–25 m close-range in both directions is large, and the value is in the range of 10.22–49.30%. The variation of mid-shot from 25 to 100 M ranged from 36.82 to 73.88%. Distant-view above 100 M has little variation, ranging from 0.51 to 10.64%. In addition, according to the data changes in observation points, it can be found that Largo do Senado (View point-3) is also the cut-off point, and there are two obvious trends of change: The "high-rise section" between Avenida da Praia Grande and Largo do Senado has a relatively small close-range area, ranging from 10.22 to 21.21%, and its color is mainly blue and gray. The area of the mid-shot is generally larger, with values ranging from 53.64 to 73.88%. The value and saturation of colors are relatively high, especially in the vicinity of observation point 3 (Figure 13).

While the close view of the "Macau Varanda Building Section" from Largo Do Senado to Rua do Visconde Paco de Arcos is more than that of the "High-rise Section," the close-range value is between 16.73 and 49.30%, and the color is mainly yellow and gray. The mid-shot is less than the "tall building section," the value is between 36.87 and 69.33%, and the color is mainly yellow-green. From another point of view, the spatial color perceptive area of the "High-rise section" is characterized by less close-range and great mid-shots, while the proportion of close-up views and mid-shots of the "Macau Varanda Building Section" is large and fluctuates steadily.

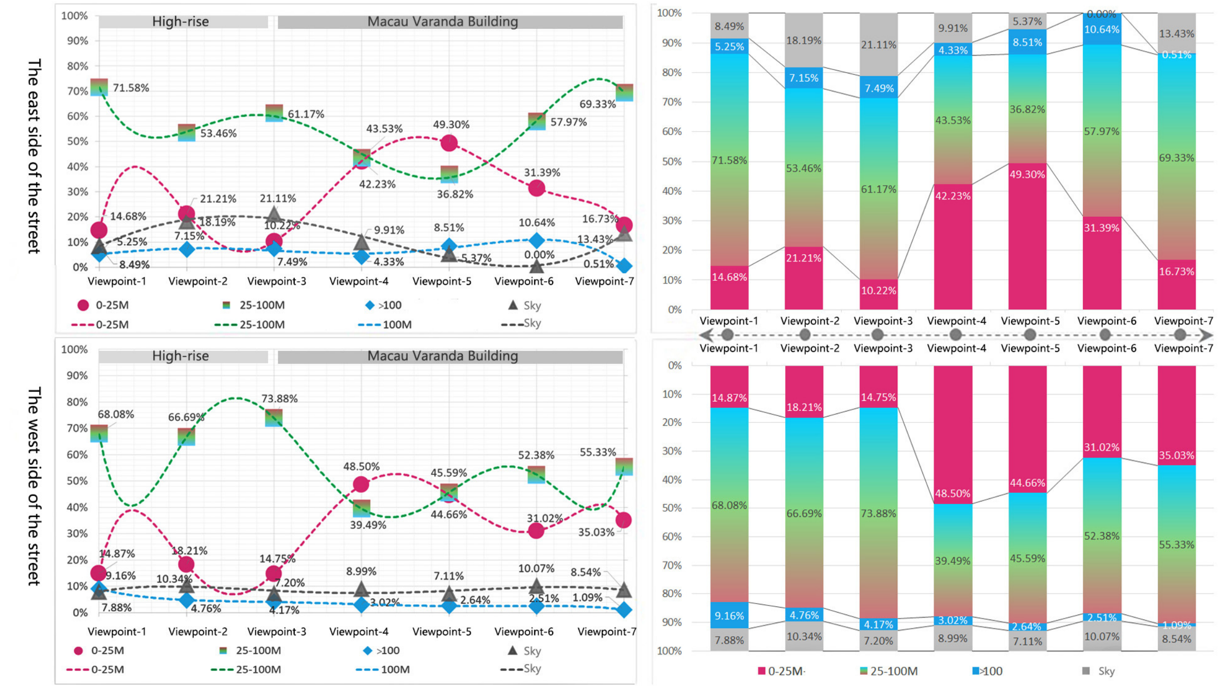


FIGURE 13
Space color perception area and trend of observation points.

In addition, as the space color perception of "Distant-view" is affected by the height of buildings at both ends of the road, the spatial color perception shows different rules. Avenida de Almeida Ribeiro is surrounded by tall buildings at both ends, so the spatial color area of the "Distant-view" is affected by the height of distant tall buildings both in the forward and backward directions. The range of "Distant-view" in two directions was 0.51–10.64% and 1.09–9.16%, respectively.

The data in both directions have their characteristics. According to distant-view spatial color statistics from Avenida da Praia Grande to Rua do Visconde Paco de Arcos, except for the endpoint of Rua do Visconde Paco de Arcos, the area of the distant-view in this direction remains between 5.25 and 10.64%. As the endpoint of this direction is a T-junction and the buildings at the end are in large yellow tones, the distant-view is almost blocked by the tall buildings on the opposite side of the road, and the color perception level of the view space is mainly yellow in close-up view and mid-shot. Although the blue-gray color of distant-view is only 0.51%, it continues the yellow color characteristics of this section of the Macau Varanda building.

From Rua do Visconde Paco de Arcos to Avenida da Praia Grande, the trend is obvious, showing a gradually increasing trend from 1.09 to 9.16%. This is because Avenida da Praia Grande is directly connected to Avenida do Infante D. Henrique (road name) with the same long value, which further extends the distant space. Avenida do Infante D. Henrique is dominated by tawdry high-rise commercial buildings and hotels. The sense of space of color spreads from blue-gray in close-up views to yellow-gray of distant-view.

Conclusion

This article takes the color of the Avenida de Almeida Ribeiro in Macao as the research object and analyzes the color classification from the three aspects of "inherent color," "environment color," and "space color" of the street facade. After extracting the "inherent color" from the CBCC Chinese architecture color card and quantifying the "environmental color" by the Python program, the phenomenon and rules of color classification research on the facade of the Macau World Cultural Heritage street building are discussed in terms of color types, color sequences, and color levels. Thus, a multi-angle street building facade color research system is formed, which can comprehensively summarize the color characteristics of the road-building facade.

From the perspective of color classification, this article classifies and arranges the main and auxiliary colors according to color types to find out the combination rules of architectural colors. The results show that the color of Macau's World Cultural Heritage Street buildings is characterized by clear hue type, low overall saturation (S = 0–0.6), and medium-high value (V =

0.2–0.8). At the same time, the number of buildings identified as yellow is 33.07%, followed by gray at 31.43%, green at 17.15%, blue at 15.89%, and red at 2.79%.

From the perspective of horizontal space of color, the article summarizes the relationship between architectural style and color sequence. Buildings of artistic value, including baroque municipal offices (Instituto Para Os Assuntos Municipais), the post office of the Macao special administrative and Banco Nacional Ultramarino, 8.7%; Portuguese classical historical buildings accounted for 42.3%, mainly in the warm color of yellow and green; Buildings with Chinese Lingnan style accounted for 11.2%, mainly gray; Modern buildings, moreover, are 35.6 percent blue. Architectural types in different periods on the same road can not only blend in style but also maintain their color characteristics and coordinate with each other in color.

From the perspective of vertical space of color, GH software is used to sort out the color horizon level of "space color" and the change in the color space level. The overall color level perception degree of street building facade space color is affected by distance, street width, and building height. Largo do Senado is taken as the turning point to present the distinctive spatial layers of the "High-rise Road section" and "Macau Varanda Building Road Section." Space color is greatly influenced by the distance between the two directions of the street and the height of the building at the end of the road.

In conclusion, based on the characteristics of human vision, a new method of color quantization combining color cards and computer vision analysis can be used to comprehensively analyze and comb the street facade colors of Avenida de Almeida Ribeiro in Macao. This method can reduce the error of traditional empirical visual recognition and has ease of use and universality in practice. The conclusion can provide a reference for color planning and facade color restoration of Macao, and has reference significance for the color selection of urban renewal. Nevertheless, the accuracy of this color recognition method needs to be further improved. In future research, the computer vision color perception analysis method used in this article will continue to be enhanced in a more accurate and intelligent direction, providing new ideas for the study of urban color.

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/s.

Ethics statement

This study was reviewed and approved by the Institutional Research Committee of Macau University of Science and Technology.

Author contributions

Conceptualization: LY, CZ, and QL. Methodology, formal analysis, data curation, writing—original draft preparation, visualization, and project administration: LY. Software: LY, QL, and YZ. Validation: QL and YZ. Investigation and writing—review and editing: LY and QL. Resources: LY and CZ. Supervision: CZ. All authors have read and agreed to the published version of the manuscript.

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

Ashihara, Y. (1981). *Exterior Design in Architecture*. New York City, NY: Van Nostrand Reinhold Company.

Hsiao, S. W. (1995). A systematic method for color planning in product design. *Color Res. Appl.* 20, 191–205. doi: 10.1002/col.5080200309

Hsiao, S. W., Hsu, C. F., and Tang, K. W. (2013). A consultation and simulation system for product color planning based on interactive genetic algorithms. *Color Res. Appl.* 38, 375–390. doi: 10.1002/col.21730

Lee, S., Maisonneuve, N., Crandall, D., Efros, A. A., and Sivic, J. (2015). "Linking past to present: discovering style in two centuries of architecture," in *IEEE International Conference on Computational Photography*. doi: 10.1109/ICCPHOT.2015.7168368

Lenclos, J. P. (2019). "The geography of colour," in *Colour for Architecture Today* (Milton Park: Taylor and Francis), 39–44. doi: 10.4324/9781315881379-10

Mollon, J. D. (1982). Color vision. *Annu. Rev. Psychol.* 33, 41–85. doi: 10.1146/annurev.ps.33.020182.000353

Shi, Z., and Huang, H. (2015). Demarcating algorithm for camera shooting viewpoints of target capsules based on golden cut theory. *Modern Computer.* 35, 47–51. doi: 10.3969/j.issn.1007-1423.2015.35.010

Srane96 (2019). *HSV-Color-Range-Calculator*. Available online at: https://github.com/srane96/HSV-Color-Range-Calculator/blob/master/HSV_Calculator_Webcam.py

Sural, S., Qian, G., and Pramanik, S. (2002). "Segmentation and histogram generation using the HSV color space for image retrieval," in *Proceedings. International Conference on Image Processing* (New York, NY: IEEE), Vol. 2, pp. II–II. doi: 10.1109/ICIP.2002.1040019

Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Cham: Springer Science and Business Media.

Tang, J., Lin, L., Ma, J. R., and Bai, J. (2020). An empirical research on the urban color characteristics of Dalian. *Urban. Architect.* 2, 45–46. doi: 10.19892/j.cnki.csjz.2020.02.014

Wickersham, J. (2001). Jane Jacob's critique of zoning: from Euclid to Portland and beyond. *BC Envtl. Aff. L. Rev.* 28, 547. Available online at: https://lawdigitalcommons.bc.edu/ealr/vol28/iss4/5/

World Data (2020). *Average Sizes of Men and Women.* https://www.worlddata.info/averagebodyheight.php

Yang, J. Y., Sun, X., Pan, Y. W., and Xia, G. Y. (2020). Landscape and view: spatial analysis and construction ways of view system in city. *City Plan. Rev.* 12, 103–112. doi: 10.11819/cpr20201213a

TYPE Original Research
PUBLISHED 07 October 2022
DOI 10.3389/fncom.2022.964686

Check for updates

OPEN ACCESS

EDITED BY
Deepika Koundal,
University of Petroleum and Energy Studies, India

REVIEWED BY
Loknath Sai Ambati,
Indiana University Kokomo,
United States
Ahmed J. Obaid,
University of Kufa, Iraq

*CORRESPONDENCE
Jerry Chun-Wei Lin
jerrylin@ieee.org

RECEIVED 08 June 2022
ACCEPTED 09 September 2022
PUBLISHED 07 October 2022

CITATION
Saikumar K, Rajesh V, Srivastava G and Lin JC-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

Heart disease detection based on internet of things data using linear quadratic discriminant analysis and a deep graph convolutional neural network

K. Saikumar[1], V. Rajesh[1], Gautam Srivastava[2,3,4] and Jerry Chun-Wei Lin[5]*

[1]Department of ECE, Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram, Andhra Pradesh, India, [2]Department of Mathematics and Computer Science, Brandon University, Brandon, MB, Canada, [3]Research Centre for Interneural Computing, China Medical University, Taichung, Taiwan, [4]Department of Mathematics and Computer Science, Lebanese American University, Beirut, Lebanon, [5]Western Norway University of Applied Science, Bergen, Norway

Heart disease is an emerging health issue in the medical field, according to WHO every year around 10 billion people are affected with heart abnormalities. Arteries in the heart generate oxygenated blood to all body parts, however sometimes blood vessels become clogged or restrained due to cardiac issues. Past heart diagnosis applications are outdated and suffer from poor performance. Therefore, an intelligent heart disease diagnosis application design is required. In this research work, internet of things (IoT) sensor data with a deep learning-based heart diagnosis application is designed. The heart disease IoT sensor data is collected from the University of California Irvine machine learning repository free open-source dataset which is useful for training the deep graph convolutional network (DG_ConvoNet) deep learning network. The testing data has been collected from the Cleveland Clinic Foundation; it is a collection of 350 real-time clinical instances from heart patients through IoT sensors. The K-means technique is employed to remove noise in sensor data and clustered the unstructured data. The features are extracted to employ Linear Quadratic Discriminant Analysis. DG_ConvoNet is a deep learning process to classify and predict heart diseases. The diagnostic application achieves an accuracy of 96%, sensitivity of 80%, specificity of 73%, precision of 90%, F-Score of 79%, and area under the ROC curve of 75% implementing the proposed model.

KEYWORDS

heart disease, detection, IoT - internet of things, sensor data, deep learning, artificial neural network

Introduction

According to WHO, cardiovascular disease (CVD) is a significant reason of death worldwide, with 17.8 million deaths every decade (Rath et al., 2021). The American Cardiac Organization (Zhang and Xu, 2021) specifies detailed indications like sleep disorders, slight pain increase as well as a drop-in heart rate and fast weight improvement (up to 1.5-2.5 kg per 7 days) (Vincent Paul et al., 2021). However, more study data and patient records from hospitals become available as time goes on. Machine learning (ML) and artificial intelligence (AI) are now widely recognized as able to play a vital role in the medical industry. ML and deep learning (DL) methods are often used to diagnose conditions as well as classify or anticipate results. ML algorithms can do a complete examination of genetic data in a short amount of time. Medical records are modified and analyzed extra thoroughly for improved predictions, and methods are trained for knowledge pandemic predictions (Liu et al., 2022). Heart disorders are identified with congenital, coronary and rheumatic events, and 370,000 Americans died due to coronary heart disease (HD) type heart attacks in 2015. Annually Americans are spending $250 billion USD on HD diagnosis and treatment. According to the American heart association, medical HD disorders will be able to be predicted by 2030.

Exercise stress tests, chest X-rays, CT scans, MRI, coronary angiograms, and electrocardiograms (ECG) are currently used to diagnose the severity of HD in patients. Patients need early and precise diagnoses of coronary HD to receive timely and effective treatment and boost their chances of long-term survival. Unfortunately, cardiovascular specialists may not be available in many resource-limited places worldwide to do these diagnostic tests. Missing diagnoses, incorrect diagnoses, and therapies put patients' health in danger in many circumstances. In addition, early detection of HD causes preventative interventions such as drugs, lifestyle changes, angioplasty, or surgery, which can help to slow disease development as well as minimize morbidity (Morris and Lopez, 2021). As a result, precise and timely heart disease diagnostics are critical for lowering mortality as well as enhancing long-term survival rates in patients. Because early detection of coronary HD is challenging, computer-assisted techniques for detecting and diagnosing heart disease in people have been developed. In medical institutions, ML methods that analyze clinical data, evaluate it, and diagnoses medical conditions is becoming increasingly common in healthcare fields.

The research contributions of this paper are as follows:

1. Collect internet of things (IoT) sensor-based heart disease data in the detection of heart disease using a deep learning architecture.
2. Process input data for noise removal and cluster the data using K-means clustering.
3. Extract the features using Linear Quadratic Discriminant Analysis.
4. Classify the extracted data using a deep graph convolutional network (DG_ConvoNet).

Appendix

Internet of things (IoT), World Health Organizations (WHO), cardiovascular disease (CVD), Receiver Operating Characteristic Curve (ROC), Machine learning (ML), Artificial Intelligence (AI), Deep learning (DL), Heart-Disease (HD), Support Vector Machine (SVM), Heart Rate Variability (HRV), Convolution Neural Network (CNN), Magnetic Resonance Imaging (MRI), Deep Graph Convolutional Network (DG_ ConvoNet).

Related work

In this section, a brief literature has been employed from the latest research papers related to heart disease prediction using IoT sensor data. Feature extraction, classification and predictions are the major steps involved in intelligence algorithms. Manogaran et al. (2017) utilized a variety of big data methods to detect cardiac illness, as well as hyperparameter tuning to improve the accuracy of results. Kanksha Aman et al. (2021) employed generalized discriminant analysis for extracting nonlinear HD features. A binary classifier with extreme ML has been used to reduce overfitting issues as well as increase training time on finding heart disorders prediction. For detecting coronary HD, the accuracy was 73% had been attained which was very less. Heart rate variability was classified as an arrhythmia by Divya et al. (2021). The heart abnormality disorders classification was done with a multilayer perceptron neural network, and 91% accuracy was reached by decreasing features or using Gaussian Discriminant Analysis, in this research work hidden features haven't been included. Hasan and Bhattacharjee (2019) employed Gaussian discriminant analysis to reduce HRV signal characteristics to 15 and an SVM classifier to obtain 70% precision, this research work cannot solve unstructured sensor data from IoT networks. An enhanced CNN model is proposed by Huang et al. (2019), in which 92.35% accuracy had been detected, the main limitation of this work is STFT-based spectrogram analysis. The STFT model is very old and faces clustering issues when large datasets have been applied to it. The Fruit classification is a complex process to predict heart diseases through IoT sensor data. The following challenge was solved by using a CNN-based technique by Wang et al. (2020). According to the researchers, designed past HD detection methods has a less classification accuracy, which is get improved than the existing methods. Zhang et al. (2019) proposed a comprehensive description of multimodal data fusion of heart-related sensor data. A combination of

CT, MRI, PET, optical imaging and radionuclide datasets has resulted in complete pathology of heart disorders in a radiology manner. The image fusion-based approach has been found to improve clinical diagnosis in recent years but failed at emergency diagnosis conditions. The CNN-based diagnosis algorithm implemented by Zhang et al. (2020a). In this research, stochastic pooling, as well as optimization of hyperparameters connected with CNN. The major drawback of this study is neuroimages orientation is altered from patient to patient so that when applying a new image to the designed application, the HD abnormality detection rate had been getting changes extremely.

The realized methods which are shown in **Table 1** have less operational sensitivity, specificity, and accuracy. Zhang et al. (2020b) introduced an FGCNet-based HD features extraction from GCN and CNN models. This method is used to diagnose chest CT scan-based heart disorders prediction but fails at noise-based CT scan radiology images specified as test input. The FGCNet is said to aid quick COVID 19 detection utilizing chest CT scans. Wang et al. (2021a) presented the CCSHNet method for heart disorders detection, which combines deep fusion. The designed CCSHNet models failed at large data samples applied at the training stage. The DCA and transfer learning-based models are very critical to detecting HD at large dimensional data. The CCSHNet is a viable option for detecting infectious heart illnesses, including COVID 19, according to deep exhaustive analysis. The literature review from many latest articles identified that traditional ML-based detection of arrhythmia with ECG signals analysis methods are outdated. However, fewer research works have been published on HD detection utilizing ECG signals and DL techniques are trending but IoT-related works are not much efficient to predict HD. Wang et al. (2021b) evaluate classification algorithms using an ML technique to predict cardiac disease. This work demonstrated the bagging technique prediction for HD with a good performance rate, as well as accuracy level. Superior HD prediction models other than past techniques are necessary. Martins et al. (2021) offer a genetic approach for predicting human heart disease through echocardiographic, the designed method is limited to huge unstructured data. The implemented method might reduce the number of test cases required to detect HD issues based on Ali et al. (2021) and Ladefoged et al. (2021). The successful HD abnormality prediction based on the radiology dataset is outdated as well as latest IoT-based techniques are required. Saikumar et al. (2022) aim to develop a precise categorization algorithm for accurately predicting cardiac disease but are unable to work on IoT sensor data. The following work concluded that regression classification is used to predict HD more accurately than other techniques by Saikumar and Rajesh, 2020a,b. R-C4.5 is proposed, and its features are extractí from the given technique by Koppula et al. (2021). The study used their equipment and found it a very beneficial machine in the healthcare industry for predicting ML-based approaches Garigipati et al. (2022). The above discussions are providing information about earlier HD prediction models and its limitations. It is clear that many cardiac diagnosis models are facing various low-level and high-level issues under dynamic conditions. This research work looks to solve some of the indicated issues from the related works.

System model

This section discusses the proposed DL technique based on feature extraction as well as classification in heart disease diagnosis. Here, the input data has been collected as IoT sensor data from a patient monitoring system.

The collected data has been processed for noise abstraction using a clustered-based K-means algorithm. Gaussian noise that was present in the medical images was removed at this block. Clustered information is used to extract the features utilizing Linear Quadratic Discriminant Analysis. Finally, the extracted features have been classified using the DG_ConvoNet. The architecture of the proposed method is shown in **Figure 1**. The pre-processing unit categorizes image registration from the medical raw image data (University of California Irvine machine learning repository). The registration enhancement process is used to line up the image for de-noise processing. Due to speckle disturbances, medical images get damaged and hinder the ability to identify deep features needed for DL. As a result, medical images are de-specked using a filtering approach technique to improve categorization results.

K-means clustering

Since k represents the number of clusters, there are k centroids, one for every cluster. After the Euclidean distance between each data point and the centroid has been evaluated, the assignment of data points to the centroid is based on the shortest Euclidean distance from that centroid. An early grouping is done when no point is left unassigned. Now, k new centroids are generated, and the iteration continues until the k centroids' positions do not change. In this stage, 256 clusters had to be created and processed for the centroid calculation of the cluster.

Let $Y = \{x_1, x_2, x_3, \ldots, \ldots, x_n\}$ are set of dataset opinions as well as $Z = \{z1, z2, \ldots, \ldots z_c\}$ be set of centers.

1. Arbitrarily choose 'c' cluster centers.
2. Evaluate the distance among each information point as well as cluster centers.
3. Allot data points to the cluster center with the shortest distance between it and all other cluster centers.
4. Again, evaluate the original cluster center using the following Eq. (1):

$$Z_i = (1/c_i) \,.\, \Sigma_{j=1}^{\mathbb{E}_1} x_{\mathbb{I}} \tag{1}$$

TABLE 1 Recent studies related to heart abnormality prediction.

S No	Author	Techniques	Performance accuracy	Limitation
1	Rahmani et al., 2018	IoT based e-health	Accuracy = 93.24% HD Detection rate = 0.76 Application score = 0.86	Data clustering is a complex issue
2	Majumder et al., 2019	Smart IoT-based cardiac disorders detection	Accuracy = 95.23% HD Detection rate = 0.79 Application score = 0.72	Limited large dimensional data
3	Golande et al., 2019	Smart medical applications using IoT	Accuracy = 91.47% HD Detection rate = 0.86 Application score = 0.83	Network issue due to conventional data analysis
4	Haq et al., 2018	ML-based HD detection	Accuracy = 95.42% HD Detection rate = 0.83 Application score = 0.71	Radiology data-based analysis is sometimes altered from sample to sample
5	Hinton and Salakhutdinov, 2006	High dimensional HD data-based abnormality detection	Accuracy = 93.68% HD Detection rate = 0.69 Application score = 0.82	Limited to structured HD data

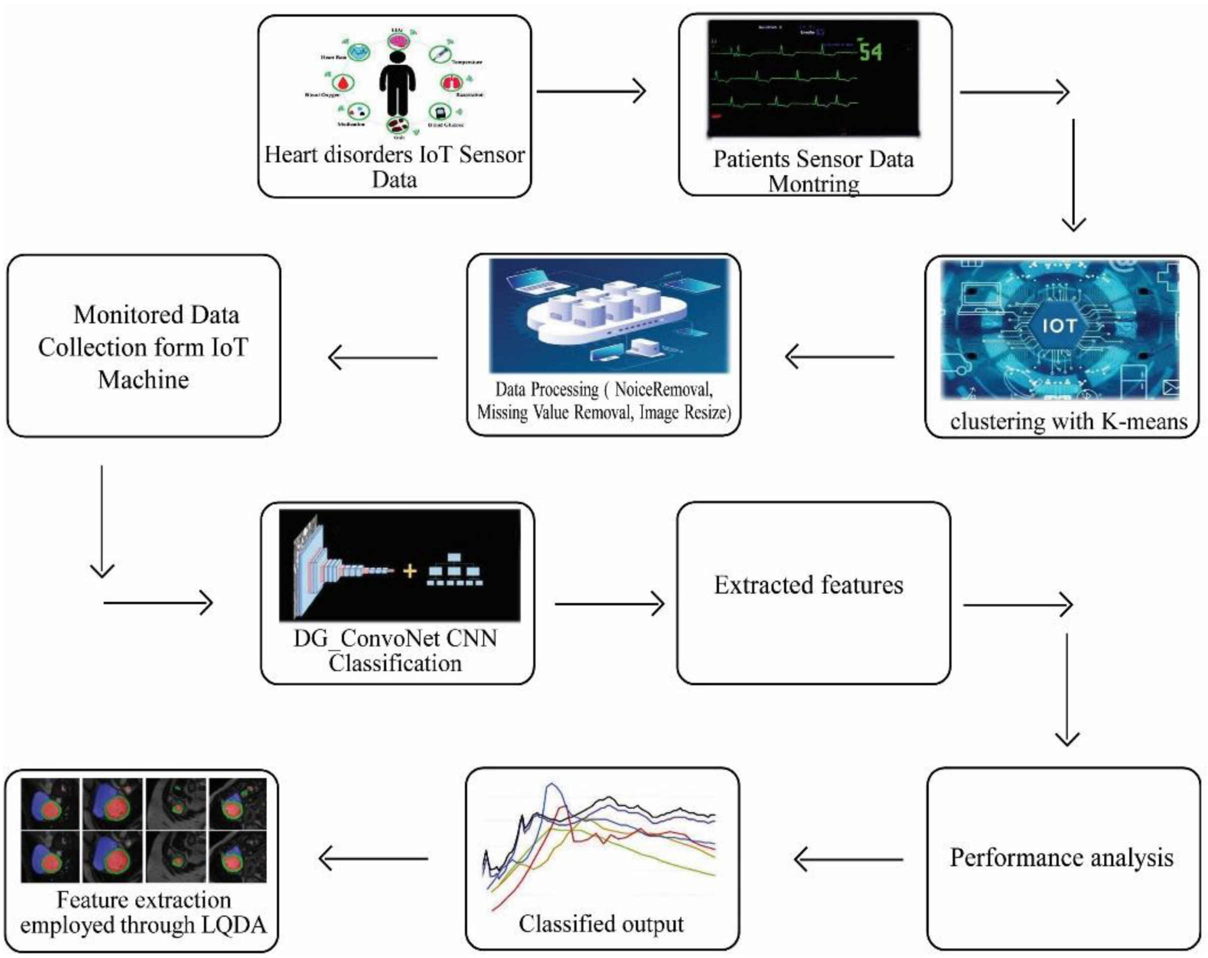

Heart disorders IoT Sensor Data
54
Patients Sensor Data Montring
IOT
clustering with K-means
Data Processing (NoiceRemoval, Missing Value Removal, Image Resize)
Monitored Data Collection form IoT Machine
DG_ConvoNet CNN Classification
Extracted features
Performance analysis
Classified output
Feature extraction employed through LQDA

FIGURE 1
Proposed IoT sensor data-based heart disease (HD) prediction.

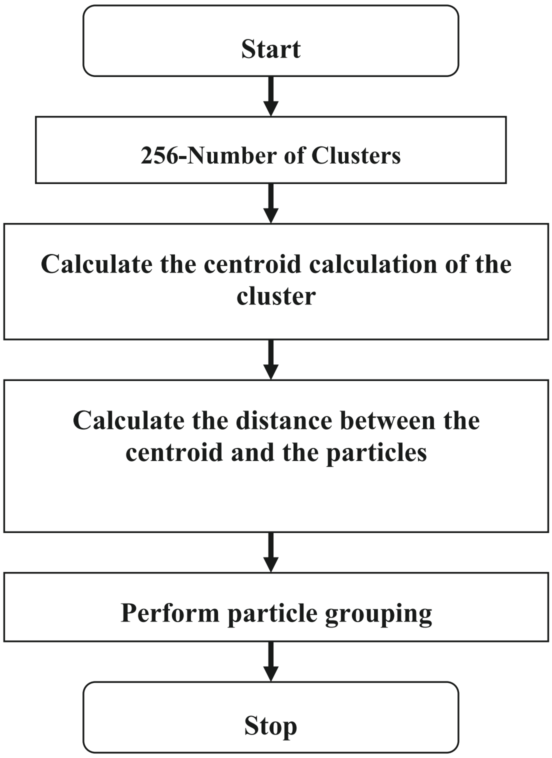

Start
256-Number of Clusters
Calculate the centroid calculation of the cluster
Calculate the distance between the centroid and the particles
Perform particle grouping
Stop

FIGURE 2
Flow chart K-means.

5. Where 'c$_i$' indicates the number of data opinions in the ith cluster.
6. Again, calculate the distance between every data point as well as the original cluster centers.
7. Stop if no information points were reallocated; otherwise, start over at step 3.

The flow chart of K-means clustering is shown in Figure 2. In this K-means flow is explained with clustered extraction on the dataset. The centroid, Euclidean and particle estimation parameters have been providing information about deep dataset information. The dataset consists of shape-based image features which are processed by the K-means algorithm.

Linear quadratic discriminant analysis based feature extraction

Let S_b and S_w be among and within-class scatter matrices, low-dimensional complement space of null space of S_b, related as $\mathcal{S}'$, is first extracted. Let $V_b = [v_{b1}, \dots, v_{bM}]$ be M eigenvectors of S_b corresponding to M non-zero eigenvalues $A = [\lambda_{b1}, \dots, \lambda_b M]$, where $M = \min(C-1, J)$. The S_b subspace $\mathcal{B}'$ is thus spanned by V_b, which is further scaled by $U = V_b A_b^{-1/2}$ so that $U^T S_b U = \mathcal{I}$, where $A_b = \text{diag}\ (A)$, diag()indicates the diagonalization operator and $\mathcal{I}$ is the $(M = M)$ identity matrix by Eq. (2):

$$\grave{\Sigma}_i(\alpha, \gamma) = (1-\gamma)\grave{\Sigma}_i(\alpha) + \frac{\gamma}{M}\,\text{tr}\left[\grave{\Sigma}_i(\alpha)\right] I,$$

$$\grave{\Sigma}_i(\alpha) = \frac{1}{C_i(\alpha)}\left[(1-\alpha)S_i + \alpha S\right], \tag{2}$$

M is the dimensionality of $\mathcal{B}'.C_i(\alpha) = (1-\alpha)C_i + \alpha N$ and S_i is the covariance matrix of ith class evaluated in $\mathcal{B}'$, i.e., $S_i = \Sigma_{j=1}^{C_i}\left(y_{ij} - \bar{y}_i\right)\left(y_{ij} - \bar{y}_i\right)^T$, $y_{ij} = U^T z_{ij}$, $\bar{y}_i = (1/C_i)\ \Sigma_{j=1}^{C_i} y_{ij}$ and $S = \Sigma_{i=1}^{C} S_i$.

Let $\Phi = [\phi(z_{11}), \dots, \phi(z_{CCc})]$ be corresponding feature representations of training samples in kernel space $\mathbb{F}^F$. Let K be $N = N$ Gram matrix, i.e., $K = (K_{lh})_{l=1,,C}^{h=1,,C_{Ih}}$ is a $C_l \times C_h$ sub – matrix of K composed of samples from classes $\mathcal{I}_l$ and $\mathcal{Z}_h$, i.e., $K_{lh} = (k_{ij})_{i=1,.,C_l}^{j=1,}$, where $k_{ij} = k\left(z_{li}, z_{hj}\right)$ and $k(\cdot)$indicates kernel function defined in $\mathbb{R}^J$. Let $\bar{S}_b$ be between-class scatter in $\mathbb{F}^F$, described as Eq. (3)

$$\grave{S}_b = \frac{1}{N}\sum_{i=1}^{C} C_i\left(\overleftarrow{\phi}_i - \overleftarrow{\phi}\right)\left(\overleftarrow{\phi}_i - \overleftarrow{\phi}\right)^T \tag{3}$$

where $\overleftarrow{\phi}_i = (1/C_i)\ \Sigma_{j=1}^{C_i}\phi\left(z_{ij}\right)$ is the mean of $\mathcal{Y}_i$ in $\mathbb{F}$F and $\overleftarrow{\phi} = (1/N)\Sigma_{i=1}^{C}\Sigma_{j=1}^{C_i}\phi\left(z_{ij}\right)$ is mean of training samples FF.

Eigenvectors of S_b, i.e., $\grave{V}_b = [\bar{v}_{b1}, \dots, \bar{v}_{bM}]$, corresponding to M largest eigenvalues. $\grave{V}_b$is obtained by solving the eigenvalue issue of $\bar{S}_b$, which is represented as Eq. (4):

$$\bar{s}_b = \sum_{i=1}^{c}\left(\sqrt{\frac{ci}{N}}(\overleftarrow{\phi}_i - \overleftarrow{\phi})\right)\left(\sqrt{\frac{ci}{N}}(\overleftarrow{\phi}_i - \overleftarrow{\phi})\right)^T$$

$$= \sum_{i=1}^{c}\grave{\overleftarrow{\phi}}_i\grave{\overleftarrow{\phi}}_i{}^T = \Phi_b\Phi_b^T \tag{4}$$

where $\grave{\phi}_i = \sqrt{C_i/N}\left(\overleftarrow{\phi}_i - \overleftarrow{\phi}\right)$ and $\Phi_b = \left[\grave{\phi}_1, \dots, \grave{\phi}_C\right]$. It is given that $\bar{S}_b$ is a matrix of size $F \times F$, where F indicates kernel space dimensionality. Due to HD of $\mathbb{F}^F$, a direct computation of eigenvectors of $\bar{S}_b$ is impossible$\left(\Phi_b\Phi_b^T\right)(\Phi_b\bar{e}_{bi}) = \overleftarrow{\lambda}_{bi}(\Phi_b\bar{e}_{bi})$. Therefore, it is deduced that $(\Phi_b\bar{e}_{bi})$ is the i th eigenvector of $\bar{S}_b = \Phi_b\Phi_{b-}^T$

$$\Phi_b^T\Phi_b = \frac{1}{N}B\left(A_{NC}^T\cdot K\cdot A_{NC} - \frac{1}{N}\left(A_{NC}^T\cdot K\cdot 1_{NC}\right)\right.$$
$$\left. - \frac{1}{N}\left(1_{NC}^T\cdot K\cdot A_{NC}\right) + \frac{1}{N^2}\left(1_{NC}^T\cdot K\cdot 1_{NC}\right)\right)B \tag{5}$$

where $B = \text{diag}\left[\sqrt{C_1}, \dots, \sqrt{C_C}\right]$, 1_{NC} is an $N \times C$ matrix with all elements equal to $1, A_{NC} = \text{diag}\left[a_{C_1}, \dots, a_{C_C}\right]$ is an $N \times C$ block diagonal matrix and a $C_i$is a $C_i = 1$ vector with all elements equal to $1/C_i$. Let $\overline{E}_{bM} = \left[\bar{e}_{b1}, ..., \grave{e}_{bM}\right]$ consist of M significant eigenvectors of $\Phi_b^T\Phi_b$ corresponding to M largest eigenvalues $\overleftharpoon{\lambda}_{b1} > ,\cdots, > \overleftharpoon{\lambda}_b M$ and $\grave{V}_b = \Phi_b\overline{E}_{bM}$, it is not difficult to derive that $\overline{V}_b^T\overline{S}_b\overline{V}_b = \overleftharpoon{\Lambda}_b$, where $\overleftharpoon{\Lambda}_b = \text{diag}\left[\overleftharpoon{\lambda}^2_{b,1}, ..., \overleftharpoon{\lambda}^2_{b,M}\right]$. Thus, the transformation matrix $\vec{U}$ such that $\overline{U}^T\grave{S}_b\overline{U} = I$is evaluated as Eqs. (6), (7):

$$\overline{U} = \overline{V}_b \overleftharpoon{A}_b^{-1/2}, \overline{V}_b = \Phi_b\grave{E}_{bM} \tag{6}$$

$$\grave{y}_{ij} = \overline{U}^T\phi\left(z_{ij}\right) = \overleftharpoon{A}_b^{-1/2}\overline{E}_{bM}^T\Phi_b^T\phi\left(z_{ij}\right) \tag{7}$$

where $\Phi_b^T\phi\left(z_{ij}\right)$ can be expressed as Eq. (8)

$$\Phi_b^T\phi\left(z_{ij}\right) = \frac{1}{\sqrt{N}}B\left(A_{NC}\cdot v\left(\phi\left(z_{\mathbf{ij}}\right)\right) - \frac{1}{N}1_{NC}^T\cdot v\left(\phi\left(z_{ij}\right)\right)\right) \tag{8}$$

where $v\left(\phi(z_{ij})\right) = \left[\phi(z_{11})\phi(z_{ij}), \phi(z_{12})\phi(z_{ij}), ..., \phi\left(z_{CC_C}\right)\phi(z_{ij})\right]^T$is evaluated implicitly through the kernel function described in $\mathbb{R}^J$, i.e., $\phi\left(z_{mn}\right)\phi\left(z_{ij}\right) = k\left(z_{mn}, z_{ij}\right)$.

$$\grave{\Sigma}_i(\alpha,\gamma) = (1-\gamma)\grave{\Sigma}_i(\alpha)+\frac{\gamma}{M}\text{tr}\left[\overline{\Sigma}_i(\alpha)\right]I,$$

$$\grave{\Sigma}_i(\alpha) = \frac{1}{C_i(\alpha)}\left[(1-\alpha)\overline{S}_i+\alpha\overline{S}\right],$$

$$C_i(\alpha) = (1-\alpha)C_i+\alpha N,$$

$$\grave{S}_i = \sum_{j=1}^{C_i}\left(\bar{y}_{ij}-\bar{\bar{y}}_i\right)\left(\bar{y}_{ij}-\bar{\bar{y}}_i\right)^T,$$

$$\overline{S} = \sum_{i=1}^{C}\grave{S}_i.$$

$$\bar{\bar{y}}_i = (1/C_i)\sum_{j=1}^{C_i}\bar{y}_{ij} \tag{9}$$

and $(\boldsymbol{\alpha}, \boldsymbol{\gamma})$ is a pair of regularization parameters.

The key component in the evaluation of $\overline{\Sigma}_i(\alpha,\gamma)$is to arise covariance matrix of ith class, i.e., $\overline{S}_i$which is given as Eq. (10):

$$\grave{S}_i = \sum_{j1}^{C_i}\left(\grave{y}_{ij}-\bar{\bar{y}}_i\right)\left(\bar{y}_{ij}-\bar{\bar{y}}_i\right)^T$$

$$= \sum_{j1}^{C_i}\bar{y}_{ij}\bar{y}_{ij}^T - \sum_{j1}^{C_i}\bar{\bar{y}}_i\bar{y}_{ij}^T - \sum_{j1}^{C_i}\bar{y}_{ij}\bar{\bar{y}}_i^T + \sum_{j1}^{C_i}\grave{\bar{y}}_i^T\bar{\bar{y}}_i^T$$

$$= \sum_{j1}^{C_i}\bar{y}_{ij}\grave{y}_{ij}^T - C_i\bar{\bar{y}}_i\overline{\overleftharpoon{y}}_i^T - C_i\bar{\bar{y}}_i\bar{\bar{y}}_i^T + C_i\bar{\bar{y}}_i\bar{\bar{y}}_i^T$$

$$= \sum_{j1}^{C_i}\grave{y}_{ij}\bar{y}_{ij}^T - C_i\bar{\bar{y}}_i\bar{\bar{y}}_i^T$$

$$= J_1 - C_i \times J_2, \tag{10}$$

where $J_1 = \Sigma_{j=1}^{C_i}\grave{y}_{ij}\bar{y}_{ij}^T$ and $J_2 = \bar{\bar{y}}_i\bar{\bar{y}}_i^T$. The detailed derivation of J_1 and $J_2$is determined in Appendices A and B.

Mahalanob is distance between feature representation of test image $\bar{q}$ and each class centre $\bar{\bar{y}}_i$ is then used to identify the test image. i.e., $ID\left(p\right) = \arg\min_i d_i\left(\bar{q}\right)$, that can be calculated in Eq. (11) as:

$$d_i(\bar{q}) = \left(\bar{q}-\bar{\bar{y}}_i\right)^T\grave{\Sigma}_i^{-1}(\alpha,\gamma)\left(\bar{q}-\bar{\bar{y}}_i\right) + \ln\left|\overline{\Sigma}_i(\alpha,\gamma)\right| - 2\ln\pi_i, \tag{11}$$

where $\pi_i = C_i/N$.

$$\left(\grave{A} = \arg\max_{\overline{A}}\left|\grave{A}^T\grave{S}_b\grave{A}\right| / \left|\grave{A}^T\overline{S}_b\overline{A}\right| + \left|\grave{A}^T\grave{S}_w\grave{A}\right|\right)$$

$$\textit{when}\left(\alpha = 1, \gamma = \left(\text{tr}\left(\overline{S}_i/N\right)+M\right)/M\right)$$

Classification using deep graph ConvoNet (convolutional network)- DG_ ConvoNet:

$\mathcal{G} = (\mathcal{Y}, \mathcal{E}, \mathcal{H})$ defines an undirected and connected graph, Here A and S are limited sets of $|A| = S$ vertices as well as edges$W \in \mathbb{R}^{N\times N}$. Numerous variables in each vertex represent the graph signals. $\mathcal{L} = D-W$, where $D = \text{diag}\left(\grave{d}_0, \cdots, d_{N-1}\right)$ is a grading matrix designed in steps $d_i = \Sigma_j\mathcal{W}_{i,j}$of vertex i. $\{\chi_l\}_{/=0}^{N-1}$, as well as nonnegative eigenvalues $0 \leq \lambda_0 \leq \cdots \lambda_{N-1}\cdot\mathcal{L}$. L is verified by a matrix of eigenvectors $\mathcal{X} = [\chi_0, \cdots, \chi_{N-1}]$ such that $\mathcal{L} = \mathcal{X}\Lambda\mathcal{X}^T$where $\mathcal{L}$ is a diagonal matrix of eigenvalues.

Instead of complex exponentials, the eigenvectors, $\left\{\chi_,\right\}_{/=0}^{N-1}$ of Laplacian matrix L that meet perpendicularity criteria are utilized as breakdown bases for graph-structured data is defined as Eq. (12):

$$\widehat{f}(\lambda_{\prime}) = \Sigma_{n=0}^{N-1}\chi_,^T(n)f(n) = \mathcal{X}^Tf \tag{12}$$

Inverse Fourier transformation is shown in Eq. (12):

$$f(n) = \Sigma_{l=0}^{N-1}\widehat{f}(\lambda_{\prime})\chi_{\prime}(n) = \widehat{xf} \tag{13}$$

In the Fourier domain, convolution is converted to a point-wise product, which can then be reconverted to vertex domain utilizing graph Fourier transform as well as convolution theorem, as shown in Eq. (14):

$$f{*}g = \Sigma_{/=0}^{N-1}\widehat{f}(\lambda_/)\grave{g}(\lambda_\zeta)\chi_{\lambda}(n) = \mathcal{X}\left(\left(\mathcal{X}^Tf\right)\cdot\left(\mathcal{X}^Tg\right)\right)$$

$$= \mathcal{X}\,\textit{dıag}\left(\grave{g}(\lambda_0), \cdots, \grave{g}(\lambda_{N-1})\right)\mathcal{X}^Tf \tag{14}$$

The graph convolution process of 2 graph signals f(n) and g(n) is shown in Figure 3, and its transform, g () l, is called a Conv kernel. A set of free parameters θ_{N-1} in Fourier domain, i.e., Laplacian eigenspace is used to build this kernel. It can also be thought of as a function of eigenvalues, written as g(A). Convolution is then written as Eq. (15):

$$\mathbf{f}{*}\mathbf{g} = x d\imath ag\left(\theta_{\mathbf{0}}, \cdots, \theta_{\mathbf{N-1}}\right) \mathcal{X}^{\mathbf{T}}\mathbf{f} = \mathcal{X}\mathcal{G}(\Lambda)\mathcal{X}^{\mathbf{T}}\mathbf{f} \tag{15}$$

The convolution mentioned above on a graph has two drawbacks: (1) Each process involves an Eigen decomposition, which incurs high computational costs; (2) after this operation, the variable value of a vertex is associated with global vertices without considering its locality in space, which is inconsistent with CNNs' local connections.

suggested a low-order polynomial approximation based on rapid localized convolution that depicts g(A) as a polynomial function of eigenvalues Eq. (16):

$$\mathcal{G}(\Lambda) = \boldsymbol{\Sigma}_{\mathbf{k=0}}^{\mathbf{K}} \theta_{\mathbf{k}} \Lambda^{\mathbf{k}} \tag{16}$$

θ_k is the polynomial order, and _k is a vector of polynomial coefficients. The convolution is then rewritten where K is a small positive integer, such as Eq. (17).

$$\begin{aligned} f{*}g &= \mathcal{X}\left(\Sigma_{k=0}^{K}\theta_k\Lambda^k\right)\mathcal{X}^T f = \left(\Sigma_{k=0}^{K}\theta_k\left(\mathcal{X}\Lambda^k\mathcal{X}^T\right)\right) f \\ &= \Sigma_{k=0}^{K}\theta_k\mathcal{L}^k f \end{aligned} \tag{17}$$

The convolution is performed by K multiplications of sparse matrix L, which speeds up computation by avoiding the Eigen decomposition procedure.

Update equation for a layer l is defined as Eq. (18):

$$\begin{aligned} \grave{h}_i^{l+1} &= O_h^l H_{k=1}\left(\Sigma_{j\in N_i} w_{ij}^{k,l} V^{k,l} h_j^l\right) \grave{e}_i^{l+1} = O_e^l H_{k=1} \\ \left(\grave{w}_{i,j}^{k,l}\right) w_{ij}^{k,l} &= \text{softmax}_j\left(\grave{w}_{i,j}^{k,l}\right) \\ w\grave{v}_{i,j}^{k,l} &= \left(\frac{Q^{k,l}h_i^l \cdot K^{k,l}h_j^l}{\sqrt{d_k}}\right) E^{k,l} e_{i,j'}^l \end{aligned} \tag{18}$$

with $Q^{k,l}, K^{k,l}, V^{k,l}, E^{k,l} \in Rd_k, O_{h'}^l, O_e^l \in R^{d\times d}, k \in \{1, 2, \ldots, H\}$ represents the number of attention heads, and where $O_h^l \in R^{d\times d}, V^{k,l} \in R^{d_k \times d}H$ indicates the number of heads, L number of layers, d is the hidden dimension and $d_k$is the dimension of a head d H = dk. Note that $h^{l}{}_{i}$ is ith node's feature at lth layer Eq. (19).

$$\text{cut}\left(S_k, \grave{S}_k\right) = \sum_{v_i \in S_k, v_j \in S_j} e\left(v_i, v_j\right) \tag{19}$$

where S_k is the k_{th} set of a given eigenvector, $\grave{S}_k$indicates residual sets excluding S_k and $e\left(v_i, v_j\right)$ is an edge among vertex v_i and v_j.The cut problem can be rewritten as follows when referring to several sets Eq. (20):

$$\text{cut}\left(S_1, S_2, S_3 \ldots S_g\right) = \frac{1}{2}\sum_{i=k}^{g} \text{cut}\left(S_k, S_k\right) \tag{20}$$

The minimum cut problem is extensively researched in literature, with normalized cut reflecting a separate direction Eq. (21):

$$\text{Ncut}\left(S_1, S_2 \ldots S_g\right) = \sum_{k=1}^{g} \frac{\text{cut}\left(S_k, \grave{S}_k\right)}{\text{vol}\left(S_k, V\right)} \tag{21}$$

wherever $\text{vol}\left(S_k, V\right) = \Sigma_{v_i \in S_k, v_i \in V} e\left(v_i, v_j\right)$ is the entire grade of bulges from S_k in diagram g.

utilizing DL optimization to turn the minimum cut issue into a DL format Eq. (22):

$$\mathbf{L_{cut}} = \sum_{\text{lower sum}} [(\mathbf{Y} \oslash \Gamma)(\mathbf{1}-\mathbf{Y})^{\mathbf{T}}] \bigodot \mathbf{A} + \sum_{\text{lower sum}} \left(\mathbf{1}^{\mathbf{T}}\mathbf{Y} - \frac{\mathbf{n}}{\mathbf{g}}\right)^{\mathbf{2}} \tag{22}$$

The normalized cut is the first term, and Y is defined as an n * g dimension matrix that indicates the neural network's output. Finally, Γ, Y calculates A, which is the adjacency matrix Eq. (23).

$$H_j^{[l+1]} = \sigma\left(\sum_{i=1}^{F_{in}}\left(\sum_{k=0}^{K} \theta_{i,j_k} \mathcal{L}^k H_i^{[l]}\right) + b_j^{[l]}\right) \tag{23}$$

Manifold convolutional and pooling layers, as well as one fully associated layer, make up the model. Figure 4 depicts the model's architecture with two convolutional layers.

Convolutional network: Convolutional layers are the foundation of a convolutional neural network. It has some filters (or kernels) whose settings will be figured out as the training progresses. Typically, its filter's size will be less than that of the image it's applied. Each filter performs a convolution on the image, yielding an activation map. For convolution, the filtration is moved throughout the height & width of the image, and at each point in space, the dot product between each component of the filter & the input is measured. The implemented design with the Deep Graph CNN model can provide better heart disease prediction compared to earlier models. The main features of this design are to give less ToC and accurate diagnosis results compared to earlier models. Heart diseases had been predicted at the classification stage using the GS-CNN process. The shape-based features are more helpful to find the information medical image such that getting differentiation with training data.

Performance analysis

A thorough experimental analysis was used to calculate the suggested hybrid technique performance. The proposed hybrid technique was tested on a PC with the following parameters: Intel(R) Core (TM) i5-7500 CPU, 32-bit Windows 7 OS, 4 GB RAM with SciPy, NumPy, Pandas, Keras and Matplotlib frameworks and Python 2.7 as the programming language.

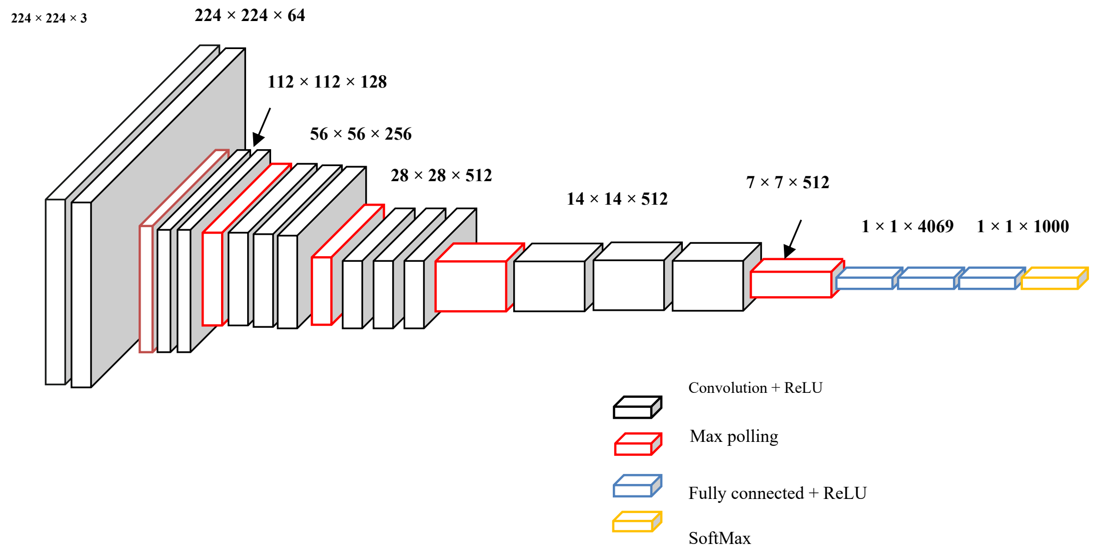


FIGURE 3
Graphical illustration of convolution f (n) and g (n).

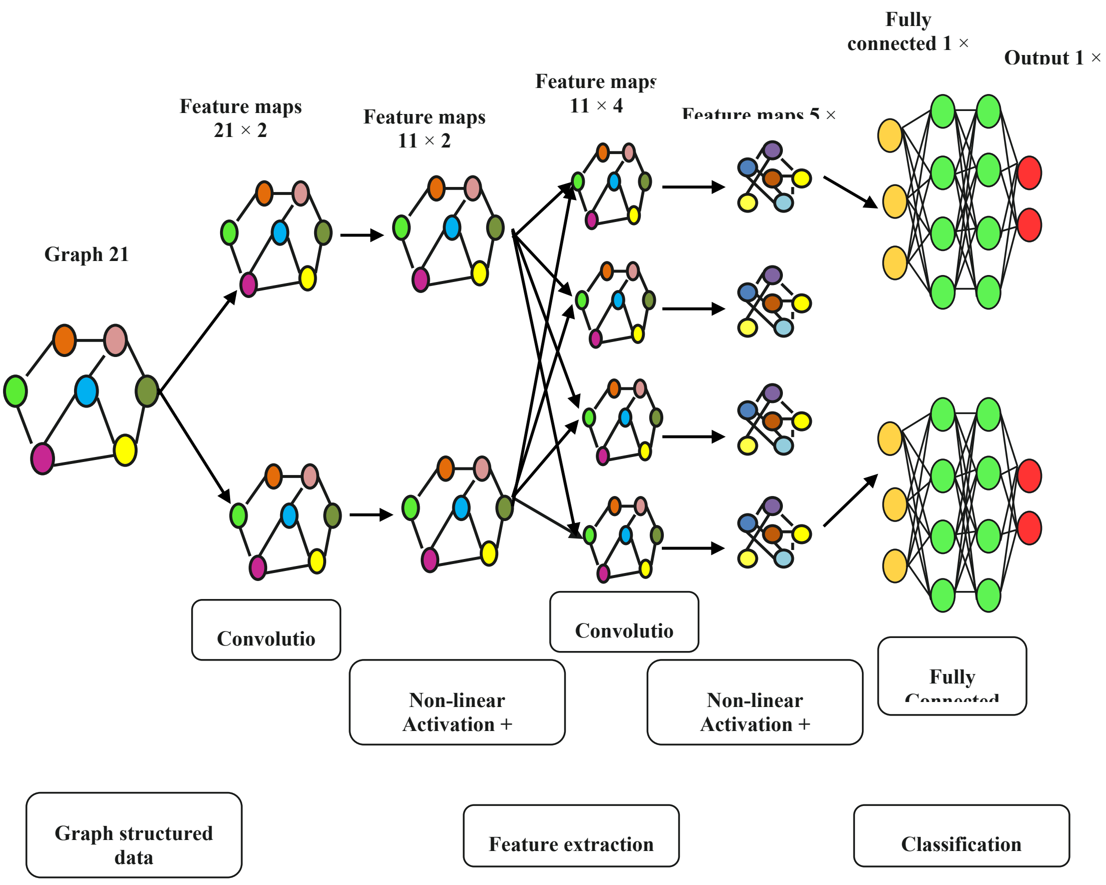


FIGURE 4
Model's architecture with two convolutional layers.

TABLE 2 Comparative analysis of diagnostic accuracy.

Number of epochs	SVM	CNN	FGCNet	K-means_LQDA_ DG_ConvoNet
100	45	52	59	65
200	49	55	63	72
300	52	59	65	79
400	55	61	69	85
500	59	65	72	96

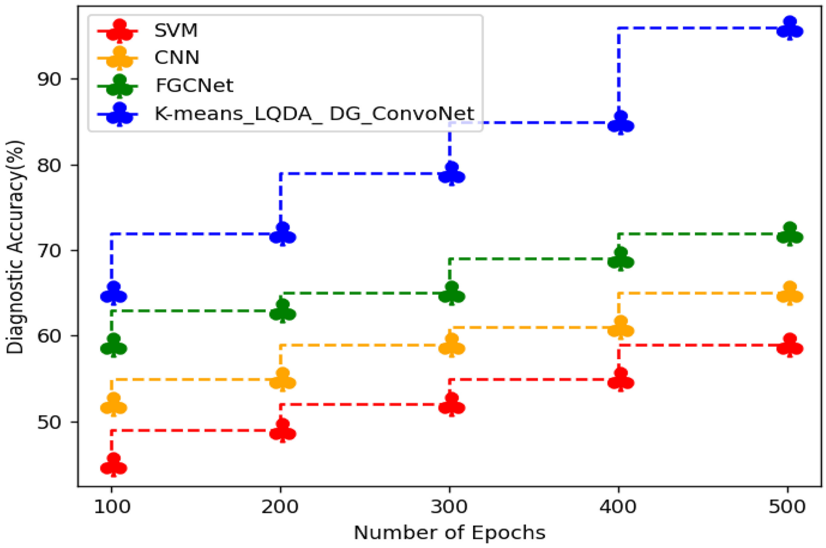


FIGURE 5
Comparative analysis of diagnostic accuracy.

TABLE 3 Comparative analysis of sensitivity.

Number of epochs	SVM	CNN	FGCNet	K-means_LQDA_ DG_ConvoNet
100	52	55	57	60
200	59	61	63	65
300	63	65	66	69
400	65	69	72	75
500	66	70	75	80

Dataset description

Public Health Dataset, which dates from 1988 and consists of four databases: Cleveland, Hungary, Switzerland and Long Beach V, was used for this study. Even though there are 76 qualities in total, including expected attributes, all published studies only utilize a selection of 14 of them.

Information on heart disease

The clinical HD data used in this study came from 303 patients at CCF in Cleveland, Ohio, in the US. Dataset was collected from UCI_MLRepository (Hinton and Salakhutdinov, 2006), part of the Heart Disease Database. There were 75 attributes and a target attribute in each of the 303 clinical situations. The target attribute was an integer ranging from 0 to 4, indicating whether a patient had HD [0] or not [1, 2, 3]. Target qualities for the absence or presence of cardiac disease in patients were ascribed to binary values of 0 and 1 for this study. There were 125 cases with heart disease (44.33%) and 157 cases without heart disease (55.67%) among the 282 total clinical episodes. A total of 76 raw attributes were used to describe each clinical event. Due to missing values among other raw variables, only 29 of the raw attributes were used in the building of DNN models (Djenouri et al., 2022; Mezair et al., 2022).

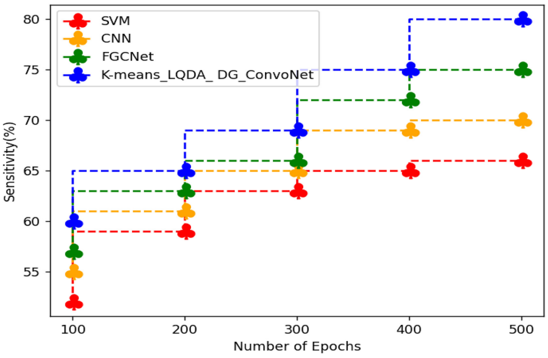


FIGURE 6
Comparative analysis of sensitivity.

TABLE 4 Comparative analysis of specificity.

Number of epochs	SVM	CNN	FGCNet	K-means_LQDA _DG_ConvoNet
100	41	45	51	55
200	45	49	55	59
300	49	51	61	62
400	52	55	63	65
500	55	59	67	73

Table 2 and Figure 5 show comparative analysis in diagnostic accuracy for proposed K-means_LQDA_ DG_ConvoNet. The diagnostic accuracy has been analyzed based on the number of epochs the neural network carries out. The epochs are taken as 100, 200, 300, 400 and 500. For all the iterations of the neural network, the proposed K-means_LQDA_ DG_ConvoNet obtained optimal results than the existing technique. The accuracy obtained in the diagnosis of disease by proposed K-means_LQDA_ DG_ConvoNet is 96% and existing SVM achieved 59% for 500 epochs and CNN obtained 65%, FGCNet attained 72%.

Table 3 and Figure 6 show comparative sensitivity analysis for proposed K-means_LQDA_ DG_ConvoNet. The sensitivity calculation refers prediction of the true positive and false positive rate of the proposed technique in diagnosing heart disease. The sensitivity obtained in disease diagnosis by proposed K-means_LQDA_ DG_ConvoNet is 80% for 500 epochs and existing SVM achieved 66% for 500 epochs and CNN obtained 70%, FGCNet attained 75%.

Table 4 and Figure 7 show comparative analysis in terms of specificity for proposed K-means_LQDA_ DG_ConvoNet. The specificity calculation relates to the percentage of real negatives projected as negatives. This means that a part of true negatives is forecasted as positives, which is denoted as false positives in the suggested method for identifying HD. The specificity obtained in the diagnosis of disease by proposed K-means_LQDA_ DG_ConvoNet is 73% for 500 epochs and existing SVM achieved 55% for 500 epochs and CNN obtained 57%, FGCNet attained 67%.

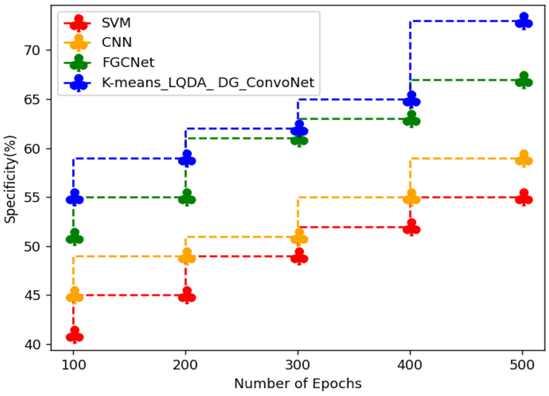


FIGURE 7
Comparative analysis of specificity.

TABLE 5 Comparative analysis of precision.

Number of epochs	SVM	CNN	FGCNet	K-means_LQDA _DG_ConvoNet
100	55	59	63	76
200	59	63	67	79
300	62	66	71	82
400	65	69	75	85
500	71	73	79	90

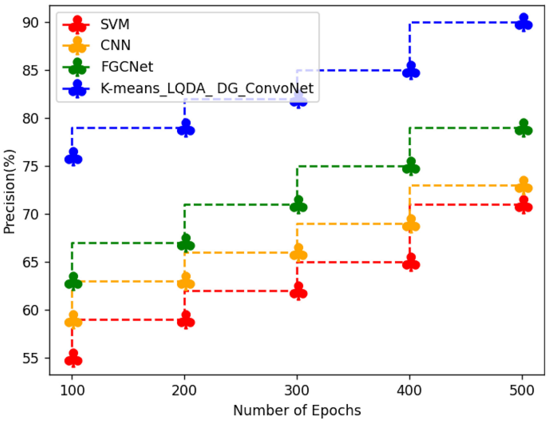


FIGURE 8
Precision analysis differentiation.

Table 5 and Figure 8 show qualified examination in terms of Precision for proposed K-means_LQDA_ DG_ConvoNet. The precision calculation mentions the number of true positives separated by the whole number of positive calculations made by the suggested technique in diagnosing heart disease, as well as the superiority of a positive forecast made

TABLE 6 Comparative analysis of *F*-Score.

Number of epochs	SVM	CNN	FGCNet	K-means_LQDA_ DG_ConvoNet
100	51	55	59	63
200	56	61	65	66
300	59	63	69	71
400	63	66	72	75
500	65	71	79	79

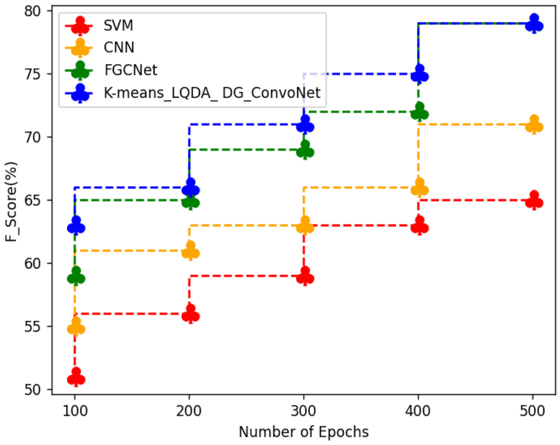


FIGURE 9
Comparative analysis of F-Score.

TABLE 7 ROC curve on various methods.

Number of epochs	SVM	CNN	FGCNet	K-means_LQDA_ DG_ConvoNet
100	31	36	45	61
200	35	39	49	65
300	39	42	53	69
400	42	49	56	72
500	45	53	62	75

by the proposed technique. The precision obtained in the diagnosis of disease by proposed K-means_LQDA_ DG_ConvoNet is 90% for 500 epochs and existing SVM achieved 71% for 500 epochs and CNN obtained 73%, FGCNet attained 79%.

Table 6 and Figure 9 show a comparative analysis in terms of F-Score for proposed K-means_LQDA_ DG_ConvoNet. The F-Score computation is utilized to assess binary classification techniques which categorize examples as "positive" or "negative." *F*-score is shown as the harmonic mean of precision and recall. For example, *F*-Score obtained in the diagnosis of disease by proposed K-means_LQDA_ DG_ConvoNet is 79% for 500 epochs and existing SVM achieved 65% for 500 epochs and CNN obtained 71%, FGCNet attained 79%.

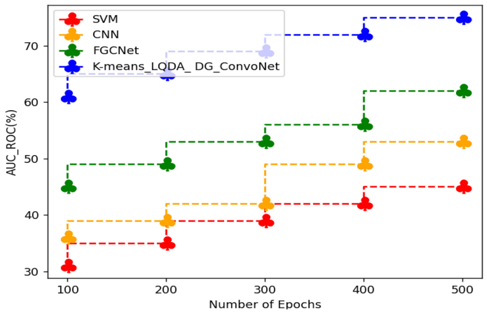


FIGURE 10
ROC curve analysis.

Table 7 and Figure 10 show an examination of the area under the ROC curve for proposed K-means_LQDA_ DG_ConvoNet. The calculation of the extent under the ROC curve is a measure of a classifier's ability to distinguish between classes as well as used as an instant of the ROC curve.

AUC indicates how well the method differentiates between positive and negative classes. F-Score

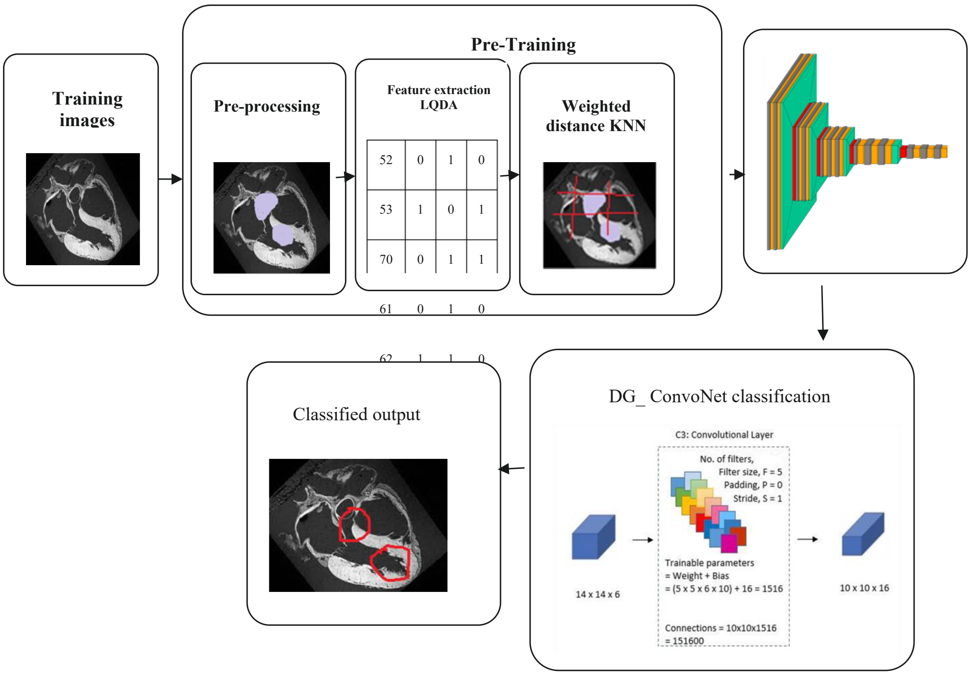


FIGURE 11
Classification of heart disease prediction.

obtained in the diagnosis of disease by proposed K-means_LQDA_ DG_ConvoNet is 75% for 500 epochs and existing SVM achieved 45% for 500 epochs and CNN obtained 53%, FGCNet attained 62% shown in **Figure 11**.

Conclusion

The proposed work is a novel technique for detecting heart disease based on IoT sensor data with a monitoring application using deep learning architectures. Here, the input data has been collected from IoT sensor data from the University of California Irvine machine learning repository. The collected data has been processed for noise removal and clustered based on K-means clustering. The clustered data has been extracted using Linear Quadratic Discriminant Analysis where the features of clustered data have been extracted. The extracted features have been classified using the deep graph ConvoNet (convolutional network)- DG_ConvoNet. The diagnostic accuracy of 96%, sensitivity of 80%, specificity of 73%, precision of 90%, *F*-Score of 79%, and area under the ROC curve of 75% are obtained by the proposed classification and prediction model, according to the testing findings. Our strong results clearly show the strength of our methodology and DG_ConvoNet. In the future, we wish to test our system model on other datasets and also look at implementing the DG_ConvoNet for other diseases.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here at doi: 10.1136/bmjopen-2020-044070.

Author contributions

KS and VR contributed to the conception and design of the study. GS performed the statistical analysis. KS and JL wrote the first draft of the manuscript. All authors contributed to the manuscript revision, read, and approved the submitted version.

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

Ali, F., Hasan, B., Ahmad, H., Hoodbhoy, Z., Bhuriwala, Z., Hanif, M., et al. (2021). Protocol: Detection of subclinical rheumatic heart disease in children using a deep learning algorithm on digital stethoscope: A study protocol. *BMJ Open* 11:e044070. doi: 10.1136/bmjopen-2020-044070

Divya, K., Sirohi, A., Pande, S., and Malik, R. (2021). "An IoMT assisted heart disease diagnostic system using machine learning techniques," in *Cognitive internet of medical 4ings for smart healthcare*, Vol. 311, eds A. E. Hassanien, A. Khamparia, D. Gupta, K. Shankar, and A. Slowik (Cham: Springer), 145–161. doi: 10.1007/978-3-030-55833-8_9

Djenouri, Y., Belhadi, A., Srivastava, G., and Lin, J. C. (2022). When explainable AI meets IoT applications for supervised learning. *Cluster Comput.* 17:1. doi: 10.1007/s10586-022-03659-3

Garigipati, R. K., Raghu, K., and Saikumar, K. (2022). "Detection and identification of employee attrition using a machine learning algorithm," in *Handbook of research on technologies and systems for E-collaboration during global crises*, eds J. Zhao and V. Vinoth (Pennsylvania, PA: IGI Global), 120–131. doi: 10.4018/978-1-7998-9640-1.ch009

Golande, A., Sorte, P., Suryawanshi, V., Yermalkar, U., and Satpute, S. (2019). Smart hospital for heart disease prediction using IoT. *Int. J. Inform. Vis.* 3, 198–202.

Haq, A. U., Li, J. P., Memon, M. H., Nazir, S., and Sun, R. (2018). A hybrid intelligent system framework for the prediction of heart disease using machine learning algorithms. *Mob. Inf. Syst.* 2018:3860146. doi: 10.1155/2018/3860146

Hasan, N. I., and Bhattacharjee, A. (2019). Deep learning approach to cardiovascular disease classification employing modified ECG signal from empirical mode decomposition. *Biomed. Signal Process. Control* 52, 128–140. doi: 10.1016/j.bspc.2019.04.005

Hinton, G. E., and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science* 313, 504–507. doi: 10.1126/science.1127647

Huang, J., Chen, B., Yao, B., and He, W. (2019). ECG arrhythmia classification using STFT-based spectrogram and convolutional neural network. *IEEE Access* 7, 92871–92880. doi: 10.1109/ACCESS.2019.2928017

Kanksha, Aman, B., Sagar, P., Rahul, M., and Aditya, K. (2021). An intelligent unsupervised technique for fraud detection in health care systems. *Intell. Decis. Technol.* 15, 127–139. doi: 10.3233/IDT-200052

Koppula, N., Sarada, K., Patel, I., Aamani, R., and Saikumar, K. (2021). "Identification and recognition of speaker voice using a neural network-based algorithm: Deep learning," in *Handbook of research on innovations and applications of AI, IoT, and cognitive technologies*, eds J. Zhao and V. Vinoth Kumar (Pennsylvania, PA: IGI Global), 278–289. doi: 10.4018/978-1-7998-6870-5.ch019

Ladefoged, C. N., Hasbak, P., Hornnes, C., Højgaard, L., and Andersen, F. L. (2021). Low-dose PET image noise reduction using deep learning: Application to cardiac viability FDG imaging in patients with ischemic heart disease. *Phys. Med. Biol.* 66:054003. doi: 10.1088/1361-6560/abe225

Liu, J., Wang, H., Yang, Z., Quan, J., Liu, L., and Tian, J. (2022). Deep learning-based computer-aided heart sound analysis in children with left-to-right shunt congenital heart disease. *Int. J. Cardiol.* 348, 58–64. doi: 10.1016/j.ijcard.2021.12.012

Majumder, A. K. M., ElSaadany, Y. A., Young, R., and Ucci, D. R. (2019). An energy efficient wearable smart IoT system to predict cardiac arrest. *Adv. Hum.Comput. Interact.* 2019:1507465. doi: 10.1155/2019/1507465

Manogaran, G., Lopez, D., Thota, C., Abbas, K. M., Pyne, S., and Sundarasekar, R. (2017). "Big data analytics in healthcare internet of things," in *Innovative healthcare systems for the 21st century*, ed. H. Qudrat-Ullah (Cham: Springer), 263–284. doi: 10.1007/978-3-319-55774-8_10

Martins, J. F. B., Nascimento, E. R., Nascimento, B. R., Sable, C. A., Beaton, A. Z., Ribeiro, A. L., et al. (2021). Towards automatic diagnosis of rheumatic heart disease on echocardiographic exams through video-based deep learning. *J. Am. Med. Inform.Assoc.* 28, 1834–1842. doi: 10.1093/jamia/ocab061

Mezair, T., Djenouri, Y., Belhadi, A., Srivastava, G., and Lin, J. C. (2022). Towards an advanced deep learning for the internet of behaviors: Application to connected vehicle. *ACM Trans. Sens. Netw.* 1–18. doi: 10.1145/3526192

Morris, S. A., and Lopez, K. N. (2021). Deep learning for detecting congenital heart disease in the fetus. *Nat. Med.* 27, 764–765. doi: 10.1038/s41591-021-01354-1

Rahmani, A. M., Gia, T. N., Negash, B., Anzanpour, A., Azimi, I., Jiang, M., et al. (2018). Exploiting smart e-health gateways at the edge of healthcare internet-of-yhings: A fog computing approach. *Future Gener. Comput. Syst.* 78, 641–658. doi: 10.1016/j.future.2017.02.014

Rath, A., Mishra, D., Panda, G., and Satapathy, S. C. (2021). Heart disease detection using deep learning methods from imbalanced ECG samples. *Biomed. Signal Process. Control* 68:102820. doi: 10.1016/j.bspc.2021.102820

Saikumar, K., and Rajesh, V. (2020a). A novel implementation heart diagnosis system based on random forest machine learning technique. *Int. J. Pharm. Res.* 12, 3904–3916. doi: 10.31838/ijpr/2020.SP2.482

Saikumar, K., and Rajesh, V. (2020b). Coronary blockage of artery for heart diagnosis with DT Artificial Intelligence Algorithm. *Int. J. Res. Pharma. Sci.* 11, 471–479. doi: 10.26452/ijrps.v11i1.1844

Saikumar, K., Rajesh, V., and Babu, B. S. (2022). Heart disease detection based on feature fusion technique with augmented classification using deep learning technology. *Trait. Signal* 39, 31–42. doi: 10.18280/ts.390104

Vincent Paul, S. M., Balasubramaniam, S., Panchatcharam, P., Malarvizhi Kumar, P., and Mubarakali, A. (2021). Intelligent framework for prediction of heart disease using deep learning. *Arab. J. Sci. Eng.* 47, 2159–2169. doi: 10.1007/s13369-021-06058-9

Wang, H., Shi, H., Chen, X., Zhao, L., Huang, Y., and Liu, C. (2020). An improved convolutional neural network based approach for automated heartbeat classification. *J. Med. Syst.* 44:35. doi: 10.1007/s10916-019-1511-2

Wang, S. H., Govindaraj, V. V., Gorriz, J. M., Zhang, X., and Zhang, Y. D. (2021a). Covid-19 classification by FGCNet with deep feature fusion from graph convolutional network and convolutional neural network. *Inf. Fusion* 67, 208–229. doi: 10.1016/j.inffus.2020.10.004

Wang, S. H., Nayak, D. R., Guttery, D. S., Zhang, X., and Zhang, Y. D. (2021b). COVID-19 classification by CCSHNet with deep fusion using transfer learning and discriminant correlation analysis. *Inf. Fusion* 68, 131–148. doi: 10.1016/j.inffus.2020.11.005

Zhang, P., and Xu, F. (2021). Effect of AI deep learning techniques on possible complications and clinical nursing quality of patients with coronary heart disease. *Food Sci. Technol.* 42, 1–6. doi: 10.1590/fst.42020

Zhang, Y. D., Dong, Z., Chen, X., Jia, W., Du, S., Muhammad, K., et al. (2019). Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation, multimed. *Tools Appl.* 78, 3613–3632. doi: 10.1007/s11042-017-5243-3

Zhang, Y. D., Dong, Z., Wang, S. H., Yu, X., Yao, X., Zhou, Q., et al. (2020a). Advances in multimodal data fusion in neuroimaging: Overview, challenges, and novel orientation. *Inf. Fusion* 64, 149–187. doi: 10.1016/j.inffus.2020.07.006

Zhang, Y. D., Nayak, D. R., Zhang, X., and Wang, S. H. (2020b). Diagnosis of secondary pulmonary tuberculosis by an eight-layer improved convolutional neural network with stochastic pooling and hyperparameter optimization. *J. Ambient Intell. Humaniz. Comput.* 1, 1–18. doi: 10.1007/s12652-020-02612-9

TYPE Original Research
PUBLISHED 16 November 2022
DOI 10.3389/fncom.2022.1016516

Check for updates

OPEN ACCESS

EDITED BY
Dan Chen,
Wuhan University, China

REVIEWED BY
B. Nataraj,
Sri Ramakrishna Engineering College, India
Mario Versaci,
Mediterranea University of Reggio Calabria, Italy

*CORRESPONDENCE
Dong-Ok Won
dongok.won@hallym.ac.kr

†These authors have contributed equally to this work

RECEIVED 11 August 2022
ACCEPTED 10 October 2022
PUBLISHED 16 November 2022

CITATION
Prabhakar SK, Ju Y-G, Rajaguru H and Won D-O (2022) Sparse measures with swarm-based pliable hidden Markov model and deep learning for EEG classification.
Front. Comput. Neurosci. 16:1016516.
doi: 10.3389/fncom.2022.1016516

Sparse measures with swarm-based pliable hidden Markov model and deep learning for EEG classification

Sunil Kumar Prabhakar[1†], Young-Gi Ju[1†], Harikumar Rajaguru[2†] and Dong-Ok Won[1*†]

[1]Department of Artificial Intelligence Convergence, Hallym University, Chuncheon, South Korea, [2]Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Sathyamangalam, India

In comparison to other biomedical signals, electroencephalography (EEG) signals are quite complex in nature, so it requires a versatile model for feature extraction and classification. The structural information that prevails in the originally featured matrix is usually lost when dealing with standard feature extraction and conventional classification techniques. The main intention of this work is to propose a very novel and versatile approach for EEG signal modeling and classification. In this work, a sparse representation model along with the analysis of sparseness measures is done initially for the EEG signals and then a novel convergence of utilizing these sparse representation measures with Swarm Intelligence (SI) techniques based Hidden Markov Model (HMM) is utilized for the classification. The SI techniques utilized to compute the hidden states of the HMM are Particle Swarm Optimization (PSO), Differential Evolution (DE), Whale Optimization Algorithm (WOA), and Backtracking Search Algorithm (BSA), thereby making the HMM more pliable. Later, a deep learning methodology with the help of Convolutional Neural Network (CNN) was also developed with it and the results are compared to the standard pattern recognition classifiers. To validate the efficacy of the proposed methodology, a comprehensive experimental analysis is done over publicly available EEG datasets. The method is supported by strong statistical tests and theoretical analysis and results show that when sparse representation is implemented with deep learning, the highest classification accuracy of 98.94% is obtained and when sparse representation is implemented with SI-based HMM method, a high classification accuracy of 95.70% is obtained.

KEYWORDS

EEG, sparse representation, hidden Markov model, swarm intelligence, deep learning

Introduction

In order to capture the activity of the brain, electroencephalography (EEG) signals are used which are nothing but the electrophysiological recordings of electrical potentials across the cortical regions of the brain (Lee et al., 2018). The spontaneous electrical activity of the brain in a very short span of time is thus measured by EEG. For analyzing various neurological-related disorders, such as coma, anesthesia, epilepsy, sleep disorders, schizophrenia, alcoholism, brain death, and encephalopathies, EEGs are widely used (Chen et al., 2016). During earlier times, the analysis was based only on visual inspection and interpretation that lead to more errors and also it required extensive training by the clinicians. With the advent of both specialized data acquisition devices and computer technology, identifying abnormalities have been incorporated very successfully (Lee et al., 2019). As EEG signals are extremely complex when compared to other biomedical signals, specialized and versatile feature extraction and selection methods incorporated with classification techniques have to be utilized. In this process, the selection of the most important features is highly useful and significant as it depicts the subsets of discriminant patterns (Won et al., 2018). Once that is achieved, the classification accuracy can be enhanced, the curse of the dimensionality problem can be alleviated, and thus the generalization capability of the system enhances gradually (Lee et al., 2015). This kind of methodology is adopted in a typical biomedical signal processing work and in this work since epilepsy classification and schizophrenia classification from EEG signals are discussed, a few important and relevant past literature in recent years is discussed as follows.

Plenty of articles are available online for epilepsy classification as it is a well-established research field nearly for the past two decades, and only a few articles are available online for schizophrenia classification as it has triggered interest among researchers very recently. A comprehensive review of the different machine learning techniques for epilepsy classification was reported in Sharmila and Geethanjali (2019), and the latest deep learning techniques utilized for epilepsy classification from EEG signals were analyzed thoroughly in Shoeibi et al. (2007). These two survey articles published in 2019 and 2020 review all the past works, working methodologies, statistical feature analysis techniques used, and datasets analyzed along with the comparison of classification accuracies obtained by every method, thereby easing the work of other researchers to not reproduce the past literature over and over again. However, some prominent ideas reported in high-quality literature during 2020 and 2021 for both epilepsy classification and schizophrenia classification are discussed as follows. An automated classification of epilepsy from EEG signals based on spectrogram and CNN was utilized in Mandhouj et al. (2021) reporting a classification accuracy of 98.25%. By means of integrating the property of convolutions with Support Vector Machine (SVM), a hybrid methodology called as Convolution SVM (C-SVM) was developed in Xin et al. (2021) reporting a classification accuracy of 99.56%. The optimal wavelet features were selected and combined with Long-Short Term Memory (LSTM) for epilepsy classification from EEG signals reporting a classification accuracy of 99% (Aliyu and Lim, 2021). Based on Jacobi polynomial transforms and Least Squares SVM, the classification of epilepsy was done in Nkengfack et al. (2021), reporting a classification accuracy ranging from 88.75 to 100%. The concept of synchrosqueezing transforms was utilized with standard machine learning techniques reporting a classification accuracy of 95.1% (Cura and Akan, 2021). A deep neural network model based on CNN is utilized for the analysis of robust detection of epileptic seizures from EEG signals reporting classification accuracy in the ranges of 97.63–99.52% (Zhao et al., 2020). A deep CNN with 10-fold cross-validation methodology was also implemented for epilepsy classification reporting a high classification accuracy of 98.67% (Abiyev et al., 2020). Other works discussed in this study are for the sake of comparing the proposed results with the previous works as the results implemented in this work were done with those same datasets. Different approaches for epilepsy classification included the usage of genetic programming (Bhardwaj et al., 2016), complex-valued classifiers (Peker et al., 2016), Empirical Mode Decomposition (EMD) based supervised learning (Riaz et al., 2016), weighted complex networks analysis (Diykh et al., 2017), Support Vector Machine (SVM) based automated seizure analysis (Zhang and Chen, 2017), and Recurrent Elman neural network classifier (Raghu et al., 2017) are some of the prominent works in this field of epilepsy classification. Recent approaches utilized for epilepsy classification in the past three years involve the usage of deep learning by means of proposing a Pyramidal 1D-CNN (Ullah et al., 2018), Continuous Wavelet Transforms with CNN (Turk and Ozerdem, 2019), and a simple normalization with a 1D-CNN (Zhao et al., 2020). Entropy-based analysis included the usage of fuzzy entropy and distribution entropy for seizure classification (Li et al., 2018) and a Fourier–Bessel series expansion-based rhythms splitting depending on weighted multiscale Renyi Permutation Entropy for epilepsy classification (Gupta and Ram, 2019). Other approaches incorporated are the usage of orthogonal wavelet filtering methodology (Sharma et al., 2018), matrix determinant approach (Raghu et al., 2019), and alpha band statistical feature-based detection of epileptic seizures (Sameer and Gupta, 2020). All these recent previous literature works are done on different epileptic datasets depending on their classification problem requirement, with some researchers focusing only on a single epileptic dataset while other authors concentrate on multiple epileptic datasets. When it comes to schizophrenia classification, many research results reported in high-quality literature are not available, and therefore, a selected few ones are presented in this study to get a clear understanding. An interesting methodology of schizophrenia classification from

EEG was reported in Prabhakar et al. (2020a), where using three different features such as isometric mapping features, nonlinear regression features, and expectation maximization based principal component features was optimized using nature-inspired algorithms and classified with Modest Adaboost classifier reporting a classification accuracy of 98.77%. Another methodology for schizophrenia classification from EEG utilizes the standard statistical features such as Hurst exponent, Sample Entropy, and Detrend Fluctuation Analysis (DFA) with four kinds of optimization techniques, and finally, when it was classified with SVM, a classification accuracy of 92.17% was reported (Prabhakar et al., 2020b). Finally, a deep learning methodology was also involved using a 11-layer CNN for schizophrenia classification in Oh et al. (2019) and they reported a classification accuracy of 81.26% for subjects-based testing and 98.51% for non-subject-based testing. All the works proposed in the literature have its own merits and demerits, and consistent improvement is being made by researchers constantly with the usage of new ideas and methods so that the performance is improved.

In recent years, the sparse representation of the signals has received huge attention (Schoellkopf et al., 2007). The most compact signal representation is solved by a sparse theory that models a signal in the context of the linear combination of atoms in an overcomplete dictionary. The signals when represented in both multi-scale and multi-orientation aspects such as contourlet, ridgelet, wavelet, and curvelet transforms play an important role in the progress of research on sparse representation. For efficient signal modeling, a better performance is provided by sparse representation when compared to techniques based on direct time domain processing. On three different aspects of the sparse representation, the focus of sparse representation research is usually concerned, (a) pursuit techniques for solving the optimization problems, (b) dictionary design techniques, and (c) application of sparse representation for various tasks (Schoellkopf et al., 2007). The primary objective in the standard theory of sparse representation is to mitigate the signal reconstruction errors utilizing a very few number of atoms. In literature too, the application of sparse representation for modeling and classification has been well explored. Sparse representation for signal classification (Schoellkopf et al., 2007) and EEG classification based on sparse representation with deep learning (Gao et al., 2018) are the two most important applications of sparse concepts in biomedical signal processing. A widely utilized generative model is HMM which usually deals with sequential data and it assumes that based on a specific state of hidden Markov chain, the conditioning of every observation is done (Rezek and Roberts, 2002). It is a very famous probabilistic model where the general assumption is that a signal is generated by means of the utilization of a double-embedded stochastic process. For analyzing sequential data, HMMs are highly useful as the dynamics of the signal is encoded by a discrete-time hidden state process which projects as a Markov chain. At each instant of time, the appearance of the signal is encoded by an observation process and it is conditioned on the present state. For biomedical signal analysis especially the EEG, HMMs are highly useful and a few applications utilizing them for various aspects of EEG signal processing are ensemble HMM for analyzing EEG, parallel HMM to classify the multichannel EEG patterns, detection of various brain diseases from EEG signals using HMM and an obstructive sleep apnea detection approach using a discriminative HMM from EEG (Eberhart et al., 2001). Swarm Intelligence combined with HMMs serves as a good combination and has been successfully implemented in our work.

The main contributions of this work are as follows:

a) An efficient sparse representation model with sparseness measures analysis with the usage of Analysis Dictionary Learning Algorithm (ALDA) for the biosignal datasets has been implemented and no literature in the past have reported it for epileptic EEG signal classification and schizophrenia EEG signals classification.
b) A swarm intelligence–based pliable HMM has been developed and incorporated in this study and it is the first of its kind to do after the sparse representation analysis is done, as no literature in the past has proceeded in this methodology.
c) The sparse-modeled features are also classified with deep learning methodology using CNN and other traditional pattern recognition techniques for providing a comprehensive analysis.
d) Overall, the amalgamation of these techniques in this proposed kind of methodology is totally new and it can be successfully implemented in other biosignal processing datasets, imaging applications, speech signal processing, financial risk level assessment classification, biometrics, etc.

In this work, sparse modeling is implemented with HMM ideology controlled by SI techniques and it is the first of its kind to adopt this methodology for biosignal processing datasets, making the system more versatile and adaptable. The organization of the work is as follows. The simplified block diagram of the work for an easy understanding is projected in **Figure 1**. Section "Sparse representation model" explains the sparse representation model of the EEG signals. Section "Hidden Markov model analysis" explains the modeling of HMM followed by the usage of swarm intelligence techniques and the incorporation of the deep learning methodology is explained in section "Deep learning–based methodology." The results and discussion with experimentation and dataset details are projected in section "Results and discussion" and conclusion in section "Conclusion and future work."

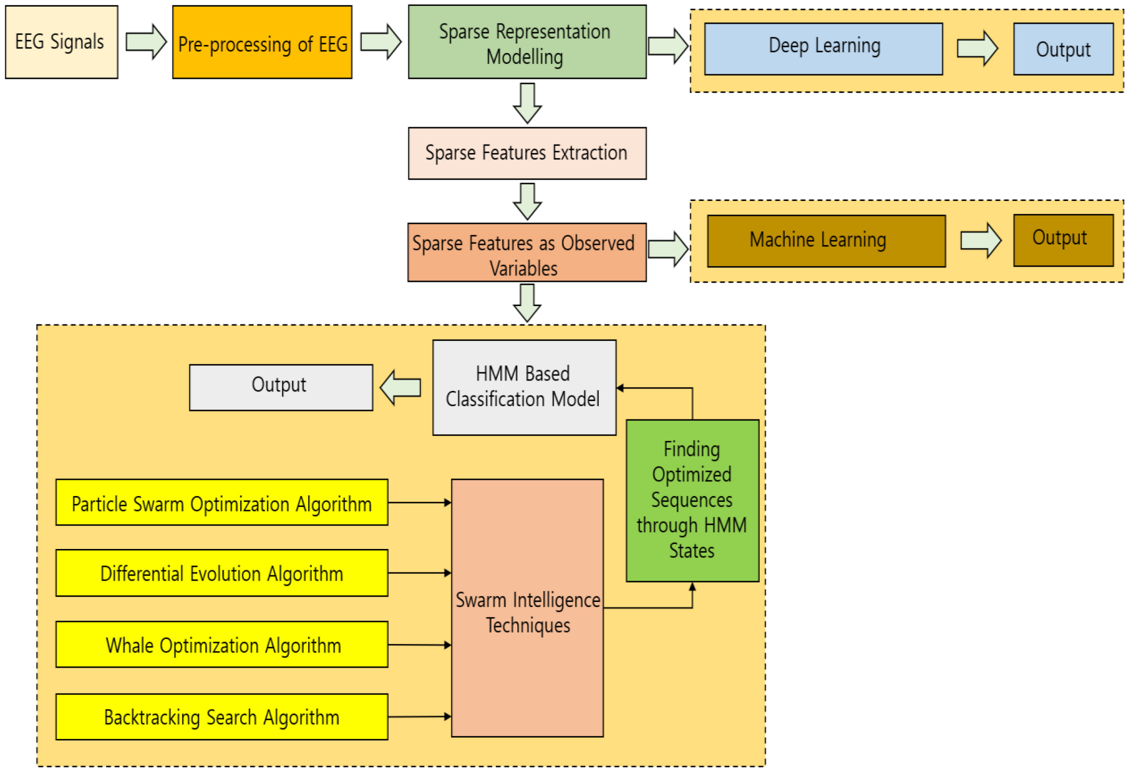

EEG Signals
Pre-processing of EEG
Sparse Representation Modelling
Deep Learning
Output
Sparse Features Extraction
Sparse Features as Observed Variables
Machine Learning
Output
Output
HMM Based Classification Model
Finding Optimized Sequences through HMM States
Particle Swarm Optimization Algorithm
Differential Evolution Algorithm
Whale Optimization Algorithm
Backtracking Search Algorithm
Swarm Intelligence Techniques

FIGURE 1
A simplified block diagram of the work for easy understanding.

Sparse representation model

The notations utilized in analyzing the sparse representation concept are explained as follows. An upper case alphabet Z denotes a matrix and lower case letter z_{ij} expresses the ij^{th} entry of Z. A vector is defined by the lower-case letter, such as z. The j^{th} entry of z is expressed as z_j. The i^{th} row and j^{th} column of a particular matrix Z is defined by the matrix slices $Z_{i:}$ and $Z_{:j}$, respectively. For a matrix Z, the Frobenius norm is expressed as $||Z||_F = \left(\Sigma_{i,j} \left|z_{ij}\right|^2\right)^{1/2}$. To indicate the determinant value of a specific matrix, det (•) is utilized.

Sparse signal representation

With the help of sparse representation, the observed signals are decomposed into a unique product of a dictionary matrix which will have the signal base and along with it a sparse coefficient matrix will also be present (Schoellkopf et al., 2007). A synthesis model and analysis model are the two various structures of the sparse representation model. The firstly initiated sparse model is the synthesis model and it is very widely utilized. Assuming that the modeling of signals to be done as $Z \in \Re^{p\times N}$, where the signal dimensionality is represented as p and the total number of measurements are represented as N. The signals could be expressed in the synthesis sparse model as

$$Z = DG \tag{1}$$

$$Z \approx DG \tag{2}$$

$$\text{such that } ||\text{Z-DG}||_\text{F}^2 \leq \varepsilon, \tag{3}$$

where $D \in \Re^{p\times n}$ is considered as a dictionary, $G \in \Re^{n\times N}$ denotes a representation co-efficient matrix, and a very small residual factor is given by $\varepsilon \geq 0$. The number of bases is represented by $'n'$ and it is termed as dictionary atoms. To obtain the sparse representation of the signals, it is assumed that from the dictionary matrix D, the representation matrix G is sparse in nature (i.e., numerous zero entities). From equations (1) and (2), it implies that the representation of every signal is done as a linear combination of a few atoms.

The choice of solution for the dictionary is the most important key issue of the sparse representation which the discovered signals are utilized to decompose. The famous choices are either a pre-defined dictionary such as wavelets, Discrete Fourier Transform (DFT), and Discrete Cosine Transform (DCT) or a learned dictionary which results to match the contents of the signals in a better manner (Schoellkopf et al., 2007). In real-world applications, a better performance is exhibited by the learned dictionary when compared to the pre-defined dictionaries. The analysis model is a simple and interesting twin of the synthesis model and it should be considered important. Supposing that there is a matrix $\Omega \in \Re^{n \times p}$ that gives a sparse coefficient matrix $_G$ by means of being multiplied by the signal matrix $G = \Omega Z$.

For the error function $||G - \Omega Z||_F$, there is a minimization problem and the equation $G = \Omega Z$ can be utilized as a solution to it. The standardized optimization methods can be very well deployed in this study as the error function is convex. To perform optimization in the analysis model is very easy as the error function present in the synthesis model is non-convex in nature. Now the analysis dictionary is represented as $\Omega \in \Re^{n \times p}$. In the analysis dictionary Ω, the atoms are considered as its rows rather than the consideration of atoms as columns in the synthesis dictionary D. In order to assemble a sparse result, the dictionary analyses the signal and so the term "analysis" is used. To clearly distinguish and stress the importance between analysis and synthesis models, a co-sparsity has been utilized (Gao et al., 2018), which helps in counting the number of zero-valued elements of ΩZ, which is nothing but the zero elements co-produced by Ω and Z. Therefore, the cosparse model can also be used instead of sparse model, and cosparse dictionary can also be used instead of analysis dictionary.

Now analysis sparse model is examined more carefully. The analysis model represented for one signal $z \in \Re^p$, which is a column in the signal matrix Z is now indicated utilizing an acceptable analysis dictionary $\Omega \in \Re^{n \times p}$. The i^{th} row termed as the i^{th} atom in Ω is specified by q_i. Now the analysis representation vectors $g = \Omega z$ should be made sparse and it is done by means of introducing a sparse measure $M(g)$, so that the behavior becomes negatively influenced by the sparsity nature of g and therefore by mitigating $M\left(g\right)$, it gives the sparsest solution represented as

$$\begin{aligned} \Omega &= \arg\min_{\Omega} M(g) \\ s.t \;\; g &= \Omega z \end{aligned} \tag{4}$$

By utilizing l_0 norm thoroughly by means of setting $M(g) = \left|\left|g\right|\right|_0$, the sparsest solution is obtained. Such a constraint leads to often NP hard problem and the optimization problem becomes combinatorial. To have easier optimization problems, the other sparsity measures such as the l_1 norm are utilized. It is also known that utilizing l_1 norm can lead to the solution becoming too sparse as it often over-penalizes large elements.

Sparseness measures analysis

For estimation and appraisal of the sparseness of a vector, the l_p norms are highly useful and are popularly used, where $p = 0, 1,$ or 2. An NP-hard problem is often yielded by the l_0 norm, and therefore, l_1 norm has its convex evaluation utilized often (Gao et al., 2018). For a vector g, the $_{l_1}$ -norm is expressed to be the total sum of the absolute values of g; i.e., $\left|\left|g\right|\right|_1 = \Sigma_i \left|g_i\right|$.

For non-negative vectors, $g \in \Re_+$, the l_1 -norm of g is expressed as $\left|\left|g\right|\right|_1 = \Sigma_i g_i$. The l_1 -norm is usually smooth and differentiable for non-negative vectors and therefore such gradient techniques are utilized in optimization. The introduction of l_2 -norm with non-negative matrix factorization is sometimes considered as its yields sparse solutions. The results with l_0 -norm or l_1 -norm are more sparser than the results with l_2 norm. The instantaneous sparsity nature of only one signal can be expressed by the sparsity measures mentioned above and are generally not utilized for covering and evaluating the sparsity across various sources of measurement.

For non-negative sources, a determinant type of sparsity measure is employed to express the joint sparseness. The sparseness of non-negative matrices can be explicitly measured by the determinant-sparse type and measures as it has various good qualities. The determinant value of a non-negative matrix is well bounded if the normalization of a non-negative matrix is done, thereby interpolating its value between two extremes 0 and 1, and thus enhancing the sparsity. Supposing if the non-negative matrix Y is non-sparse, then the determinant of YY^T, $\det(YY^T)$ addresses toward 1. If all the entries of YY^T are similar, then the determinant value acts in a manner such that $0 \leq \det\left(YY^T\right) \leq 1$, where $\det\left(YY^T\right) = 0$. The following two conditions are fully complacent at the time when $\det\left(YY^T\right) = 1$ and are mentioned as follows:

(i) For all $i \in \left\{1, 2, ..., p\right\}$, only a single element in y_i is non-zero

(ii) For all $i, j \in \left\{1, 2, ..., p\right\}$, and $i \neq j$, y_i, and y_j are orthogonal in nature, $y_i^T y_j = 0$

Thus, in the cost function, the determinant measure can be utilized. If the determinant measure has a larger value, then the matrix is more sparse. Therefore, with these determinant constraints, the sparse coding problem can be now modeled as an optimization problem and represented as

$$\max_y \det\left(YY^T\right) = \min_y - \det\left(YY^T\right) \tag{5}$$

Formulation of sparse representation problem

The analysis sparse representation problem description is explained as follows. It is assumed that the observed signal vector $t \in \Re_+^p$ is present and it is a noisy aspect of a signal $z \in \Re_+^p$. Therefore, $t = z + v$, where v denotes additive positive white Gaussian noise. With the help of an analysis dictionary $\Omega \in \Re^{n\times p}$, every row which explains $1 \times p$ analysis atom is considered so that z satisfies $||\Omega z||_0 = p - s$, where s expresses the cosparsity of the signal which is matched to be the total number of zero elements. To define the signals with every column as one signal, a matrix Z is utilized so that the signals matrix can be extended. In this study, the sparse measure is analyzed as $M(.)$. The noise in the measured signals is considered, and therefore, for analyzing dictionary learning, an optimization task is formulated as

$$\min_{\Omega,Z} M\left(\Omega Z\right) \tag{6}$$

such that $||\text{T-Z}||_F^2 \leq \sigma$.

The noise level parameter is denoted by σ, the sparse regularization is expressed as M. With the help of penalty multipliers, a regularized version of the above equation can be done. In such a case, X is considered as an approximation of ΩZ, which tends to make the learning fast and easy. By means of thresholding the sparsity measure on X and the product of ΩZ, the analysis sparse coding is obtained.

The analysis sparse representation is expressed as

$$\min_{\Omega,Z,X} M(X) + \lambda\, ||\text{T-Z}||_F^2 + \beta\, ||\Omega\text{Z-X}||_F^2 \tag{7}$$

such that $\left|\left|q_i\right|\right|_2 = 1, \forall_i$, where the representation coefficient matrix is denoted by $X \in \Re^{n\times N}$. Now the representation matrix X is considered as sparse. The λ and β in (7) are estimated with the help of the famous Lower Upper (LU) decomposition technique. To remove the scale ambiguity, a normalization constraint is introduced $\left(\forall_i \left|\left|q_i\right|\right|_2 = 1\right)$. The analysis dictionary learning procedure is summarized in Algorithm 1.

```
Initialization: $\Omega_0, X_0, Z_0 = T, i = 0$
While convergence is not achieved do
    $\Omega_{i+1} = \min_\Omega ||\Omega Z - X||_F^2\, s.t.\forall_i \left|\left|q_i\right|\right|_2 = 1$
    $X_{i+1} = \min_X M(X) + \beta\, ||\Omega Z - X||_F^2$
    $Z_{i+1} = \min_Z \lambda\, ||T - Z||_F^2 + \beta\, ||\Omega Z - X||_F^2$
    $i = i + 1$
```

Algorithm 1. Analysis dictionary learning algorithm (ADLA).

For the sparse represented EEG signal, the statistical feature parameters such as mean, variance, skewness, kurtosis, sample entropy, approximate entropy, Shannon entropy, Hurst exponent, Largest Lyapunov Exponent, Fractal Dimension, Recurrence Quantification Analysis, Higher Order Cumulants, Lempel Ziv Complexity, Kolmogorov Complexity, and Hjorth exponent are computed. **Table 1** shows the average statistical feature parameter values for sparse represented EEG data signals. It is noted from **Table 1** that low values of mean, variance, and skewness are observed among the Bonn dataset (Andrzejak et al., 2001) (normal, inter-ictal, and ictal categories) and schizophrenia dataset, while the kurtosis parameter reached a high value in the Bonn dataset and schizophrenia dataset (Olejarczyk and Jernajczyk, 2017). Bonn dataset does not differentiate among the entropy features, but in the case of schizophrenia dataset, there exists a difference in the entropy features. All the statistical parameters for the features such as Hurst Exponent, Largest Lyapunov Exponent, Fractal Dimension, Recurrence Quantification Analysis, Higher Order Cumulants, Lempel Ziv Complexity, Kolmogorov Complexity, and Hjorth Exponent show the nonlinear behavior and it has very close values among the group of datasets. This justification indicates that the sparse represented data should be further processed through the HMM with bio-inspired learning algorithms.

Hidden Markov model analysis

To express a Markov process with unknown parameters, HMM is often used (Rezek and Roberts, 2002). Through observable parameters, it is hectic to understand the implicit parameter of the process, and so it is utilized to proceed with further in-depth analysis. Two discrete-time stochastic processes that are related to each other are described by HMM. Hidden state variables are applicable to the first process and denoted as $(V_1, V_2, ..., V_n)$, which emits the observed variables with various probability factors. The second process is applicable and related to observed variables $(w_1, w_2, ..., w_n)$. The transition probability and the emission probability are the two main parameters of HMM.

Transition Probability: $P\left(V_l = v_p \middle| V_{l-1} = v_m\right)$ It implies that the current state depends on the previous state v_m.

Emission Probability: $P\left(w_l \middle| V_l = v_p\right)$ The current state v_p is used to release the observation symbol. In our model, for every extracted sparse signal feature f_i, an HMM $\lambda^{(f_i)}$ is built. The observed variables are nothing but the sparse representation features extracted from the signal $'s'$, while every hidden state $V_l^{(f_i)}$ of $\lambda^{(f_i)}$ is assured as a state related to the feature w_l. If the sparse representation S obtains a very high probability for the model $\lambda^{(f_i)}$, it implies that S is related to the sparse signal feature f_i.

Consideration of sparse features as observed variables

The features extracted from the sparse signal model are termed as sparse features, and these are considered as observed variables. Under this domain, category-based extraction and global-based extraction are the two main categories of feature extraction techniques. As global-based extraction methods cannot be utilized to differentiate the various sparse features so well, in our work we adopted category-based feature extraction techniques.

TABLE 1 Average statistical feature parameters for sparse represented EEG dataset signals.

Sl. No.	Statistical parameters	Bonn EEG dataset			Schizophrenia dataset	
		A	C	E	Schizophrenia	Normal
1	Mean	0.10049	0.300822	0.850364	0.971108	0.189563
2	Variance	0.001707	0.001091	0.000505	3.9E-05	5.31E-05
3	Skewness	0.561381	1.5384	−0.69973	0.596389	1.523915
4	Kurtosis	64.37504	48.20223	29.77208	59.32448	77.81783
5	Sample entropy	11.7308	11.5397	11.3726	6.8751	10.289
6	Approximate entropy	1.986	1.648	1.461	1.7916	2.041
7	Shannon entropy	10.87	6.69	5.421	5.832	11.67
8	Hurst exponent	0.734	0.582	0.348	0.231	0.831
9	Largest Lyapunov	0.839	0.2311	0.469	0.415	0.942
10	Fractal dimension	0.2769	0.281	0.286	0.341	0.242
11	Recurrence quantification	0.1208	0.1176	0.2177	0.2307	0.098
12	Higher order cumulants	0.2495	0.482	0.725	0.774	0.2116
13	Lempel–Ziv complexity	341.33	334.9	326.58	406.91	312.21
14	Kolmogorov complexity	11.039	9.873	9.5684	7.8002	7.6749
15	Hjorth exponent	1.528	1.7153	1.6887	1.6002	1.726

Considering a set of n feature extraction techniques $\{F_1, F_2, \ldots, F_n\}$, a sparse representation S is divided into n terms $\left(k_1, k_2, .., k_n\right)$. Assuming z_{jl} is the l^{th} feature which is extracted by the method F_l. For computing the sparse representation feature vector, an intermediate $h \times n$ matrix of term-level feature is utilized. For every sparse signal feature f_i, the sparse representation feature vector of the signal S is represented as follows:

$$\begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_h \end{bmatrix} \rightarrow \begin{bmatrix} z_{11} & z_{12} & .. & z_{1n} \\ z_{21} & z_{22} & .. & z_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ z_{h1} & z_{h2} & .. & z_{hn} \end{bmatrix} \rightarrow [w_1, w_2, .., w_n]^{(f_i)} \tag{8}$$

where $w_l = \sum_{j=1}^{h} z_{jh}/h, \left(1 \leq l \leq\ n\right)$.

Over all the signal features,$_{w_l}$ is a mean value of $_{l^{th}}$ features over all the extracted signal features.

Development of hidden Markov model-based signal classification model

A value is supposed to be emitted by each hidden state and so the sequence of values is generated by the whole model that constitutes and manages the sparse representation feature vector. The representation of the best signal category is done by a set of values and it is considered to be as a state in our work. Between the sparse representation and the HMM states, there is a one-to-one mapping that requires the transition of hidden states to be in a stationary mode and the states to indicate the start level v_1. During the working of the classifier, the features of test sparse representations being drawn closer to the signal are done by the transition probability and are expressed as follows:

$$P\left(V_l = v_p \middle| V_{l-1} = v_m\right) = \begin{cases} 1, \left(p = m+1\right) \\ 0, \left(p \neq m+1\right) \end{cases} \tag{9}$$

With the help of known state V_l, the feature w_l and the training data, the emission probability $P\left(w_l / V_l\right)$is calculated. The HMM model $\lambda^{(f_i)}$ for every feature f_i is expressed in Algorithm 2 as follows:

```
The signal level feature vector is
expressed as $[w_1, w_2, \ldots, w_n]^{(f_i)}$
Input: feature vector $[w_1, w_2, \ldots, w_n]^{(f_i)}$
Output: Probability
$P\left(\left[w_1, w_2, \ldots, w_n\right]^{(f_i)} \middle| \lambda^{(f_i)}\right)$
For $l = 1$ to $n$ do
  Calculate $P\left(w_l / V_l\right)^{(f_i)}$
End for
  Calculate $P\left(\left[w_1, w_2, \ldots, w_n\right]^{(f_i)} \middle| \lambda^{(f_i)}\right)$
  using forward algorithm.
```

Algorithm 2. Expression of every feature in the HMM Model.

For each of the signal features, the HMM concept is constructed and implemented. The calculation of the probabilities of the sparse representations on the signal feature models is done when a new sparse representation arrives. The sparse representation is labeled with the signal features whose model is highly related to the maximum probability.

To compute $P\left(w_l \middle| V_l\right)^{(f_i)}$, a Jaccard similarity (J) is utilized which helps in testing the correlation between the value w_l and V_l.

$$P\left(w_l|\,V_l\right)^{(f_i)} = J\left(w_l|\,V_l\right)^{(f_i)} = \frac{H_{11}}{H_{11}+H_{10}+H_{01}} \quad (10)$$

where H_{11} indicates the number of sparse representations contained with w_l and V_l in f_i; H_{01} indicates the number of sparse representations which has only w_l in f_i; H_{10} indicates the number of sparse representations which has only V_l in f_i.

To assess whether the signal feature f_i relates to the observation w_l or the state V_l, an associated factor is included and it is specified by δ_l. The feature extracted from the training data be $w_l^{'}$ and the association is described as follows:

$$\left|w_l^{'} - w_l\right| \leq \delta_l \ \ or \ \left|w_l^{'} - V_l\right| \leq \delta_l \quad (11)$$

If the above inequalities are satisfied, then it is understood that the observation w_l or the state V_l is related to f_i.

Self-Pliable mechanism of hidden Markov model by swarm intelligence techniques- computation of parameters

It is very important to build a versatile HMM classifier, and it is significant to trace the optimized sequences of the HMM states. For the optimization of state parameters, various strategies are utilized by means of utilizing SI techniques. In this work, PSO, DE, WOA, and BSA are utilized. The main reasons for selecting these four SI techniques are because they are very easy and have a simple implementation with fast convergence and good computational efficiency. As HMM can adapt itself to the various optimization techniques, the HMM techniques can be called as self-pliable one.

Particle swarm optimization

A famous population-dependent stochastic optimization is PSO (Eberhart et al., 2001). The candidate solution of an HMM parameter is represented by every particle in PSO. Around the search space, the movement of the particles takes place. With the help of the local best-known position of a particle, the best-known positions are found. The best parameters can be iteratively found by this technique. PSO has the extreme power to achieve global optimization and it has a good application in our study. The mathematical expressions concerning it are as:

$$\begin{aligned} v_e\,[\,] = \ & W_e \times v_e\,[\,] + ac_1 \times r \times \left(pbest\,[\,] - presentposition[\,]\right) \\ & + ac_2 \times R \times \left(gbest\,[\,] - presentposition[\,]\right) \end{aligned} \quad (12)$$

$$presentposition[\,] = present\,[\,] + v_e\,[\,] \quad (13)$$

[] specifies that its specific variable is a vector. In the range of [0,1], the variables *rand* and *R* are represented.

The individual extremes are recorded by *pbest*[], and the global extremes are recorded by *gbest*[]. The inertia weight is represented by the constant W_e. The acceleration constants are represented as ac_1 and ac_2.

Based on the previous velocity value and its corresponding distance to the best particle, the updates of the velocities of particles are done. The present [] particle's position is updated by (13) based on the current velocity and the previous position value.

Parameter settings of PSO: To have various impacts on optimization performances, various PSO parameters are considered. The PSO parameters are selected on the following basis:

$V_{\max}$: It is set by values of training data and it implements the searching space granularity.

W_e : It decides the motion inertia of the particles and the value is set as 0.5 in our experiment.

ac_1, ac_2 : indicates the accelerated weight so that it could propagate each particle to *pbest*[] and *gbest*[], the weights of both of them are set to 4 after a lot of trial and error basis.

To find the two types of parameters in HMM, a fitness function is used; (i.e.) the reduced associated factor δ_l and the $V_l^{(f_i)}$, which indicates the l^{th} hidden state of the HMM $\lambda^{\,(f_i)}$.

The definition of fitness function is done as follows:

$$fitness\left(\delta_l,\, V_l^{(f_1)},\, \ldots,\, V_l^{(f_6)}\right) = \ F_1 - Measure, \quad \left(1 \leq l \leq n\right) \quad (14)$$

where F_1 measure is one of the metrics used for classification accuracy. The exhaustive search is done for a total number of the involved parameters. The set of parameters is divided into n independent parts as training the whole parametric set is time-consuming by PSO. Depending on the fitness function, the parameters are thoroughly learned.

Differential evolution

It is a famous population-based approach that is widely used by everyone and is a promising global search technique and can be used well for HMM (Sarker et al., 2014). The candidate solution of an HMM parameter is represented by every evolution process in DE. Once the initial population is generated, then by looping mutation, selection, and crossover operations, the updation of the population is done. In the following four steps, the DE procedure is summarized as follows:

(A) Initialization: By utilizing random number distributions, the generations of an initial population are done. The j^{th} dimension of the i^{th} individual is initialized as

$$z_{i,j} = B_j + rand(0,1)^{*}\left(U_j - L_j\right), \quad i = 1, 2, \ldots, S, j = 1, 2, \ldots, D \quad (15)$$

where the population size is S and the dimension of individual is represented as D, A random number in [0,1] range which is uniformly distributed is expressed by rand (0,1). The upper bound of the j^{th} dimension is expressed as U_j and the lower bound of the j^{th} dimension is expressed as L_j, respectively.

(B) Mutation: The differential evolution enters the main loop after the initialization is done. A mutant individual m_i through mutation operators $\left(DE/rand\right)/1$ is generated by every target individual z_i in the population. The generated m_i is represented as

$$m_i = z_{r1} + C^* \left(z_{r2} - z_{r3}\right), r_1 \neq r_2 \neq r_3 \neq i \tag{16}$$

where r_1, r_2, and r_3 are selected randomly from the present population. To scale the difference vector, C is utilized and is termed as the mutation control parameters.

The other generally used mutation operators for DE are expressed as follows:

(1) "$DE/best/1$"

$$m_i = z_{best} + C^* \left(z_{r1} - z_{r2}\right), r_1 \neq r_2 \neq i \tag{17}$$

(2) "$DE/rand/2$"

$$m_i = z_{r1} + C^* \left(z_{r2} - z_{r3}\right) + C^* \left(z_{r4} - z_{r5}\right), \quad r_1 \neq r_2 \neq r_3 \neq r_4 \neq r_5 \neq i \tag{18}$$

(3) "$DE/best/2$"

$$m_i = z_{best} + C^* \left(z_{r1} - z_{r2}\right) + C^* \left(z_{r3} - z_{r4}\right), \quad r_1 \neq r_2 \neq r_3 \neq r_4 \neq i \tag{19}$$

(4) “$DE/Current - to - best/1''$

$$m_i = z_i + C^* \left(z_{r1} - z_i\right) + C^* \left(z_{r2} - z_{r3}\right), \quad r_1 \neq r_2 \neq r_3 \neq i \tag{20}$$

(5) “$DE/rand - to - best/1''$

$$m_i = z_i + C^* \left(z_{best} - z_i\right) + C^* \left(z_{r1} - z_{r2}\right), \quad r_1 \neq r_2 \neq i \tag{21}$$

z_{best} represents the individual with the best fitness function value.

However in this work, all the above-mentioned five combinations were utilized and upon analysis,$\left(DE/rand\right)/1$ was finally chosen and implemented as it was very convenient to set and alter the values after the initialization process is done.

(C) Crossover:

To generate a trial individual t_i, a crossover operation which is binomial in nature is implemented to the target individual z_i and the mutant individual m_i as follows:

$$t_{i,j} = \begin{cases} m_{i,j} & if \quad rand(0,1) \leq LR \quad or \quad j = j_{rand} \\ z_{i,j} & \qquad\qquad\qquad otherwise \end{cases} \tag{22}$$

where a randomly chosen integer in the range of $[1, D]$ is expressed as j_{rand}. The crossover control parameters are expressed as CR and it is in the range of $CR \in [0, 1]$.

(D) Selection:

Selection of the better one from the target individuals z_i and crossover individual t_i into the upcoming generations is important and so the greedy selection operator is utilized in this study. Based on the primary comparison of fitness values, this operation is performed, and it is computed as:

$$z_i^{t+1} = \begin{cases} t_i, & if\,fit(t_i) < fit(z_i) \\ z_i, & otherwise \end{cases} \tag{23}$$

where the fitness function is denoted by fit.

Whale optimization algorithm

A famous swarm-based metaheuristic algorithm is WOA (Mirjalili and Lewis, 2016). The candidate solution of an HMM parameter is represented by every whale in WOA. The intelligent foraging behavior of hump back whales is mimicked in it, and this algorithm is influenced by bubble net hunting strategy (Mirjalili and Lewis, 2016). The main operators are included in WOA such as

(i) Simulation and searching the prey.
(ii) Encircling behavior of the prey.
(iii) Bubble net foraging behavior of the whales.

The exploration phase is nothing but searching for prey, and the exploitation phase is the encircling prey and spiral bubble net attacking method. For the two phases, the mathematical model is presented below.

(I) Initial stage: Exploitation stage

This includes the encircling prey phase/bubble net attacking method. Based on two mechanisms, the updation of their positions is done by the hump back whales during the exploitation phases such as shrinking with encircling mechanism and the spiral updation position. The former is called encircling prey, and the latter is called spiral bubble net attacking method. Using the following equations, the representation of the shrinking mechanism is done as follows:

$$\vec{Z}(t+1) = \vec{Z}^*(t) - \vec{M}.\overrightarrow{Dis}, \tag{24}$$

$$\overrightarrow{Dis} = \left| \vec{N}.\vec{Z}^*(t) - \vec{Z}(t) \right| \tag{25}$$

where the current iteration is represented by t.

The best solution of the position vector obtained so far is represented by $\vec{Z}^*(t)$, and the position vector is indicated as $\vec{Z}(t)$.

The coefficient vectors are denoted as ${}_{\overrightarrow{M}}$and ${}_{\overrightarrow{N}}$, and it is calculated as follows:

$$\begin{aligned} \overrightarrow{M} &= 2\overrightarrow{m}.\overrightarrow{r} - \overrightarrow{m} \\ \overrightarrow{N} &= 2.\overrightarrow{r} \end{aligned} \tag{26}$$

In both the phases, over the period of iterations, $\overrightarrow{m}$is linearly decreased from 2 to 0. Here, $\overrightarrow{r}$ represents a random vector in the range of [0.1]. Using the following equation, the mathematical representation of the spiral updating position is expressed as follows:

$$\overrightarrow{Z}(t+1) = \overrightarrow{Dis}'.e^{cq}.\cos\left(2\pi l\right) + \overrightarrow{Z}^{*}(t) \tag{27}$$

$$\overrightarrow{Dis}' = \left|\overrightarrow{Z}^{*}(t) - \overrightarrow{Z}(t)\right| \tag{28}$$

The distance of the x^{th} humpback whale to the best solution derived is represented by $\overrightarrow{Dis}'$.

The logarithmic spiral shape is defined by a constant c and the random number in the range of [–1,1] and is expressed by q. The element-by-element multiplication is given by $(\cdot)$. The mechanism exhibited by whale when catching a prey such as shrinking encircling mechanism and spiral updating positions are accomplished at the same time. The assumption is that a probability of 50% is chosen between them so that this behavior could be initiated. This mathematical modeling is expressed as follows:

$$\overrightarrow{Z}(t+1) = \begin{cases} \overrightarrow{Z}^{*}(t) - \overrightarrow{M}.\overrightarrow{Dis} \;\; if \;\; k < 0.5 \\ \overrightarrow{Dis}'.e^{cq}.\cos\left(2\pi q\right) + \overrightarrow{Z}^{*}(t) \;\; if \;\; k \geq 0.5 \end{cases} \tag{29}$$

where the random number k is in the range of [0,1].

(II) Prey Searching Phase (Exploration Phase):

In order to increase the exploration capability of WOA, based on randomly selected whale, the position of the whale is updated instead of utilizing the best whale food in the process. To force or to propagate away from a whale and to move very far from the best-known whale, a coefficient vector M with random values substantially greater than 1 or less than -1 is utilized.

Mathematically, it is expressed as

$$\overrightarrow{Z}(t+1) = \overrightarrow{Z}_{rand}(t) - \overrightarrow{M}.\overrightarrow{Dis} \tag{30}$$

$$\overrightarrow{Dis} = \left|\overrightarrow{N}.\overrightarrow{Z}_{rand}(t) - \overrightarrow{Z}(t)\right| \tag{31}$$

where a random position vector selected from the current population is expressed as $\overrightarrow{Z}_{rand}$.

Backtracking search optimization algorithm

A famous population-based metaheuristic algorithm is BSA (Beek, 2006). The candidate solution of an HMM parameter is represented by every search in backtracking mechanism of BSA. By means of implementing mutation, crossover, and selection of population, this algorithm achieves the optimization purpose similar to other meta-heuristic algorithms. It has the unique quality to remember historical populations and therefore by completely mining the historical information, previous generations can be benefitted. Five steps are present in the original BSA, namely, (i) initialization (ii) Selection Phase I (iii) Mutation (iv) Crossover, and (v) Selection Phase II. The explanation for the 5 steps is as follows:

Step 1: Initialization:

At the outset, with the following formula, the population A and the historical population $oldA$ is initialized by BSA, respectively.

$$\begin{aligned} A_{i,j} &\sim W\left(low_j, up_j\right) \\ OldA_{i,j} &\sim W\left(low_j, up_j\right) \end{aligned} \tag{32}$$

where $i = 1, 2, ..., S, j = 1, 2, ..., D$.

The population size is represented by S, and the population dimension is represented by D, respectively. The uniform distribution is denoted by W. The lower boundaries of variables are denoted as low_j, and the upper boundaries of variable are denoted as upp_j.

Step 2: Selection Phase I:

Based on equation (32), the updation of the historical population $oldA$ is done. Then there is a random change in the locations of individuals in $oldA$ as projected in equation (33):

$$if \; p < q\left(p, q \sim W(0,1)\right), \; then \; oldA = A \tag{33}$$

$$oldA = permuting\left(oldA\right) \tag{34}$$

where a random permutation of the integers from 1 to $N$is done by permuting $(\cdot)$ operations.

Step: 3 Mutation Process

The initial trial population is generated by the mutation operator of BSA so that there is complete control of the documented and authentic information along with the current information. The expression of mutual operation is expressed as:

$$M_{i,j} = A_{i,j} + C^{*}(oldA_{i,j} - A_{i,j}) \tag{35}$$

where the control parameters are denoted by C, and the value of C is chosen to be 5 in our experiment after a lot of trial and error basis. A powerful global search ability is obtained by this operation.

Step: 4 Crossover:

Here, it comprises of 2 steps:

(1) Initially, a binary integer value matrix map is generated which is of size $S^{*}D$

2) Secondly, depending on the matrix map generated, the location of crossover individual elements are determined in population A

3) Therefore, to get the final trial population T_p, the individual elements in A are exchanged with the respective collaborating positive elements in population V. The expression of crossover operation is expressed as

$$R_{i,j} = \begin{cases} A_{i,j} & if \quad map_{i,j} = 1 \\ V_{i,j} & \quad otherwise \end{cases} \tag{36}$$

Sometimes they might be an overflow of few individuals of the trial population T than the allowed search space limits after the crossover operation. There will be a regeneration of individuals present beyond the boundary control based on equation (32).

Step 5: Selection II phase:

To preserve the best favorable trial individuals, a greedy selection mechanism is utilized. For the trial individuals and the target individuals, the fitness values are compared. The trial individual can get accepted to the next generation if the fitness value of trial individuals is much less than the target individuals. If the fitness merit and utility of trial individuals are more than the target individual, then the target individual is retained in the population. The definition of selection operation is expressed as follows:

$$A_i = \begin{cases} T_{p_i}, \ if \quad fitness\left(T_{p_i}\right) < fitness\left(A_i\right) \\ A_i, \qquad\qquad\qquad\qquad otherwise \end{cases} \tag{37}$$

where the objective function value of a particular individual is $f(\cdot)$.

Feedback mechanism for swarm computing techniques

To manually label all the sparse feature representations, it is pretty time-consuming and very expensive too. Therefore, a feedback technique is introduced that can automatically deal and relate whether an unlabeled sparse representation is chosen and present in a training data pool once the HMM assigns it with the signal feature. Therefore, the best strategy is to calculate the entropy measures of a sparse representation S so that the signal is more discriminating than all the other signal features on the sparse representations S. A famous information theoretic measure it is expressed as

$$\phi(S) = -\sum_i P\left(f_i \middle| S\right) \log P\left(f_i \middle| S\right) \tag{38}$$

where $P\left(f_i \middle| S\right)$ expresses the probability of the sparse representations S recognized as a signal feature f_i. If $\phi(s)$ is less, then the certainty about the sparse representations S on the signal feature f_i is more. To decide whether a sparse representation should be present in the training data set, the algorithm of feedback-based mechanism is utilized as shown in Algorithm 3.

```
 Input: training data $D$, test pool data
$N$, query strategy parameter $\phi(\bullet)$, query
batch size parameter $B_s$
Repeat
   For i=1 to |F| do
Optimized $\lambda^{(f_i)}$ by utilizing current $D$
and PSO/DE/WOA/BSA algorithm
   End for
   For $b_s = 1$ to $B_s$ do
   $S^*_{b_s} = \arg\max_{S\in U} \phi(S)$
   Move $S^*_{bs}$ from $N$ to $D$
   End for
Utilizing some stopping criterion
```

Algorithm 3. Feed back mechanism.

The gist of EEG signal classification with sparse representation measures and a swarm computing-based HMM methodology is as follows:

(a) Preprocessing of signals is done initially by using Independent Component Analysis (ICA).
(b) Sparse Modeling of the signals is done.
(c) Computation and extraction of sparse feature vectors of the entire dataset are done.
(d) Building an HMM for the assessed sparse signal features as observed variables.
(e) The hidden states of each $\lambda^{(f_i)}$ are optimized by PSO/DE/WOA/BSA.
(f) For every $\lambda^{(f_i)}\left(f_i \in |F|\right)$, the signal vector $\left[w_1, w_2, ..., w_n\right]^{(f_i)}$ of S in sparse signal feature f_i is computed, and the output values $P\left(\left[w_1, w_2, ..., w_n\right]^{(f_i)} \middle| \lambda^{(f_i)}\right)$ are calculated through model $\lambda^{(f_i)}$.
(g) Return $f^* = \arg\max_{f_i \in |F|}\left\{P\left(\left[w_1, w_2, ..., w_n\right] | \, \lambda^{(f_i)}\right)\right\}$.

To test the performance of every HMM, several sparse representation features represented as observed variables are selected randomly. For each signal representation, the test dataset contains numerous sparse representation features. For about ten times, each test result is executed, and the evaluation is based on the average results.

Deep learning–based methodology

Generally, to perform the classification in an end-to-end manner, the deep CNN model (Zhao et al., 2020) is utilized but in this work, once the sparse representation modeling to EEG signals is done, then deep feature extraction happens through the developed deep learning model, and finally, it is fed to classification. The utilized 1D-CNN deep learning architecture is expressed in Figure 2 as follows:

The sparse represented EEG signals are fed into the four convolution blocks where every block is comprised of five different layers so that the sparse representation can be learned more deeply. For the generation of a group of linear activation responses, multiple convolutions in parallel are computed by

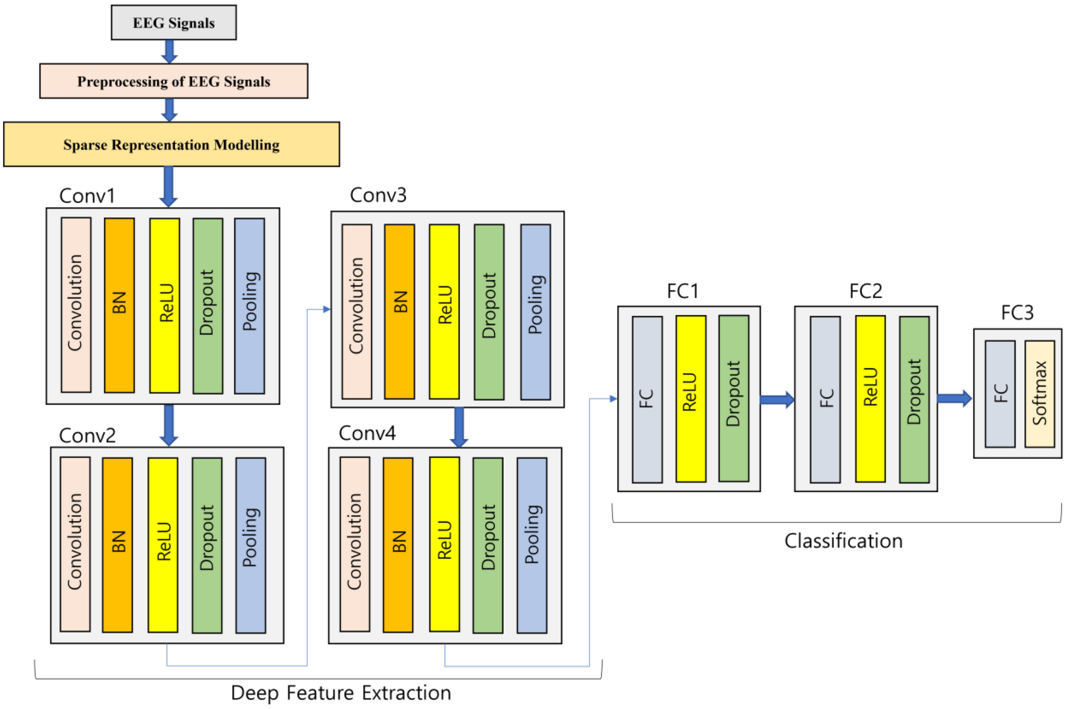


FIGURE 2
Deep learning 1D-CNN for the classification of EEG.

the first layer. In order to solve the internal variable shift, the second layer utilized is Batch Normalization (BN). A nonlinear activation function in the layer is passed by each linear activation response. Rectified Linear Unit (ReLU) is the chosen activation function and is implemented in this work. To avoid overfitting, the concept of dropout methodology is used in the fourth layer. Finally, translation invariance is introduced by the max pooling layer, which serves as the last layer in the block. In the developed deep learning architecture, the second, third, and fourth convolution blocks are same as the first convolution block repeating the same actions. The flattening of the feature maps is done into a one-dimensional vector at the end of the fourth convolution block which is connected to the Fully Connected (FC) layer so that the features are integrated. The activation function is chosen as ReLU for the first two FC layers which are accompanied by a dropout layer. Softmax activation function is implemented in the third FC layer so that a vector of probabilities communicating to every category is given as output. The experiments were tried with various model parameters and the one which produced good results is provided in this work.

Convolution layer

In order to process the data with same network structures, CNN is widely preferred. By means of regular sampling on time axis, the consideration of the time series data can be done as a one-dimensional grid. The important three layers, namely, convolution layer, activation function layer, and pooling layer are present in any convolutional block of the standard CNN model. The convolution operation for the 1D EEG data utilized in this article is expressed as:

$$s(t) = \left(x^{*}w\right)(t) = \sum_{a} x(a)w\,(t-a) \tag{39}$$

The attributes of the sparse interaction are present in the convolutional network that helps to mitigate the storage requirements of the developed deep learning model. This ensures that all the memory parameters are thoroughly learned with the parameters shared by the convolution kernel. Convolution is actually a special type of linear operation and it is only with the help of activation function, the nonlinear characteristics are bought in the network. The commonly utilized activation function in CNN is ReLU, which helps to solve the vanishing gradient issue so that the models can learn faster and enhance the overall performance. The spatial size of the representation is mitigated with the help of pooling function so that the total number of parameters along with the computation is reduced in the network. At specific portions, the output of the system is replaced by the pooling function, thereby making the representation roughly invariant to minor input translations.

Computation of batch normalization

To the standard convolution blocks, the addition of the BN layer along with the dropout layer is done. There is always a close relation between the parameters of every layer where the training of the deep neural network is done. When the input layers are distributed, an inconsistency occurs causing an issue called as internal covariate shifts, making it hectic to choose a suitable learning rate. Therefore, BN process is used in this study so that almost any deep network can be reparametrized quite easily by means of coordinating the updation process between multiple layers of the network. Therefore, the normalization is considered as part of the deep learning model architecture and it helps to normalize every mini-batch. For the mini-batch response H, the computation of the sample mean (μ) and standard deviation (σ) in backpropagation during training is done as follows:

$$\mu = \frac{1}{m}\sum_{i} H_i \tag{40}$$

$$\sigma = \sqrt{\delta + \frac{1}{m}\sum_{i} (H-\mu)_i^2} \tag{41}$$

To prevent the gradient from becoming undefined, the delta component δ is usually added and it is a very small positive value. In order to normalize H, the following expression is utilized as:

$$H^{'} = \frac{H-\mu}{\sigma} \tag{42}$$

The convergence of the training phase can be well accelerated by BN so that overfitting can be avoided easily and therefore BN is employed after every convolution layer.

Fusion of features along with classification

A large number of parameters need to be learned by the deep neural networks and in the case of smaller datasets, there is a high chance for occurrence of overfitting. Therefore, to solve this issue, dropout technology was added so that the coadaptation of feature detection is avoided fully. The random dropping of units with a predefined probability from the neural network seems to be the main intention of dropout layer during the training process. When compared to other regularization methods, this technique can reduce the overfitting to a great extent and therefore after each ReLU activation function, a dropout layer is added. The high-level features of the EEG signals are indicated by the output of the final convolutional block. The FC layer can easily learn all the nonlinear combinations of these functions. In this work, three FC layers have been developed. The connection of all the neurons in the last max-pooling layer is done with the neurons of the first FC layer. Depending on the final classification problem, the determination of the total number of neurons in the final FC layer is done and since a two-class epilepsy classification problem and a two-class schizophrenia classification problem is dealt in this study, the number of neurons in FC3 layer is chosen to be two. A generalized form of the binary manifestation of logistic regression is the softmax activation function. In order to assemble a categorical distribution over the class labels and to trace the probability of every input element belonging to a particular label, this softmax function is usually implemented in the ultimate layer of a deep neural network. The respective probability of the i^{th} sample expressed by $x^{(i)}$ which belongs to each category and is indicated by the softmax function $h_\theta\left(x^{(i)}\right)$ as follows:

$$h_\theta\left(x^{(i)}\right) = \begin{bmatrix} p(y^{(i)} = 1|\, x^{(i)}; \theta) \\ p(y^{(i)} = 2|\, x^{(i)}; \theta) \\ \vdots \\ p(y^{(i)} = k|\, x^{(i)}; \theta \end{bmatrix} = \frac{1}{\sum_{l=1}^{k} e^{\theta_l^T x^{(i)}}} \begin{bmatrix} e^{\theta_1^T x^{(i)}} \\ e^{\theta_2^T x^{(i)}} \\ \vdots \\ e^{\theta_k^T x^{(i)}} \end{bmatrix} \tag{43}$$

where the softmax model parameters are expressed by $\theta_1, \theta_2, \ldots, \theta\ _k$.

Model training

The weight parameters are required to be learned from the EEG data for the training of the proposed model. The standard Backpropagation algorithm was used and the loss function utilized is cross entropy. The stochastic gradient descent technique with Adam optimization is utilized to learn the parameters. The hyperparameters of Adam are set as follows: learning rate is 0.0001, beta1 value is set at 0.5 and beta2 value is set at 0.55. The batch size is considered as 200 in our experiment which helps in the updation of the training process. The total number of epochs utilized in this work is expressed as 250 so that the training of the model can be done well.

Results and discussion

For evaluating and validating this proposed model, it has been tested on University of Bonn dataset (Andrzejak et al., 2001) which deals with epilepsy classification and the schizophrenia dataset from Institute of Psychiatry and Neurology in Warsaw, Poland, which deals with schizophrenia classification (Olejarczyk and Jernajczyk, 2017). There are five sets of epileptic data available such as A, B, C, D, and E. Set A and B belongs to the normal category, Set C and D belongs to the inter-ictal category, and set E belongs to the ictal category. The classification problem considered in epileptic dataset are A-E, AC-E, B-E, CD-E, ACD-E, and ABCD-E, and the classification problem considered in schizophrenia datasets are normal versus schizophrenia. The elaborate details of both datasets are given in Andrzejak et al. (2001) and Olejarczyk and Jernajczyk (2017). For both datasets, the Independent Component Analysis (ICA) is utilized as a common pre-processing technique. As far as the epilepsy dataset is considered, 100 single-channel recordings of EEG signals are present in each of these sets with a sampling rate of 173.61 Hz and time duration of 23.6 s. The respective time series is sampled into 4097 data points and further every 4097 data point is divided into 23 chunks, thereby the total number in each category has about 2,300 samples. For deep learning methodology, once the sparse modeling is implemented to it, the random division of the 2,300 EEG samples is done into ten non-overlapping folds as a 10-fold cross-validation is adopted here for evaluation. As far as the SI-based HMM along with the conventional machine learning is considered, the 2,300 samples are reduced by means of sparse feature extraction eliminating the redundant ones. Only the essential sparse features are considered as observed variables as expressed in the sparse representation modeling concept and then it is proceeded for classification by the SI-based HMM and the conventional machine learning models. As far as the schizophrenia dataset is concerned, there are about 225,000 samples with each channel, and the data are represented in this study with a matrix of [5,000 × 45]. As there are 19 such channels available there, it is represented as [5,000 × 45 × 19]. For the deep learning methodology, once the sparse modeling is implemented to it, the random division of the schizophrenia EEG samples is done into ten non-overlapping folds as a 10-fold cross-validation is adopted in this study for evaluation. As far as the SI-based HMM along with the conventional machine learning is considered for schizophrenia EEG signal classification, the [5,000 × 45] data are reduced by means of sparse feature extraction eliminating the redundant ones. Only the essential sparse features are considered as observed variables as represented in the sparse representation modeling concept and then it is proceeded for classification by the SI-based HMM and the conventional machine learning models. The performance metrics analyzed are the general measures used widely such as Classification accuracy, Sensitivity, and Specificity. The details of the 1D-CNN model utilized in this research are tabulated in Table 2.

Table 3 indicates the performance analysis of the proposed SI-based HMM for different datasets with optimization techniques. The highest sensitivity of 100% is attained for Schizophrenia dataset with DE-HMM, WOA-HMM, and BSA-HMM methods. In the case of epileptic dataset (AC-E) with BSA-HMM, and epileptic dataset (B-E) with WOA-HMM also, it reached 100% sensitivity. The lower sensitivity value of 69.86% is reached for epileptic dataset (AC-E) with WOA-HMM method. The highest specificity of 100% is obtained for epileptic dataset (AC-E) with PSO, DE, and WOA-based HMM methods. As in the case of epileptic dataset (A-E) with DE-HMM and epileptic dataset (B-E) with DE and BSA-based HMM methods, it reached 100% specificity. A low specificity value of 76.83% is reached for schizophrenia dataset with DE-HMM method. A high classification accuracy of 95.70% is attained for epileptic dataset (A-E) with DE-HMM method and low classification accuracy of 82.43% is reached for epileptic dataset (ABCD-E) with BSA-HMM method. For schizophrenia datasets, a high classification accuracy of 91.41% is obtained with PSO-HMM, and a low classification accuracy of 88.41% is obtained from DE-HMM.

Table 4 shows the performance analysis of the proposed methodology for the biosignal processing datasets in terms of accuracy using swarm intelligence–based HMM, conventional machine learning, and deep learning techniques. If the proposed flow of methodology is implemented with NBC for the datasets, then a high classification accuracy of 92.12% is obtained for the B-E dataset. When the standard LDA is utilized, then a high classification accuracy of 92.34% is obtained for the A-E dataset, and low classification accuracy of 80.5% is obtained for the ACD-E dataset. When KNN methodology is utilized, a high classification accuracy of 90.23% is obtained for A-E dataset and a low classification accuracy of 79.98% is obtained for ABCD-E dataset. If the proposed flow of methodology is implemented with Adaboost classifier for the datasets, then a high classification accuracy of 89.34% is obtained for the B-E dataset. When comparing all the conventional classifiers, the SVM performs better as a higher classification accuracy of 93.49% is obtained for the schizophrenia dataset and low classification accuracy of 87.9% is obtained ABCD-E dataset. Before computing the swarm intelligence–based HMM model, the methodology was tested for the ordinary HMM model and the highest result of only 87.34% was obtained for the A-E dataset. This seemed to motivate the researchers to undergo more research in fine-tuning HMM so that a better result could be obtained. The swarm techniques were successfully computed with HMM, and much better results were obtained. For the PSO-HMM combination, a higher classification accuracy of 92.45% was obtained for the B-E combination and a lower classification accuracy of 85.86% was obtained for the ACD-E combination. For the DE-HMM combination, a higher classification accuracy of 95.7% was obtained for the A-E combination and a lower classification accuracy of 86.8% was obtained for the ABCD-E combination. For the WOA-HMM combination, a higher classification accuracy of 89.9% was obtained for the schizophrenia dataset and a lower classification accuracy of 82.87% was obtained for the ABCD-E combination. For the BSA-HMM combination, a higher classification accuracy of 92.97% was obtained for the A-E dataset and a lower classification accuracy of 82.43% was obtained for ABCD-E combination. For the proposed 1D-CNN combination, a higher classification accuracy of 98.94% was obtained for the A-E dataset, and a lower classification accuracy of 97.05% was obtained for ACD-E combination.

TABLE 2 Convolutional neural network (CNN) structure details utilized in this work.

Name of the block	Types of layer	Number of neurons	Kernel size (output feature map)	Stride
Conv1	Convolution	179 × 20	60	1
	BN	179 × 20	–	–
	ReLU	179 × 20	–	–
	Dropout	179 × 20	–	–
	Max-pooling	90 × 20	2	2
Conv2	Convolution	71 × 40	40	1
	BN	71 × 40	–	–
	ReLU	71 × 40	–	–
	Dropout	71 × 40	–	–
	Max-pooling	36 × 40	2	2
Conv3	Convolution	31 × 60	20	1
	BN	31 × 60	–	–
	ReLU	31 × 60	–	–
	Dropout	31 × 60	–	–
	Max-pooling	18 × 60	2	2
Conv4	Convolution	13 × 80	10	1
	BN	13 × 80	–	–
	ReLU	13 × 80	–	–
	Dropout	13 × 80	–	–
	Max-pooling	5 × 80	2	2
FC1	FC	64	–	–
	ReLU	64	–	–
	Dropout	64	–	–
FC2	FC	32	–	–
	ReLU	32	–	–
	Dropout	32	–	–
FC3	FC	2	–	–

Comparison of results with previous works associated with similar datasets

The authors in recent years have dealt with classification problems as per their wish depending on their problem requirement, and therefore, it was not mandatory to perform the analysis of classification on every available subset of the epileptic data. Therefore, the available results are compared with our works and projected in Table 5.

On analyzing Table 5, it is quite evident that a wonderful attempt has been made by the authors to attain good classification accuracy results. As far as the A-E epileptic dataset is considered, among the proposed methodology, the sparse representation measures with 1D-CNN surpassed all the other results proposed in this work and gave the highest classification accuracy of 98.94% for A-E dataset, 97.15% for AC-E dataset, 98.56% for B-E dataset, 97.56% for CD-E dataset, 97.05% for ACD-E dataset, and 97.34% for ABCD-E dataset. When the swarm intelligence–based HMM is concerned, the highest classification accuracy of 95.70% is obtained when the sparse representation measures are implemented with DE-HMM for the A-E dataset. Similarly, the DE-HMM gives a high classification accuracy of 94.92% in AC-E dataset, 95.44% in B-E dataset, 90.65% in CD-E dataset, and 88.81% in ACD-E dataset when compared to other swarm-based HMM methods. For the ABCD-E dataset, the sparse representation measures with PSO-HMM provided a high accuracy of 88.9% when compared to other swarm-based methods. It is commonly known that deep learning outperforms most of the conventional pattern recognition techniques and so in this work also, the highest classification accuracy of 98.94% is obtained with the novel idea of sparse modeling with deep learning. When the results of the present work are compared to the previous works, the deep learning results obtained by us have matched more or less similar to the results obtained by the previous methods though at many places, the classification accuracy obtained by this work is slightly lower than the earlier proposed works by a range of two to four percent. In such a case, it should not induce the research community into thinking that as the classification results are slightly lower, the proposed methodology is not as versatile as the other methods. It has to be observed and

TABLE 3 Performance analysis of the proposed swarm intelligence based HMM for different datasets.

Performance metrics (%)	Datasets	Swarm intelligence based HMM			
		PSO-HMM	DE-HMM	WOA-HMM	BSA-HMM
Sensitivity	Epileptic dataset (A-E)	93.26936	91.40875	83.12805	95.83567
	Epileptic dataset (AC-E)	79.03875	89.84875	69.86516	100
	Epileptic dataset (B-E)	94.53375	90.88625	100	85.69125
	Epileptic dataset (CD-E)	88.38541	89.83978	83.87825	84.27894
	Epileptic dataset (ACD-E)	85.34346	88.38376	82.47892	82.47892
	Epileptic dataset (ABCD-E)	87.46243	85.38761	81.17835	81.48923
	Schizophrenia dataset	92.97457	100	100	100
Specificity	Epileptic dataset (A-E)	90.36875	100	86.52344	90.1125
	Epileptic dataset (AC-E)	100	100	100	83.4325
	Epileptic dataset (B-E)	90.36875	100	77.02032	100
	Epileptic dataset (CD-E)	89.19385	91.46782	87.47892	85.56672
	Epileptic dataset (ACD-E)	86.37892	89.23872	86.37892	86.38997
	Epileptic dataset (ABCD-E)	90.34678	88.22389	83.35781	84.37781
	Schizophrenia dataset	89.85625	76.8375	79.81938	81.27457
Classification accuracy	Epileptic dataset (A-E)	91.81906	95.70435	84.82575	92.97409
	Epileptic dataset (AC-E)	89.51938	94.92435	84.93258	91.71625
	Epileptic dataset (B-E)	92.45125	95.44312	88.51016	92.84563
	Epileptic dataset (CD-E)	88.78963	90.65381	85.67858	84.92283
	Epileptic dataset (ACD-E)	85.86119	88.81124	84.42892	84.43445
	Epileptic dataset (ABCD-E)	88.90460	86.80575	82.87462	82.434445
	Schizophrenia dataset	91.41541	88.41875	89.90969	90.63729

TABLE 4 Performance analysis of sparse representation based swarm HMM and deep learning for the biosignal processing datasets in terms of accuracy.

Classifier	A-E	AC-E	B-E	CD-E	ACD-E	ABCD-E	Schizophrenia
NBC	91.37582	87.34981	92.12783	85.01358	82.10368	81.89451	87.03481
LDA	92.34589	86.24951	91.34591	86.93169	80.50275	83.67912	85.56921
KNN	90.23578	85.34917	89.25791	85.87615	81.28507	79.98205	88.45917
Adaboost	88.98659	83.45691	89.34725	87.58941	77.28905	75.91632	86.68113
SVM	93.45781	91.87543	92.46915	91.34721	88.56891	87.90982	93.49812
HMM	87.34591	81.12678	83.45916	84.33861	79.48697	71.26748	81.36991
PSO-HMM	91.81906	89.51938	92.45125	88.78963	85.86119	88.90460	91.41541
DE- HMM	95.70435	94.92435	95.44312	90.65381	88.81124	86.80575	88.41875
WOA-HMM	84.82575	84.93258	88.51016	85.67858	84.42892	82.87462	89.90969
BSA-HMM	92.97409	91.71625	92.84563	84.92283	84.43445	82.43444	90.63729
1D-CNN	98.94919	97.15912	98.56781	97.56789	97.05981	97.34862	98.19864

noted that in the field of machine learning, the classification accuracies may be more or less in the range of plus or minus 3–5%, but what has to be observed carefully is the ease of methodology and implementation strategy. If that aspect is considered, the proposed methodology surpasses many earlier techniques as no strong mathematical model has been built in earlier models, whereas a strong mathematical model for sparse representation with the hybrid SI-based HMM along with deep learning is done in this work. Moreover, swarm intelligence field is like an ocean and there are hundreds of algorithms developed in the past two decades by various researchers. This work is just a starting step to use the concept of sparse modeling with SI-based HMM. In the upcoming years, a variety of other SI algorithms shall be implemented to HMM to test its ability and check its performance with the sparse representation models, and the authors are confident of obtaining much higher classification accuracy. As far as the schizophrenia classification analysis is concerned, very high-quality literature is not available online as it is an emerging field. All the important works in schizophrenia classification are

TABLE 5 Performance comparison of our works with the previous works – Epilepsy dataset.

References	A-E	AC-E	B-E	CD-E	ACD-E	ABCD-E
Bhardwaj et al., 2016	98.64	–	–	–	98.61	98.89
Peker et al., 2016	99.50	–	–	–	–	99.13
Riaz et al., 2016	99.00	–	–	–	–	96.00
Diykh et al., 2017	100	–	99.76	–	96.50	94.00
Zhang and Chen, 2017	–	–	–	–	–	98.87
Raghu et al., 2017	99.70	–	–	–	–	–
Ullah et al., 2018	100	–	99.6	99.7	–	99.7
Li et al., 2018	–	–	–	91.00	–	–
Sharma et al., 2018	100	–	–	–	–	–
Raghu et al., 2019	99.45	96.50	96.06	96.85	96.00	97.20
Gupta and Ram, 2019	99.50	–	99.50	99.00	–	98.60
Turk and Ozerdem, 2019	99.50	–	99.50	–	–	–
Sameer and Gupta, 2020	98	–	96	96.33	–	97.40
Zhao et al., 2020	99.52	–	99.11	98.03	–	98.76
Proposed technique 1: Sparse representation measures with PSO-HMM [2022]	91.81	89.51	92.45	88.78	85.86	88.90
Proposed technique 2: Sparse representation measures with DE-HMM [2022]	95.70	94.92	95.44	90.65	88.81	86.80
Proposed technique 3: Sparse representation measures with WOA-HMM [2022]	84.82	84.93	88.51	85.67	84.42	82.87
Proposed technique 4: Sparse representation measures with BSA-HMM [2022]	92.97	91.71	92.84	84.92	84.43	82.43
Proposed technique 5: Sparse representation measures with 1D-CNN [2022]	**98.94**	**97.15**	**98.56**	**97.56**	**97.05**	**97.34**

discussed in the introduction section of the article with their respective classification accuracies, where reference (Prabhakar et al., 2020a) reported 98.77%, reference (Prabhakar et al., 2020b) reported 92.17%, and reference (Oh et al., 2019) reported 81.26% for subject-based testing and 98.51% for non-subject-based testing. However, when comparing our results with the previous works, the concept of sparse representation with 1D-CNN produced a very high classification accuracy of 98.19%, and the concept of sparse representation with SI-based HMM produced an accuracy of 91.41% for PSO-HMM, 88.41% for DE-HMM, 89.90% for WOA-HMM, and 90.9% for BSA-HMM. Every swarm intelligence technique is so inspiring and it would take a life time to understand why a particular combination with sparse representation measures performs better with HMM or deep learning. Possible ways to obtain better results in SI-based HMM is to fine-tune the parameters much more carefully, varying the hyperparameters depending on the problem requirement, increasing the iteration numbers if the pre-requisite conditions are not satisfied, enhancing the essential parameters of the algorithm depending on the SI techniques considered, and updating the state space model of the HMM effectively by efficient techniques. Better results could also be obtained by means of utilizing other hybrid deep learning methods for the efficient classification of biomedical signals. An interesting classification tool based on fuzzy similarities which are characterized by a low computational complexity, and high utility for real-time applications is proposed in Versaci et al. (2022). Although it was tested on a NdT problem, due to the transversality of the approach, the method could be easily applied to the problem studied in this work too, and the authors wanted to implement a similar strategy utilized in Versaci et al. (2022) to the analysis of neurological disorders in future.

Conclusion and future work

An efficient modality through which brain signals corresponding to different states can be acquired easily is by means of using EEG. In this article, sparse representation and modeling of EEG signals are done initially, and later, an HMM classification model was proposed to compute the hidden states in the HMM, four different types of SI techniques were incorporated to make the HMM very flexible. This kind of methodology involving sparse representation with a pliable HMM for biosignal classification seems to be very efficient and easy to handle. An exhaustive analysis of the proposed SI-based HMM for epileptic and schizophrenia datasets was computed and comprehensively analyzed. The sparse representation modeling was also combined with deep learning, and conventional machine learning techniques and exhaustive analysis are provided. When the proposed sparse representation measures were combined with SI-based HMM, the highest accuracies reported are 92.45% for PSO-HMM, 95.7% for DE-HMM, 89.9% for WOA-HMM, and 92.97% for BSA-HMM. When the proposed sparse representation measures were utilized with deep learning by utilizing a CNN, high accuracy of 98.94% was obtained. Future works aim to develop more efficient sparse representation models by means of introducing more advanced concepts in the synthesis and analysis side. Though the sparse representation–based swarm HMM methods did not provide very high classification accuracy when compared to other previous works, the careful selection of the swarm intelligence algorithm with HMM would aid a very high classification accuracy with less error rate in the upcoming years. Future works also aim to hybrid the sparse representation measures with other nature-inspired algorithms such as Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), Genetic Bee Colony (GBC), Cuckoo Search Optimization (CSO), Spider Monkey Optimization (SPO), Bat algorithm, and Firefly algorithm, so that the hidden states of the HMM can be well computed in order to assess its performance on the biomedical signal datasets. Other work plans to incorporate in future include the usage of sparse representation measures with efficient deep learning techniques, such as Long Short-Term Memory (LSTM), Bidirectional LSTM, Gated Recurrent Unit (GRU), Bidirectional GRU, and hybrid deep learning techniques, for efficient classification of epilepsy and schizophrenia from its respective datasets. This proposed kind of methodology is also planned to be implemented in other image processing datasets, stock market datasets, speech processing datasets, and other beneficial datasets to check its performance assessment. In the upcoming years, the work can be integrated with Very Large Scale Integration (VLSI) technology to produce some good advancement in medicine and technology for the betterment of human health care.

Data availability statement

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

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work, and approved it for publication.

Funding

This work was supported by the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korea government [the Ministry of Science and ICT (MSIT)] (No. 2017-0-00451, Development of BCI-Based Brain and Cognitive Computing Technology for Recognizing User's Intentions using Deep Learning), partly supported by the Korea Medical Device Development Fund grant funded by the Korea government (MSIT, the Ministry of Trade, Industry and Energy, the Ministry of Health and Welfare, the Ministry of Food and Drug Safety) (No. 1711139109, KMDF_PR_20210527_0005), and partly supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2022R1A5A8019303).

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

Abiyev, R., Arslan, M., Idoko, J. B., Sekeroglu, B., and Ilhan, A. (2020). Identification of epileptic EEG signals using convolutional neural networks. *Appl. Sci.* 10:4089. doi: 10.3390/app10124089

Aliyu, I., and Lim, C. G. (2021). Selection of optimal wavelet features for epileptic EEG signal classification with LSTM. *Neural Comput. Appl.* doi: 10.1007/s00521-020-05666-0

Andrzejak, R. G., Lehnertz, K. C., Rieke, F., Mormann, P., and Elger, C. E. (2001). Indications of non linear deterministic and finite dimensional structures in time series of brain electrical activity: dependence on recording region and brain state. *Phys. Rev. Statistical Non Linear Soft. Matter. Phys.* 64:061907. doi: 10.1103/PhysRevE.64.061907

Beek, P. V. (2006). Backtracking search algorithms. *Foundations Art. Intell.* 2, 85–134. doi: 10.1016/S1574-6526(06)80008-8

Bhardwaj, A., Tiwari, A., Krishna, R., and Varma, V. (2016). A novel genetic programming approach for epileptic seizure detection. *Comp. Methods Prog. Biomed.* 124, 2–18. doi: 10.1016/j.cmpb.2015.10.001

Chen, Y., Atnafu, A. D., Schlattner, I., Weldtsadik, W. T., Roh, M.-C., Kim, H. J., et al. (2016). A high-security EEG-based login system with RSVP stimuli and dry electrodes. *IEEE Trans. Inform. Forensics Security* 11, 2635–2647. doi: 10.1109/TIFS.2016.2577551

Cura, O. K., and Akan, A. (2021). Classification of epileptic EEG signals using synchrosqueezing transform and machine learning. *Int. J. Neural Syst.* 31:2150005. doi: 10.1142/S0129065721500052

Diykh, M., Li, Y., and Wen, P. (2017). Classify epileptic EEG signals using weighted complex networks based community structure detection. *Experts Systems Appl.* 90, 87–100. doi: 10.1016/j.eswa.2017.08.012

Eberhart, R., Shi, Y., and Kennedy, J. (2001). *Swarm Intelligence*. Burlington, MA: Morgan Kaufmann.

Gao, G., Shang, L., Xiong, K., and Fang, J. (2018). EEG classification based on sparse representation and deep learning. *NeuroQuantology* 16, 789–795. doi: 10.14704/nq.2018.16.6.1666

Gupta, V., and Ram, P. B. (2019). Epileptic seizure identification using entropy of FBSE based EEG rhythms. *Biomed. Signal Proc. Control* 53:101569. doi: 10.1016/j.bspc.2019.101569

Lee, M.-H., Fazli, S., Mehnert, J., and Lee, S.-W. (2015). Subject-dependent classification for robust idle state detection using multi-modal neuroimaging and data-fusion techniques in BCI. *Pattern Recogn.* 48, 2725–2737. doi: 10.1016/j.patcog.2015.03.010

Lee, M.-H., Kwon, O.-Y., Kim, Y.-J., Kim, H.-K., Lee, Y.-E., Williamson, J., et al. (2019). EEG dataset and OpenBMI toolbox for three BCI paradigms: an investigation into BCI illiteracy. *GigaScience* 8, 1–16. doi: 10.1093/gigascience/giz002

Lee, M.-H., Williamson, J., Won, D.-O., Fazli, S., and Lee, S.-W. (2018). A high performance spelling system based on EEG-EOG signals with visual feedback. *IEEE Trans. Neural Systems Rehabilitation Eng.* 26, 1443–1459. doi: 10.1109/TNSRE.2018.2839116

Li, P., Karmakar, C., Yearwood, J., Venkatesh, S., Palaniswami, M., and Liu, C. (2018). Detection of epileptic seizure based on entropy analysis of short-term EEG. *PLoS One* 13:e0193691. doi: 10.1371/journal.pone.0193691

Mandhouj, B., Cherni, M. A., and Sayadi, M. (2021). An automated classification of EEG signals based on spectrogram and CNN for epilepsy diagnosis. *Analog Integr. Circ. Sig. Process.* 108, 101–110. doi: 10.1007/s10470-021-01805-2

Mirjalili, S., and Lewis, A. (2016). The whale optimization algorithm. *Adv. Eng. Software* 95, 51–67. doi: 10.1016/j.advengsoft.2016.01.008

Nkengfack, L. C. D., Tchiotsop, D., Atangana, R., Door, V. L., and Wolf, D. (2021). Classification of EEG signals for epileptic seizures detection and eye states identification using Jacobi polynomial transforms-based measures of complexity and least-squares support vector machines. *Inform. Med. Unlocked* 23:100536. doi: 10.1016/j.imu.2021.100536

Oh, S. L., Vicnesh, J., Ciaccio, E. J., Yuvaraj, R., and Acharya, U. R. (2019). Deep convolutional neural network model for automated diagnosis of schizophrenia using EEG signals. *Appl. Sci.* 9:2870. doi: 10.3390/app9142870

Olejarczyk, E., and Jernajczyk, W. (2017). Graph-based analysis of brain connectivity in schizophrenia. *PLoS One* 12:e0188629. doi: 10.1371/journal.pone.0188629

Peker, M., Sen, B., and Delen, D. (2016). A novel method for automated diagnosis of epilepsy using complex-valued classifiers. *IEEE J. Biomed. Health Inform.* 20, 108–118. doi: 10.1109/JBHI.2014.2387795

Prabhakar, S., Rajaguru, K. H., and Lee, S.-W. (2020a). A framework for schizophrenia EEG signal classification with nature inspired optimization algorithms. *IEEE Access* 8, 39875–39897. doi: 10.1109/ACCESS.2020.2975848

Prabhakar, S. K., Rajaguru, H., and Kim, S.-H. (2020b). Schizophrenia EEG signal classification based on swarm intelligence computing. *Comp. Intell. Neurosci.* 2020:8853835. doi: 10.1155/2020/8853835

Raghu, S., Sriraam, N., Hegde, A. S., and Kubben, P. L. (2019). A novel approach for classification of epileptic seizures using matrix determinant. *Expert Systems Appl.* 127, 323–341. doi: 10.1016/j.eswa.2019.03.021

Raghu, S., Sriraam, N., and Pradeep Kumar, G. (2017). Classification of epileptic seizures using wavelet packet log energy and norm entropies with recurrent Elman neural network classifier. *Cogn. Neurodynamics* 11, 51–66. doi: 10.1007/s11571-016-9408-y

Rezek, I., and Roberts, S. (2002). "Learning ensemble hidden markov models for biosignal analysis," in *Proceedings of the 14th International Conference on Digital Signal Processing*, (Greece).

Riaz, F., Hassan, A., Rehman, S., Niazi, I. K., and Dremstrup, K. (2016). EMD Based temporal and spectral features for the classification of EEG signals using supervised learning. *IEEE Trans. Neural Systems Rehabilitation Eng.* 24, 28–35. doi: 10.1109/TNSRE.2015.2441835

Sameer, M., and Gupta, B. (2020). Detection of epileptical seizures based on alpha band statistical features. *Wireless Pers. Commun.* 115, 909–925. doi: 10.1007/s11277-020-07542-5

Sarker, R. A., Elsayed, S. M., and Ray, T. (2014). Differential evolution with dynamic parameters selection for optimization problems. *IEEE Trans. Evol. Comp.* 18, 689–707. doi: 10.1109/TEVC.2013.2281528

Schoellkopf, B., Platt, J., and Hofmann, T. (2007). "Sparse representation for signal classification, in advances in neural information processing systems 19," in *Proceedings of the 2006 Conference*, (Germany: MITP).

Sharma, M., Bhurane, A. A., and Acharya, U. R. (2018). MMSFL-OWFB: a novel class of orthogonal wavelet filters for epileptic seizure detection. *Knowledge Based Systems* 160, 265–277. doi: 10.1016/j.knosys.2018.07.019

Sharmila, A., and Geethanjali, P. (2019). A review on the pattern detection methods for epilepsy seizure detection from EEG signals. *Biomed. Tech.* 64, 507–517. doi: 10.1515/bmt-2017-0233

Shoeibi, A., Ghassemi, N., and Khodatars, M. (2007). *Application of Deep Learning Techniques for Automated Detection of Epileptic Seizures: a Review*.

Turk, O., and Ozerdem, 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. E., Qazi, U.-H., and Aboalsamh, H. (2018). An automated system for epilepsy detection using EEG brain signals based on deep learning approach. *Expert Systems Appl.* 107, 61–71. doi: 10.1016/j.eswa.2018.04.021

Versaci, M., Angiulli, G., Crucitti, P., De Carlo, D., Laganà, F., Pellicanò, D., et al. (2022). A fuzzy similarity-based approach to classify numerically simulated and experimentally detected carbon fiber-reinforced polymer plate defects. *Sensors* 22:4232. doi: 10.3390/s22114232

Won, D.-O., Hwang, H.-J., Kim, D.-M., Müller, K.-R., and Lee, S.-W. (2018). Motion-based rapid serial visual presentation for gaze-independent brain-computer interfaces. *IEEE Trans. Neural Systems Rehabilitation Eng.* 26, 334–343. doi: 10.1109/TNSRE.2017.2736600

Xin, Q., Hu, S., Shuaiqi, M., Ma, X., Lv, H., and Zhang, Y. D. (2021). Epilepsy EEG classification based on convolution support vector machine. *J. Med. Imaging Health Inform.* 11, 25–32. doi: 10.1166/jmihi.2021.3259

Zhang, T., and Chen, W. (2017). LMD based features for the automatic seizure detection of EEG signals using SVM. *Trans. Neural Systems Rehabilitation Eng.* 28, 1100–1108. doi: 10.1109/TNSRE.2016.2611601

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. *Comp. Mathematical Methods Med.* 2020:9689821. doi: 10.1155/2020/9689821

TYPE Technology and Code
PUBLISHED 26 January 2023
DOI 10.3389/fncom.2023.1091180

OPEN ACCESS

EDITED BY
Zhanhui Wang,
China Agricultural University, China

REVIEWED BY
Liang Xue,
Shanghai University of Electric Power, China
Shouyu Wang,
Jiangnan University, China

*CORRESPONDENCE
Yan Wang
✉ wangyan@tju.edu.cn
Xinquan Liu
✉ liuxinquantju@163.com

RECEIVED 06 November 2022
ACCEPTED 13 January 2023
PUBLISHED 26 January 2023

CITATION
Liu X, Du K, Lin S and Wang Y (2023) Deep learning on lateral flow immunoassay for the analysis of detection data.
Front. Comput. Neurosci. 17:1091180.
doi: 10.3389/fncom.2023.1091180

Deep learning on lateral flow immunoassay for the analysis of detection data

Xinquan Liu[1]*, Kang Du[2], Si Lin[1,3] and Yan Wang[1]*

[1]School of Precision Instrument and Optoelectronics Engineering, Tianjin University, Tianjin, China, [2]Tianjin Boomscience Technology Co., Ltd., Tianjin, China, [3]Beijing Savant Biotechnology Co., Ltd., Beijing, China

Lateral flow immunoassay (LFIA) is an important detection method *in vitro* diagnosis, which has been widely used in medical industry. It is difficult to analyze all peak shapes through classical methods due to the complexity of LFIA. Classical methods are generally some peak-finding methods, which cannot distinguish the difference between normal peak and interference or noise peak, and it is also difficult for them to find the weak peak. Here, a novel method based on deep learning was proposed, which can effectively solve these problems. The method had two steps. The first was to classify the data by a classification model and screen out double-peaks data, and second was to realize segmentation of the integral regions through an improved U-Net segmentation model. After training, the accuracy of the classification model for validation set was 99.59%, and using combined loss function (WBCE + DSC), intersection over union (IoU) value of segmentation model for validation set was 0.9680. This method was used in a hand-held fluorescence immunochromatography analyzer designed independently by our team. A Ferritin standard curve was created, and the T/C value correlated well with standard concentrations in the range of 0–500 ng/ml (R^2 = 0.9986). The coefficients of variation (CVs) were $\leq$ 1.37%. The recovery rate ranged from 96.37 to 105.07%. Interference or noise peaks are the biggest obstacle in the use of hand-held instruments, and often lead to peak-finding errors. Due to the changeable and flexible use environment of hand-held devices, it is not convenient to provide any technical support. This method greatly reduced the failure rate of peak finding, which can reduce the customer's need for instrument technical support. This study provided a new direction for the data-processing of point-of-care testing (POCT) instruments based on LFIA.

KEYWORDS
lateral flow immunoassay, data processing, point of care testing, deep learning, convolutional neural network, U-Net model

1. Introduction

In vitro diagnosis (IVD) generally refers to detecting targets in the blood, urine, sweat, saliva, tissue fluid, or tissue outside the body, and is mainly used to diagnose diseases, prevent infections, manage chronic diseases, track pathological changes, evaluate therapeutic effects, and other aspects of health care (Yang et al., 2021; Peng et al., 2022). Currently, the instruments used for IVD include biochemical, immunological, molecular, microbial, and blood diagnosis as well as point-of-care testing (POCT) (Haung and Ho, 1998; Xiao and Lin, 2015;

Chen et al., 2017; Vila et al., 2017; Li et al., 2020; Liao et al., 2021). Compared with previous instruments, POCT has the characteristics of high speed, convenience, and low cost; therefore, it has received considerable attention from the medical industry (Singer et al., 2005; Damhorst et al., 2019).

Point-of-care testing is a patient-centered method for rapid sample detection using portable analytical instruments or simple reagents (Luppa et al., 2011; Florkowski et al., 2017). There are many kinds of POCT instruments, among which the lateral flow immunoassay (LFIA), based on paper-based and fluorescence detection technology, is increasingly being applied (Chen and Yang, 2015). It has the advantages of being cheap, lightweight, and easy to handle, and the fluorescence detection method can realize the quantitative detection of the sample. Both of them make LFIA highly competitive, especially for developing countries where budget is an important criterion, which is a good choice (Wu et al., 2018).

According to the published literature, LFIA technology has successfully realized the detection of biomarkers in many fields. Our research group combined many medical units using fluorescent microsphere labeling and immunochromatography technology to successfully detect COVID-19 and evaluated the analytical ability and clinical application of this technology (Zhang et al., 2020). Hu et al. (2016) developed a highly sensitive quantitative lateral flow analysis method for protein biomarkers using fluorescent nanospheres (FNs) as materials, which can be used to detect the concentration of CRP in the human body with a detection limit of 27.8 pM. Lee et al. developed a novel portable fluorescence sensor that integrates a lateral flow assay with quantum dots (Qdots) labeling and a mobile phone reader for the detection of Taenia solium T24H antibodies in human serum (Lee et al., 2019). Huang et al. (2020) used a double-antibody sandwich immunofluorescence method based on the combination of nano europium (EUNP) and lateral flow immunoassay (LFIA) to detect IL6 with a wide linear range (2–500 pg/ml) and high sensitivity (0.37 pg/ml) (Huang et al., 2020). Shao et al. (2017) used the double-antibody sandwich immunofluorescence method combined with the time-resolved immunofluorescence (TRFIA) and lateral flow immunoassay (LFIA) to detect human procalcitonin with high sensitivity (0.08 ng/ml). Gong et al. (2019) developed a miniaturized and portable UCNP-LFA platform that can be used to detect small molecules (ochratoxin A, OTA), heavy metal ions (Hg2+), bacteria (Salmonella, SE), nucleic acids (hepatitis B virus, HBV), and proteins (growth-stimulating expressed gene 2, ST-2).

As shown in **Figure 1**, there are two schemes of fluorescence detection technology for LFIA: a photoelectric scanning data acquisition platform based on Si photodiode, which is the current mainstream technology because of better performance, and a data acquisition platform based on CCD photography (Shao et al., 2019). The classical method of LFIA data processing is to obtain the C-/T-lines of the strip by peak-finding method. In this way, the normal peak and interference peak or noise peak cannot be distinguished, and wrong peak is easy to be regarded as normal peak, thus giving wrong detection result. These methods still perform poorly in effectively identifying weak and overlapping peaks while maintaining a low false-discovery rate. Qin et al. (2020) used a U-Net neural network, a variant of the convolutional neural network (CNN), to achieve the region of interest (ROI) containing T-/C-lines of test strips, and which was only used for CCD photography. In this study, we proposed a novel data processing method, which can be applied to both CCD photography and photoelectric scanning data acquisition platform. When applied to CCD photography, it only needed to convert the data to one dimension, which can be done by averaging the same row pixels parallel to the fluorescent band. This method greatly reduced the failure rate of peak finding, which can reduce the customer's need for instrument technical support, and provided a new direction for the data processing of POCT instruments based on LFIA.

Compared with the classical peak-finding method, method proposed in this study has the following advantages:

(1) Classical peak-finding methods combined with threshold-based techniques do not have the ability to identify peak shapes. They can only find local maxima according to certain rules, and cannot accurately identify certain noise signals as invalid data. For example, according to the setting rules in section "3.4. Comparison with classical methods," they will misjudge peak 1 as C-peak in **Figures 2A–G**, and misjudge peak 2 as T-peak, resulting in incorrect detection results. They will also misjudge peak 1 as C-peak in **Figure 2H**, and no T-peak can be found, resulting in a false concentration of 0. In fact, all the data listed in **Figure 2** were judged invalid by the technician. Due to the diversity of sample types and detection items, coupled with some problems in user operation, various invalid data could be generated. The classification model based on deep learning proposed in this study has ability to distinguish peak shape, and it can identify these invalid data as noise (class 1) or only T-peak (class 3), thus solving this problem well.

(2) Classical peak-finding methods cannot solve the problem of interference peaks, especially the interference peaks around weak T-peak, as shown in **Figure 4**. Interfering peaks may appear anywhere, to the left or right of valid peak. Classic peak-finding methods combined with threshold-based techniques, such as setting an interval range for the positions of C-peak and T-peak or setting a threshold for the height of C-peak, are not completely reliable. Because the positions of C- and T- peaks will change with assembly position of nitrocellulose membrane, insertion position of test strip, difference between different instruments, sampling speed and so on, errors will occur when the set range is exceeded. For example, the classic peak-finding methods will misjudge peak 1 as C-peak in **Figure 4B**, and misjudge peak 2 as T-peak in **Figure 4C** and peak 1 or 2 as T-peak in **Figure 4D**. In addition, the classic peak-finding methods perform poorly when looking for weak T-peak. They often fail to find T-peak and misjudge the tailing peak (peak 2 in **Figures 4E, F**) as T-peak. Similarly, the improved U-net segmentation model proposed in this study has ability to distinguish shape of peaks, which can solve this problem well.

(3) For classical methods, a minimum threshold is generally set for the height of C-peak. If the threshold is too small, accuracy will be greatly reduced due to presence of interference peaks or invalid data. If the threshold is too large, it will be unfavorable to process data with low height of C-peak in test strips of competition method. This is an unavoidable shortcoming of classical methods, but the method proposed in this study does not have this problem.

(4) Method proposed in this study can enhance its generalization ability by constantly learning new type data, but classical algorithm obviously does not have this ability. They are only some fixed peak-finding rules and threshold judgments, and cannot accurately identify some noise peaks similar to valid

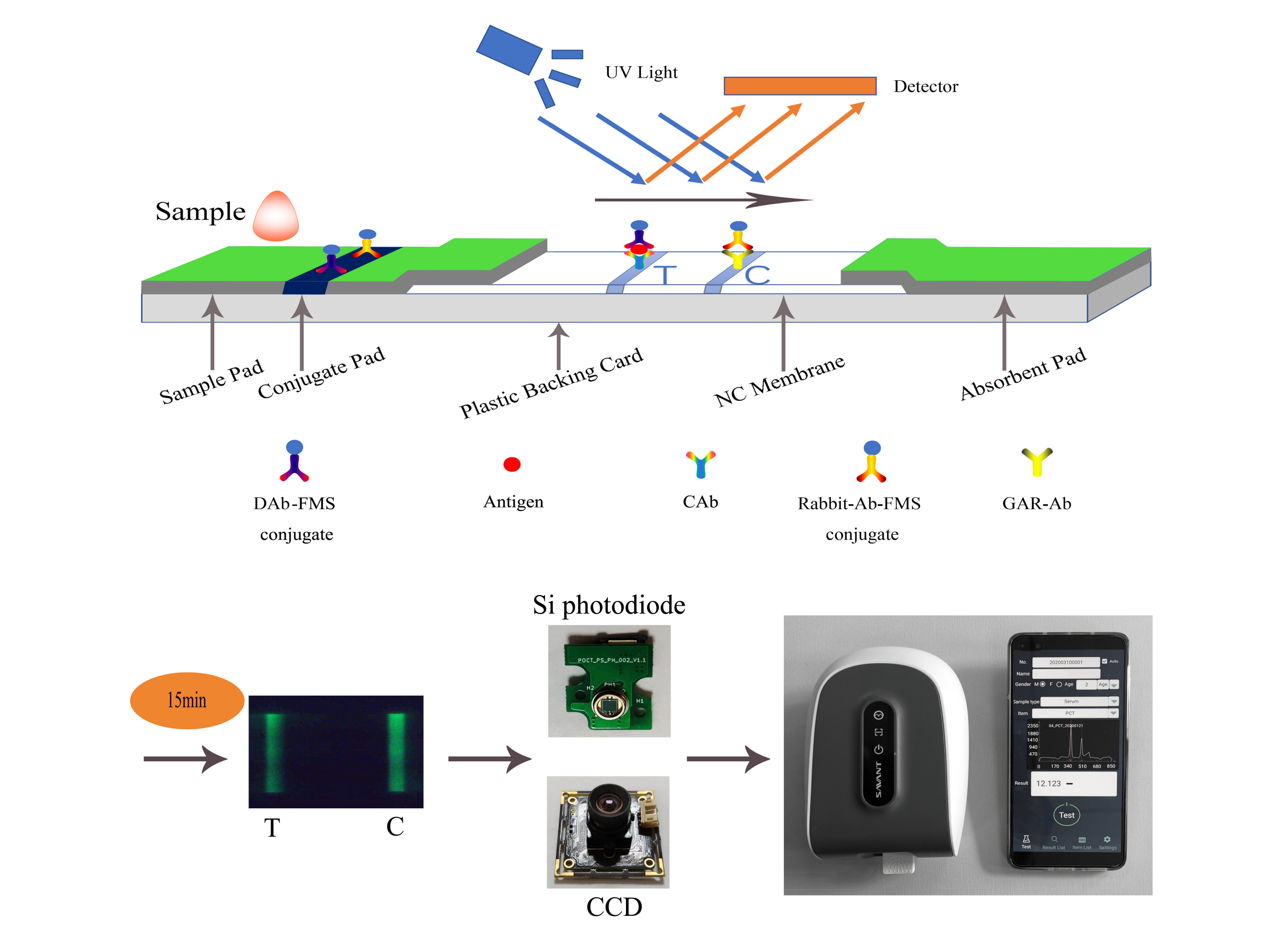


FIGURE 1
Schematic diagram of LFIA. A hand-held fluorescence immunoassay analyzer which was used to measure fluorescent intensity controlled by a mobile phone *via* Bluetooth. Its sensor can be CCD or Si photodiode.

peaks. In particular, the noise data is ever-changing, and it is difficult for classical methods to be suitable for every new type of data.

2. Materials and methods

2.1. Materials

The data used for training, validation, and testing in this study were obtained from Beijing Savant Biotechnology Co., Ltd. These data are the result of testing a variety of items. The detection items mainly included human ferritin, vitamin D, D-dimer, and C-reactive protein and so on. The sample types mainly included whole blood, serum, and plasma.

2.2. Principle of LFIA

A double-antibody sandwich test strip with fluorescent microspheres (FMS) as the carrier was used to illustrate the detection principle of LFIA. The double-antibody sandwich structure is shown in **Figure 1**. The test strip was composed of a sample pad, conjugate pad, nitrocellulose membrane (NC membrane), absorbent pad, and plastic backing card. After the sample was dripped into the sample pad, it was subjected to immunochromatography under capillarity. The detection antibody-FMS (DAb-FMS) and rabbit IgG antibody-FMS (Rabbit-Ab-FMS) were placed on the conjugate pad. There are T and C lines on the NC membrane; the T line is coated with capture antibody (CAb), and the C line is coated with goat anti-rabbit IgG antibody (GAR-Ab). The absorbent pad causes liquid to flow *via* capillary action. The plastic backing card plays the role of fixing and supporting.

When the sample solution containing the analyte was added to the sample pad, it was laterally transferred along the NC membrane *via* capillary action. When the sample flowed through the conjugate pad, the Antigen in the sample reacted with DAb to form a DAb-FMS/Antigen complex. When the complex flows to the T line in the NC membrane, the Antigen and CAb on the T line are immunized to form a DAb-FMS/Antigen/CAb complex. Rabbit-Ab-FMS, which does not participate in the reaction, continues to flow forward to the C line and reacts with GAR-Ab.

Generally, the entire reaction process takes approximately 15 min. After immunochromatography is completed, the excitation light generated by the scanning mechanism irradiates the T and C lines, and fluorescence is generated. In the process of scanning the

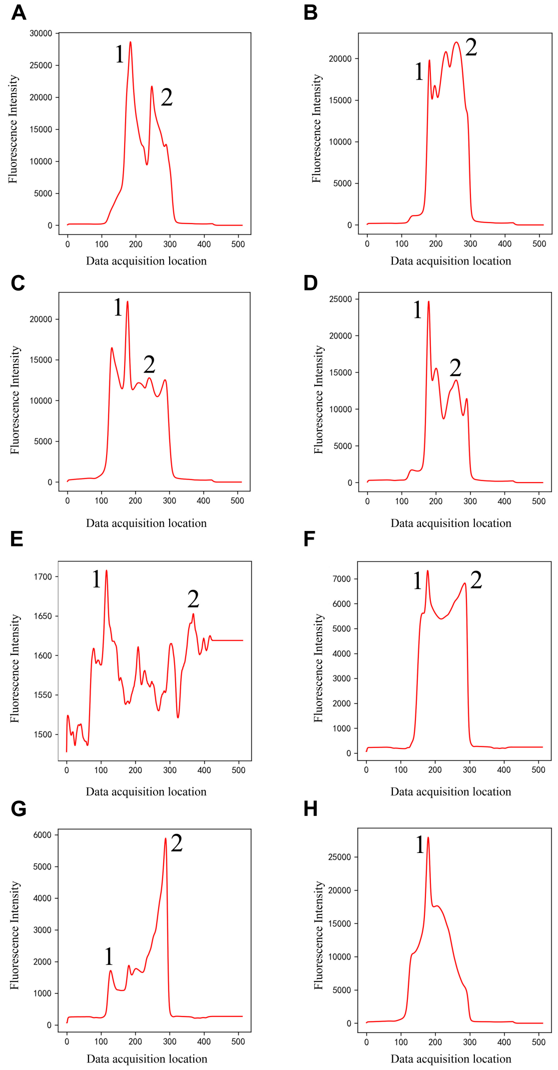


FIGURE 2
Noise (Class 1) of different shapes **(A–H)**. The classical methods will misjudge peak 1 as C-peak in panels **(A–G)**, and misjudge peak 2 as T-peak, resulting in incorrect detection results. They will also misjudge peak 1 as C-peak in panel **(H)**, and no T-peak can be found, resulting in a false concentration of 0.

NC membrane, the fluorescence intensity produced at each point of the scan was recorded using a photodiode, and the peak data shown in Figure 1 were finally formed. The ratio of the fluorescence intensities of the two lines can be obtained by calculating the ratio of the peak areas of the T- and C-peaks. The concentration of the antigen detected in the sample was proportional to the T/C. By establishing a standard curve, the concentration of the antigen detected in the sample can be calculated.

2.3. Data augmentation

During the testing of clinical samples, four different peak shape data were obtained: noise (class 1), only C-peak (class 2), only T-peak (class 3), and double-peaks (C-peak and T-peak, class 4). Class 1 was generated by a fluorescence analyzer scanning the fouled NC membrane, whereas class 2 was generated by detecting the sample with a concentration of 0. However, the data of class 3 were very few, and were generally generated from the test strips with the disappearance of the C-peak. To better train the model, the C-peak of the double-peaks (class 4) was deleted and transformed into the background by a cubic spline interpolation method; thus, a large amount of data containing only the T-peak was generated manually.

2.4. Label annotation

To train the model, a large amount of labeled data is required. Data annotation is a complex process, and the quality of the annotation directly affects the results of the model training. This method includes two steps corresponding to a classification and a segmentation model, and the training data of the two models must be annotated separately. The labeled dataset was randomly divided into training and verification sets.

The entire dataset for the classification model includes approximately 4,100 detection data, including four types of peak shapes, namely, noise, only C-peak, only T-peak, and double-peaks. These four types of data were encoded according to one-hot, which were noise (class 1), only C-peak (class 2), only T-peak (class 3), and double-peaks (class 4), as shown in Figures 2–4. There were approximately 900 data for noise (class 1), 900 for only C-peak (class 2), 900 for only T-peak (class 3), and 1,400 for double-peaks (class 4). The peak shape of the detection data is particularly complex and diverse, and only a few typical ones are selected for display here.

The dataset of the segmentation model includes approximately 1,400 pieces of detection data, that is, all the data of class 4. In this study, based on the Python language, software was designed to annotate the integral regions of the T-peak and C-peak, and the integral regions of the C-peak and T-peak of 1,400 fluorescence detection data were annotated.

2.5. Network architecture

Convolutional neural network is an artificial neural network specially designed to process data such as images or videos. It generally has three layers, namely, convolution, pooling and full connection layer. In the convolution layer, input samples are convolved with kernel, and the discrete convolution function is defined as:

$$(f * g)(x) = \sum_{\tau} f(\tau) \cdot g(x - \tau)$$

where f and g are two functions.

Pooling is used to extract high-dimensional features, and the most commonly used ones are maximum and average pooling. In a fully connected layer, all neurons in the current layer are interconnected with every neuron in the next layer.

As shown in Figure 5, the entire data-processing flow consists of two steps. First, a classification model was used to classify the input data. Second, after analyzing the input data, if the output result was class 4 (double-peaks), the data were imported into the next segmentation model to realize the data segmentation of the C-peak and T-peak areas.

The input of the classification model had two channels. Because the fluorescent signal has strong background noise, we subtracted

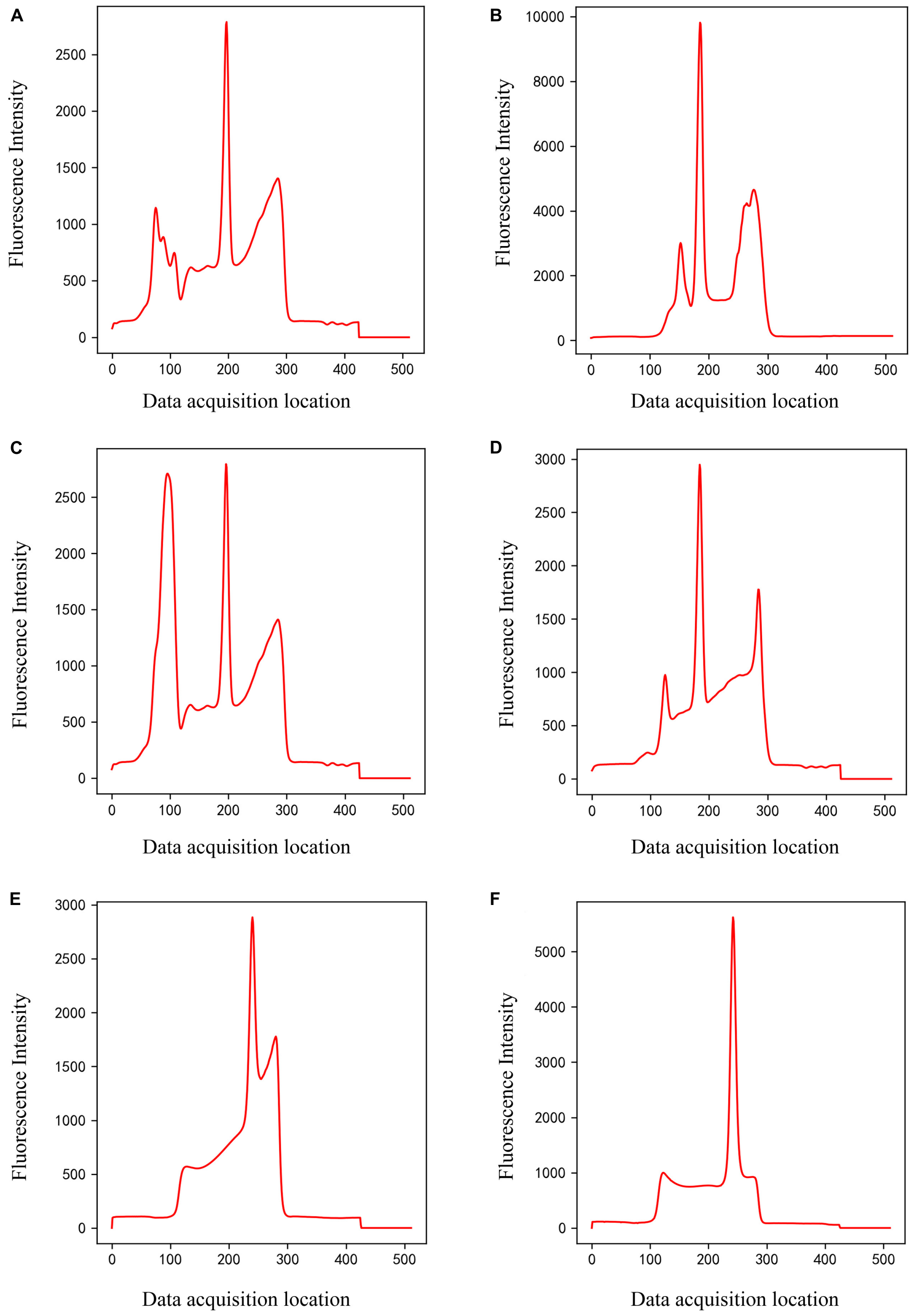


FIGURE 3
Panels **(A–D)** are only C-peak (Class 2) of different shapes, and panels **(E,F)** are only T-peak (Class 3) of different shapes.

the background and then normalized it as the first channel. It was achieved by the following formula.

$$Y_1 = \frac{X - x_{min}}{x_{max} - x_{min}}$$

where Y_1 is the first channel, X is raw input data, x_{min} is minimum value, and x_{max} is maximum value of raw data. In order to make the model learn the peak shape rather than intensity, we performed a logarithmic operation on the signal which was deducted background as the second channel. It was achieved by the following formula.

$$Y_2 = \frac{log_{10}(X - x_{min})}{log_{10}(x_{max} - x_{min})}$$

where Y_2 is the second channel, X is raw input data, x_{min} is minimum value, and x_{max} is maximum value of raw data.

The network architecture of the classification model is illustrated in **Figure 5A**. The entire network architecture consisted of 10 layers; the first seven layers were conv1d + ReLU + MaxPool1d

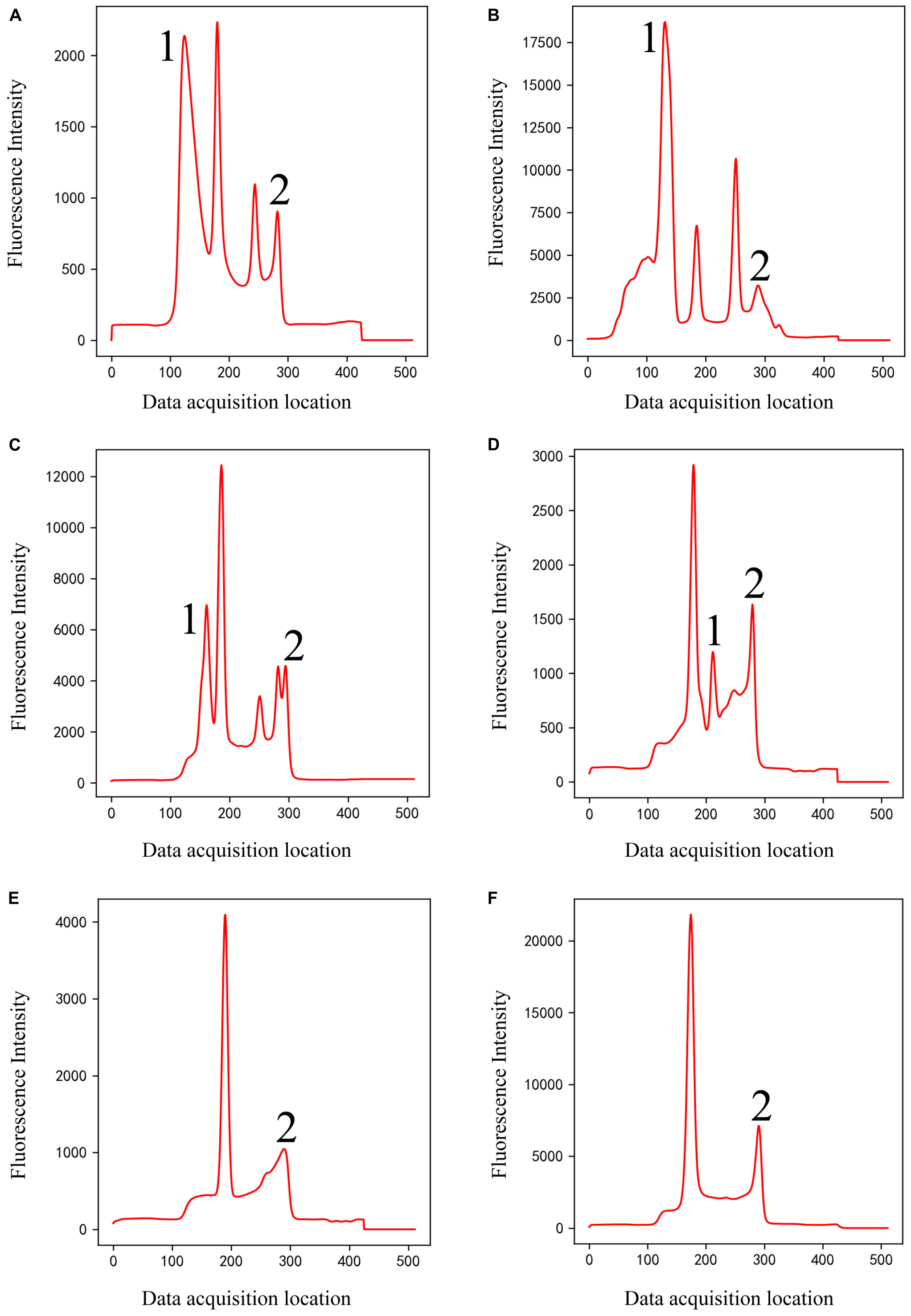


FIGURE 4
Double-peaks (Class 4) of different shapes **(A–F)**. Peaks 1 and 2 are interference peaks. The classic methods will misjudge peak 1 as C-peak in panel **(B)**, misjudge peak 2 as T-peak in panel **(C)**, and peak 1 or 2 as T-peak in panel **(D)**. They often fail to find T-peak and misjudge the tailing peak [peak 2 in panels **(E,F)**] as T-peak.

(Acharya et al., 2017; Gu et al., 2018; Zhang et al., 2019), and the input data were extracted into four features of high-dimensional 1,024 channels. The eighth layer extended the number of channels to 2,048. Next, Max + Transposition was used to extract the maximum value from the four high dimension features (Gu et al., 2018). To improve the accuracy of classification, we used dropout layer before fully connected layers. The last layer (Dropout + Fully-connected + SoftMax) classified the data into one of four classes (Srivastava et al., 2014).

We designed an improved U-Net segmentation model with reference to the classic U-Net model (Ronneberger et al., 2015); the network architecture is shown in **Figure 5B**. We changed

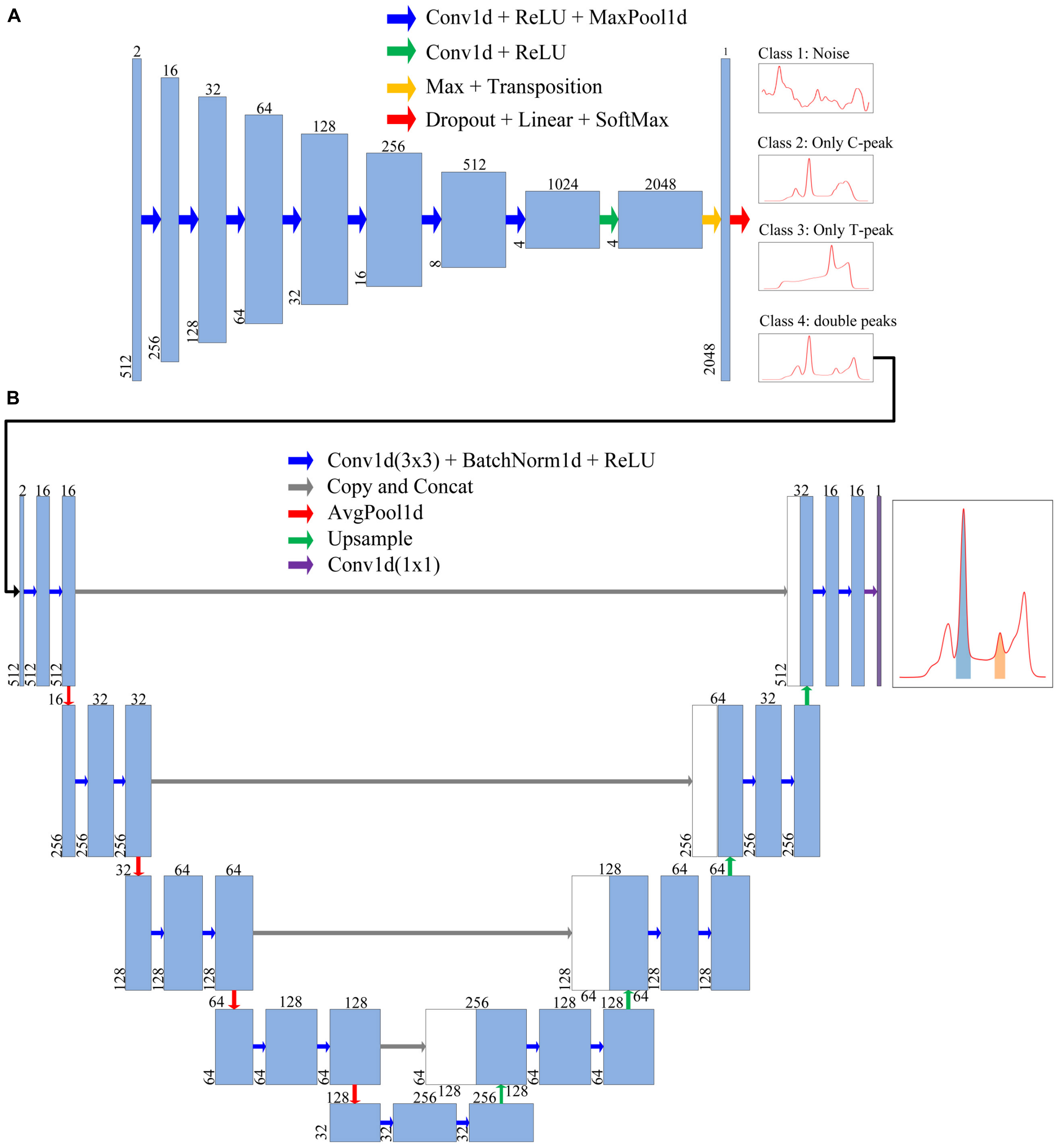


FIGURE 5

Each blue box represents a feature map of a layer. Number above the blue box is the number of channels, whereas number in lower left corner is the number of data points. The arrows represent different operations. **(A)** Neural network architecture of the classification model. The first blue box represents the format of input data. After being processed by the classification model, the input data were finally classified into four classes, namely, Class 1 (Nosie), Class 2 (Only C-peak), Class 3 (Only T-peak), and Class 4 (double-peaks). **(B)** Neural network architecture of segmentation model, through which the ROI of test strip containing T-/C- peak can be extracted and obtained.

the input data into two channels. This model had four parts: input unit, encoding structure, decoding structure, and output unit (Ronneberger et al., 2015; Oh et al., 2019; Wang et al., 2021; Zheng et al., 2021; Zunair and Ben Hamza, 2021). The encoding structure used four units to reduce the dimensions, and the number of feature maps was increased gradually. In order to reduce training time, we added batch normalization after each convolution (Melnikov et al., 2020). In the decoding structure, each step was symmetrical with the encoding part to recover data. The upsampling section allowed the network to propagate the context information to a higher-resolution layer. In the last layer, the discrimination of whether each point in fluorescence data belonged to an integral region was realized.

2.6. Loss function

The classification model classified the data into one of four classes, which is a problem of four classes. Multi-classification neural networks generally use cross-entropy loss as a loss function. The mathematical expression of this loss function in the program is:

$$L_{CE} = -\frac{1}{M}\sum_{j=1}^{M}\sum_{i=1}^{C} y_{ij}\, log\, o_{ij}$$

where M is the batch size, C is the total number of classes (four), y_{ij} is the real label, and o_{ij} is the predictive output.

The Dice coefficient (also known as the Dice score or DSC) is a function of the set similarity measurement, which is usually used to calculate the similarity between two sets (Saeedizadeh et al., 2021), with values ranging from 0 to 1. Here, it was used to measure the overlap between the ground-truth and predicted masks, where 0 indicates no overlap and 1 indicates complete overlap.

$$DSC\,(A, B) = \frac{2\,|A \cap B|}{|A| + |B|}$$

where A and B denote the predicted and ground-truth masks.

To minimize the loss function, we used the 1-DSC as the final loss function. The mathematical expression of this loss function in the program is:

$$L_{DSC} = 1 - \frac{1}{M}\sum_{j=1}^{M}\frac{2\sum_{i=1}^{N} y_{ij}\, o_{ij}}{\sum_{i=1}^{N} y_{ij} + \sum_{i=1}^{N} o_{ij}}$$

where M is the batch size, N is the number of sample data, y_{ij} is the ground-truth mask, and o_{ij} is the predictive mask.

For unbalanced sample data, weighted binary cross entropy can be used as the training loss function. Therefore, compared with the standard cross-entropy loss, better results can be obtained when the number of positive and negative points is unbalanced (Zhu et al., 2019). The mathematical expression of this loss function in the program is:

$$L_{WBCE} = -\frac{1}{M \times N}\sum_{j=1}^{M}\sum_{i=1}^{N}\left(w_1 y_{ij}\, log\, o_{ij} + w_0 (1 - y_{ij})\, log(1 - o_{ij})\right)$$

where M is the batch size, N is the number of sample data, y_{ij} is the ground-truth mask, and o_{ij} is the predictive mask. w_1 and w_0 correspond to the weights labeled 1 and 0, respectively.

In this study, the mathematical expression for the weight parameter w_c is:

$$w_c = \frac{N - N_c}{N}$$

where N represents the total number of data points for each sample and N_c represents the number of data points in class c.

2.7. Model hyper-parameters of models

After labeling the data, we trained the model. The classification and segmentation models were trained separately. The training parameters of the classification and segmentation model are listed in Table 1.

3. Results

3.1. Evaluation metrics of models

Accuracy, which is the proportion of correctly predicted samples to the total number of samples, is generally used as the evaluation metric of a multi-classification model. The mathematical expression of accuracy in the program is:

$$Accuracy\,(y, o) = \frac{1}{M}\sum_{i=1}^{M} 1\left(o_i = y_i\right)$$

where M denotes the batch size, y_i denotes the real label, and o_i is the predictive output.

The intersection over union (IoU), also known as the Jaccard index, calculates the ratio of the intersection and union of the ground-truth and predicted segmentation masks (Saeedizadeh et al., 2021). It can be used to measure the similarity between the ground-truth and predicted segmentation masks; the higher the similarity, the higher the value.

$$IoU\,(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

where A and B denote the predicted and ground-truth masks. The mathematical expression of IoU in the program is:

$$IoU\left(y, o\right) = \frac{1}{M}\sum_{j=1}^{M}\frac{\sum_{i=1}^{N} y_{ij}\, o_{ij}}{\sum_{i=1}^{N} y_{ij} + \sum_{i=1}^{N} o_{ij} - \sum_{i=1}^{N} y_{ij}\, o_{ij}}$$

where M is the batch size, N is the number of sample data, y_{ij} is the ground-truth mask, and o_{ij} is the predictive mask.

3.2 Model hyper-parameters optimization of segmentation model

Both the weight coefficients of the weighted binary cross-entropy and cut-off threshold have a certain influence on the performance of the model. To obtain appropriate weights and cut-off thresholds, this study conducted cross experiments on weights and cut-off thresholds. As presented in Table 2, when $w_0 : w_1 = 0.6 : 0.4$ and the cut-off threshold = 0.6, the IoU achieved a maximum value of 0.9680. The other parameters used during the training are listed in Table 1.

We also compared the three loss functions of WBCE, DSC, and WBCE + DSC. When the other conditions were the same, the combined loss function (WBCE + DSC) was used to obtain the maximum IoU, as illustrated in Table 3.

TABLE 1 Important parameters used in two models training.

Network parameters	Classification model	Segmentation model
Batch size	8	8
Epoch	30	100
Activation function	ReLU	ReLU
Padding mode	MaxPool	AvgPool
Pooling size	2	2
Optimizer	Adam	Adam
Learning rate	0.001	0.001
Convolution kernel	3	3
Upsample	–	Nearest
Input size	512 × 2	512 × 2

TABLE 2 Cross experiments result on weights of the weighted binary cross entropy and cut-off threshold.

w_0:w_1	IoU (Cut-off = 0.3)	IoU (Cut-off = 0.4)	IoU (Cut-off = 0.5)	IoU (Cut-off = 0.6)	IoU (Cut-off = 0.7)
0.3:0.7	0.9668	0.9652	0.9665	0.9677	0.9660
0.4:0.6	0.9674	0.9657	0.9658	0.9674	0.9674
0.5:0.5	0.9668	0.9674	0.9674	0.9665	0.9670
0.6:0.4	0.9677	0.9676	0.9666	0.9680	0.9674
0.7:0.3	0.9668	0.9674	0.9652	0.9669	0.9654

When $w_0 : w_1 = 0.6 : 0.4$ and the cut-off threshold = 0.6, the IoU achieved a maximum value of 0.9680.

TABLE 3 Overall performance with different loss functions, $w_0 : w_1 = 0.6 : 0.4$ and cut-off threshold = 0.6.

Loss	w_0:w_1	Cut-off	IoU
WBCE	0.6:0.4	0.6	0.9652
DSC			0.9586
WBCE + DSC			0.9680

3.3. Training convergence analysis of models

Figure 6A shows the loss curves of different epochs during the classification model training process, and Figure 6B shows the accuracy of the training and validation sets corresponding to different epochs. The maximum accuracy of the model validation set was 99.59%.

To analyze which samples were misclassified, we built confusion matrix. As in Figure 6C, only five samples were misclassified, two class 1 and three class 2 data were misclassified as class 4. These five samples had the characteristics of two different classes, which leaded to misclassification. In general, such samples are rare.

Figure 7A shows the loss curves of different epochs during the segmentation model training process, and Figure 7B shows the IoU of the training and validation sets corresponding to different epochs. The maximum IoU of the model validation set was 0.9680.

3.4. Comparison with classical methods

There are many types of peak detection methods, such as the direct peak location, Fourier transform, cumulative sum derivative, curve fitting, devolution, and wavelet transform (CWT) methods (Deng et al., 2021). The direct peak location according to the properties of peak and continuous wavelet transform are two classical methods in traditional methods. The principle of direct peak location is to find out all the local maxima of the signal through the simple comparison method, and then select the subset of these peaks according to the specified peak properties. The method principle of CWT is that the signal is first transformed by CWT in certain scales, and then the ridges are found in the CWT matrix. The positions of these ridges correspond to the positions of all peaks (Du et al., 2006). Using the verification set, method proposed in this paper was compared with the two traditional methods. These two methods have been implemented in SciPy library based on Python, so we directly used the related functions (find_peaks() and find_peaks_cwt()) in SciPy library.

Classical peak-finding methods can only find the local maxima of the signal, and do not have the ability to classify the signal. Here, after obtaining the local maxima through the classical methods, some subsequent processing steps were adopted to make it have the classification ability, and then compared with the classification model proposed in this study. These subsequent processing steps are as follows:

(1) According to the characteristics of the strip, the data of 512 sample points are divided into C peak region (0–220) and T peak region (221–511).
(2) Judge whether there are local maxima in the C peak region (0–220), and if so, take the maximal local maximum as the C peak. Judge whether the height of the C peak is greater than 1,000, and if it is greater than 1,000, it is considered to be an effective C peak (according to the characteristics of the strip, the height of the C peak is usually greater than 1,000).
(3) Judge whether there are local maxima in the T peak region (0–220), and if so, take the maximal local maximum as the T peak.
(4) According to the results of (2) and (3), the signal is classified to noise (class 1), only C-peak (class 2), only T-peak (class 3), and or double-peaks (class 4).

The comparison results are shown in Table 4. It can be seen that the performance of the two classical methods is similar in term of accuracy, one is 80.10%, the other is 80.76%. Accuracy of the method proposed in this study is 99.59%, which is much better than classical methods.

For two classical methods, the function of peak_widths() in the SciPy library can be used to identify the integral region. Compared with the segmentation method in this study in terms of IoU, Dice, Recall and Precision. The results are shown in Table 4. As can be seen from the table, no matter which evaluation term it is, the method proposed in this paper is much better than two classical methods.

3.5. Test of the method

The method proposed in this study was tested using instrument test data. First, the ability of the segmentation model to segment various peak shapes was tested. Next, three most important indicators (standard curve, repeatability, and recovery) were tested.

After training, the method can classify raw input data into one of four classes and perform data segmentation on data belonging to Class 4. The segmentation model could effectively segment C- and T-peak regions from fluorescence intensity of 512 data points. Figure 8 shows examples of data segmentation results for some

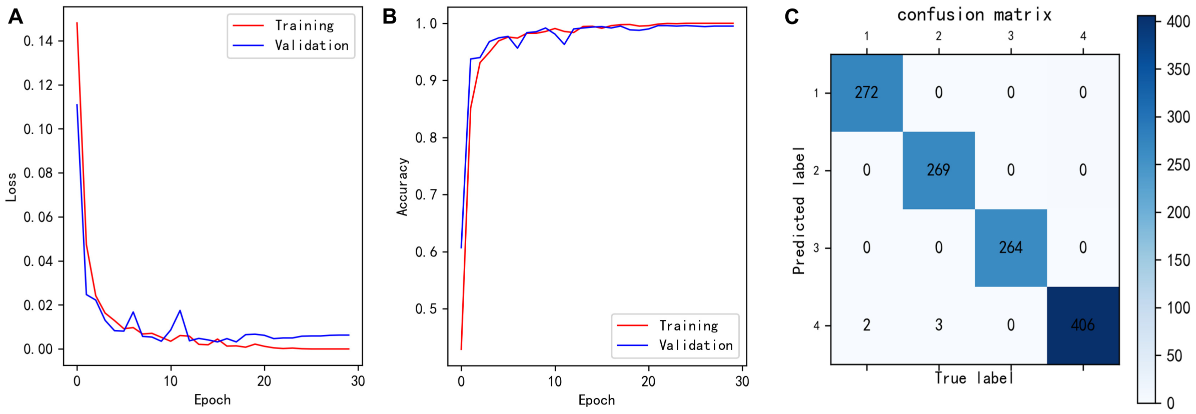


FIGURE 6
(A) Loss of classification model during training. **(B)** Training and validation accuracy of classification model during training. **(C)** Confusion matrix showing the result of trained classification model for validation set. The row number reflects the predicted label, and column number reflects the true label.

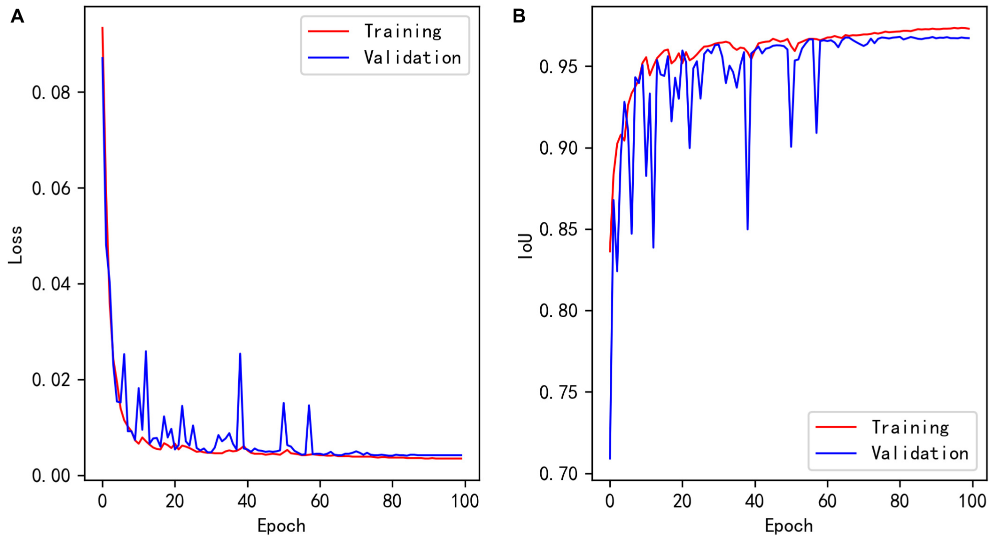


FIGURE 7
(A) Loss of segmentation model during training. **(B)** Training and validation IoU of segmentation model during training.

TABLE 4 **Comparison of classical peak-finding methods and proposed method performance in terms of accuracy, IoU, dice, recall and precision.**

Method	Accuracy	IoU	Dice	Recall	Precision
Direct peak location	80.10%	0.7753	0.8509	0.8801	0.8391
CWT	80.76%	0.7597	0.8423	0.8510	0.8541
Our method	99.59%	0.9680	0.9836	0.9857	0.9821

typical peak shapes, where the orange shaded areas are segmented C- and T-peak regions. Figure 8A shows segmentation of the normal peak shape, and C -and T-peak regions were accurately extracted and obtained. Figures 8B, C show that in the presence of overlapping and interference peaks, C- and T-peaks can be accurately segmented. Figures 8D–F show the segmentation results for weak T-peak with baseline drift, tailing or interference peak. As shown in the figure, baseline drift, tailing and interference peaks did not affect accurate segmentation of the data; the detection of weak T-peak region is also excellent. After data were imported into the segmentation model, they were first normalized. The network model only focused on learning the shape of entire data set and did not learn the value of fluorescence intensity. The experimental results indicate that it can meet the requirements of LFIA for data processing.

The method was tested using Ferritin. A standard curve was established using a range of concentrations (0, 15, 50, 200, 300, and 500 ng/ml) of the standards. Each concentration of the standard was tested three times using test strips. The detection data were processed using proposed method. First, data were classified, then segmented, and finally, the segmented regions were integrated and T/C was calculated. The method accurately classified the detection data of 0 ng/ml as class 2 (only C-peak), and the corresponding

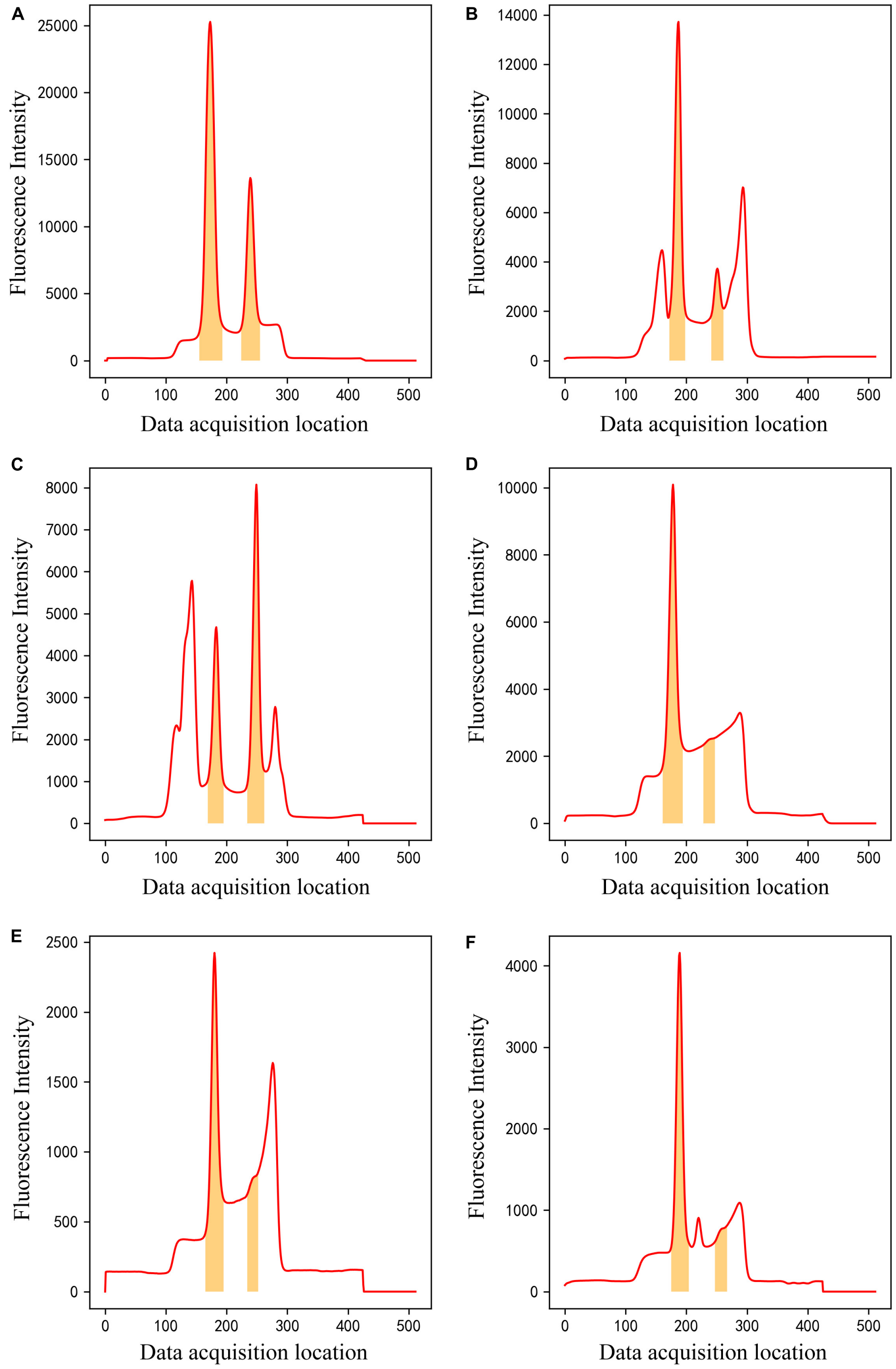


FIGURE 8
The results of ROI extraction by segmentation model on different kinds of data. **(A)** Normal peak data, **(B,C)** overlapping and interference peak data, **(D–F)** Weak T-peak with baseline drift, tailing or interference peak data.

T/C values were 0. The remaining data were classified as class 4 (double-peaks) and then segmented. Using T/C as the ordinate and concentration as the abscissa, a standard curve was established using four parameters, as shown in **Figure 9**. It can be observed that the T/C and concentration have a good correlation with a correlation coefficient of 0.9986. This shows that the method is effective in dealing with LFIA data.

Three concentrations (20, 220, and 400 ng/ml) of the reference standards were tested for repeatability using the same batch of test strips. Each concentration was tested 10 times, and the CV values were calculated. The data were processed using the method described in this study. The data for all the three concentrations were classified as class 4 (double-peaks). The data were segmented, and concentrations were calculated; the results are listed in **Table 5**. It can be observed that the CV values of three concentrations are all good, and the maximum does not exceed 1.37%. This shows that the stability of the method is good.

Recovery was tested using samples of three concentrations (40, 100, and 150 ng/ml). Each sample was tested thrice. The method in

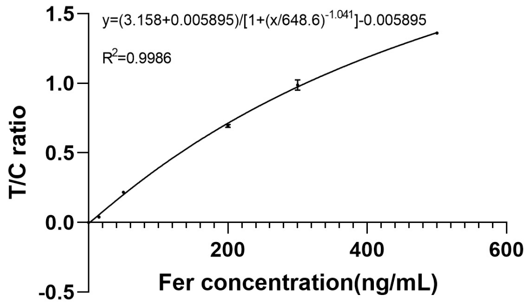


FIGURE 9
Four parameter fitting line for ferritin detection in the range of 0–500 ng/ml.

TABLE 5 Precision results of ferritin test strips.

Mean (ng/ml)	SD	CV (%)
17.561	0.240	1.37
212.541	1.274	0.60
369.034	1.401	0.38

TABLE 6 Recovery rates results of Ferritin test strips.

Concentration (ng/ml)	Mean (ng/ml)	Recovery rate (%)
40	42.030	105.07
100	96.371	96.37
150	149.502	99.67

this study was used to process the data, and all the data were classified as class 4 (double-peaks); the results are listed in Table 6. The calculated recovery rates were 105.07, 96.37, and 99.67%, respectively. This shows that concentration calculated by the method is very accurate.

4. Discussion

Because POCT instruments based on LFIA detection technology are used in a variety of situations and there are many different types of samples, the peak shape of the test data is complex. It is difficult for classical peak-finding methods to deal with all peak shapes. The data-processing method proposed in this study has several advantages.

First, through a classification network, the peak types were classified into four classes, and the peak types that needed to be calculated for the concentration were screened. In this manner, the data processing difficulty of the segmentation model is reduced, and the model can easily achieve better performance. Second, an improved U-Net-based segmentation model directly identifies the integration regions, replacing the operations of the peak finding, peak start and end location in the classical method, which makes the data processing process more accurate and convenient. It is very difficult to determine the starting point and ending point of the peak accurately by the traditional method. Our segmentation model can easily solve this problem. Third, through experiments, it was found that this U-Net -based segmentation method also performs well in effectively identifying weak and trailing peaks. Forth, the classical peak-finding methods can only find the local maxima of the signal, and do not have the ability to classify the signal. In this case, it is difficult to distinguish the noise peak from the effective peak. Our classification model has perfectly solved this problem.

The method was applied to the hand-held immunofluorescence analyzer developed by ourselves and good results were obtained. Interference peaks are the biggest obstacle in the use of hand-held instruments, and often lead to peak-finding errors. The use environment of hand-held instruments is flexible and changeable, which makes it inconvenient to provide technical support. This method greatly reduced the failure rate of peak finding, which can reduce the customer's need for instrument technical support. This is a great advantage for hand-held instruments sold in large quantities.

5. Conclusion

In this study, a deep-learning-based LFIA photoelectric scanning data-processing method was proposed. The entire method had two steps. The first step was to build a CNN classification model to classify the LFIA peak shape and screen out the data required to calculate the concentration. The second step was to build an improved 1D U-Net segmentation model to achieve the segmentation of C- and T-peak integration regions for data containing double-peaks and then perform calculations such as T/C and concentration. A large amount of experimental data were used to train the two models. The accuracy of classification model on validation set was 99.59% and the IoU of segmentation model on validation set was 0.9680. Using the data-processing method, a standard curve was established for Ferritin, and the CV and recovery rate, the two most relevant indicators in clinical testing, were tested. The CV values corresponding to the three concentrations of 20, 220, and 400 ng/ml were 1.37, 0.60, and 0.38%, respectively. The recovery rates corresponding to the three concentrations of 40, 100, and 150 ng/ml were 105.07, 96.37, and 99.67%, respectively. These experimental results show that the data-processing method proposed in this study can be used for the processing of LFIA photoelectric scanning data, and the obtained results are accurate and reliable, which proposes a new direction for POCT instrument data processing.

Data availability statement

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

Author contributions

XL and YW involved in the conception and research design. XL, KD, and SL collected the data and annotated the data. XL performed the statistical analysis and wrote the manuscript. XL, KD, SL, and YW revised it for publication. All authors contributed to the article and approved the submitted version.

Acknowledgments

We would like to thank the colleagues in Tianjin Boomscience Technology Co., Ltd., and Beijing Savant Biotechnology Co., Ltd., for their friendly help. We would also like to thank Editage (www.editage.cn) for English language editing.

Conflict of interest

KD was employed by Tianjin Boomscience Technology Co., Ltd. SL was employed by Beijing Savant Biotechnology Co., Ltd.

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

Acharya, U. R., Oh, S. L., Hagiwara, Y., Tan, J. H., Adam, M., Gertych, A., et al. (2017). A deep convolutional neural network model to classify heartbeats. *Comput. Biol. Med.* 89, 389–396. doi: 10.1016/j.compbiomed.2017.08.022

Chen, A., and Yang, S. (2015). Replacing antibodies with aptamers in lateral flow immunoassay. *Biosens. Bioelectron.* 71, 230–242. doi: 10.1016/j.bios.2015.04.041

Chen, F.-H., Li, N., Zhang, W., Zhang, Q.-Y., Wang, Y., Ma, Y.-Y., et al. (2017). A comparison between China-made Mindray BS-2000M biochemical analyzer and Roche cobas702 automatic biochemical analyzer. *Front. Lab. Med.* 1:98–103. doi: 10.1016/j.flm.2017.06.006

Damhorst, G. L., Tyburski, E. A., Brand, O., Martin, G. S., and Lam, W. A. (2019). Diagnosis of acute serious illness: the role of point-of-care technologies. *Curr. Opin. Biomed. Eng.* 11, 22–34. doi: 10.1016/j.cobme.2019.08.012

Deng, F., Li, H., Wang, R., Yue, H., Zhao, Z., and Duan, Y. (2021). An improved peak detection algorithm in mass spectra combining wavelet transform and image segmentation. *Int. J. Mass Spectrom.* 465:116601. doi: 10.1016/j.ijms.2021.116601

Du, P., Kibbe, W. A., and Lin, S. M. (2006). Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching. *Bioinformatics* 22, 2059–2065. doi: 10.1093/bioinformatics/btl355

Florkowski, C., Don-Wauchope, A., Gimenez, N., Rodriguez-Capote, K., Wils, J., and Zemlin, A. (2017). Point-of-care testing (POCT) and evidence-based laboratory medicine (EBLM) – does it leverage any advantage in clinical decision making? *Crit. Rev. Clin. Lab.* 54, 471–494. doi: 10.1080/10408363.2017.1399336

Gong, Y., Zheng, Y., Jin, B., You, M., Wang, J., Li, X., et al. (2019). A portable and universal upconversion nanoparticle-based lateral flow assay platform for point-of-care testing. *Talanta* 201, 126–133. doi: 10.1016/j.talanta.2019.03.105

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., et al. (2018). Recent advances in convolutional neural networks. *Pattern Recognit.* 77, 354–377. doi: 10.1016/j.patcog.2017.10.013

Haung, M. L., and Ho, C. H. (1998). Diagnostic value of an automatic hematology analyzer in patients with hematologic disorders. *Adv. Therapy* 15:137.

Hu, J., Zhang, Z. L., Wen, C. Y., Tang, M., Wu, L. L., Liu, C., et al. (2016). Sensitive and quantitative detection of C-reaction protein based on immunofluorescent nanospheres coupled with lateral flow test strip. *Anal. Chem.* 88, 6577–6584. doi: 10.1021/acs.analchem.6b01427

Huang, D., Ying, H., Jiang, D., Liu, F., Tian, Y., Du, C., et al. (2020). Rapid and sensitive detection of interleukin-6 in serum via time-resolved lateral flow immunoassay. *Anal. Biochem.* 588:113468. doi: 10.1016/j.ab.2019.113468

Lee, C., Noh, J., O'Neal, S. E., Gonzalez, A. E., Garcia, H. H., Cysticercosis Working Group in Peru, et al. (2019). Feasibility of a point-of-care test based on quantum dots with a mobile phone reader for detection of antibody responses. *PLoS Negl. Trop. Dis.* 13:e0007746. doi: 10.1371/journal.pntd.0007746

Li, N., Wang, P., Wang, X., Geng, C., Chen, J., and Gong, Y. (2020). Molecular diagnosis of COVID-19: current situation and trend in China (Review). *Exp. Ther. Med.* 20:13. doi: 10.3892/etm.2020.9142

Liao, M., Zheng, J., Xu, Y., Qiu, Y., Xia, C., Zhong, Z., et al. (2021). Development of magnetic particle-based chemiluminescence immunoassay for measurement of human procalcitonin in serum. *J. Immunol. Methods* 488:112913. doi: 10.1016/j.jim.2020.112913

Luppa, P. B., Muller, C., Schlichtiger, A., and Schlebusch, H. (2011). Point-of-care testing (POCT): current techniques and future perspectives. *Trends Analyt. Chem.* 30, 887–898. doi: 10.1016/j.trac.2011.01.019

Melnikov, A. D., Tsentalovich, Y. P., and Yanshole, V. V. (2020). Deep learning for the precise peak detection in high-resolution LC-MS data. *Anal. Chem.* 92, 588–592. doi: 10.1021/acs.analchem.9b04811

Oh, S. L., Ng, E. Y. K., Tan, R. S., and Acharya, U. R. (2019). Automated beat-wise arrhythmia diagnosis using modified U-net on extended electrocardiographic recordings with heterogeneous arrhythmia types. *Comput. Biol. Med.* 105, 92–101. doi: 10.1016/j.compbiomed.2018.12.012

Peng, P., Liu, C., Li, Z., Xue, Z., Mao, P., Hu, J., et al. (2022). Emerging ELISA derived technologies for in vitro diagnostics. *TrAC Trends Analyt. Chem.* 152:116605. doi: 10.1016/j.trac.2022.116605

Qin, Q., Wang, K., Xu, H., Cao, B., Zheng, W., Jin, Q., et al. (2020). Deep Learning on chromatographic data for segmentation and sensitive analysis. *J. Chromatogr. A* 1634:461680. doi: 10.1016/j.chroma.2020.461680

Ronneberger, O., Fischer, P., and Brox, T. (2015). *U-Net: convolutional networks for biomedical image segmentation.* New York, NY: Springer International Publishing.

Saeedizadeh, N., Minaee, S., Kafieh, R., Yazdani, S., and Sonka, M. (2021). COVID TV-Unet: segmenting COVID-19 chest CT images using connectivity imposed Unet. *Comput. Methods Programs Biomed. Update* 1:100007. doi: 10.1016/j.cmpbup.2021.100007

Shao, L., Zhang, L., Li, S., and Zhang, P. (2019). Design and quantitative analysis of cancer detection system based on fluorescence immune analysis. *J. Healthc. Eng.* 2019:1672940. doi: 10.1155/2019/1672940

Shao, X. Y., Wang, C. R., Xie, C. M., Wang, X. G., Liang, R. L., and Xu, W. W. (2017). Rapid and sensitive lateral flow immunoassay method for procalcitonin (PCT) based on time-resolved immunochromatography. *Sensors* 17:480. doi: 10.3390/s17030480

Singer, A. J., Ardise, J., Gulla, J., and Cangro, J. (2005). Point-of-care testing reduces length of stay in emergency department chest pain patients. *Ann. Emerg. Med.* 45, 587–591. doi: 10.1016/j.annemergmed.2004.11.020

Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15, 1929–1958.

Vila, J., Gómez, M. D., Salavert, M., and Bosch, J. (2017). Methods of rapid diagnosis in clinical microbiology: clinical needs. *Enferm. Infecc. Microbiol. Clin.* 35, 41–46. doi: 10.1016/j.eimce.2017.01.014

Wang, Z., Zou, Y., and Liu, P. X. (2021). Hybrid dilation and attention residual U-Net for medical image segmentation. *Comput. Biol. Med.* 134:104449. doi: 10.1016/j.compbiomed.2021.104449

Wu, M., Lai, Q., Ju, Q., Li, L., Yu, H. D., and Huang, W. (2018). Paper-based fluorogenic devices for in vitro diagnostics. *Biosens. Bioelectron.* 102, 256–266. doi: 10.1016/j.bios.2017.11.006

Xiao, Q., and Lin, J.-M. (2015). Advances and applications of chemiluminescence immunoassay in clinical diagnosis and foods safety. *Chin. J. Anal. Chem.* 43, 929–938. doi: 10.1016/s1872-2040(15)60831-3

Yang, J., Cheng, Y., Gong, X., Yi, S., Li, C.-W., Jiang, L., et al. (2021). An integrative review on the applications of 3D printing in the field of in vitro diagnostics. *Chin. Chem. Lett.* 33, 2231–2242. doi: 10.1016/j.cclet.2021.08.105

Zhang, C., Zhou, L., Du, K., Zhang, Y., Wang, J., Chen, L., et al. (2020). Foundation and clinical evaluation of a new method for detecting SARS-CoV-2 antigen by fluorescent microsphere immunochromatography. *Front. Cell Infect. Microbiol.* 10:553837. doi: 10.3389/fcimb.2020.553837

Zhang, Q., Zhang, M., Chen, T., Sun, Z., Ma, Y., and Yu, B. (2019). Recent advances in convolutional neural network acceleration. *Neurocomputing* 323, 37–51. doi: 10.1016/j.neucom.2018.09.038

Zheng, S., Lin, X., Zhang, W., He, B., Jia, S., Wang, P., et al. (2021). MDCC-Net: multiscale double-channel convolution U-Net framework for colorectal tumor segmentation. *Comput. Biol. Med.* 130:104183. doi: 10.1016/j.compbiomed.2020.104183

Zhu, Q., Du, B., and Yan, P. (2019). Boundary-weighted domain adaptive neural network for prostate MR image segmentation. *arXiv [preprint]* Available online at: https://doi.org/10.48550/arXiv.1902.08128 doi: 10.1109/TMI.2019.2935018 (accessed August 15, 2019).

Zunair, H., and Ben Hamza, A. (2021). Sharp U-Net: depthwise convolutional network for biomedical image segmentation. *Comput. Biol. Med.* 136: 104699 doi: 10.1016/j.compbiomed.2021.104699

TYPE Original Research
PUBLISHED 20 February 2023
DOI 10.3389/fncom.2023.1110889

EDITED BY
Wujie Zhou,
Zhejiang University of Science and Technology, China

REVIEWED BY
Arpit Sharma,
Manipal University Jaipur, India
Manoj Kumar,
University of Wollongong in Dubai, United Arab Emirates

*CORRESPONDENCE
Yuan Li
✉ sczyxxzx@163.com

RECEIVED 29 November 2022
ACCEPTED 16 January 2023
PUBLISHED 20 February 2023

CITATION
Zhang Y, Xie L, Li Y and Li Y (2023) A neural learning approach for simultaneous object detection and grasp detection in cluttered scenes. *Front. Comput. Neurosci.* 17:1110889.
doi: 10.3389/fncom.2023.1110889

A neural learning approach for simultaneous object detection and grasp detection in cluttered scenes

Yang Zhang[1], Lihua Xie[1], Yuheng Li[2] and Yuan Li[1]*

[1]China Tobacco Sichuan Industrial Co., Ltd, Chengdu, Sichuan, China, [2]Qinhuangdao Tobacco Machinery Co., Ltd, Qinhuangdao, Hebei, China

Object detection and grasp detection are essential for unmanned systems working in cluttered real-world environments. Detecting grasp configurations for each object in the scene would enable reasoning manipulations. However, finding the relationships between objects and grasp configurations is still a challenging problem. To achieve this, we propose a novel neural learning approach, namely SOGD, to predict a best grasp configuration for each detected objects from an RGB-D image. The cluttered background is first filtered out via a 3D-plane-based approach. Then two separate branches are designed to detect objects and grasp candidates, respectively. The relationship between object proposals and grasp candidates are learned by an additional alignment module. A series of experiments are conducted on two public datasets (Cornell Grasp Dataset and Jacquard Dataset) and the results demonstrate the superior performance of our SOGD against SOTA methods in predicting reasonable grasp configurations "from a cluttered scene."

KEYWORDS

grasp detection, object detection, RGB-D image, deep neural network, robotic manipulation

1. Introduction

Automated object grasping is essential and challenging to robots or unmanned systems working in real-world cluttered scenarios. As a core component of autonomous grasping, grasp detection, which outputs the most possible grasp configuration for the manipulator, has attracted great attention from both academic and industrial communities. Existing methods often predict a series of possible grasp configurations based on the input images (Depierre et al., 2018; Zhang et al., 2019; Wang et al., 2021; Yu et al., 2022b). When encountered with a cluttered scene, which is a common case in our daily life, we humans often identify the target object first and then determine the best pose to grab the object. This provides two kinds of benefits: (1) we can easily explain why the predicted grasp configuration is the best, and (2) our efforts will be focused on the object area instead of the cluttered background to make a better decision. However, most previous studies do not have a strong ability to model the relationship between the target objects and the predicted grasp configurations. In order to make grasp detection more accurate and reasonable, we investigate the problem of simultaneous object detection and grasp detection, where the best grasp configuration is predicted for each detected object in the cluttered scene.

Since object manipulation is performed in a 3D space, using a 3D representation for grasp detection is a natural way. A grasp candidate is a 6-DOF gripper pose $g = (x, y, z, r_x, r_y, r_z)$, $g \in SE(3)$, with the 3D position and rotation angles along each axis of the gripper. Methods based on this 3D representation (Pas et al., 2017; Liang et al., 2019) often generate a large number of candidates and then evaluate whether it is a good grasp according to a specific criterion. These methods are easy to understand but often suffer efficiency problems due to 3D operations. Motivated by the superior performance of deep learning technology on detection or segmentation tasks (Cheon et al., 2022; Huang et al., 2022; Khan et al., 2022), image-based deep models have become popular for grasp detection (Chu et al., 2018; Zhang et al., 2019; Dong et al., 2021; Yu et al., 2022a). These methods often use a rectangle representation $g = (x, y, h, w, \theta)$, where (x, y) is the center pixel location of a grasp candidate, (h, w) are height and width of the gripper, and θ is the rotation of the gripper. This representation is widely used in end-to-end deep networks. Some other studies (Wang et al., 2019, 2022) also used a score map with the same size as the image to represent the quality of grasp configurations at each pixel.

A number of existing grasp detection methods are inspired by object detection (Zhou et al., 2018; Zhang et al., 2019; Park et al., 2020). These two problems share a similar output, which consists of a regression of a rectangle (a grasp configuration for grasp detection or a bounding box for object detection) and a classification score (quality of the grasp or confidence in the predicted category). Thus, one straightforward way for designing a grasp detection model is to modify it from an object detector. For example, Zhang et al. (2019) propose an ROI-based grasp detection method which is a modification from Faster R-CNN. They use the region proposal network (RPN) to generate graspable proposals and an ROI-pooling layer to extract features for each proposal. Then grasp configurations and corresponding successful rates are estimated with the local features. However, grasp detection differs from object detection in the additional prediction of orientation. To predict the orientation of the gripper, serval existing methods (Chu et al., 2018; Dong et al., 2021; Yu et al., 2022a) convert this regression problem into a classification problem by discretizing continuous angles into angle anchors. This makes orientation prediction much more convenient but will also cause a loss of accuracy. To overcome this short back, other studies (Park et al., 2020) use classification and regression processes to predict the final angles. Another kind of grasp detection method (Yu et al., 2022b) makes dense predictions at each pixel and outputs a set of heatmaps representing the grasp configurations and quality.

To generate more reasoning and human-like grasp candidates, we investigate the problem of simultaneously detecting objects and grasp configurations from an RGB-D image and propose a novel neural learning approach, namely SOGD, for this task. Our SOGD model takes an RGB-D image as input and outputs a set of tuples $(x_o, y_o, w_o, h_o, cls_o, x_g, y_g, w_g, h_g, \theta_g, s_g)$, representing the joint prediction of the object detection result and the grasp detection result. To this end, features extracted by the top stages of a backbone and feature pyramid network (FPN) are used for both detection tasks. Two separate detection branches are designed to detect objects and grasp them, respectively. The correspondences between object proposals and grasp candidates are modeled by an alignment module. In addition, we present a depth-based method to filter out backgrounds in a cluttered scene. This would enable our detectors to focus on features from the target objects other than the texture from the environment.

Our main contributions are summarized as follows:

(1) We propose a novel neural learning approach to detect target objects and their best grasp configurations in cluttered environments simultaneously.
(2) An alignment module is designed to estimate the correlations between the separately detected objects and grasp configurations. This module enables our model to predict more reasonable grasp configurations for each detected object.
(3) A 3D-plane-based pre-processing is presented to filter out cluttered backgrounds from the RGB-D image.
(4) A series of experiments are conducted on two publicly available datasets (the Cornell Grasp Dataset and the Jacquard Dataset). Our method achieves +0.7 to +1.4% improvement in average accuracy compared with the existing RGB-D-based grasp detection methods.

2. Related studies

Grasp detection methods can be divided into traditional methods and learning-based methods. The traditional methods are mainly divided into the template matching method and the feature point matching method. The template-based pose estimation algorithm (Georgakis et al., 2019) needs to build the template of the object in advance, which can be strongly applicable to objects with regular shapes and has a good effect on targets without texture. However, when the object is blocked and the light is insufficient, the matching will be too low, leading to the failure of prediction. The pose estimation method based on feature points can extract effective feature points from images and match them with standard images. Since descriptors can describe visual features stably and robustly, this method is not susceptible to illumination. However, this method only uses the information of feature points in the image, so the utilization rate of information is very low. If there are not many feature points in the image, this method will have a high probability of deviation from the predicted capture rectangle.

Motivated by the superior performance of deep learning technology (Chhabra et al., 2022; Motwani et al., 2022; Shailendra et al., 2022; Singh et al., 2022), it has been applied in grasping detection to improve the accuracy of grasping in recent years. In order to improve the generalization of 3D models, some grab detection methods based on 3D reconstruction are proposed. According to Yang et al. (2021), this method uses 3D reconstruction to optimize the candidate grasping objects generated by the grasping suggestion network and improves the grasping accuracy of unknown objects. According to Jiang et al. (2021), this method uses implicit neural representation and studies synergies between affordance and geometry to improve the accuracy of grasping detection. Sundermeyer et al. (2021) used a 3D point cloud to predict the 3D points of grasping contact and reduce the dimension from 6-Dof to 4-Dof to make the learning process more convenient. However, the method based on 3D reconstruction needs a certain amount of time to build the 3D model, and the real-time capturing will be affected to some extent.

Since deep learning has shown excellent results in detection and segmentation tasks, image-based capture detection methods have become increasingly popular. But different from object detection, grasp detection needs to predict the angle of the gripper. Therefore, some of the methods (Chu et al., 2018; Dong et al., 2021; Yu et al., 2022a) discretized continuous angles into angle anchors and transformed the regression problem into a classification problem. However, these methods may cause a lack of accuracy. To solve this problem, Park et al. (2020) predicted the final angle using classification and regression processes. The method provided by Yu et al. (2022b) intensively predicts that each pixel represents the heat map of the capture configuration and quality. Zhang et al. (2018) divided the grasping problem into two separate tasks (object detection and grab detection) and then integrated them as the final solution. Yu et al. (2022b) proposed a module that extracts feature mappings from bidirectional feature pyramid networks, object detection, and grab detection, and outputs the optimal grab position and appropriate operational relations. Park et al. (2020) predicted the boundary box, the category of objects, and the direction of the grab rectangle and grab configuration using a global feature map. Ainetter and Fraundorfer (2021) designed an end-to-end CNN-based network architecture and designed a refinement module to improve the accuracy of prediction.

Most deep-learning-based methods directly output grasp candidates without recognizing the target object. They cannot answer the question that what is the best grasp configuration for every single object in a cluttered scene. Unfortunately, this is a common case an unmanned system needs to deal with. To generate a more reasoning prediction, Zhang et al. (2018) designed a recognize-and-then-grasp approach, which divides the problem into two separate tasks (object detection and grasp detection) and then integrates them as the final solution. Another way is to perform grasp detection together with object detection or segmentation tasks (Park et al., 2020; Ainetter and Fraundorfer, 2021; Yu et al., 2022a). For example, Park et al. (2020) generated a global feature map to predict the bounding box, the category of an object, the grasp rectangle, and the orientation of a grasp configuration. Then, non-maximum suppression is applied to both bounding boxes and grasp rectangles to filter out unnecessary predictions. The relationship between the bounding boxes and the grasp rectangles is built *via* computing the Intersection over Union (IoU) of the two areas. If the IoU is greater than a certain threshold, the grasp will be assigned to the detected object. However, choosing an appropriate threshold is not easy. When the graspable area is much smaller than the entire object, the IoU between the grasp rectangle and the bounding box of the object will be too small. As a result, such a strategy cannot avoid filtering out possible solutions.

3. Materials and methods

3.1. Problem formulation and reparameterization

In the process of object grabbing, humans usually first identify the object to be grabbed in the scene and then select an appropriate grabbing position for the target object. Whether a grasping pose is appropriate is directly related to the target object to be grasped. Motivated by this fact, this study investigates the problem of robotic manipulation by detecting the target object in the scene and its grasp position simultaneously.

Given an RGB-D image, the goal of the simultaneous object and grasp detection is to identify every single object in the scene and find out a grasp configuration for it. To this end, we formulate the representation of the simultaneous object and grasp detection det as:

$$\mathrm{det} = \left(\mathbf{od}, \mathbf{gd}\right)$$

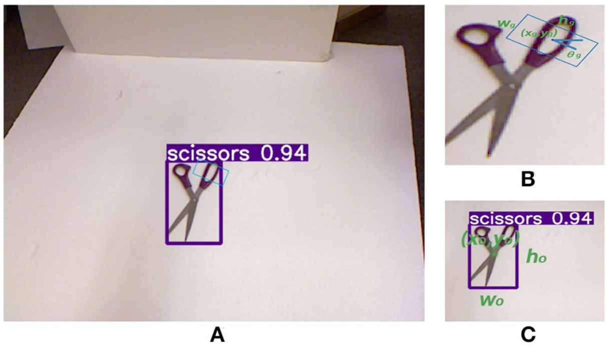


FIGURE 1
An example of our simultaneous object and grasp detection representation. **(A)** 11D object and grasp detection representation. **(B)** 5D object detection representation with location (x_o, y_o), width w_o, height h_o, and the category of the object cls_o. **(C)** 6D grasp detection representation with location (x_g, y_g), gripper width w_g, plate size h_g, orientation θ_g, and its success rate s_g.

$$\mathbf{od} = \left(x_o, y_o, w_o, h_o, cls_o\right)$$

$$\mathbf{gd} = \left(x_g, y_g, w_g, h_g, \theta_g, s_g\right)$$

The representation consists of two parts: object detection and grasp detection. Figure 1 shows an example of this representation. For the object detection part, we use (x_o, y_o, w_o, h_o) to represent the location of a bounding box and cls_o to represent the category of the object inside of it. For the grasp detection part, we adopt the famous 5-dimensional rectangular representation (Lenz et al., 2013), which consists of the location and orientation $(x_g, y_g, w_g, h_g, \theta_g)$ of the gripper for a grasp configuration. In addition, we add s_g, a value between 0 and 1, to represent the score of a grasp. The higher s_g is, the greater chance of the grasp being a success.

Similar to Park et al. (2020), we formulate the estimation of θ_g as a classification + regression problem instead of a single regression problem. According to the symmetry of the gripper, the range of θ_g is $[0, \pi]$. We convert this range into several bins $\left\{0, \frac{\pi}{k_a}, \frac{2\pi}{k_a}, ..., \left(k_a - 1\right)\frac{\pi}{k_a}\right\}$ to be angle anchors, with k_a is the number of bins. The classification problem is to predict a one-vs.-all vector to determine which bins θ_g belongs to. The regression problem is to estimate the angle offset to the anchors.

Inspired by Redmon and Farhadi (2018) and Ge et al. (2021), we reparametrize the regression problem of $(x_j, y_j, w_j, h_j, \theta_j)$ as estimation of $(t_j^x, t_j^y, t_j^w, t_j^h, t_j^\theta)$ to the location of the grid cell (a_j^x, a_j^y), bounding box prior width and height (a_j^w, a_j^h), and orientation angle bin a_j^θ. The relationship between $(x_j, y_j, w_j, h_j, \theta_j)$ and $(t_j^x, t_j^y, t_j^w, t_j^h, t_j^\theta)$ is defined as follows.

$$x_j = \sigma(t_j^x) + a_j^x$$

$$y_j = \sigma(t_j^y) + a_j^y$$

$$w_j = a_j^w \times e^{t_j^w}$$

$$h_j = a_j^h \times e^{t_j^h}$$

$$\theta_j = \sigma(t_j^\theta)\times\left(\frac{\pi}{k_a}\right) + a_j^\theta$$

This reparameterization is applied to both the object bounding box regression and the grasp rectangle regression.

3.2. Overview of the SOGD model

The architecture of our SOGD model is shown in Figure 2. It takes an RGB-D image as input, and outputs the detected object's bounding $(t_o^x, t_o^y, t_o^w, t_o^h)$ and category *cls* together with the corresponding grasp position $(t_g^x, t_g^y, t_g^w, t_g^h)$, orientation t_g^θ, and success rate s_g. Our model consists of five parts: a pre-processing for background removal, a backbone and a feature pyramid network (FPN) for image feature extraction, an object detection head, a grasp detection head, and an alignment module for candidate objects and grasp configurations.

Motivated by Dong et al. (2021), we design a pre-processing module to remove the background from the cluttered scene. According to Dong et al. (2021), backgrounds are recognized by an encoder–decoder network to segment the original image. However, our module utilizes the priors of the scene that objects to be grabbed are laid on a 3D plane, such as the surface of a desk. According to this fact, we categorize pixels on and under the 3D plane as background and filter them out. We argue that this background removal strategy is more reasonable and robust than the U-net-based method (Dong et al., 2021). Details about the 3D-plane-based background removal are discussed later.

For multi-scale feature extraction, various deep models [e.g., Darknet (Wood, 2009) or ResNet (He et al., 2016)] can be utilized. In this study, ResNet-50 is used as the backbone to release the computational burden of deep models during feature extraction and facilitate real-time performance. The very last feature map learned by different stages (e.g., *conv1*, *conv2*, and *conv3*) is used as multi-scale features. We denote these feature maps as $\{C_1, C_2, C_3, C_4, C_5\}$. The stride steps of these feature maps are $\{2, 4, 8, 16, 32\}$ with respect to the original image. Only $\{C_3, C_4, C_5\}$ are used in FPN for feature fusion and the fused feature maps are denoted as $\{P_3, P_4, P_5\}$. Our feature extraction and fusion module can be formulated as follows:

$$image\_f = BackgroundRemoval\left(image\_rgbd\right)$$

$$F_{backbone} = \{C_3, C_4, C_5\} = ResNet\left(image\_f\right)$$

$$F_{FPN} = \{P_3, P_4, P_5\} = FPN\left(F_{backbone}\right)$$

Inspired by the fact that grasp rectangles are often much small than the object's boundaries, we use different prior rectangle sizes for object detection and grasp detection. The object detection head and the grasp detection head share a similar structure with the detection head (Redmon and Farhadi, 2018). The output tensor for object detection is in the shape of $N \times N \times k_o \times (4 + 1 + C_o)$, where $N \times N$ is the size of the feature map, k_o is the number of predicted bounding boxes, 4 stands for the number of parameters of a bounding box, 1 stands for the channel of confidence, and C_o is the number of object categories. Similarly, the output tensor for grasp detection is in the shape of $N \times N \times k_g \times (5 + C_a + 1)$, where 5 stands for the location, width, height, and orientation of a grasp rectangle, C_a is the number of angle bins, and 1 stand for the successful rate prediction.

3.3. Background removal

Robotic manipulation often encounters a cluttered environment. The captured RGB-D images include both the target objects to be grasped and the background surroundings. To achieve an accurate object and grasp detection, we should focus on the pixels belonging to the targets. The additional observation on backgrounds may distract our attention from the targets. Figure 3 presents a quantitative analysis of this additional information. When the background is removed, the detection model only needs to learn features from the target objects and all learned features are valuable for final manipulation. However, if the RGB-D image encounters a cluttered background, the model will have to learn features from both the targets and the background, and distinguish which feature contributes to the downstream tasks. This will increase the burden of the model for feature extraction and also increase the number of parameters to learn task-specific features. According to Dong et al. (2021), these additional cluttered background pixels will even lead to a false grasp detection.

To eliminate the cluttered background and let the model focus on the targets, Dong et al. (2021) adopt an encoder–decoder-based U-net model to segment the input image into foreground and background. It is indeed a potential way to filter out the background in an image. But the U-net model needs to be trained on a large dataset and its generalizability to new observations is limited. Instead of recognizing the background in the image domain, we present a background removal method in 3D space. We assume that objects to be grasped are laid on a 3D plane (such as the surface of a desk), which is the common case in robotic manipulation. Under this assumption, pixels that are up to the 3D plane are defined as the foreground, while pixels in or under the 3D plane are defined as the background. This could separate the targets from the cluttered background in most cases. In the top-left image in Figure 3, the white vertical surface will also be considered as the foreground using the aforementioned approach. But this mis-segmentation will not affect the detection of the targets since the vertical surface is disconnected from the targets.

For the 3D plane estimation, we use a model-based method to fit the unknown parameters. In 3D spaces, a plane is defined as $aX + bY + cZ + d = 0$, with $\left(a, b, c, d\right)$ as the plane parameters. Given three points, we can fit a plane for it. Since pixels belonging to the 3D plane are dominant in the image, we adopt a RANSAC-based method to fit the parameters of the largest 3D plane in the image. The method achieves its goal by iteratively selecting a random subset of the original 3D points. The selected subset is assumed to be inliers and the plane parameters are fitted with respect to these inlier points. Then all other points are tested against the fitted model. If a point fits well to the estimated model, it will be considered as inliers to the model. The fitted 3D plane is reasonably good if sufficiently many points are classified as inliers. In this study, 3D coordinates are computed from the depth image with a fixed camera intrinsic parameter when 3D points are not presented in the dataset.

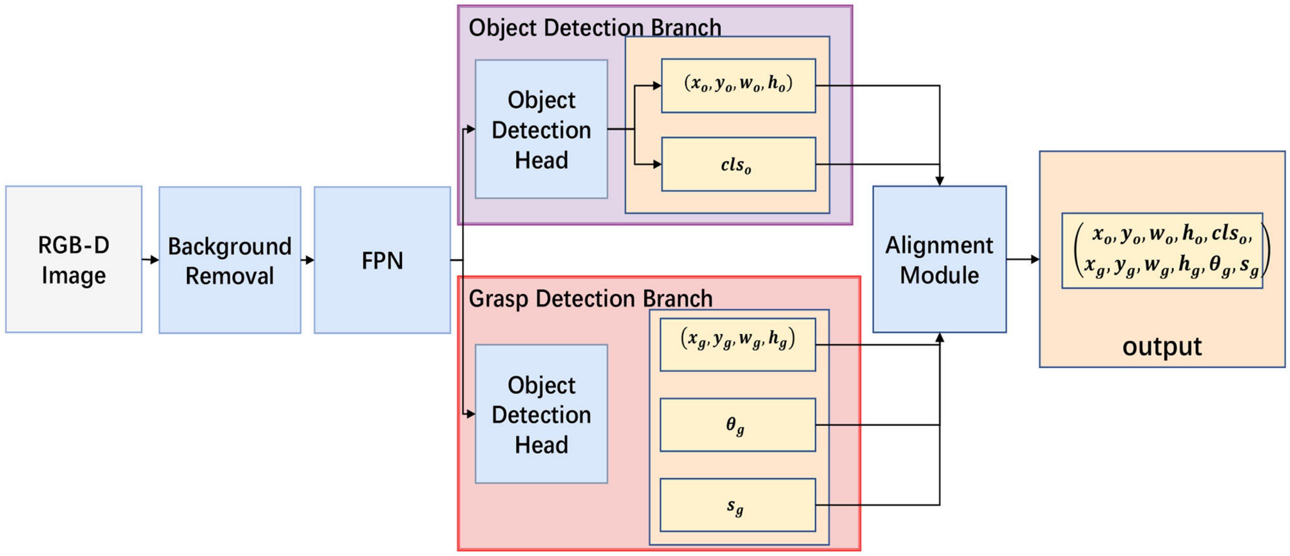


FIGURE 2
Outline of our Simultaneous Object and Grasp Detection (SOGD) model. SOGD takes an RGB-D image as input and outputs a series of recognized objects together with the most appropriate grasp for every single object. It consists of five parts: a pre-processing module to remove background from the cluttered scene, a backbone (e.g., Darknet or ResNet) and FPN for hierarchical feature extraction, two separate branches for object detection and grasp detection, and an alignment module to assign a most appropriate grasp for each detected object.

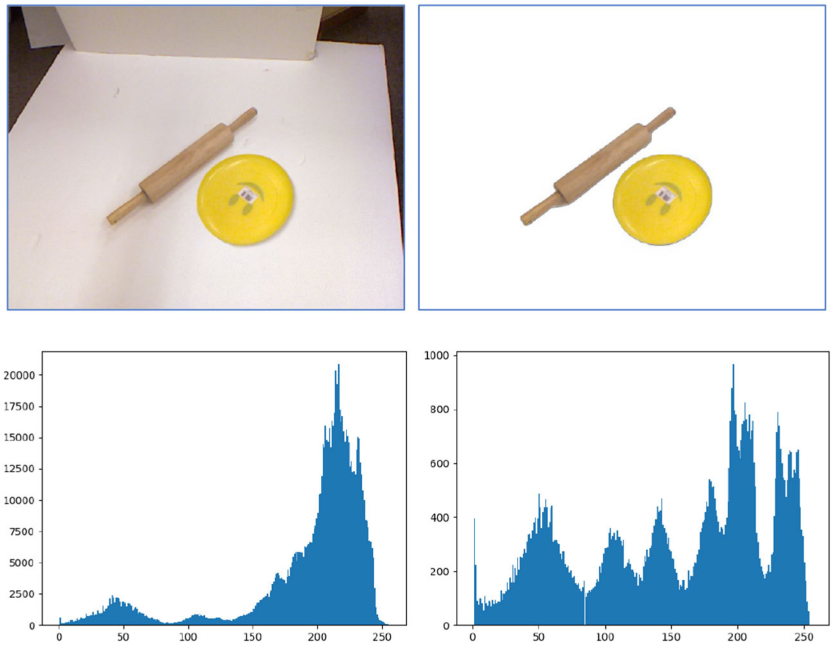


FIGURE 3
Illustration of the unnecessary efforts spent on the cluttered background during feature extraction. **Top** lines are original images and foreground images. **Bottom** lines are the corresponding histograms of the top images. From the histograms, it can be seen that the additional information which is useless to target detection will significantly increase when a cluttered background is encountered in the image.

We also applied voxelization to speed up the process and generate a finer plane.

3.4. Separate object and grasp detection branches

Object detection and grasp detection are both detection problems. These two tasks share a similar output in the regression of location, width and height of a rectangle (as the bounding box for object detection and the grasp rectangle for grasp detection), and a confidence score (as the classification for object detection and the grasp quality for grasp detection). According to observation, we design two separate detection branches for these two tasks, but the branches share a similar architecture as shown in Figure 4.

The structure of our two detection branches is motivated by modern object detectors (Redmon and Farhadi, 2018). The detection head takes multi-stage outputs from FPN as inputs to detect objects

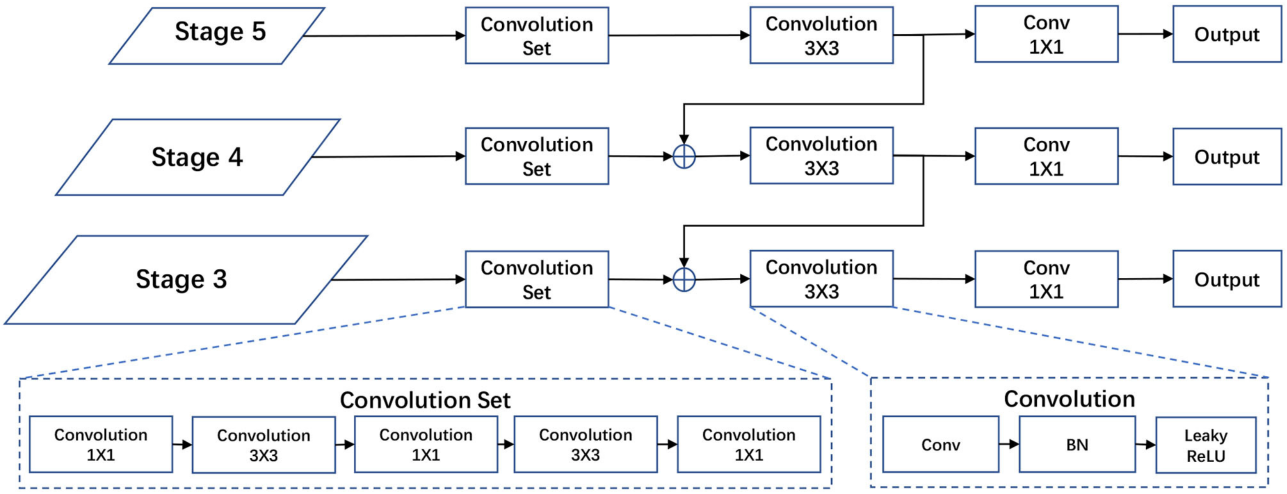


FIGURE 4
Structure of the object/grasp detection head. Our object detection head and grasp detection head share a similar structure except for the output channels. Both heads take $\{P_3, P_4, P_5\}$ from the FPN as inputs. The output of object detection includes the bounding box and object categories, while the output of grasp detection has more channels for orientation and grasp score.

or grasp configurations at different scales. In fact, a gripper only needs to grab a small part of an object to take it up, instead of grabbing the whole of it. As a result, the detected rectangle for grasp is often smaller than the bounding box of the object. Thus, we use the same inputs as modern detectors for object detection and grasp detection, but a relatively small-scale prior size for grasp detection. This enables our model to use a same-level semantic feature for both tasks.

Our detection head consists of a convolution set, a 3 × 3 convolution block, and a 1 × 1 convolution layer for prediction. The multi-stage outputs of FPN are treated as fused features which include both texture and semantic information extracted from the input image. The convolution set is designed to learn a task-correlated feature representation from the texture and semantic information. Then the 3 × 3 convolution block fuses task-corelated features at the top and current scales. The 1 × 1 convolution layer is used to match the number of channels to the final predictions. The number of channels of outputs for object detection is $k_o \times (4 + 1 + C_o)$, where k_o is the number of predicted bounding boxes for each grid cell; 4 stands for $(t_o^x, t_o^y, t_o^w, t_o^h)$, parameters of a bounding box; 1 stands for the confidence of the prediction; and C_o is the number of object categories. Similarly, the number of channels of output for grasp detection is $k_g \times (5 + C_a + 1)$, where k_g is the number of predicted grasp rectangles; 5 stands for $(t_g^x, t_g^y, t_g^w, t_g^h, t_g^\theta\ )$, parameters of a grasp rectangle; 1 stands for the score of the grasp configuration; and C_a is the number of angle bins. The mathematical computation of our detection head is as follows:

$$\mathbf{F_{task}} = \{\mathbf{T_3,T_4,T_5}\} = \mathbf{ConvolutionSet}\,(F_{FPN})$$

$$\mathbf{F_{task\_fusion}} = \{\mathbf{TF_3,TF_4,TF_5}\}$$

$$\mathbf{TF_i} = \mathbf{Convolution}\,(\mathbf{TF_{i+1}} + \mathbf{T_i})$$

$$\mathbf{Pro} = \mathbf{Conv_{1\times1}}\left(\mathbf{F_{task\_fusion}}\right)$$

3.5. Alignment between objects and grasp configurations

The two detection branches make predictions for objects and grasp configurations separately. To model the correspondence between detected objects and grasp configurations, we design an alignment module. Given an object prediction $\mathbf{Pro_o} \in \mathbf{R}^{N\times N\times k_o}$ and a grasp prediction $\mathbf{Pro_g} \in \mathbf{R}^{N\times N\times k_g}$, the correspondences between all possible pairs are defined as $\mathbf{Pro_{corr}} \in \mathbf{R}^{(N\times N\times k_o)\times(N\times N\times k_g)}$. Objects and grasp configurations are detected at different scales. Generating correspondences across multi scales would significantly increase the computational complexity. Thus, we only consider possible object-grasp pairs within the same scale.

Our alignment module takes the task-correlated features from object detection head $\mathbf{TF_o} \in \mathbf{R}^{N\times N\times c_o}$ and grasp detection head $\mathbf{TF}_g \in \mathbf{R}^{N\times N\times c_g}$ as input. Then two 1 × 1 convolution layers are applied to the features separately, resulting in the outputs of $\acute{\mathbf{F}}_\mathbf{o} \in \mathbf{R}^{N\times N\times k_o\times f}$ and $\acute{\mathbf{F}}_g \in \mathbf{R}^{N\times N\times k_g\times f}$. The two feature maps are reshaped into a 2D matrix and transpose matrix multiplication is applied to generate the output $\mathbf{F_{corr}} \in \mathbf{R}^{(N\times N\times k_o)\times(N\times N\times k_g)}$. Finally, we use a sigmoid activation function to model the joint possibility of the detection pairs. The mathematical computation of our alignment module can be formulated as follows:

$$\acute{\mathbf{F}}_\mathbf{o} = \mathbf{Conv_{1\times1}}\,(\mathbf{TF_o})$$

$$\acute{\mathbf{F}}_g = \mathbf{Conv_{1\times1}}\left(\mathbf{TF_g}\right)$$

$$\mathbf{F}_{corr} = \mathbf{reshape}(\acute{\mathbf{F}}_\mathbf{o}) \bullet (\mathbf{reshape}(\acute{\mathbf{F}}_g))^T$$

$$\mathbf{Pro}_{corr} = sigmoid(\mathbf{F}_{corr})$$

Though our two detection heads make predictions separately, our model is forced to learn a better $\mathbf{Pro}_{corr}$ to model the correlation between the predictions. At inference, we use two additional parameters k_o and k_c to control the number of final predictions. First, top k_o object predictions are selected from $\mathbf{Pro}_o$, then top k_c correlated grasp predictions are selected and assigned to the detected objects. If the quality of a grasp prediction is smaller than a threshold, the grasp prediction will be filtered out from the alignment results. In this way, our model can be easily extended to multi-object detection and multi-grasp detection cases.

3.6. Loss function

The loss of our SOGD model consists of three parts: object detection loss L_o, grasp detection loss L_g, and alignment loss L_{corr}. The loss of object detection is defined as:

$$L_o = L_o^{reg} + \alpha \times L_o^{cls}$$

$$L_o^{reg} = \frac{1}{N_o^{reg}} \sum_i smooth_{L1}\left(t_o^i, \hat{t}_o^i\right)$$

$$L_o^{cls} = \frac{1}{N_o^{cls}} \sum_i Loss_{focal}\left(cls_i, \hat{cls}_i\right)$$

where t_o^i and $\hat{t}_o^i$ are ground truth and predictions for a bounding box, respectively. We use the smooth L1 loss for regression and focal loss (Lin et al., 2017) for classification. N_o^{reg} and N_o^{cls} are the normalizers. α is the weight of classification loss.

Similarly, the loss of grasp detection is defined as:

$$L_g = L_g^{reg} + \beta \times L_g^{angle} + \gamma \times L_g^{score}$$

$$L_g^{reg} = \frac{1}{N_g^{reg}} \sum_i smooth_{L1}\left(t_g^i, \hat{t}_g^i\right)$$

$$L_g^{angle} = -\sum_i \left[a_i \log\left(\hat{a}_i\right) + (1-a_i) \log\left(1-\hat{a}_i\right)\right]$$

$$L_g^{score} = -\sum_i \left[s_i \log\left(\hat{s}_i\right) + (1-s_i) \log\left(1-\hat{s}_i\right)\right]$$

where t_g^i and $\hat{t}_g^i$ are ground truth and predictions for a grasp rectangle, respectively. β and γ are hyperparameters.

The loss of object and grasp alignment is defined as:

$$L_{corr} = -\delta \times \sum_i \left[p_i \log\left(\hat{p}_i\right) + \left(1-p_i\right) \log\left(1-\hat{p}_i\right)\right]$$

where p_i and $\hat{p}_i$ are ground truth and predictions of a candidate pair for object and grasp, respectively. δ controls the weights of alignment loss to the total loss.

The total loss function is the summation of the three losses:

$$L = L_o + L_g + L_{corr}$$

4. Results

To evaluate the performance of our proposed SOGD model against previous methods, we test it on two publicly available datasets: the Cornell Grasp Dataset (Lenz et al., 2013) and the Jacquard Dataset (Depierre et al., 2018). Our model is designed for the task of grasp detection, but it also outputs predictions for object detection. Thus, the metrics used in our experiments consist of two parts. For the grasp detection task, we use the popular Jaccard Index and angle difference as metrics, consistent with previous methods (Jiang et al., 2011; Chu et al., 2018; Kumra et al., 2020; Yu et al., 2022b). A predicted grasp configuration is considered as correct if and only if the following two conditions are satisfied.

(1) Jaccard Index of the predicted grasp rectangle and the ground truth is >0.25. Assuming that $\hat{b}_g$ is the predicted grasp rectangle and b_g is the ground truth, Jaccard Index is defined as:

$$Jaccard\ Index = \frac{\left|\hat{b}_g \cap b_g\right|}{\left|\hat{b}_g \cup b_g\right|}$$

(2) the difference between the predicted orientation angle and the ground truth is $<30°$.

For object detection tasks, we use accuracy instead of the commonly used mAP as metrics. In this research, object detection is an additional output only to achieve the final goal of predicting the most possible grasp for each individual object in a cluttered scene. Our method only needs to know where the object is, and what kind of object it is does not matter too much. So, we consider a prediction as correct if the intersection over union of the predicted bounding box and ground truth is >0.5.

4.1. Grasp detection results on cornell grasp dataset

There are 878 images together with the corresponding depth image and 3D point clouds in the Cornell Grasp Dataset (Lenz et al., 2013). The resolution of the images is 640 × 480. Each image contains a single graspable object at different positions and orientations. The dataset is manually annotated with many positive and negative grasp rectangles. Following previous research (Zhang et al., 2019; Dong et al., 2021), we use a five-fold cross-validation strategy to evaluate the performance of our method and report the average detection accuracy in this section. The reported results include both image-wise (IW) and object-wise (OW) detection accuracy. In image-wise experiments, all images are randomly divided into a train set and a test set. The object in the test set may have been learned during training but at different poses and views. This is mainly to test the generalization ability of our method when objects are captured from multiple points of view. In object-wise experiments, images are divided according to the object categories. Objects in the test set have never been seen during training. This is to test the generalization ability of our method when it faces a new kind of object.

The evaluation of grasp detection accuracy and efficiency are summarized in Table 1. Our method achieves 98.9 and 98.3% accuracy on image-wise and object-wise detection tasks, respectively. Compared with the state-of-the-art RGB-D-based method (Yu et al., 2022b), our SOGD shows an improvement of +0.7 and +1.2% in accuracy. The efficiency of our method is relatively low than SE-ResUNet (Yu et al., 2022b). In SE-ResUNet, a squeeze-and-excitation residual network is designed to predict the width and orientation of the grasp rectangle and the quality of the grasp outputs. It does not involve the detection of the target object. As a result, the complexity of SE-ResUNet is smaller than ours and their detection ability is not strong as ours. Similar results are observed when our SOGD is compared with Kumra et al. (2020). Compared with the state-of-the-art RGB-based method (Yu et al., 2022a) and RG-D-based method (Park et al., 2020), our SOGD shows a similar performance in the image-wise detection task, but a superior performance in the object-wise detection task. This is mainly because our SOGD not only learns possible grasp configurations but also the correspondences between the target object and grasp candidates. In this way, it has the ability to figure out what is the best grasp configuration for a specific kind of object. Therefore, when facing new objects, it can benefit from learned knowledge of the relationship between grasp configurations and objects.

TABLE 1 Grasp detection results on the Cornell Grasp Dataset.

Methods	IW/%	OW/%	FPS	Input
Chu et al. (2018)	94.4	95.5	8.3	RGB
Wang et al. (2021)	96.1	95.5	-	RGB
Asif et al. (2019)	96.7	-	-	RGB
Yu et al. (2022a)	**98.9**	**97.8**	**50.0**	RGB
Dong et al. (2021)	96.4	96.5	9.4	RG-D
Song et al. (2020)	95.6	97.1		RG-D
Zhang et al. (2019)	92.3	91.7	25.2	RG-D
Park et al. (2020)	**98.6**	**97.8**	**62.5**	RG-D
Jiang et al. (2011)	60.5	58.3	0.2	RGB-D
Lenz et al. (2013)	73.9	75.6	0.7	RGB-D
Chu et al. (2018)	96.0	96.1	8.3	RGB-D
Kumra et al. (2020)	97.7	96.6	**50.0**	RGB-D
Yu et al. (2022b)	98.2	97.1	40.0	RGB-D
SOGD (ours)	**98.9**	**98.3**	9.6	RGB-D

The best performance of methods within a same kind of input is shown as bold.

Visualization of typical grasp detection results is shown in Figure 5. The three lines in the figure are ground truth annotations, predicted grasp configurations of our SOGD, and the corresponding grasp quality predicted by our SOGD. For each detected object, our model outputs the best grasp rectangle for it. In this experiment, the quality is not the direct output from the grasp detection branch in our SOGD model. Our grasp detection branch outputs a score map for each grasp candidate that can be treated as the quality of the grasp, as mentioned in a previous study (Yu et al., 2022a,b). But our model includes an alignment module to learn the correspondences between predicted objects and grasp configurations. The quality is the product of the score map and the correspondences. It represents the success rate of a grasp if there is a detected target object. From the figure, it can be seen that our SOGD has a good ability in detecting grasp rectangles and the output quality map is able to provide a clear reason for the grasp decision.

4.2. Grasp detection results on Jacquard Dataset

The Jacquard Dataset (Depierre et al., 2018) is collected from a simulator with CAD models of the ShapeNet dataset. There are 54k images with 11k different kinds of objects in the dataset. A large number of samples facilitate our model training. However, we still use some data augmentation strategies (like random rotation) to increase

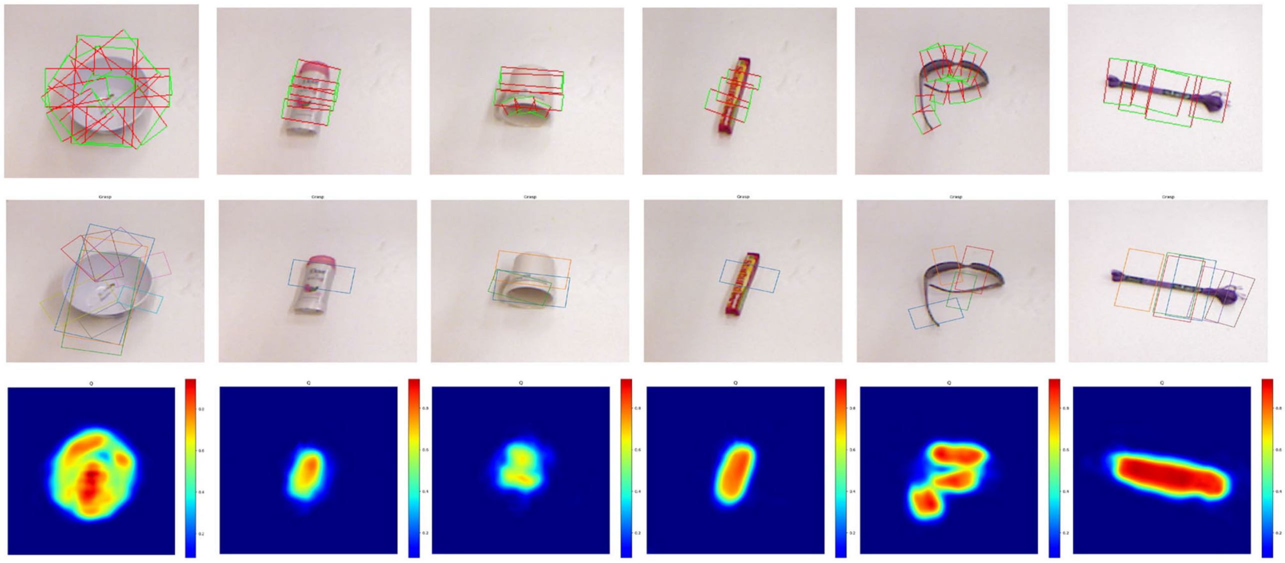


FIGURE 5
Typical grasp detection results of our SOGD on the Cornell Grasp Dataset. **Top** line is ground truth annotation; **middle** line is the prediction of our SOGD; and **bottom** line is the estimated grasp quality of the corresponding detection.

the robustness of the learned model. The resolution of images in this dataset is 1,024 × 1,024. We down-sample the original image to a size of 512 × 512 for both training and testing. Unlike results on the Cornell Dataset, we report the overall accuracy of the grasp detection results on the Jacquard Dataset.

Table 2 reports the performance of our SOGD against the state-of-the-art methods. Our SOGD shows a 99% accuracy on the Jacquard Dataset, which is higher than both RGB-based and RG-D-based state-of-the-art methods. Compared with MASK-GD (Dong et al., 2021), our method achieves a +1.4% performance boost. MASK-GD also involves pre-processing for background removal. Background removal is treated as an image segmentation problem and a deep network is trained for it in MASK-GD. This strategy has the potential to recognize the foreground targets from the cluttered scenes, and may also suffer the problem of limited generalization ability. In addition, our model has an additional alignment module to learn the relationship between object candidates and grasp candidates, while MASK-GD cannot. Compared with other grasp detection methods, the improvement of our SOGD is more significant.

TABLE 2 Grasp detection results on the Jacquard Dataset.

Methods	Accuracy/%	Input
Zhou et al. (2018)	91.8	RGB
Zhang et al. (2019)	90.4	RGB
Dong et al. (2021)	**97.1**	RGB
Zhou et al. (2018)	92.8	RG-D
Zhang et al. (2019)	93.6	RG-D
Dong et al. (2021)	**97.6**	RG-D
SOGD (ours)	**99.0**	RGB-D

The best performance of methods within a same kind of input is shown as bold.

Figure 6 shows some typical grasp detection results of our SOGD on this dataset. Both the detected grasp configurations and the quality maps are visualized to give a better understanding of the results. From the figure, it can be seen that our SOGD can well detect grasp candidates and outputs a reasonable quality map on this dataset.

4.3. Object detection results

For object detection evaluation, our SOGD model is trained and tested on the two datasets mentioned earlier. We use the Labelme tools from MIT to manually annotate bounding boxes and class labels on the Cornell Dataset. To prevent overfitting, the pre-trained

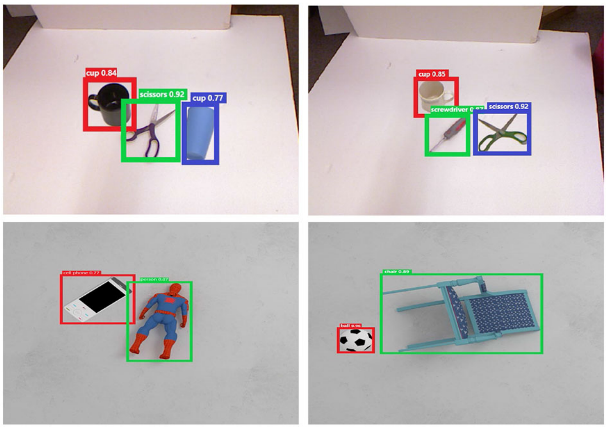


FIGURE 7
Typical object detection results.

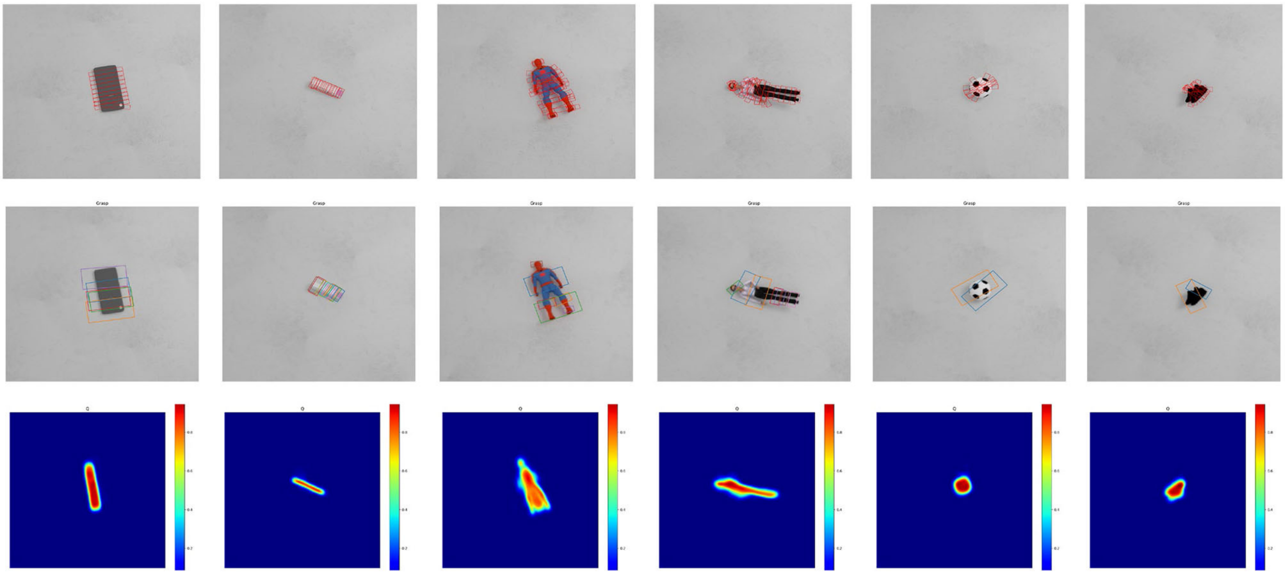


FIGURE 6
Grasp detection results of our SOGD on the Jacquard Dataset. **Top** line is ground truth annotation; **middle** line is the prediction of our SOGD; and **bottom** line is the estimated grasp quality of the corresponding detection.

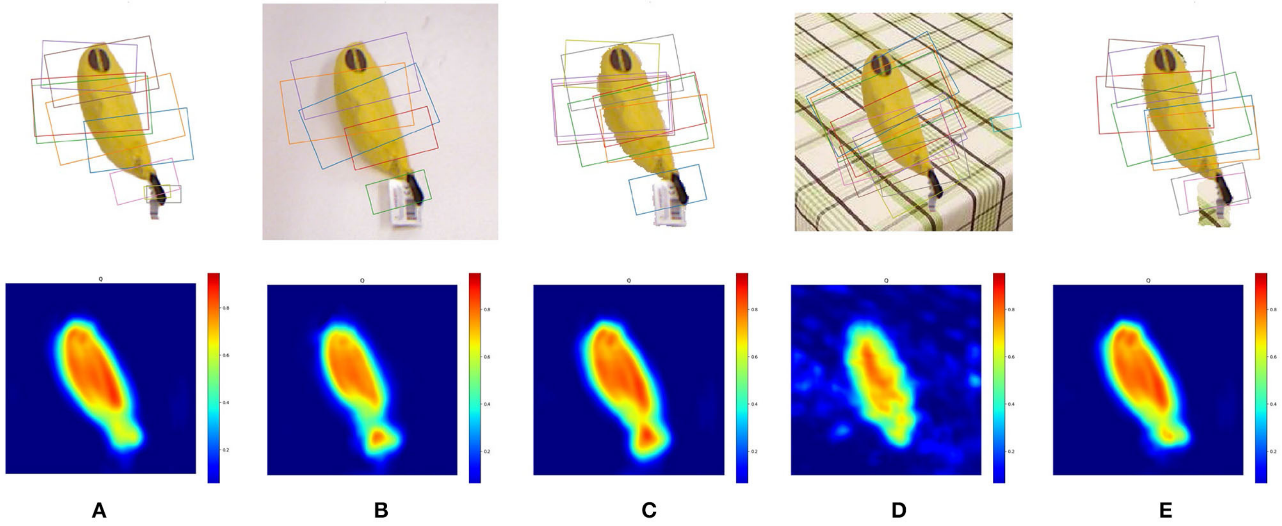


FIGURE 8
Grasp detection results on images with various kinds of backgrounds. **(A)** SOGD without background removal (BR) on images without background. **(B)** SOGD without BR on original images from Jacquard Dataset. **(C)** SOGD with BR on the original images. **(D)** SOGD without BR on images with the background being replaced. **(E)** SOGD with BR on images with the background being replaced. From left to right the background of the image becomes more cluttered.

parameters of ResNet-50 are fixed for the backbone and several data augmentation strategies are involved, such as rotation, translation, flip, random crop, and illumination change. Typical experimental results are shown in Figure 7. From the figure, it can be seen that after removing the background boundaries foreground objects are much easier to detect, releasing the burden of the object detection branch and resulting in more accurate bounding box predictions. In our experiments, though we pay more attention to the prediction accuracy of where the object is, the classification confidence of our SOGD is almost above 0.75 on the two datasets. The above-mentioned observations show that our SOGD model has a good performance in detecting objects from a cluttered scene, especially in identifying the boundaries of the objects.

4.4. Discussion on background removal

From the results in Tables 1, 2, it has already been seen that our SOGD has superior performance than existing methods without background removal (Zhang et al., 2019; Kumra et al., 2020; Yu et al., 2022b). But to investigate how much the background removal contributes to this performance boost, we conduct an ablation study on this pre-processing. The Jacquard Dataset is used in this experiment since it provides a ground truth mask for foreground target objects. With this mask, we generate two additional types of images from the original dataset to test the performance of our SOGD on it. The first one is to filter out backgrounds with the ground truth mask. The second one is to fill the background with a cluttered background. To achieve this, we download a number of images from the Internet as the background image datasets and randomly choose one to replace the background images from the Jacquard Dataset.

We provide a comparison among five types of grasp detection configurations: (1) SOGD without background removal on images with background filtered out, (2) SOGD without background removal on the original image, (3) SOGD with background removal on the original image, (4) SOGD without background removal on images with background being replaced, and (5) SOGD with background removal on images with background being replaced. Typical results are shown in Figure 8. The background of the scene in Figure 8 becomes more cluttered from left to right. From the figure, we observe that the predicted grasp configurations will be much better when the background is removed from the image.

5. Conclusion and future study

This study is focused on the problem of grasp detection from an RGB-D image. Unlike previous methods, we solve this problem by simultaneously detecting the target object and the corresponding grasp configurations. This is motivated by the fact that when grabbing an object, we humans first identify where the object is and then make a decision on which part of the object to grab. To this end, a novel neural network SOGD together with its learning method is proposed. In SOGD, object and grasp configurations are first detected by two separate branches, and then the relationship between object candidates and grasp configurations is learned by an alignment module. The best grasp configuration is predicted according to the grasp score and its correspondence to the target object. Our method is tested on two publicly available datasets. A series of experiments are conducted and both qualitative and quantitative experimental results are presented. The results demonstrate the validity and practicability of our method.

To deal with grabbing in a cluttered scene, a pre-processing for background removal is designed. Unlike previous methods where background removal is treated as an image segmentation problem, we propose to leverage the prior knowledge that objects to be grabbed are often placed on a 3D plane. Therefore, we adopt a RANSAC-based plane fitting method to detect the largest 3D plane in the scene. All pixels laid in or under the plane are considered background. The experimental results show that our strategy makes grasp detection more robust in cluttered environments.

The stacked scene is not considered in this research. In daily life cases, it is common that objects to be grabbed are laid on top of each other. This is more challenging for the grasp detection method because it has to figure out the execution order of the predicted grasp configurations. This is an interesting topic for our future study. In addition, the kind of object for model training is limited. It has to face a large number of unknown objects when the learned model is deployed to real devices. It is interesting to extend our model with the life-long learning ability after deployment. We will explain it in our future study.

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

YuaL contributed the main ideas and designed the algorithm. YuhL organized the database. LX performed the statistical analysis. YZ wrote the first draft of the manuscript. YZ, LX, YuhL, and YuaL wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

Conflict of interest

YZ, LH, and YuaL were employed by China Tobacco Sichuan Industrial Co., Ltd. YuhL was employed by Qinhuangdao Tobacco Machinery Co., Ltd.

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

Ainetter, S., and Fraundorfer, F. (2021). "End-to-end trainable deep neural network for robotic grasp detection and semantic segmentation from RGB," in *Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA)* (Sanya, China: IEEE), 13452–13458.

Asif, U., Tang, J., and Harrer, S. (2019). "Densely Supervised Grasp Detector (DSGD)," in *Proceedings of the AAAI Conference on Artificial Intelligence* (Hawaii, USA: AAAI), 33, 8085–8093.

Cheon, J., Baek, S., and Paik, S. B. (2022). Invariance of object detection in untrained deep neural networks. *Front. Comput. Neurosci.* 16, 1030707. doi: 10.3389/fncom.2022.1030707

Chhabra, M., Ravulakollu, K. K., Kumar, M., Sharma, A., and Nayyar, A. (2022). Improving automated latent fingerprint detection and segmentation using deep convolutional neural network. *Neural Comput. Appl.* 2022, 1–27. doi: 10.1007/s00521-022-07894-y

Chu, F. J., Xu, R., and Vela, P. A. (2018). Real-world multiobject, multigrasp detection. *IEEE Robot. Autom. Lett.* 3, 3355–3362. doi: 10.1109/LRA.2018.2852777

Depierre, A., Dellandréa, E., and Chen, L. (2018). "Jacquard: A large scale dataset for robotic grasp detection," In Proceedings of the *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid, ES: IEEE), 3511–3516.

Dong, M., Wei, S., Yu, X., and Yin, J. (2021). Mask-GD Segmentation Based Robotic Grasp Detection. *Comp. Commun.* 178, 124–130. doi: 10.1016/j.comcom.2021.07.012

Ge, Z., Liu, S., and Wang, F. (2021). Yolox: exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*

Georgakis, G., Karanam, S., Wu, Z., and Kosecka, J. (2019). "Learning Local RGB-to-CAD Correspondences for Object Pose Estimation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (Seoul, Korea (South): IEEE), 8966–8975.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, USA: IEEE), 1063–6919. doi: 10.1109/CVPR.2016.90

Huang, Z. J., Hui, B. W., and Sun, S. J. (2022). An infrared sequence image generating method for target detection and tracking. *Front. Comput. Neurosci.* 16, 930827. doi: 10.3389/fncom.2022.930827

Jiang, Y., Moseson, S., and Saxena, A. (2011). "Efficient grasping from rgbd images: Learning using a new rectangle representation," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA)* (Shanghai, CHN: IEEE), 3304–3311.

Jiang, Z., Zhu, Y., Svetlik, M., Fang, K., and Zhu, Y. (2021). "Synergies between affordance and geometry: 6-DoF grasp detection via implicit representations," in *Robotics: Science and Systems XVII* (Robotics: Science and Systems Foundation).

Khan, A., Khan, A., Ullah, M., Alam, M. M., Bangash, J. I., Suud, M. M., et al. (2022). A computational classification method of breast cancer images using the VGGNet model. *Front. Comput. Neurosci.* 16, 1001803. doi: 10.3389/fncom.2022.1001803

Kumra, S., Joshi, S., and Sahin, F. (2020). "Antipodal Robotic Grasping using Generative Residual Convolutional Neural Network," In *Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Prague, CZ: IEEE), 9626–9633.

Lenz, I., Lee, H., and Saxena, A. (2013). *Deep Learning for Detecting Robotic Grasps*. London, England: SAGE Publications Sage UK.

Liang, H., Ma, X., Shuang, L., Grner, M., and Zhang, J. (2019). "PointNetGPD: detecting grasp configurations from point sets," In *Proceedings of the 2019 International Conference on Robotics and Automation (ICRA)* (Montreal, CAN: IEEE), 3629–3635.

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. *IEEE Trans. Patt. Anal. Mach. Intell.* 42, 318–327. doi: 10.1109/TPAMI.2018.2858826

Motwani, A., Shukla, P. K., Pawar, M., Kumar, M., Ghosh, U., Al Numay, W., et al. (2022). Enhanced framework for COVID-19 prediction with computed tomography scan images using dense convolutional neural network and novel loss function. *Comput. Electr. Eng.* 105, 108479. doi: 10.1016/j.compeleceng.2022.108479

Park, D., Seo, Y., Shin, D., Choi, J., and Chun, S. Y. (2020). "A single multi-task deep neural network with post-processing for object detection with reasoning and robotic grasp detection," in *Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA)* (Paris, FR: IEEE), 7300–7306.

Pas, A. T., Gualtieri, M., Saenko, K., and Platt, R. (2017). Grasp pose detection in point clouds. *Int. J. Robot. Res.* 36, 1455–1473. doi: 10.1177/0278364917735594

Redmon, J., and Farhadi, A. (2018). Yolov3: an incremental improvement. *arXiv preprint arXiv:1804.02767*

Shailendra, R., Jayapalan, A., Velayutham, S., Baladhandapani, A., Srivastava, A., Kumar Gupta, S., et al. (2022). An IoT and machine learning based intelligent system for the classification of therapeutic plants. *Neural Process. Lett.* 2022, 1–29. doi: 10.1007/s11063-022-10818-5

Singh, M., Shrimali, V., and Kumar, M. (2022). Detection and classification of brain tumor using hybrid feature extraction technique. *Multimedia Tools Appl.* 2022, 1–25. doi: 10.1007/s11042-022-14088-0

Song, Y., Gao, L., Li, X., and Shen, W. (2020). A novel robotic grasp detection method based on region proposal networks. *Robot. Cim-Int. Manuf.* 65, 101963. doi: 10.1016/j.rcim.2020.101963

Sundermeyer, M., Mousavian, A., Triebel, R., and Fox, D. (2021). "Contact-GraspNet: Efficient 6-DoF Grasp Generation in Cluttered Scenes," in *2021 IEEE International Conference on Robotics and Automation (ICRA)* (Xi'an, China: IEEE), 13438–13444.

Wang, D., Liu, C., Chang, F., Li, N., and Li, G. (2022). High-performance pixel-level grasp detection based on adaptive grasping and grasp-aware network. *IEEE T. Ind. Electron.* 69, 11611–11621. doi: 10.1109/TIE.2021.3120474

Wang, S., Jiang, X., Zhao, J., Wang, X., Zhou, W., Liu, Y., et al. (2019). "Efficient fully convolution neural network for generating pixel wise robotic grasps with high resolution images," in *Proceedings of the 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (Yunnan, CHN: IEEE), 474–480.

Wang, Y., Zheng, Y., Gao, B., and Huang, D. (2021). "Double-Dot Network for Antipodal Grasp Detection," in *Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Prague, CZ: IEEE), 4654–4661.

Wood, J. A. (2009). The Darknet: A digital copyright revolution. *Rich. JL Tech.* 16, 1. http://jolt.richmond.edu/v16i4/article14.pdf

Yang, D., Tosun, T., Eisner, B., Isler, V., and Lee, D. (2021). "Robotic Grasping through Combined Image-Based Grasp Proposal and 3D Reconstruction," in *2021 IEEE International Conference on Robotics and Automation (ICRA)* (Xi'an, China: IEEE), 6350–6356.

Yu, S., Zhai, D. H., and Xia, Y. (2022a). EGNet: Efficient Robotic Grasp Detection Network. *IEEE T. Ind. Electron.* 2022, 1–1. doi: 10.1109/TMECH.2022.3209488

Yu, S., Zhai, D. H., Xia, Y., Wu, H., and Liao, J. (2022b). SE-ResUNet: A novel robotic grasp detection method. *IEEE Robot. Autom. Lett.* 7, 5238–5245. doi: 10.1109/LRA.2022.3145064

Zhang, H., Lan, X., Bai, S., Zhou, X., Tian, Z., Zheng, N., et al. (2019). "ROI-based Robotic Grasp Detection for Object Overlapping Scenes," in *Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Macau, CHN: IEEE), 4768–4775.

Zhang, H. B., Lan, X. G., Zhou, X. W., Tian, Z. Q., Zhang, Y., Zheng, N. N., et al. (2018). Robotic grasping in multi-object stacking scenes based on visual reasoning. *Scientia Sinica Technologica.* 48, 1341–1356. doi: 10.1360/N092018-00169

Zhou, X., Lan, X., Zhang, H., Tian, Z., Zhang, Y., Zheng, N., et al. (2018). "Fully convolutional grasp detection network with oriented anchor box," In *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid, ES: IEEE), 7223–7230.

frontiers | Frontiers in Computational Neuroscience

TYPE Original Research
PUBLISHED 04 August 2023
DOI 10.3389/fncom.2023.1115167

Check for updates

OPEN ACCESS

EDITED BY
Si Wu,
Peking University, China

REVIEWED BY
Shi Yuxin,
Fudan University, China
Sidra Abbas,
COMSATS University Islamabad, Sahiwal Campus, Pakistan

*CORRESPONDENCE
Wei Qian
✉ qianweisdfw@163.com

RECEIVED 03 December 2022
ACCEPTED 11 July 2023
PUBLISHED 04 August 2023

CITATION
Ming M, Lu N and Qian W (2023) Evaluation of computed tomography images under deep learning in the diagnosis of severe pulmonary infection.
Front. Comput. Neurosci. 17:1115167.
doi: 10.3389/fncom.2023.1115167

Evaluation of computed tomography images under deep learning in the diagnosis of severe pulmonary infection

Mao Ming[1], Na Lu[2] and Wei Qian[3]*

[1]Department of Infectious Disease, South of Guang'anmen Hospital, China Academy of Chinese Medical Sciences, Beijing, China, [2]Department of Colorectal Surgery, South of Guang'anmen Hospital, China Academy of Chinese Medical Sciences, Beijing, China, [3]Department of Intensive Care Unit, South of Guang'anmen Hospital, China Academy of Chinese Medical Sciences, Beijing, China

This work aimed to explore the diagnostic value of a deep convolutional neural network (CNN) combined with computed tomography (CT) images in patients with severe pneumonia complicated with pulmonary infection. A total of 120 patients with severe pneumonia complicated by pulmonary infection admitted to the hospital were selected as research subjects and underwent CT imaging scans. The empty convolution (EC) and U-net phase were combined to construct an EC-U-net, which was applied to process the CT images. The results showed that the learning rate of the EC-U-net model decreased substantially with increasing training times until it stabilized and reached zero after 40 training times. The segmentation result of the EC-U-net model for the CT image was very similar to that of the mask image, except for some deviations in edge segmentation. The EC-U-net model exhibited a significantly smaller cross-entropy loss function (CELF) and a higher Dice coefficient than the CNN algorithm. The diagnostic accuracy of CT images based on the EC-U-net model for severe pneumonia complicated with pulmonary infection was substantially higher than that of CT images alone, while the false negative rate (FNR) and false positive rate (FPR) were substantially lower ($P < 0.05$). Moreover, the true positive rates (TPRs) of CT images based on the EC-U-net model for patchy high-density shadows, diffuse ground glass density shadows, pleural effusion, and lung consolidation were obviously higher than those of the original CT images ($P < 0.05$). In short, the EC-U-net model was superior to the traditional algorithm regarding the overall performance of CT image segmentation, which can be clinically applied. CT images based on the EC-U-net model can clearly display pulmonary infection lesions, improve the clinical diagnosis of severe pneumonia complicated with pulmonary infection, and help to screen early pulmonary infection and carry out symptomatic treatment.

KEYWORDS

empty convolution, deep convolutional neural network, severe pneumonia, pulmonary infection, computed tomography images

1. Introduction

Inflammation in lung tissues (bronchioles, alveoli, and interstitium) caused by various etiologies and pathogens on different occasions has similar or the same pathophysiological process and can deteriorate into severe pneumonia (Zhou et al., 2021; Al Khoury et al., 2022). It can be caused by various pathogenic causes. Pneumonia with cardiopulmonary foundation or additional risk factors or infection with special pathogenic microorganisms, such as severe acute respiratory syndrome (SARS) virus, avian influenza virus, and legionella bacteria, will aggregate pneumonia and increase the risk of death (Cai and Zheng, 2020; Gerges Harb et al., 2020; Wan et al., 2021). Severe pneumonia is a serious respiratory disease, and most patients will be complicated with organ dysfunction. In addition to the common respiratory symptoms of pneumonia, there are respiratory failure and obvious involvement of the circulatory system, nervous system, and other systems. Common symptoms include fever, chills, cough, expectoration, chest pain, dyspnea, and increased respiratory rate (Issa et al., 2020). Severe pneumonia will result in various sequelae, the most common of which is lung injury (such as bullae, empyema, and pyopneumothorax) and heart-related diseases, including heart failure or pulmonary heart disease. Therefore, it is very important to pay attention to the early diagnosis and treatment of severe pneumonia.

Imaging examination is an important process in the diagnosis of pneumonia and is one of the important indexes to judge severe pneumonia. Clinical diagnosis of lung lesions often adopts X-ray, bedside ultrasound, conventional chest computed tomography (CT) plain scan, etc (Hu et al., 2021). Chest X-ray examination is relatively convenient and cost-effective, but it exhibits great limitations in the patient's position and scope of fluoroscopy, which limits the imaging results and easily leads to a false negative result (Sayad et al., 2021). Ultrasound shows the lungs clearly and features with low price, is easy to operate, and is easily disturbed by lung gas. CT images are grayscale images with high density resolution that can clearly display the lung and other soft tissue organs at low cost and have been widely used in the diagnosis of lung diseases.

Image segmentation refers to finding and distinguishing the target area according to the properties and characteristics of the image. In the medical field, segmenting the images of tissue and organ lesions is an important auxiliary means for clinical diagnosis, treatment, and efficacy evaluation of diseases. Traditional image segmentation algorithms are still widely used, even in commercial applications. However, with the exponential growth of the current data volume, the requirements for the depth of information mining and segmentation technology are increasing, so it is necessary to study higher-level technologies (Arej et al., 2022). Deep learning is a deep nonlinear structure that is based on the human neural network mechanism, layered feature extraction, and recognition. Ideally, as long as the amount of data is sufficient and the network is deep enough, an ideal effect can be achieved, and the accuracy rate of human beings can even be exceeded (Diab et al., 2020; Alimoradi et al., 2021). Due to its excellent quality, deep learning is also widely used in medical image processing. Therefore, deep learning was combined with CT imaging technology and applied in clinical diagnosis in this work. Wang et al. (2021) discussed the application of deep learning technology in conical beam computed tomography image analysis of oral lesions, and processed images by artificial segmentation, threshold segmentation algorithm, and full convolutional neural network algorithm. The results showed that the image segmentation accuracy of the full convolutional neural network algorithm was superior to the traditional manual segmentation and threshold segmentation algorithms. Wu et al. (2020) proposed a deep convolutional neural network fusion support vector machine algorithm (DCNN-F-SVM) and applied it to brain tumor image segmentation. According to the segmentation results obtained, the image segmentation performance of this model was significantly better than that of deep convolutional neural network and integrated SVM classifier.

In summary, the combination of deep learning technology and medical imaging is still the focus of clinical research. Therefore, 120 patients with severe pneumonia complicated with pulmonary infection were selected as subjects for CT imaging scanning. An EC-U-net network model based on empty convolution (EC) and the U-net network phase was constructed and applied to patient CT image processing. The diagnostic value of a deep convolutional neural network (CNN) combined with CT images for severe pneumonia complicated with pulmonary infection was discussed by analyzing the imaging characteristics of patients. In this study, deep learning technology was innovatively combined with lung CT image, which was jointly applied in clinical treatment, providing a theoretical reference for the evaluation of lung infection in patients with pneumonia.

2. Materials and methods

2.1. Research objects

In this work, 120 patients with severe pneumonia complicated with pulmonary infection, aged 20–69, admitted to the hospital

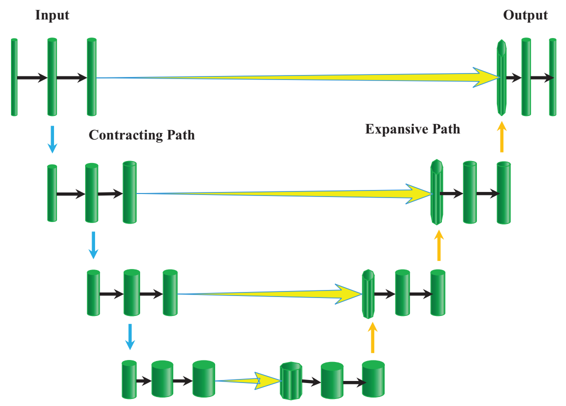


FIGURE 1
Schematic diagram of the U-net network structure.

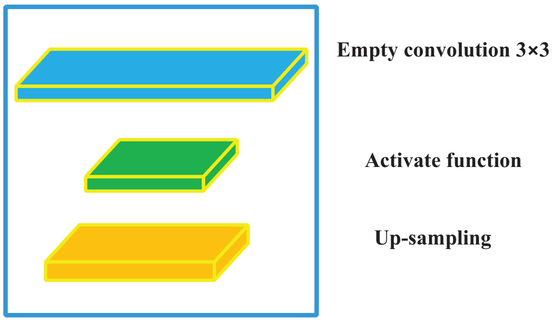


FIGURE 2
The convolution blocks of the EC-U-net network model.

from November 2019 to April 2021, were selected as the research subjects. This study was approved by the medical ethics committee of the hospital, and patients and their families were informed of this study and signed informed consent.

Inclusion criteria: (i) patients older than 18 years; (ii) patients with complete clinical data; (iii) patients who signed informed consent; (iv) patients who met the diagnostic criteria for severe pneumonia formulated by the American Society of Infectious Diseases/American Thoracic Society in 2007 (Vetrugno et al., 2020); and (v) the diagnosis of severe pneumonia was in accordance with the guidelines of the Respiratory Society of Chinese Medical Association in 2006 (Kim, 2020).

Exclusion criteria: (i) patients with autoimmune diseases; (ii) patients complicated with organ transplantation; (iii) patients with other tumors; (iv) patients with heart disease and other important organ damage; and (v) patients who had poor compliance with examination.

2.2. CT image scanning

All patients were scanned by 64-row spiral CT. The patients were placed in the supine position and scanned from the chest entrance to the bottom of the lung. The scanning parameters were as follows: layer thickness of 2.5 mm, pitch of 1.25, tube voltage of 120 kV, tube current of 120 mA, and matrix of 521 × 521.

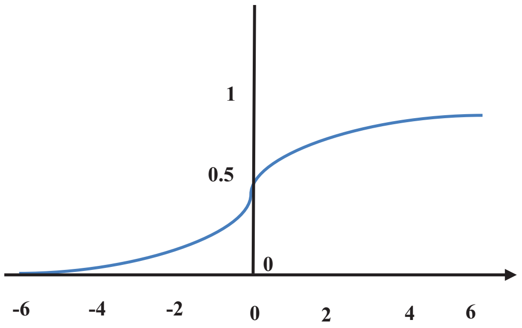


FIGURE 3
Sigmoid function.

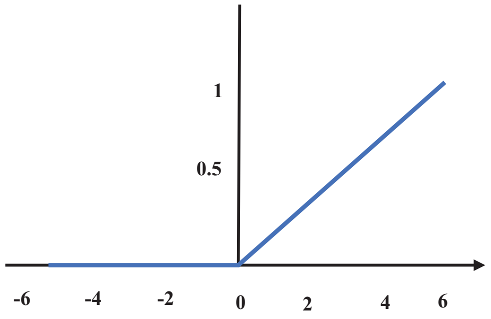


FIGURE 4
ReLU function diagram.

2.3. CT image segmentation based on the deep learning model

The U-net model (Feng et al., 2020) is an improved fully convolutional network (FCN) structure, which is generally composed of a contracting path on the left half and an expansive path on the right half (**Figure 1**). The compression channel is a typical CNN structure. It repeats the structure with two convolutional layers and one maximum pooling layer. The dimensionality of the feature map is doubled after each pooling operation. In the expansion channel, a deconvolution operation was performed first to reduce the dimensionality of the feature map by half, and then the feature maps obtained from the corresponding compression channel were spliced to reconstitute a feature map of two times the size. Then, two convolutional layers were adopted for feature extraction, and this structure was repeated. In the final output layer, two convolutional layers were employed to map the 64-dimensional feature map into a 2-dimensional output map.

Empty convolution (EC) (Moore and Gardiner, 2020) is essentially a convolution with intervals. It can enlarge the receiving field without changing the number of parameters and enhance the ability of the model to extract information. EC and the U-net network were combined to design an EC-U-net network model in this work. The convolution block of the model (**Figure 2**) mainly included the EC and activation function.

In the field of mathematics, convolution is an operation on a function. In fact, it is a weighted summation process, which is an integral operation. The convolution operation is expressed as the following equation.

$$(h_1^* h_2)(t) \triangleq \int_{-\infty}^{\infty} h_1(\upsilon) h_2(t-\upsilon) d\upsilon \tag{1}$$

In equation (1), h_1 and h_2 are functions, and two continuous functions are integrable within the real number range. When CNN is adopted to process the CT image, image pixels are used as input, the convolution kernel is an impact function that acts on the system and can extract system features, and the output is a feature map corresponding to the image. Therefore, the image convolution process is actually a linear operation, and the convolution of a two-dimensional vector is expressed as follows.

$$P^*(i,j) = (P \times C)(i,j) = \sum_a \sum_b P(a+i, b+j)^* C(a,b) \tag{2}$$

In equation (2), P represents the input, C represents the convolution kernel, with a step size of 1, P^* represents the output, and (i,j) represents the pixel coordinates. Since the input data dimension may not be an integer multiple of the convolution kernel dimension, the method of padding zeros in the edge area is usually adopted to protect effective information, and the filling column is introduced.

$$P = \left[\frac{i - c + 2k}{l}\right] + 1 \tag{3}$$

In equation (3), k represents the filling column, and l represents the step size. Weight sharing is a major feature of CNNs, which can greatly reduce the number of parameters and increase the nonlinearity of the model. Then, the total number of parameters (TNP) is calculated as shown in the following equation.

$$TNP = m + n = c^2 * z * j + j \tag{4}$$

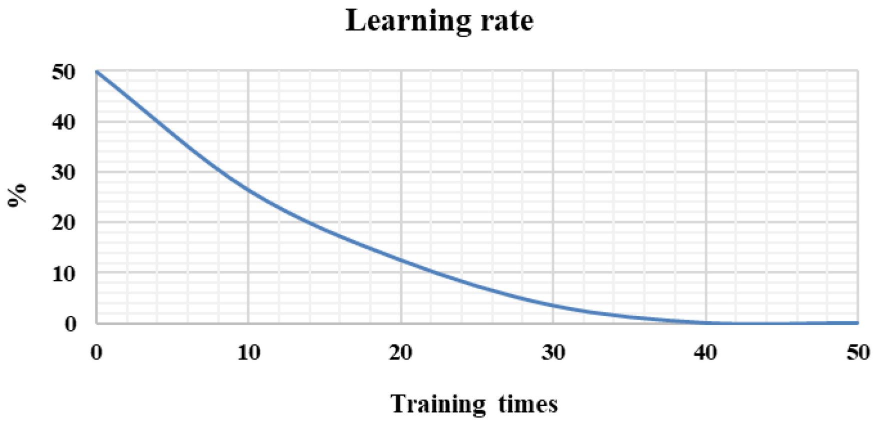


FIGURE 5
Model learning rate under different training times.

In equation (4), m represents the weight, n represents the bias value, z is the number of feature channels, and j represents the feature map. In practical applications, the EC may cause some image pixels to not participate in the convolution calculation due to the interval, thereby losing the continuity of some information. To solve this problem, the hole size of the model network is designed according to the hybrid dilated convolution (HDC) standard. The following condition is needed.

$$T_i = \max[T_{i+1} - 2e_i, T_{i+1} - 2(T_{i+1} - e_i), e_i] \tag{5}$$

In equation (5), e_i is the space interval of the i-th layer, and T_i is the space interval of the i-th layer.

Calculation of the convolutional layer is essentially a linear weighted summation, so the model lacks nonlinear expression, and the expression ability is extremely limited. Therefore, an activation function should be introduced. In this work, the sigmoid function (Ma et al., 2021) is used for classification output, and the ReLU function (Kwee and Kwee, 2020) is employed for internal feature extraction.

The sigmoid function can compress the value of the function to the range of (0, 1), and it can be derived everywhere (Figure 3), which is expressed as follows.

$$\text{Sigmoid} = {}^{1}\!\Big/\!\left(1 + e^{-(mx+n)}\right) \tag{6}$$

The sigmoid function is related to the parameter update and model optimization, and the step size of the parameter update is related to the gradient. The reverse transfer process of the gradient between the layers can be expressed as the following equation.

$$\frac{\partial D}{\partial b_u} = \text{Sigmoid}'(z_c)m_{c+1}\text{Sigmoid}'(z_{c+1})m_{c+2} \cdots \text{Sigmoid}'(z_u)\frac{\partial D}{\partial a_v} \tag{7}$$

In equation (7), a_v is the v-layer output, and $\frac{\partial D}{\partial b_u}$ is the gradient of the objective function to the bias term.

The ReLU function (Figure 4) can effectively avoid gradient disappearance. It is an optimization of the sigmoid function, which is expressed as the following equation.

$$Relu() = \max(0, z) \tag{8}$$

$$Relu()' = \begin{cases} 0 \ z < 0 \\ 1 \ z > 0 \end{cases} \tag{9}$$

After the model is constructed, a learning criterion should be set to supervise the model or select the optimal model. The cross-entropy loss function (CELF) and Dice coefficient are used as the learning criteria, which are expressed as the following equations.

$$CELF() = -\frac{\sum[X\log(F(I)) + (1-X)\log(F(I))]}{n} \tag{10}$$

$$Dice = 2^{*}\frac{M \cap N}{|M| + |N|} \tag{11}$$

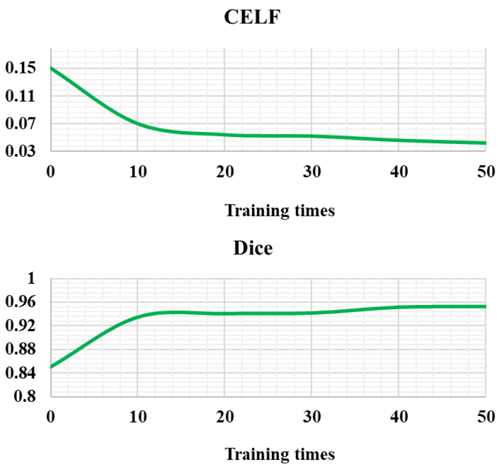


FIGURE 6
Cross-entropy loss function and Dice coefficient of the EC-U-net model.

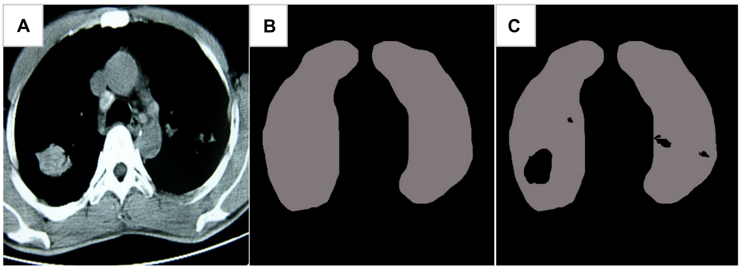


FIGURE 7
Image segmentation results of the EC-U-net model. **(A)** Lung CT; **(B)** mask diagram; **(C)** segmentation results.

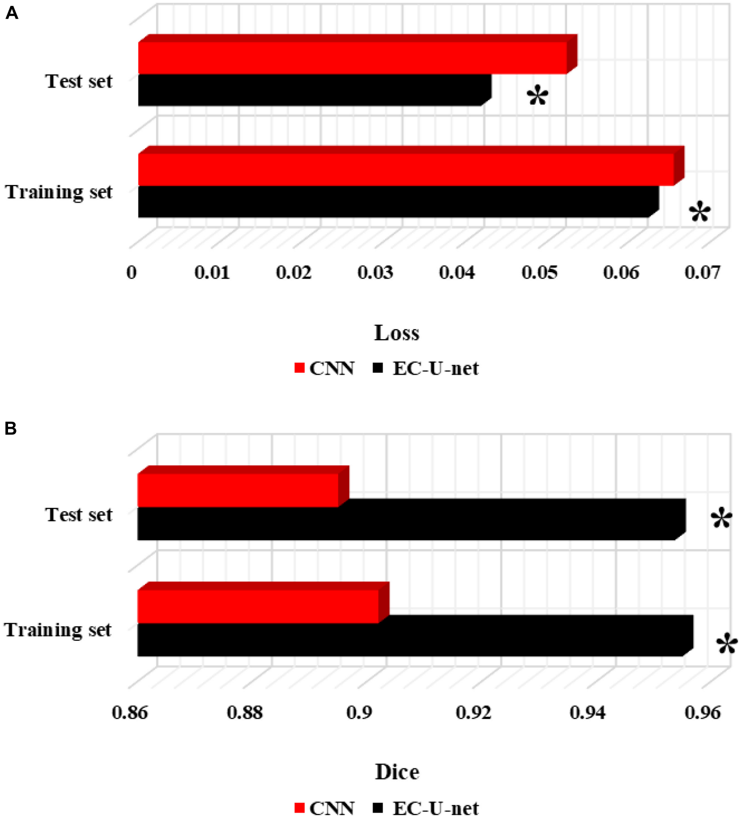


FIGURE 8
Comparison of segmentation performance between the traditional CNN algorithm and this model. **(A)** CELF; **(B)** Dice coefficient. *Compared to the CNN algorithm, $P < 0.05$.

In equation (11), M represents the pixel matrix of the image mask, N is the pixel matrix of the output predicted image, and $M \cap N$ represents the inner product of the two image matrices.

2.4. Construction of the experimental environment

The operating system is Windows 10, the processor uses Xeon CPU E5-2630, and the graphics card uses NVIDIA Quadro K2200. The framework uses TensorFlow, the language uses Python3.5, and the dependent libraries use CUDA9.0, cudnn, OpenCV, and SimpleITK.

The data set uses lung data from the Kaggle competition, which includes 2,650 lung images and corresponding 250 mask images made by experts. The training set and test set are set to 1:1.

2.5. Statistical methods

SPSS 19.0 was used for data processing in this study. The mean ± standard deviation ($\overline{X} \pm s$) was used to indicate the measurement data, and the percentage (%) was used for counting data. Pairwise comparisons were performed by one-way ANOVA. The difference was statistically significant at $P < 0.05$.

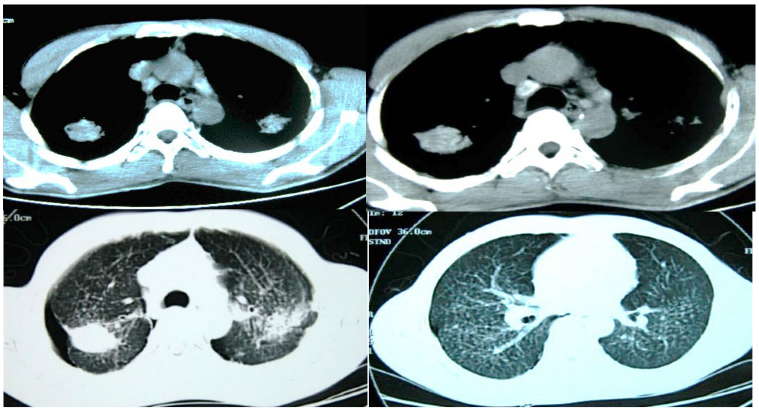


FIGURE 9
CT images of a 38-year-old male patient.

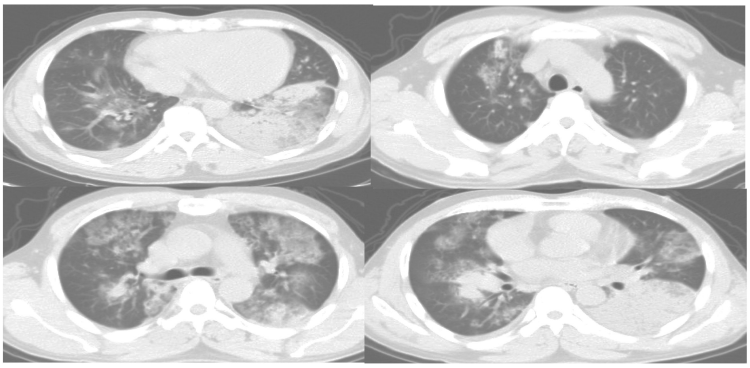


FIGURE 10
CT images of a 51-year-old male patient.

3. Results

3.1. Experimental results

In **Figure 5**, the learning rate of the EC-U-net model decreased substantially as the number of training iterations increased until it stabilized and became zero after 40 training iterations.

The CELF and Dice coefficients were compared (**Figure 6**). The CELF of the EC-U-net model attenuated with increasing training times, while the Dice coefficient increased with increasing training times (gradually approaching 1) until it was stable.

3.2. Application effect of the EC-U-net model in CT images

Figure 7 showed the CT image segmentation result of the EC-U-net model. The result of CT image segmentation using the EC-U-net model was very similar to the mask image, but there were some deviations in the segmentation at the edge.

The traditional CNN algorithm was introduced and compared with the segmentation results of the established model (**Figure 8**). The CELF of the EC-U-net model for lung CT image segmentation was observed to be substantially smaller than that of the CNN algorithm, and the difference was considerable ($P < 0.05$). The Dice coefficient of the EC-U-net model for lung CT image segmentation was substantially greater than that of the CNN algorithm ($P < 0.05$).

3.3. Patient imaging findings

Figure 9 showed the CT images of a 38-year-old male patient, showing multiple small nodules in both lungs, mostly in the upper and posterior parts of the lungs; fibrotic masses were observed in the posterior segments of the upper lobes of both lungs, with bilateral symmetry and extravasation-like changes. Pulmonary bullae were observed below the pleura in the periphery of the lungs where the nodules were concentrated, and a pneumothorax shadow was seen on the periphery of the lungs with localized pleural hypertrophy.

Figure 10 showed the CT images of a 51-year-old male patient, showing multiple segmental lesions in both lungs spreading more than before. It was considered infectious lesions, multiple small lymph nodes in the mediastinum, and a small amount of free effusion in the right pleural cavity.

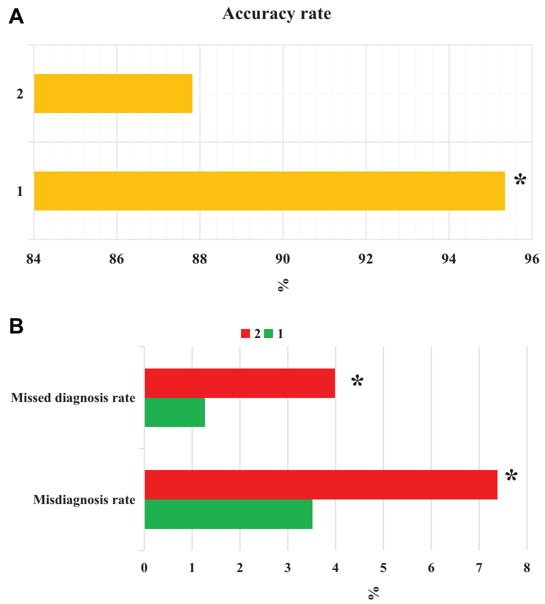


FIGURE 11
Comparison of patient diagnosis accuracy, FNR, and FPR. **(A)** accuracy; **(B)** FNR and FPR. 1: CT image based on the EC-U-net model; 2: original CT image. *Compared with 1, *P* < 0.05.

3.4. Comparison of patient diagnosis accuracy, false positive rate (FPR) and false negative rate (FNR)

From **Figure 11**, the accuracy of CT images based on the EC-U-net model in the diagnosis of severe pneumonia combined with infection was substantially higher than that of CT images ($P < 0.05$). The FPR and FNR of CT images based on the EC-U-net model for severe pneumonia complicated by infection were substantially lower than those of CT images, and the differences were considerable ($P < 0.05$).

3.5. Comparison of diagnosis results of CT imaging features

Figure 12 compared the diagnosis results of patients with CT imaging features. The CT image based on the EC-U-net model had a true positive rate (TPR) of 57.93% for patchy high-density shadows and a TPR of 75.31% for diffuse ground-glass density shadows. The TPRs were 16.39, 32.88, and 5.08% for pleural effusion, pulmonary consolidation, and reticular nodules, respectively. The original CT image had a TPR of 48.89% in the diagnosis of patchy high-density shadows, 64.03% in diffuse ground-glass density shadows, 11.27% in pleural effusion, 24.91% in pulmonary consolidation, and 4.55% in reticular nodules. In short, CT images processed by the EC-U-net model had a higher TPR for patchy high-density shadows, diffuse ground glass density shadows, pleural effusions, and lung consolidation shadows than the original CT images, and the differences were substantial ($P < 0.05$).

4. Discussion

Severe pneumonia is a very common critical symptom around the world. It usually occurs in elderly individuals. Because its onset is relatively insidious and there are no obvious symptoms in the early stage, it will lead to delayed detection of the patient's condition and endanger the life of the patient (Salerno et al., 2021). Therefore, early examination and early treatment are of great significance to patients with severe pneumonia complicated with pulmonary infection (Ding et al., 2020; Huang et al., 2021). Thanks to the continuous development of computer technology, medical imaging technology has gradually exceeded the scope of traditional X-ray photography, among which CT imaging technology is widely adopted in the diagnosis of various diseases because of its high accuracy, low cost, and convenient operation. A total of 120 patients with severe pneumonia combined with pulmonary infection were selected as the research subjects and

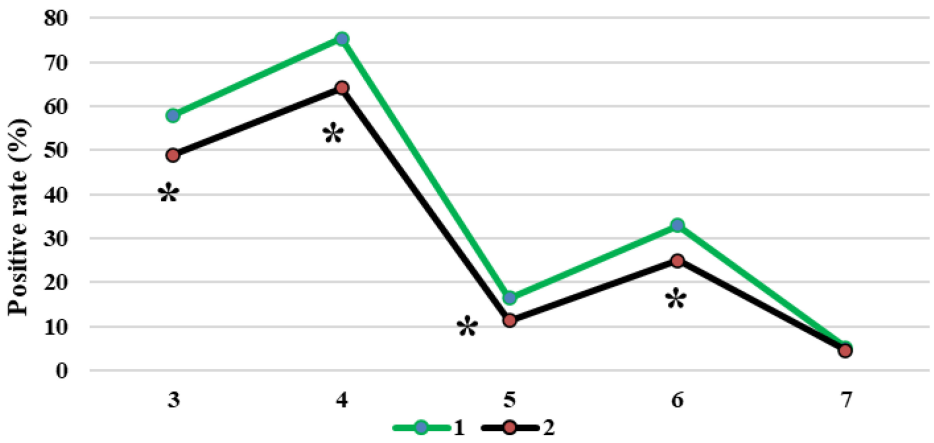


FIGURE 12
Comparison of diagnosis results of patients with CT imaging features. 1: CT image based on the EC-U-net model; 2: original CT image; 3: patchy high-density shadow; 4: diffuse ground-glass-like density shadow; 5: pleural effusion; 6: lung consolidation shadow; 7: reticular nodule shadow. *Compared with 1, *P* < 0.05.

underwent CT imaging scans. Then, an EC-U-net network model was constructed based on empty convolution and the U-net network and applied to CT image processing. First, analysis of the performance of the model suggested that the learning rate of the EC-U-net model decreased substantially as the training times increased until it stabilized, and it even became 0 when there were 40 training times. Such results indicated that the training efficiency of the model was high and local fluctuations were avoided. The segmentation result of the CT image by the EC-U-net model was very similar to the mask image, but there were some deviations in the edge segmentation. This was different from the results of Shi et al. (2021), indicating that the segmentation effect on the microstructure of the lungs in the EC-U-net model was not satisfactory. The segmentation results of the introduced traditional CNN algorithm and the proposed model were compared. It was found that the EC-U-net model for lung CT image segmentation exhibited a substantially smaller CELF and a greatly larger Dice coefficient than the CNN algorithm ($P < 0.05$). This showed that the overall segmentation performance of the EC-U-net model for CT images was better than that of traditional algorithms, and it had clinical application feasibility (Gordaliza et al., 2018).

Accuracy reflects the precision of prediction. The FNR and FPR are a pair of indicators from the perspective of prediction coverage. The EC-U-net model was applied to the CT image processing of 120 cases of severe pneumonia combined with pulmonary infection. It was found that the accuracy of CT images based on the EC-U-net model in the diagnosis of severe pneumonia complicated by infection was substantially higher than that of CT images. The FNR and FPR of CT images based on the EC-U-net model for severe pneumonia complicated by infection were substantially lower than those of CT images, and the differences were great ($P < 0.05$). This showed that the combination of the EC-U-net model and CT images can effectively improve the clinical diagnosis of severe pneumonia complicated by pulmonary infection, improve the diagnostic accuracy, and help screen early pulmonary infections for symptomatic treatment (Haas et al., 2017; Borodulina et al., 2020). Then, the CT image characteristics of patients were analyzed, and the CT images based on the EC-U-net model had a higher TPR for patchy high-density shadows, diffuse ground-glass density shadows, pleural effusions, and pulmonary consolidation shadows, and the differences were notable ($P < 0.05$). This is similar to the research results of Morris et al. (2020), indicating that CT images based on the EC-U-net model can clearly show pulmonary infection lesions and determine the scope of the lesion, thereby providing a diagnostic basis for the early diagnosis of severe pneumonia combined with pulmonary infection.

5. Conclusion

In this research, 120 patients with severe pneumonia complicated with pulmonary infection were recruited to receive CT imaging scans. Furthermore, an EC-U-net network model based on the EC and U-net network phases was constructed and applied to process the CT images of patients. The results showed that the EC-U-net model was superior to the traditional algorithm in the overall performance of CT image segmentation and had feasibility for clinical application. CT images based on the EC-U-net model can clearly display pulmonary infection lesions, improve the clinical diagnosis of severe pneumonia complicated with pulmonary infection, and help to screen early pulmonary infection and carry out symptomatic treatment. However, this study has not solved the unideal segmentation effect of the EC-U-net model on microscopic structures such as tiny pulmonary vessels, and imaging analysis of pulmonary infections caused by different pathogens is lacking. In future studies, we will include more case data of patients with severe pneumonia complicated with pulmonary infection, and conduct more image segmentation experiments with the proposed algorithm to verify the reliability of deep learning technology. In conclusion, the results provide data support for the clinical diagnosis and treatment of severe pneumonia complicated with pulmonary infection.

Data availability statement

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

Ethics statement

The studies involving human participants were reviewed and approved by the China Academy of Chinese Medical Sciences. The patients/participants provided their written informed consent to participate in this study.

Author contributions

MM: writing-original draft, conceptualization, and formal analysis. NL: software and validation. WQ: methodology, writing-review, and editing. All authors contributed to the article and approved the submitted version.

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

Al Khoury, C., Bashir, Z., Tokajian, S., Nemer, N., Merhi, G., and Nemer, G. (2022). In silico evidence of beauvericin antiviral activity against SARS-CoV-2. *Comput. Biol. Med.* 141:105171. doi: 10.1016/j.compbiomed.2021.105171

Alimoradi, M., Chahal, A., El-Rassi, E., Daher, K., and Sakr, G. (2021). Synthol systemic complications: hypercalcemia and pulmonary granulomatosis. A case report. *Ann. Med. Surg.* 69, 102771. doi: 10.1016/j.amsu.2021.102771

Arej, N., Mechleb, N., Issa, M., Cherfan, G., Tomey, K., Abdelmassih, Y., et al. (2022). Combining spectral domain optical coherence tomography of retinal nerve fiber layer and noncontact tonometry in mass glaucoma screening during the World Glaucoma Week. *J. Fr. Ophtalmol.* 45, 384–391. doi: 10.1016/j.jfo.2021.11.010

Borodulina, E., Vasneva, Z., Borodulin, B., Vdoushkina, E., Povalyaeva, L., and Mateesku, K. (2020). Hematological indicators for lung damage caused by COVID-19 infection. *Klin. Lab. Diagn.* 65, 676–682. doi: 10.18821/0869-2084-2020-65-11-676-682

Cai, Z. P., and Zheng, X. (2020). A private and efficient mechanism for data uploading in smart cyber-physical systems. *IEEE Trans. Netw. Sci. Eng.* 7, 766–775.

Diab, K., Rieger, K., and Noor, A. (2020). Endobronchial valve placement for pulmonary tuberculosis-related bronchocutaneous fistula after thoracoplasty. *J. Bronchol. Interv. Pulmonol.* 27, 294–296. doi: 10.1097/LBR.0000000000000688

Ding, X., Xu, J., Zhou, J., and Long, Q. (2020). Chest CT findings of COVID-19 pneumonia by duration of symptoms. *Eur. J. Radiol.* 127:109009. doi: 10.1016/j.ejrad.2020.109009

Feng, X., Ding, X., and Zhang, F. (2020). Dynamic evolution of lung abnormalities evaluated by quantitative CT techniques in patients with COVID-19 infection. *Epidemiol. Infect.* 148:e136. doi: 10.1017/S0950268820001508

Gerges Harb, J., Noureldine, H., Chedid, G., Eldine, M., Abdallah, D., Chedid, N., et al. (2020). SARS, MERS and COVID-19: clinical manifestations and organ-system complications: a mini review. *Pathog. Dis.* 78:ftaa033. doi: 10.1093/femspd/ftaa033

Gordaliza, P., Muñoz-Barrutia, A., Abella, M., Desco, M., Sharpe, S., and Vaquero, J. (2018). Unsupervised CT lung image segmentation of a mycobacterium tuberculosis infection model. *Sci. Rep.* 8:9802. doi: 10.1038/s41598-018-28100-x

Haas, B., Clayton, J., Elicker, B., Ordovas, K., and Naeger, D. (2017). CT-guided percutaneous lung biopsies in patients with suspicion for infection may yield clinically useful information. *Am. J. Roentgenol.* 208, 459–463. doi: 10.2214/AJR.16.16255

Hu, M., Zhong, Y., Xie, S., Lv, H., and Lv, Z. (2021). Fuzzy system based medical image processing for brain disease prediction. *Front. Neurosci.* 15:714318. doi: 10.3389/fnins.2021.714318

Huang, T., Zheng, X., He, L., and Chen, Z. (2021). Diagnostic value of deep learning-based CT feature for severe pulmonary infection. *J. Healthc. Eng.* 2021:5359084. doi: 10.1155/2021/5359084

Issa, E., Merhi, G., Panossian, B., Salloum, T., and Tokajian, S. (2020). SARS-CoV-2 and ORF3a: nonsynonymous mutations, functional domains, and viral pathogenesis. *mSystems* 5, e00266–20. doi: 10.1128/mSystems.00266-20

Kim, E. (2020). CT diagnosis of coronavirus infection. *Curr. Med. Imaging* 16:273. doi: 10.2174/157340561604200402091854

Kwee, T., and Kwee, R. (2020). Chest CT in COVID-19: what the radiologist needs to know. *Radiographics* 40, 1848–1865. doi: 10.1148/rg.2020200159

Ma, J., Wang, Y., An, X., Ge, C., Yu, Z., Chen, J., et al. (2021). Toward data-efficient learning: a benchmark for COVID-19 CT lung and infection segmentation. *Med. Phys.* 48, 1197–1210. doi: 10.1002/mp.14676

Moore, S., and Gardiner, E. (2020). Point of care and intensive care lung ultrasound: a reference guide for practitioners during COVID-19. *Radiography* 26, e297–e302. doi: 10.1016/j.radi.2020.04.005

Morris, M., Goettel, C., Mendenhall, C., Chen, S., and Hirsch, K. (2020). Diagnosis of asymptomatic COVID-19 infection in a patient referred for CT lung biopsy. *J. Vasc. Interv. Radiol.* 31, 1194–1195. doi: 10.1016/j.jvir.2020.04.002

Salerno, D., Oriaku, I., Darnell, M., Lanclus, M., De Backer, J., Lavon, B., et al. (2021). Temple University Covid-19 Research Group. Association of abnormal pulmonary vasculature on CT scan for COVID-19 infection with decreased diffusion capacity in follow up: a retrospective cohort study. *PLoS One* 16:e0257892. doi: 10.1371/journal.pone.0257892

Sayad, E., Coleman, R., Chartan, C., and Tillman, R. (2021). Diagnostic delays and characteristics of pediatric pulmonary hypertension presenting as syncope. *Clin. Pediatr.* 60, 443–446. doi: 10.1177/00099228211037190

Shi, F., Wei, Y., Xia, L., Shan, F., Mo, Z., Yan, F., et al. (2021). Lung volume reduction and infection localization revealed in big data CT imaging of COVID-19. *Int. J. Infect. Dis.* 102, 316–318. doi: 10.1016/j.ijid.2020.10.095

Vetrugno, L., Bove, T., Orso, D., Barbariol, F., Bassi, F., Boero, E., et al. (2020). Our Italian experience using lung ultrasound for identification, grading and serial follow-up of severity of lung involvement for management of patients with COVID-19. *Echocardiography* 37, 625–627. doi: 10.1111/echo.14664

Wan, Z., Dong, Y., Yu, Z., Lv, H., and Lv, Z. (2021). Semi-supervised support vector machine for digital twins based brain image fusion. *Front. Neurosci.* 15:705323. doi: 10.3389/fnins.2021.705323

Wang, X., Meng, X., and Yan, S. (2021). Deep learning-based image segmentation of cone-beam computed tomography images for oral lesion detection. *J. Healthc. Eng.* 2021:4603475. doi: 10.1155/2021/4603475

Wu, W., Li, D., Du, J., Gao, X., Gu, W., Zhao, F., et al. (2020). An intelligent diagnosis method of brain MRI tumor segmentation using deep convolutional neural network and SVM algorithm. *Comput. Math. Methods Med.* 2020:6789306. doi: 10.1155/2020/6789306

Zhou, X., Li, Y., and Liang, W. (2021). CNN-RNN based intelligent recommendation for online medical pre-diagnosis support. *IEEE/ACM Trans. Comput. Biol. Bioinform.* 18, 912–921. doi: 10.1109/TCBB.2020.2994780

frontiers | Frontiers in Computational Neuroscience

TYPE Original Research
PUBLISHED 30 August 2023
DOI 10.3389/fncom.2023.1204445

Check for updates

OPEN ACCESS

EDITED BY
Si Wu,
Peking University, China

REVIEWED BY
Laith Abualigah,
Amman Arab University, Jordan
Saghir Alfasly,
Mayo Clinic, United States

*CORRESPONDENCE
Chibiao Liu
✉ lcbsmc@163.com

RECEIVED 12 April 2023
ACCEPTED 03 August 2023
PUBLISHED 30 August 2023

CITATION
Gezawa AS, Liu C, Jia H, Nanehkaran YA, Almutairi MS 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

An improved fused feature residual network for 3D point cloud data

Abubakar Sulaiman Gezawa[1], Chibiao Liu[1]*, Heming Jia[1], Y. A. Nanehkaran[2], Mubarak S. Almutairi[3] and Haruna Chiroma[4]

[1]College of Information Engineering, Fujian Key Lab of Agriculture IOT Application, Sanming University, Sanming, Fujian, China, [2]Department of Software Engineering, School of Information Engineering, Yancheng Teachers University, Yancheng, Jiangsu, China, [3]College of Computer Science and Engineering, University of Hafr Al-Batin, Hafar Al Batin, Saudi Arabia, [4]College of Computer Science and Engineering Technology, Applied College, University of Hafr Al-Batin, Hafar Al Batin, Saudi Arabia

Point clouds have evolved into one of the most important data formats for 3D representation. It is becoming more popular as a result of the increasing affordability of acquisition equipment and growing usage in a variety of fields. Volumetric grid-based approaches are among the most successful models for processing point clouds because they fully preserve data granularity while additionally making use of point dependency. However, using lower order local estimate functions to close 3D objects, such as the piece-wise constant function, necessitated the use of a high-resolution grid in order to capture detailed features that demanded vast computational resources. This study proposes an improved fused feature network as well as a comprehensive framework for solving shape classification and segmentation tasks using a two-branch technique and feature learning. We begin by designing a feature encoding network with two distinct building blocks: layer skips within, batch normalization (BN), and rectified linear units (ReLU) in between. The purpose of using layer skips is to have fewer layers to propagate across, which will speed up the learning process and lower the effect of gradients vanishing. Furthermore, we develop a robust grid feature extraction module that consists of multiple convolution blocks accompanied by max-pooling to represent a hierarchical representation and extract features from an input grid. We overcome the grid size constraints by sampling a constant number of points in each grid using a simple K-points nearest neighbor (KNN) search, which aids in learning approximation functions in higher order. The proposed method outperforms or is comparable to state-of-the-art approaches in point cloud segmentation and classification tasks. In addition, a study of ablation is presented to show the effectiveness of the proposed method.

KEYWORDS

point clouds, part segmentation, classification, shape features, 3D objects recognition

1. Introduction

Three-dimensional (3D) data are a great asset in the computer vision field since it contains detailed information on the whole geometry of detected objects and scenes. With the availability of massive 3D datasets and processing power, it is now possible to apply deep learning to learn specific tasks on 3D data such as segmentation with classification (Varga et al., 2020; Ergün and Sahillioglu, 2023; Qi et al., 2023), recognition, and correspondence

(Long et al., 2021). There are several categories of 3D data representations including point cloud, voxel, mesh, multi views, octree, and many others. A comprehensive overview of point clouds and other 3D data representations may be found in the study by Bello et al. (2020) and Gezawa et al. (2020). Point cloud data processing employs a variety of approaches. Following dispatching a point cloud to a voxel grid that is quantized spatially in the grid space, volumetric models use a volumetric convolution to compute (Maturana and Scherer, 2015; Choy et al., 2016). Volumetric approaches correlate points with grid positions by using grids as data structuring technique and convolutional kernels in 3D to get data from nearby voxels. Although grid data structures are efficient, to maintain the granularity of the data position, a high voxel resolution is essential. The amount of processing and memory used grows in a cubical relationship with the voxel resolution since large point clouds are expensive to process. Furthermore, most point clouds contain ∼90% empty voxels (Zhou and Tuzel, 2018), processing no data could use a lot of computing power. Point-based models are another type of point cloud data processing paradigm. Unlike volumetric models, point-based models offer effective computation but have poor data organization. For instance, PointNet (Charles et al., 2017) aggregates the data in the network's final stage using the point cloud without quantization, as a result the precise locations of the data are preserved. However, the cost of computation rises in lockstep with the point number. Subsequent studies (Qi et al., 2017; Wang et al., 2018; Yifan et al., 2018; Qiangeng et al., 2019; Wang Y. et al., 2019) aggregate information using a downsampling approach at each layer. Graph convolutional networks (GCN) have been used in the network layer to generate a local graph for each point cluster (Simonovsky and Komodakis, 2017; Kuangen et al., 2019; Wang L. et al., 2019; Li et al., 2023) that can be regarded as a variant of the PointNet++ design (Qi et al., 2017). This architecture, however, is costly in terms of data structuring [e.g., Random Point Sampling (RPS)]. As reported by Zhijian et al. (2019), data structuring costs account for up to 88% of the entire computational cost in three common point-based models (Li Y. et al., 2018; Yifan et al., 2018; Wang Y. et al., 2019). Furthermore, SO-Net (Li J. et al., 2018) employs the self-organizing map (SOM; Kohonen, 1998) to create a set of points used to model a point cloud's spatial pattern. Even though SO-Net considers a point cloud's regional correlation, SOM is trained independently. As a result, SOM's spatial modeling and a specific point cloud task are no longer coupled. DGCB-Net (Tian et al., 2020) uses cutting-edge convolutional layers built by weight-shared multiple-layer perceptrons (MLPs), to automatically extract local features from the point cloud graph structure. A feature aggregation is formed by concatenating the features received from all edge convolutional layers. Rather than stacking multiple layers deep, the DGCB-Net adopts a strategy to flatly extend point cloud feature aggregation.

In this study, we utilize deep learning to develop an approach that manage enormous 3D object datasets without compromising shape resolution. The majority of handcrafted 3D features are limited to low 3D resolutions. For example, Chiotellis et al. (2016) and Zhou and Tuzel (2018) require each 3D model in the datasets to be down-sampled to 20,000 faces with Meshlab before they can be fed into the system. Additionally, a method is provided that can handle structural variations in 3D objects without the need for data pre-processing. Many machine learning algorithms, such as the support vector machine (SVM), are effective when the datasets are small and well-curated, which implies that the data have been carefully pre-processed and requires human intervention. To address these challenges, this study offers an improved fused feature network, an end-to-end framework that solves shape classification and segmentation tasks using a two-branch technique with feature representation learning. To efficiently simplify the network, we start by developing a feature encoding network with two independent building blocks and layer skips with batch normalization and ReLU in between. Because there are few layers through which to propagate, using the layer skips speeds up learning and lessens the effect of gradients vanishing. Figure 1 presents the entire network structure of the approach. In addition, we create a detail grid feature extraction module, which comprises various convolution blocks accompanied by a max-pooling to represent a hierarchical representation of several feature representations and extracts features from the input grid. Max-pooling is used in each of the pooling layers, resulting in each spatial dimension having a smaller grid and helps to manage overfitting by gradually lowering the representation's spatial dimension, the parameters in the network, and the amount of processing. This module includes a regular-structured enclosing volumetric grid that helps capture details and features hierarchically. To extract features of high-resolution inputs, this module is utilized in conjunction with the feature encoding network. To pull through the limitation of the grid size, the local region in every grid sampled a constant number of points using a simple KNN search which aids in learning approximation functions in higher order to better characterize the details of the features.

Our major contributions are as follows:

- We design an effective module named detail grid feature extraction (DGFE) module. This module aids 3D convolutions to hierarchically capture global information and reduces the grid size in each spatial dimension as well as managing overfitting by gradually lowering the spatial dimension of the representation making it viable for high-resolution 3D objects.
- We design a feature encoding network that uses two different building blocks with layer skips containing batch normalization and ReLU in between, resulting in fewer layers in the early training phase which helps speed learning and reduces the effect of gradients vanishing since there are few layers through which to propagate.
- We built a network using the modules that have been proposed, which achieves a notable balance of accuracy and speed.

2. Related work

2.1. 3D learning using voxel-based methods

To build on the advance of CNN models on images (He et al., 2016a; Huang et al., 2017), Voxnet and its revisions (Maturana and Scherer, 2015; Wang and Posner, 2015; Wu et al., 2015; Brock et al., 2016) start by converting a point cloud to a grid occupancy and then used convolution in a volumetric form. To overcome the problem

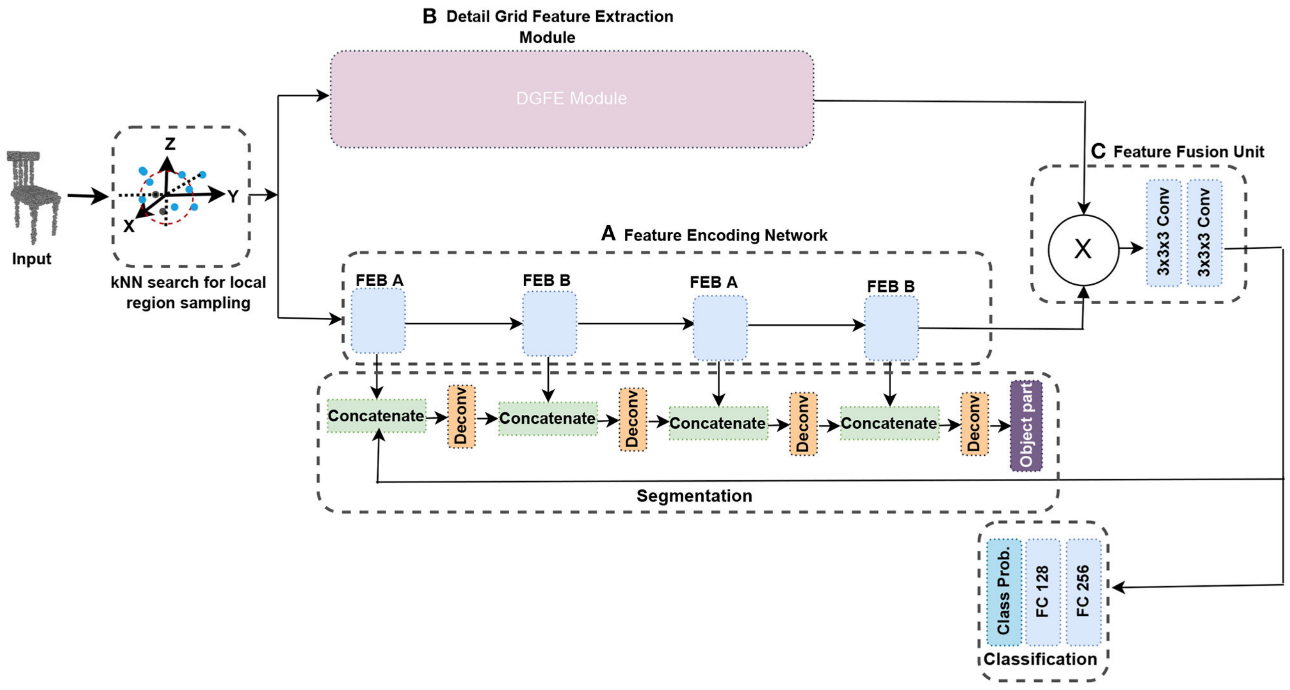


FIGURE 1
The complete architecture of the proposed method. The network is divided into three branches. The feature encoding network extract features from the input grid in **(A)**. The DGFE module exploits the detailed shape characteristics in **(B)**. The feature fusion unit which has two consecutive convolutional layers, fuses the features from the two branches to produce a feature with improved contextual representation by exploiting both local and global shape structures in **(C)**. See also Section 3.5.

of rising memory usage due to cubical expansion, OctNet creates structures like a tree for non-empty voxels to avoid computing in space. While the volumetric approach is effective at structuring data, it suffers from poor computational effectiveness and data granularity loss. Transformers have lately been incorporated into the model designs of many 3D vision approaches in response to the success of transformer-based designs in the two-dimensional (2D) domain. The transformer has improved previous 3D learning techniques because of its ability to read remote input and provide task-specific inductive biases. The point-voxel transformer for single-stage 3D detection (PVT-SSD) proposed by Yang et al. (2023) uses input-dependent query initialization and voxel-based sparse convolutions for strong feature encoding. The PVT-SSD overcame the drawbacks of both point clouds and voxels by combining their advantages. To reduce farthest point sampling (FPS) runtime, they used sparse convolutions to transform points into a limited number of voxels rather than directly sampling them. They also sampled non-empty voxels. The voxel features were adaptively blended with the point features to make up for the difficulty of quantization.

2.2. 3D learning using point cloud-based methods

Charles et al. (2017); Qi et al. (2017) pioneered the use of point-based models which used pooling to aggregate the point features to achieve the permutation invariant. To better capture local characteristics, methods such as kernel correlation (Atzmon et al., 2018; Wu et al., 2019) and extended convolutions (Thomas et al., 2019) are proposed. To resolve the ambiguity, the local point order is predicted by PointCNN (Li Y. et al., 2018) while RSNet (Huang et al., 2018) sequentially consumes points from various directions. In methods based on points, the cost of computation grows linearly with the points input. The cost of structuring data, nevertheless, turned out to be a performance bottleneck for large inputs. Recently, a dynamic sparse voxel transformer (DSVT) was presented by Wang et al. (2023) in an effort to widen the uses of transformers so that they may serve as a solid foundation for outdoor 3D perception just as they do for 2D vision. A number of local regions are split up into smaller ones in each window using DSVT based on sparsity, and each window's attributes are then computed fully in parallel. Another recent point cloud classification framework named point content-based transformer (PointConT) was introduced by Liu et al. (2023), and it employs local self-attention in the space of features rather than the 3D space. One of the main advantages of PointConT is that it takes advantage of the locality of points in the feature space by clustering sampled points with similar features into the same class and computing self-attention within each class, allowing for an efficient trade-off between collecting long-range dependencies and computational complexity.

2.3. Strategies for point data structuring

The majority of point-based methods (Qi et al., 2017; Li Y. et al., 2018; Bello et al., 2021; Gezawa et al., 2021) employ FPS (Eldar et al., 1997) to sample uniformly distributed group centers. However, it does not account for the subsequent processing of the sampled points which may result in suboptimal performance. Random point sampling (RPS) has the advantage of having a minimal downtime. It is indeed, nevertheless, sensitive to variation in density. The KNN search we used for sampling the local region in each grid cell combines sampling and neighbor querying in a single step, making it faster than RPS.

SO-Net (Li J. et al., 2018), on the other hand, creates a self-organizing map. To split the spaces, KDNet (Klokov and Lempitsky, 2017) employs kd-tree. Gumble subset sampling is used instead of FPS by Yang et al. (2019). To create super points, Landrieu and Simonovsky (2018) employs a clustering algorithm. The majority of these approaches are either too slow or necessitate structure preprocessing. VoxelNet (Le and Duan, 2018; Zhou and Tuzel, 2018), for example, blends point-based and volumetric approaches by performing voxel convolution and employing the study by Charles et al. (2017) inside each voxel. Similar concepts are used by the fast model (Zhijian et al., 2019), whereas Lu et al. (2022) made use of ball query with graph convolution layers. However, the number of points is not steadily decreased over all layers. Our DGFE module, however, utilized max-pooling in each of the pooling layers, resulting in each spatial dimension having a smaller grid allowing it to be used for high-resolution 3D objects. Apart from those features, the local region in every grid sampled a constant number of points using a simple KNN search which aids in learning approximation functions in higher order to better characterize the detailed features.

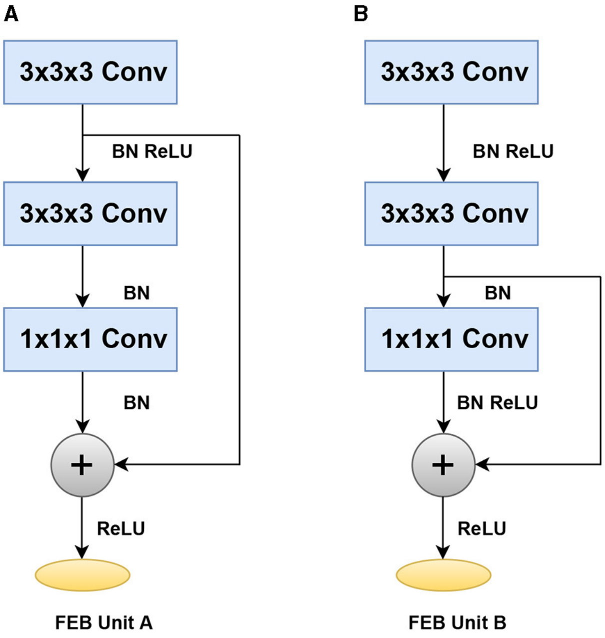


FIGURE 2
Building blocks of the feature encoding network with two different layer skips. **(A)** Feature encoding block (FEB unit A) **(B)** Feature encoding block (FEB unit B).

3. The proposed method

In this section, the KNN search for local region sampling is first introduced. Following that, we propose the feature encoding network that serves as the basis of the enhanced fused feature network. The split-transform-merge paradigm, which is based on the residual learning framework, is one of the primary building block we employ to design our feature encoding network (Figure 1A). One of the primary benefits of employing the residual network is its simplicity in training networks with many layers without raising the training error percentage. It also aids in solving the vanishing gradient problem by applying identity mapping. To compensate for structural changes in 3D objects, our feature encoding network employs two different building blocks [feature encoding block (FEB) unit A and feature encoding block (FEB) unit B], with layer skips in between. We begin with 3x3x3 convolutions twice, followed by 1x1x1 convolutions with a stride in each convolution to accommodate both small and large datasets without possible overfitting and to lower the spatial dimension of the representation. Then, we introduce the detail grid feature extraction module and finally the feature fusion unit. The complete framework is presented Figure 1.

3.1. KNN search for local region sampling

Point clouds are typically represented as raw coordinates of points in 3D space. Here, we will go over how our model extracts features from 3D objects when given a point cloud of number of points (N) as input. When provided with an input of N × 3 set of point clouds, the object is then subdivided into equal-sized 3D voxels, such as 64 × 64 × 64, 16 × 16 × 16 or 8 × 8 × 8. Using KNN, K points will be sampled from each grid cell. To avoid extra computation, those with empty points will be padded with zeros. In contrast to standard KNN, in which the search area consists of all points, it just needs to search among non-empty voxels in our situation, making the query much faster. Unlike VoxNet (Maturana and Scherer, 2015) which represents the 3D structure using an occupancy grid, we build a grid from point clouds and designate the grid's key feature to the points that are inside each grid. Some grids, on the other hand, may contain a different point number. This implies that we need a grid that will share kernels in 3D convolution. Moreover, for addressing this constraint, we utilized a sampling strategy that ensures each grid has an equal point number. In particular, if there are beyond K points in the grid, we use the KNN sampling strategy to choose K points from the total points. K points are sampled with substitution when the points inside a grid are below K. Consequently, each grid will have the same number of points, allowing us to encode the grid feature so that each grid feature has the same feature size vector which enables us to extract hierarchical features of the object using 3D convolutional kernels.

3.2. Feature encoding network

We concentrate on developing a robust network for shape classification and segmentation that achieves a notable balance of accuracy and speed. The feature encoding network is one of the key blocks that we create by making use of the split-transform-merge

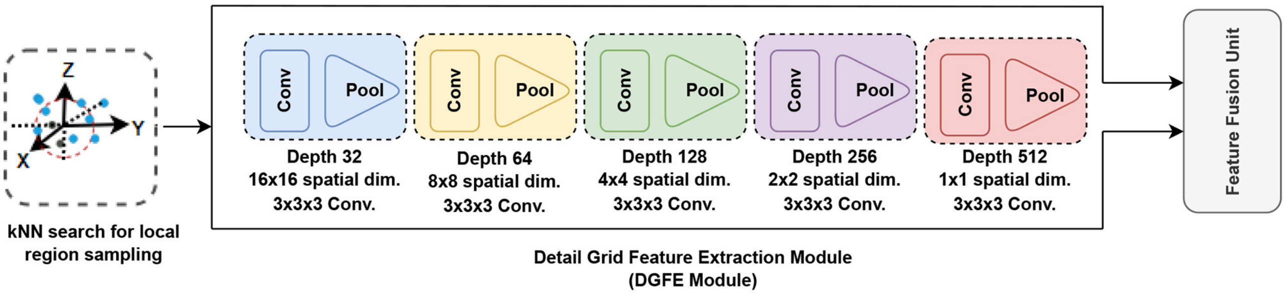


FIGURE 3
Detail grid feature extraction module (DGFE Module). This module extracts features from the input grid using many convolution blocks. Max-pooling is used in each of the pooling layers, resulting in each spatial dimension having a smaller grid and helps to manage overfitting by gradually lowering the representation's spatial dimension, the parameters in the network, and the amount of processing.

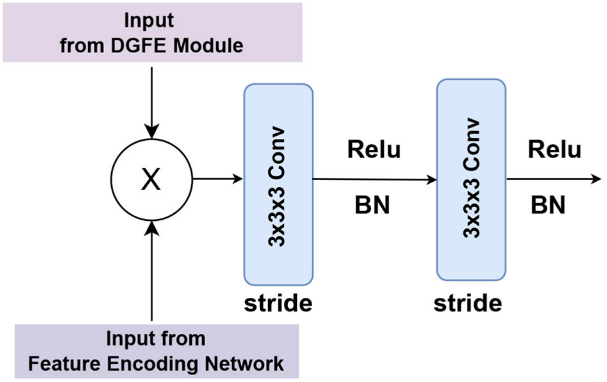


FIGURE 4
Illustration of the detailed design of the feature fusion unit, which consists of two consecutive 3x3x3 convolutions with BN and ReLU in between, as well as a stride in each convolution to help manage overfitting.

paradigm, inspired by the residual learning framework design in the study by Szegedy et al. (2015), He et al. (2016a,b), and Elhassan et al. (2021) and leveraging its powerful representational ability. These networks are scalable structures that bundle building units with the same linked shape which are referred to as residual units or blocks. The original blocks in the study by He et al. (2016b) compute as follows:

$$O_i = h(I_i) + f\left(I_i, Weights_i\right), \tag{1}$$

$$I_{i+1} = f(O_i). \tag{2}$$

In this case, I_i represents the i-th block's input feature. $Weights_i = \left\{Weights_i, k \mid 1 \leq k \leq K\right\}$ contains biases and weights connected to block i-th. K stands for total layers in a block. f signifies the block function, such as a pile of convolutional layers of two 3x3 in Equation 1. The operation following element-wise addition is represented by the function f, which is ReLU in Equation 1. The h function is designated as an identity mapping: $h(I_i) = I_i$. Similarly, if function f is identity mapping, $I_{i+1} \equiv O_i$. Putting Equation 2 into Equation 1 yields:

$$I_{i+1} = I_i + f\left(I_i, Weights_i\right). \tag{3}$$

To efficiently accelerate training and reduce the number of parameters, the feature encoding network uses two separate construction blocks, such as Feature encoding block (FEB unit A) and feature encoding block (FEB unit B), with layer skips containing batch normalization (BN) and ReLU in between. The BN and ReLU are regarded as the weight layers' pre-activation, according to He et al. (2016b). We make some minor changes here by using the ReLu with BN and Conv before the addition of operation. We start with 3x3x3 convolutions twice, followed by 1x1x1 convolutions, and then we apply the BN and ReLu before the addition. We use a stride in each convolution to help manage overfitting by gradually reducing the spatial dimension of the representation. The feature encoding network's design is shown in Figure 2.

3.3. Detail grid feature extraction module

To represent numerous hierarchical feature representations, the detail grid feature extraction module employs several convolution blocks and max-pooling and extracts features from the input grid, as shown in Figure 3. Max-pooling is used in each of the pooling layers, resulting in each spatial dimension having a smaller grid and helps to manage overfitting by gradually lowering the representation's spatial dimension, the parameters in the network, and the amount of processing. BN (Ioffe and Szegedy, 2015) can be done to any set of network activations using:

$$y = g(Hu + p) \tag{4}$$

where H and p are model parameters that have been learned, and $g(.)$ denotes a non-linearity being ReLU or sigmoid. By normalizing $z = Hu + p$, the BN transform can be introduced right before the non-linearity. Since z is normalized, $y = g(Hu + p)$ can be replaced with

$$y = g(BN(Hu)) \tag{5}$$

where the BN (Ioffe and Szegedy, 2015) is used separately for each dimension of $z = Hu$, with a distinct set of learned parameters for each dimension. We utilized a 3 × 3 × 3 kernel with stride 1 convolution and a ReLU (Nair and Hinton, 2010) in each convolution layer. The initial block employs 32-filter convolutions, which are then doubled in subsequent blocks. This module offers a regular-structured embedding volumetric grid that supports 3D convolutions in hierarchically capturing global information. To extract features of high-resolution inputs, this module is utilized in conjunction with the feature encoding network. To keep local fine details in early encoder layers, at the same spatial resolution, we connect the encoder network's encoded features to equivalent features extracted from the detail grid feature extraction module.

3.4. Feature fusion unit

The feature fusion unit is made up of two consecutive convolutional layers. We used 3 × 3 × 3 convolutions twice, with BN and ReLU in between, and a stride in each convolution to help manage overfitting. The proposed DGFE module and the encoding network outputs are fused using a cross-product in the feature fusion unit, as shown in Figure 4, to produce a feature with improved contextual representation.

3.5. Network overview

We built a 3D convolutional network with fixed points inside each grid cell, which aids in the learning of local approximation functions in high-order that better capture local shape features. Figure 1 presents a diagram of the proposed architecture. The network is made up of two major modules. A feature encoding network that serves as the foundation for extracting features from the input grid, as shown in Figure 1A in Section 3.2, and detail grid feature extraction (DGFE) module which comprises various convolution blocks accompanied with an operation of max-pooling to help in representing several relational features and pull out features from the input (Section 3.3). We hierarchically combine these two modules to form the proposed improved fused feature network. The proposed DGFE module and the encoding network outputs are fused in the feature fusion unit containing two consecutive convolutional layers (Figure 1C) to produce a feature with improved contextual representation by utilizing both local and global shape structures.

The point cloud is first normalized within the unit box. In each grid, the coordinates of the points are piled as features. accordingly, given the appropriate x, y, and z coordinates, a K-point grid has features 3K. In theory, by dividing the sum of points (P) by its grid cells, K can be approximated. To acquire classification scores, the resulting fused feature can be categorized using two fully connected layers. Finally, one additional fully connected layer is added, along with a softmax, which aids in regressing the likelihood in every group. The whole layer's nodes correspond to the set of categories of objects inside the dataset. To generate the segmentation, the segmentation network decodes the retrieved features. To create the output, this network upsamples and combines the features. For every cell inside the grid, this network produces K+1 labels, as for K points in that cell equivalent to K labels and one more label level cell. Obtaining ground truth labels of object components, we chose its greater label among the labels of points within every cell. Unoccupied Cells are tagged "no label." Before actually acquiring the object part, we perform a deconvolution operation by concatenating the feature obtained from the feature fusion unit, with the feature retrieved out of each block of the feature encoding network.

4. Experiments

In this section, a number of datasets including ModelNet10 and ModelNet40 (Wu et al., 2015) for object classification and part segmentation on ShapeNetPart (Yi et al., 2016) were used to assess the performance of the proposed network. We discuss the dataset's specifics and the evaluation metrics in Section 4.1. The implementation protocol discussion presented in Section 4.2. In Sections 4.3, 4.4, and 4.5, we discuss some experimental results from applying the proposed network to classify shapes on ModelNet, measure precision-recall on ModelNet10, and segment parts on ShapeNetPart. In Section 4.6, we demonstrate the advantages of the proposed method by conducting a good set of ablation experimental tests to evaluate various setup adjustments.

4.1. Datasets and evaluation metrics

ModelNet dataset: This is indeed a notable dataset. It comprises two datasets with CAD models in 10 and 40 categories, respectively. ModelNet10 is made up of 4,899 object instances including 2,468 training samples and 909 testing samples. ModelNet40 is made up of 12,311 object instances, 9,843 of which are in the training set and 3,991 samples in the testing set. For object classification on the ModelNet dataset, we employed accuracy as the assessment metric.

ShapeNetPart dataset: There are 16,881 shapes in this dataset, divided into 16 categories and annotated with a combined amount of 50 components. A considerable share of shape categories is partitioned into 2–5 segments. We, then, used mean intersection over union (mIoU) for evaluation. For every part shape within the object category, we calculate the union of prediction and ground truth. The mIoU was computed using Equation 6 as follows:

$$mIoU = \frac{X}{X + G - P} \tag{6}$$

where G, P, and X denote the number of ground truth points, predicted positive points, and true positive points, respectively. The mIoU is obtained by taking the average of each class's IoU.

4.2. Implementation protocol

In Python, the proposed method was implemented using the Tensorflow deep learning library. Each experiment is conducted on an Nvidia Geforce Titan GTX GPU, CUDA 10.1, and CuDNN 7.1 with RAM of 12 GB. For the classification task, we test with various parameters setup including different grid sizes and K values. Each point's location is jittered with a standard deviation of 0.02. The batch size is 32, and batch normalization is used for all layers. For both the segmentation and classification tasks, we used the cross-entropy loss to improve the discrimination of the class features. We utilized an initial learning rate of 10^{-4} and employ Adam optimizer (Kingma and Ba, 2015).

Loss function: Over the years, a wide range of loss functions have been proposed to perform 3D shape analysis tasks. For example, the cross-entropy loss was already been utilized successfully in many shape analysis tasks. Although the network can be trained using cross-entropy loss alone, we employ a combination of Shape loss (Wei et al., 2020) and modified cross-entropy loss (Huang et al., 2019) to make the class features more discriminatory. The Shape Loss is given as follows:

$$L_{shape} = L_s\left(C(S), M\right) \tag{7}$$

where M is the shapes's class label, L_s is a cross-entropy loss based on shape feature S, and C is a classifier.

Moreover, the cross-entropy loss is given as follows:

$$L_{cross-entropy} = \frac{1}{n}\sum_{y}\left(zlogQ + (1-z)log(1-Q)\right) \tag{8}$$

For each sample, $Q \in [0,1]$ is the likelihood of the network output and z represents the class ground truth. To minimize the weight of easily categorized samples, the cross-entropy function can be reshaped by inserting a hyperparameter that aids in weight balancing.

$$L_{cross-entropy} = \frac{1}{n}\sum_{y}\left[z(1-Q)^{\gamma}logQ + (1-Q)Q^{\gamma}log(1-Q)\right] \tag{9}$$

Once a sample is successfully identified, $Q \rightarrow 1$, the factor $(1-Q) \rightarrow 0$; Alternatively, when Q is small, the factor $(1-Q)$ approaches 1. Our total loss is the combination of this two losses as follows:

$$L_{total} = L_{shape} + L_{cross-entropy} \tag{10}$$

4.3. Classification on ModelNet

We use the PointNet (Charles et al., 2017) convention to prepare the data. Input points are set to 1,024 by default. Furthermore, we improve performance by incorporating more points and surface normal. To analyze various models to varying degrees of speed and accuracy, the network is trained with varying settings to balance speed and performance (Section 4.6). The variants are in different grid sizes and K values.

4.3.1. Classification on ModelNet10

Comparison: The proposed improved fused feature residual network approach was compared with a number of state-of-the-art methods, as shown in Table 1. The proposed method outperforms the majority of previous voxel-based techniques in terms of "overall accuracy" including VoxNet (Maturana and Scherer, 2015), 3DShapeNets (Wu et al., 2015), 3DGAN (Wu et al., 2016), VSL (Liu et al., 2018), and BV-CNN's (Ma et al., 2017). Although VRN (Brock et al., 2016), which combines many networks, outperforms our method in ModelNet classification, their network structure is quite complex, with each network being trained separately and taking many days to complete, making them unsuitable for large datasets. When compared with point cloud-based methods, the proposed method outperforms many of them, including Dominguez et al. (2018), OctNet (Riegler et al., 2017), ECC (Simonovsky and Komodakis, 2017), DGCB-Net (Tian et al., 2020), and VACWGAN-GP (Ergün and Sahillioglu, 2023). The DGFE module helps 3D convolutions hierarchically acquire global information, allowing the network to capture the contextual neighborhood of points. Despite using viewpoints in a predefined sequence, as opposed to any random views by DeepPano (Shi et al., 2015), Gan classifier (Varga et al., 2020), GPSP-DWRN (Long et al., 2021), OrthographicNet (Kasaei, 2019), PANORAMA-NN (Sfikas et al., 2017), and SeqViews2SeqLabels (Han et al., 2019) both of which are multi-view techniques, the method outperforms these approaches, making it suitable for high resolution input. The proposed method also outperforms PolyNet (Yavartanoo et al., 2021), a mesh-based 3D representation network that combined the features in a much smaller dimension using PolyShape's multi-resolution structure.

TABLE 1 Object classification accuracy (%) on ModelNet10.

Method	Input	Acc (%)
VoxNet (Maturana and Scherer, 2015)	Volume	92.0
3DShapeNet (Wu et al., 2015)	Volume	83.5
3DGAN (Wu et al., 2016)	Volume	91.0
VSL (Liu et al., 2018)	Volume	91.0
BV-CNNs (Ma et al., 2017)	Volume	92.3
VRN (Brock et al., 2016)	Volume	97.1
PolyNet (Yavartanoo et al., 2021)	Mesh	94.9
DeepPano (Shi et al., 2015)	Image	85.4
OrthographicNet (Kasaei, 2019)	Image	88.5
PANORAMA-NN (Sfikas et al., 2017)	Image	91.1
SeqViews2SeqLabels (Han et al., 2019)	Image	94.8
Geometry-image (Sinha et al., 2016)	Image	88.4
Gan Classifier (Varga et al., 2020)	Image	89.2
GPSP-DWRN (Long et al., 2021)	Image	92.4
G3DNet (Dominguez et al., 2018)	Point	93.1
OctNet (Riegler et al., 2017)	Point	90.4
ECC (Simonovsky and Komodakis, 2017)	Point	90.0
DGCB-Net (Tian et al., 2020)	Point	94.6
VACWGAN-GP (Ergün and Sahillioglu, 2023)	Point	91.7
(Ours)	Point	**95.6**

The bold values used to differentiate our results from the rest of the other methods.

TABLE 2 Object classification accuracy (%) on ModelNet40.

Method	Input	Acc (%)
VoxNet (Maturana and Scherer, 2015)	Volume	83.0
3DShapeNet (Wu et al., 2015)	Volume	77.0
3DGAN (Wu et al., 2016)	Volume	83.3
VSL (Liu et al., 2018)	Volume	84.5
BV-CNNs (Ma et al., 2017)	Volume	85.4
VRN (Brock et al., 2016)	Volume	95.5
NormalNet (Wang et al., 2019a)	Volume	88.6
DeepNN (Gao et al., 2022)	Mesh	91.0
PolyNet (Yavartanoo et al., 2021)	Mesh	82.8
GIFT (Bai et al., 2016)	Image	83.1
DeepPano (Shi et al., 2015)	Image	77.6
OrthographicNet (Kasaei, 2019)	Image	88.5
SeqViews2SeqLabels (Han et al., 2019)	Image	93.0
Geometry-image (Sinha et al., 2016)	Image	83.9
PointNet (Charles et al., 2017)	Point	89.2
PointConT (Liu et al., 2023)	Points	93.5
RECON (Qi et al., 2023)	Point	93.9
Pointwise (Hua et al., 2018)	Point	86.1
NPCEM (Song et al., 2020)	Point	89.4
ECC (Simonovsky and Komodakis, 2017)	Point	83.2
DGCB-Net (Tian et al., 2020)	Point	92.9
3DCTN (Lu et al., 2022)	Point	91.2
VACWGAN-GP (Ergün and Sahillioglu, 2023)	Point	81.3
(Ours)	Point	**93.1**

The bold values used to differentiate our results from the rest of the other methods.

4.3.2. Classification on ModelNet40

Comparison: We further tested the effectiveness and applicability of the proposed approach using the ModelNet40 dataset. Table 2 compares the classification accuracy of the proposed method to that of alternative scalable 3D representations techniques on the ModelNet40 datasets. As observed, the proposed method performs better than VoxNet (Maturana and Scherer, 2015), 3DGAN (Wu et al., 2016), 3DShapeNets (Wu et al., 2015), NormalNet, VACWGAN-GP (Wang et al., 2019a; Ergün and Sahillioglu, 2023), DPRNet (Arshad et al., 2019), Pointwise (Hua et al., 2018), BV-CNN's (Ma et al., 2017), NPCEM (Song et al., 2020), ECC (Simonovsky and Komodakis, 2017), PointNet (Charles et al., 2017), Geometry image (Sinha et al., 2016), VSL (Liu et al., 2018), GIFT (Bai et al., 2016), FPNN (Li et al., 2016), DGCB-Net (Tian et al., 2020), and DeepNN (Gao et al., 2022) that utilized mesh 3D data. The recent RECON (Qi et al., 2023) and PointConT (Liu et al., 2023) slightly outperformed our technique, which could be attributed to their usage of transformers and pre-train models. The improved fused feature residual network offers a significant advantage over the bulk of voxel and point cloud-based approaches, as shown in Table 2. The proposed method

TABLE 3 ModelNet40 per-class classification comparison between PointNet, Pointwise, DPRNet, and (ours).

Methods	Ours	PointNet	Pointwise	DPRNet
Avg. class	**87.4**	86.2	81.4	81.9
Airplane	100	100	100	100
Bathtub	90.0	80.0	82.0	76.0
Bed	94.0	94.0	93.0	95.0
Bench	80.0	75.0	68.4	80.0
Bookshelf	88.0	93.0	91.8	85.0
Bottle	98.0	94.0	93.9	95.0
Bowl	95.0	100	95.0	95.0
Car	99.0	97.9	95.6	91.0
Chair	97.0	96.0	96.0	97.0
Cone	100	100	80.0	90.0
Cup	90.0	70.0	60.0	70.0
Curtain	85.0	90.0	80.0	80.0
Desk	77.0	79.0	76.7	86.0
Door	92.0	95.0	75.0	85.0
Dresser	74.0	65.1	67.4	60.5
Flowerpot	44.6	30.0	10.0	25.0
Glassbox	91.0	94.0	80.8	86.0
Guiter	99.0	100	98.0	100
Keyboard	100	100	100	100
Lamp	87.0	90.0	83.3	80.0
Laptop	86.0	100	95.0	100
Mental	87.0	96.0	93.9	93.0
Monitor	71.0	95.0	92.9	96.0
Nightstand	65.0	82.6	70.2	70.9
Person	90.0	85.0	89.5	90.0
Piano	91.0	88.8	84.5	83.0
Plant	91.0	73.0	78.8	83.0
Radio	88.0	70.0	65.0	55.0
Range hood	96.0	91.0	88.9	89.9
Sink	85.0	80.0	65.0	70.0
Sofa	93.0	96.0	96.0	93.0
Stairs	90.0	85.0	80.0	75.0
Stool	90.0	90.0	83.3	70.0
Table	98.0	88.0	90.9	77.0
Tent	85.0	95.0	90.0	90.0
Toilet	98.0	99.0	94.9	95.0
TV stand	80.0	87.0	84.5	89.0
Vase	83.0	78.8	81.3	80.0
Wardrobe	65.0	60.0	30.0	20.0
Xbox	90.0	70.0	75.0	80.0

The bold values used to differentiate our results from the rest of the other methods.

performs below VRN (Brock et al., 2016), which makes usage of 24 rotating replicas for training and voting when compared with non-voxel-based approaches. Additionally, the proposed method outperformed PolyNet (Yavartanoo et al., 2021), a mesh-based 3D representation network that integrated the features in a much fewer dimension using PolyShape's multi-resolution structure. It is also worth noting that the improved fused feature residual network proposed already has a high level of accuracy, with a score of above 90%. This may be attributed to the fact that our feature encoding network together with the DGFE module, directly extracts features from the input grid and represents an organized structure of numerous feature representations.

4.3.3. ModelNet40 per-class classification accuracy comparison

Table 3 and Figure 5 compared the per-class accuracies of the proposed method to PointNet (Charles et al., 2017), Pointwise (Hua et al., 2018), and DPRNet (Arshad et al., 2019) on ModelNet40 dataset. As shown in Table 3 and Figure 5, using residual learning and extracting detail features improves per class classification accuracy. The proposed method outperforms PointNet, Pointwise, and DPRNet in key classes such as bathhub, car, bottle dresser, flowerpot, cup, and radio. In terms of average class performance, the method outperformed PointNet (1.2%), Pointwise (6%), and DPRNet (5.5%). Table 3 illustrates it.

4.4. Precision-recall on ModelNet10

Precision is a metric that assesses the accuracy of predictions, i.e., the percentage of correct predictions. It determines how many of the model's predictions were actually right. The precision was computed using Equation 11 as follows:

$$P = \frac{T_P}{T_P + F_P} \tag{11}$$

where T_P is true positive while F_P is false positive (predicted as positive but was incorrect). In the case of recall, it determines how well all of the positives are found which is given as follows:

$$R = \frac{T_P}{T_P + F_N} \tag{12}$$

where F_N is false negatives (unable to predict the presence of an object). The mAP is calculated as the average precision of all classes in the dataset while the F1-score is the harmonic mean of the precision and recall. We used these metrics to assess the efficacy and robustness of the proposed method. We used a grid size of 32 × 32 × 32 and kept the value of K at 8. As shown in Figure 6, the model can learn all 10 object class categories with high precision and recall on the ModelNet10 dataset, with 100% precision on bathtub and chair and 100% recall on bed and toilet. We can also observe that the four classes with the lowest precision and recall (desk, table, nightstand, and dresser) are highly similar which makes them difficult to distinguish even by a human expert. As shown in Figure 6, we observed that the proposed approach successfully generated results with (1) more than 90% precision on the bed, monitor, sofa, table, and toilet and more than 80% on the remaining classes, (2) 90% or higher recall of bathtub, chair, monitor, sofa, and table with more than 80% on the desk, dresser, and nightstand, and (3) 90% or higher F1-score of the bathtub, bed, chair, monitor, sofa, toilet, and table with more than 80% on the desk, dresser, and nightstand. This demonstrates that our model can learn discriminative features from 3D shapes directly across several classes.

To calculate the mAP, we perform several experiments, one of which involved using 16 × 16 × 16 voxel size combined with sampling 8 points per grid. The model was trained using ModelNet10 from scratch, which achieved a 90.2% mAP score. We, then, reduced the learning rate by half (0.5^{-5}) and retrained the model. The effect of fine-tuning improves the mAP to 90.7%. Another experiment was using a 32 × 32 × 32 grid size with the same points per grid. We train the model using the same procedure in the first experiment. We achieved 92.5% with 0.1^{-4} learning rate, and after reducing the learning rate to half and retraining the model, the result improves to 93.3%. With mAP scores of 93.3%, our model surpasses 3DShapeNets (Wu et al., 2015), PANORAMA-ENN (Sfikas et al., 2017), DeepPano (Shi et al., 2015), PolyNet (Yavartanoo et al., 2021), Multimodal (Chen et al., 2021), SeqViews2SeqLabels (Han et al., 2019), Geometry image (Sinha et al., 2016), and GIFT (Bai et al., 2016) on the ModelNet10 dataset, as shown in Table 4. Even while SeqViews2SeqLabels (Han et al., 2019) has the advantage of pre-existing 2D networks that have been pre-trained on big datasets such as ImageNet1K, we achieved a higher mean average precious mAP with 1.9% margin on ModelNet10. To further illustrate the effectiveness of the improved fused feature network, Figure 7 shows the confusion matrix. The confusion matrix was normalized to 100%. We can see that most objects from all classes are recognized correctly.

4.5. Part segmentation on ShapeNetPart

Part segmentation seems to be more difficult than classification tasks and is regarded as every-point classification. Given a triangular mesh or point cloud representation of a 3D object, the purpose of part segmentation is to give each point or triangle face a part category which makes it more challenging than object classification because of the fine-grained and dense predictions. We used the metric procedure from PointNet++ (Qi et al., 2017). For every part shape within the object category, we calculate the union of prediction and ground truth. Figure 8 shows some ShapeNetPart dataset segmentation results from our method. As observed, in most cases, the proposed method results are visually appealing.

Comparison: The segmentation performance of the proposed method is compared with that of various deep learning methods, as shown in Table 5. Although OCNN and RS-Net (Huang et al., 2018) exceed ours in terms of mIoU of all shapes, the improved fused feature residual network outperforms OCNN in specific categories, such as bag, cap, rocket, lamp, and motorbike, and achieves comparable results in the remaining categories. While OCNN has the best IoU, it also uses a conditional dense random field to rectify their network output which serve as a post-processing step, whereas our approach has no similar strategy.

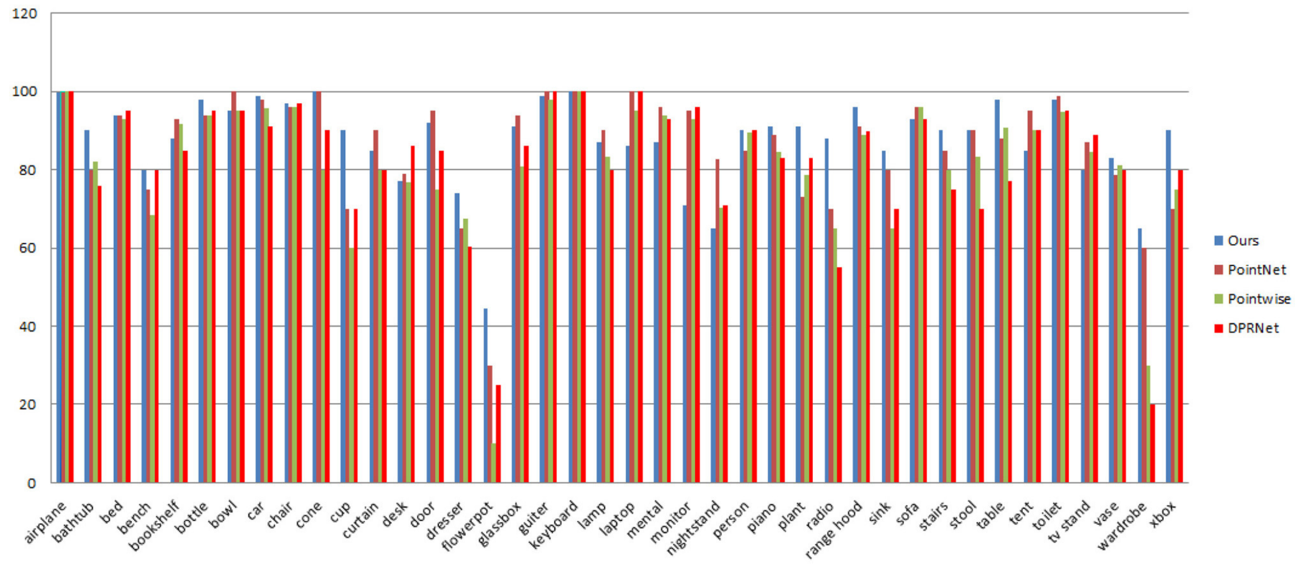


FIGURE 5
ModelNet40 per-class classification accuracy comparisons between PointNet, Pointwise, DPRNet, and (proposed).

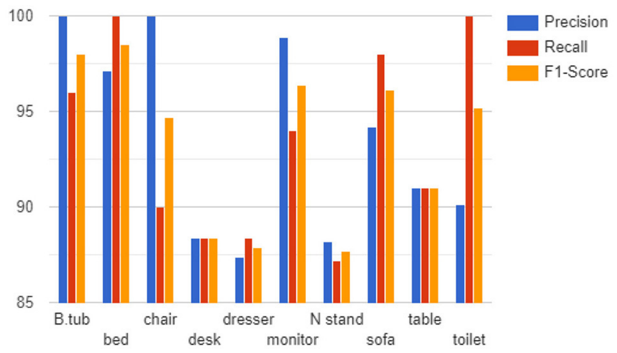


FIGURE 6
Precision, recall, and F1-score on ModelNet10.

TABLE 4 Mean average precision mAP (%) on ModelNet10.

Method	mAP (%)
3DShapeNet (Wu et al., 2015)	68.3
DeepPano (Shi et al., 2015)	84.1
PANORAMA-ENN (Sfikas et al., 2017)	93.2
SeqViews2SeqLabels (Han et al., 2019)	91.4
Geometry-image (Sinha et al., 2016)	88.4
GIFT (Bai et al., 2016)	91.1
PolyNet (Yavartanoo et al., 2021)	84.6
(Ours) (16 × 16 × 16 − *grid*)	**90.7**
(Ours) (32 × 32 × 32 − *grid*)	**93.3**

The bold values used to differentiate our results from the rest of the other methods.

4.6. Ablation experiments

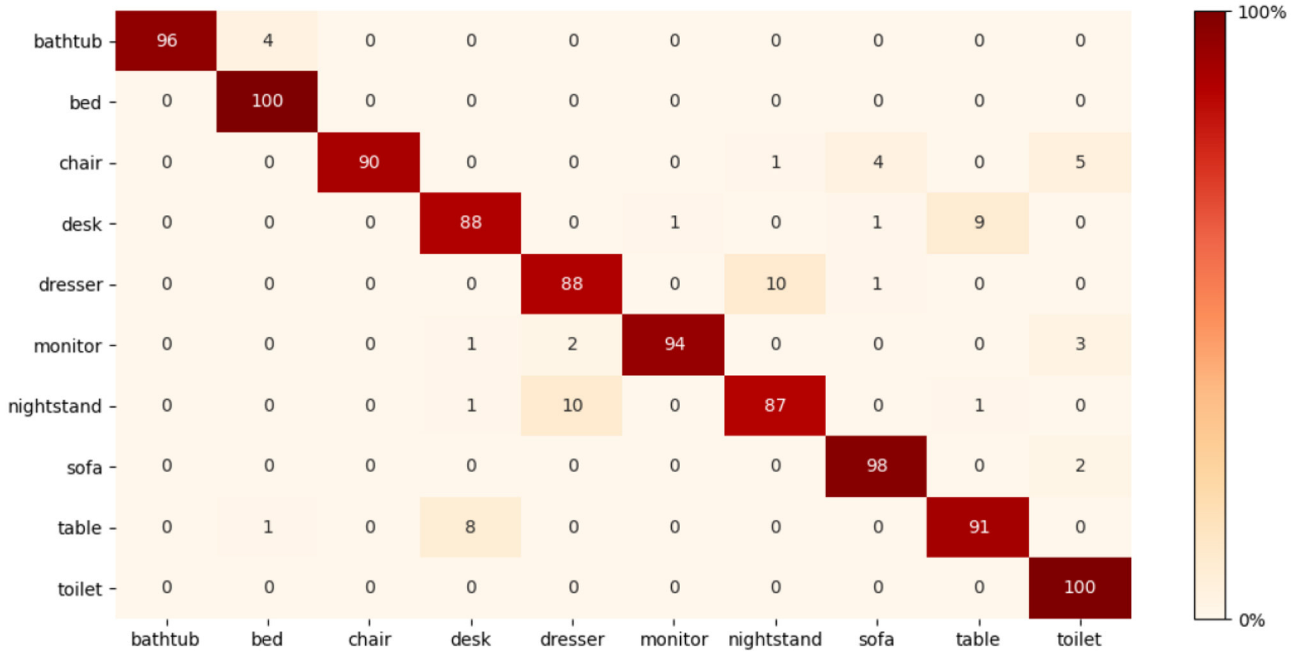
Here, we conduct some ablation experimental tests to assess various setup modifications and highlight the benefits of the improved fused feature network. The experiments were carried out using the ModelNet10 (Wu et al., 2015) dataset.

4.6.1. Effects of extracted features in the DGFE module

We present an ablation test on ModelNet10 classification to demonstrate the impact of the DGFE module's extracted features. Specifically, we experimented with many variables, including different grid sizes and K values. In the first settings, using a grid size of 16 × 16 × 16 and increasing the value of K from 2 to 8, the classification accuracy increased from 88.1% with K = 2 to 90.5% with K = 8. In the second attempt, we used a grid size of 32 × 32 × 32 and kept the values of K between 2 and 8, and the classification accuracy increased from 90.1% with K = 2 to 91.8% with K = 8. We end up using the later attempt to set the DVFE module in our approach which yields the best model result of 95.6%. Figure 9 displays the results. It shows how the proposed DGFE module encourages correlation among different point cloud regions and is useful for modeling the entire point cloud spatial distribution.

4.6.2. Effects of feature encoding network

This section analyzes the significance of the encoding branch in the proposed approach. After removing the encoding branch, the network is trained using only the DVFE module and KNN search, to sample the local region in each grid cell. We, then, repeated the tests using the same configuration as the previous ablation experiment, with a grid size of 16x16x16 and K = 2. The classification accuracy was 90.1% with K = 2 and 91.1% with K = 8. The classification accuracy improved from 91.3% with K = 2 to 92.4% with K = 8 when utilizing a grid size of 32 × 32 × 32. The results are shown in Figure 9. The model design aids in



	bathtub	bed	chair	desk	dresser	monitor	nightstand	sofa	table	toilet
bathtub	96	4	0	0	0	0	0	0	0	0
bed	0	100	0	0	0	0	0	0	0	0
chair	0	0	90	0	0	0	1	4	0	5
desk	0	0	0	88	0	1	0	1	9	0
dresser	0	0	0	0	88	0	10	1	0	0
monitor	0	0	0	1	2	94	0	0	0	3
nightstand	0	0	0	1	10	0	87	0	1	0
sofa	0	0	0	0	0	0	0	98	0	2
table	0	1	0	8	0	0	0	0	91	0
toilet	0	0	0	0	0	0	0	0	0	100

FIGURE 7
Confusion matrix on ModelNet10.

the efficient encoding of features from the input grid and DVFE module. The output features are combined to complement one another. Figure 9 demonstrates the accuracy achieved by inserting the feature encoding network into the whole network, which results in boosting the classification accuracy. The next experiments investigate the sensitivities of the feature encoding units which consist of two units (Feature Encoding Block FEB Unit A and Feature Encoding Block FEB Unit B) with layer skips containing BN and ReLU in between. In each unit, we start with 3 × 3 × 3 convolutions twice, followed by 1 × 1 × 1 convolutions. The main difference between the units is in the application of BN, a regularly used technique to speed up and stabilize the learning process of deep neural networks, and Relu, which has the advantage of allowing complicated correlations in the data to be learned. To test how resilient our approaches are to changes of this type, we swapped the units in different orders. With a 32 × 32 × 32 grid size and K = 8, we apply four possible combinations, such as ABAB, BABA, AABB, and BBAA. We train the model from the scratch. As shown in Table 6, the classification accuracy is fairly stable across the different combinations. The combination of ABAB has the highest accuracy and the lowest total log loss, with AABB coming in second. Although the two other combinations, BABA and BBAA, have lower accuracy, their overall performance is generally stable. The above result seems to indicate that, in line with He et al. (2016a), adding BN after addition forces skip connections to perturb the output, which is problematic. The main advantage of applying BN before addition here is that it speeds up training and allows a wider range of learning rates without sacrificing training convergence.

4.6.3. Time complexity

Table 7 compares the average testing time for classification and segmentation with other similar methods. TensorFlow 1.1 is used to record forward time using Nvidia Geforce Titan GTX GPU. The proposed method requires less testing time than many other methods, such as (Leng et al., 2016; Charles et al., 2017; Huang et al., 2018), DGCNN (Wang Y. et al., 2019), SpecGCN (Wang et al., 2018), and 3D-UNet (Cicek et al., 2016), because of its strong data closeness and consistency. Because zeros are padded to empty voxel, the proposed voxelization and sampling approaches both include random memory accesses, which help to decrease unnecessary computation. As observed, using the same voxel resolution of 32^3, the proposed improved fused feature residual network is faster than the 3DCNN (Leng et al., 2016) method and still outperforms it in terms of mIoU, as shown in Table 5. Another advantage of this strategy is that the same number of points is kept in each grid cell while still being able to describe neighborhood information. Now lets analyze the approach to the PointNet++ (Qi et al., 2017), set abstraction module. If we have a batch of 2,048 points with 64-channel characteristics, the technique can model the entire point cloud, but the SA module must aggressively downsample the input, resulting in information loss. The proposed method does not necessitate dynamic kernel computing, which is typically rather expensive. Even though RSNet (Huang et al., 2018) outperformed ours in terms of Mean IoU by 0.7%, the proposed improved fused feature residual network is much faster and requires less memory consumption, as shown in Table 7.

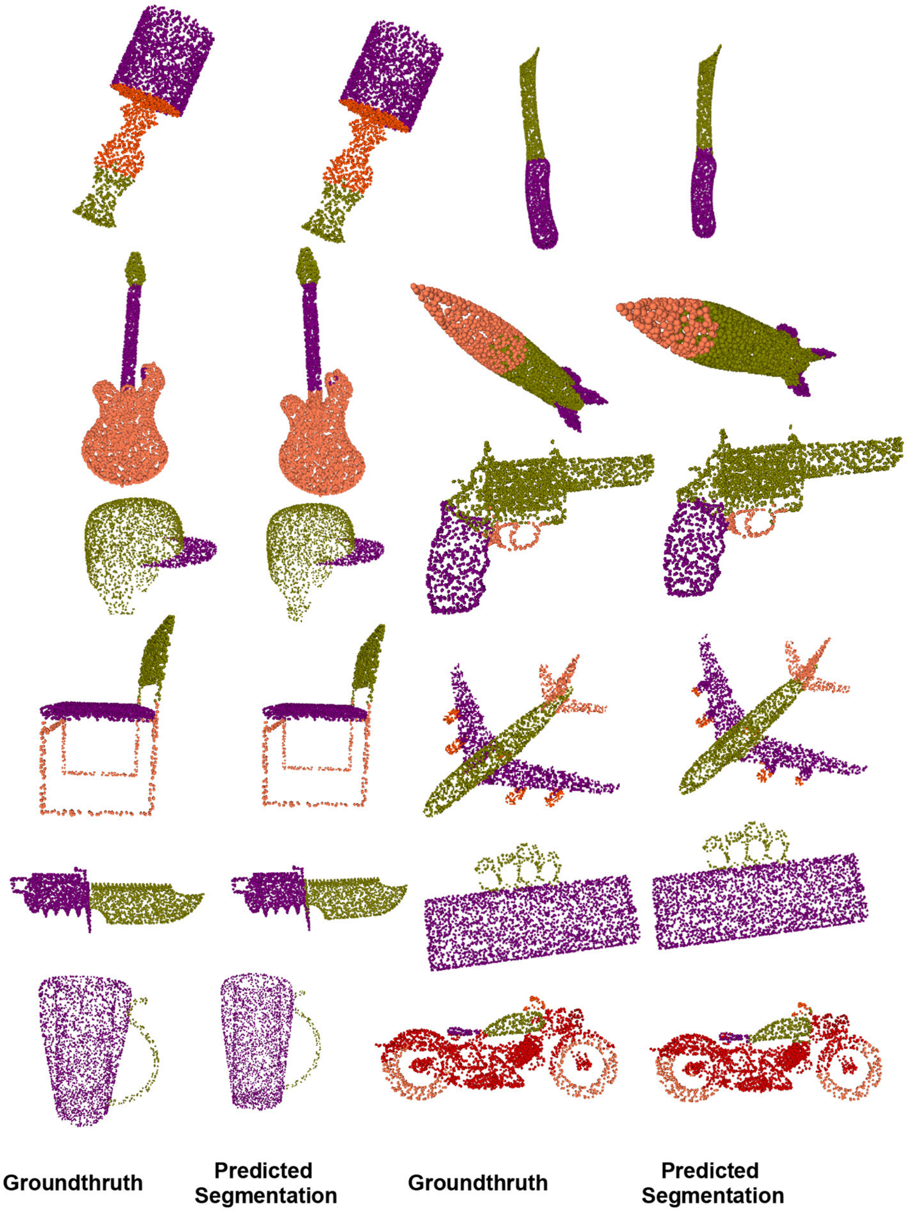


FIGURE 8
On the ShapeNet-part dataset, we compared the visual results of our object part segmentation with groundthruth.

4.6.4. Effects of neighborhood query

In this section, we experiment with ball query and sift query, two other popular neighbor querying methods to sample local areas and experiment with general search radius. For all experiments, we use a 32 × 32 × 32 grid size with a K = 8 value on the ModelNet10 dataset. Table 8 shows that KNN is more effective for our strategy. The sift query is the most inefficient method when compared with the KNN and ball query.

5. Conclusion and future work

In this study, we proposed the detail grid feature extraction (DGFE) module which is a highly efficient module. This module assists 3D convolutions in hierarchically capturing global information, reducing the grid size in each spatial dimension and managing overfitting by gradually lowering the spatial dimension of the representation, making it practical for high-resolution 3D objects. Furthermore, we design a feature encoding network that uses two different building blocks with layer skips containing batch normalization and non-linearity ReLU in between, resulting in fewer layers in the early training phase which helps speed learning and reduces the effect of gradients vanishing since there are few layers through which to propagate. The outputs of the two modules are fused in the feature fusion unit to produce a feature with improved contextual representation by utilizing both local and global shape structures. We built a network called improved fused feature

TABLE 5 Segmentation results of different methods on ShapeNet-part dataset (Yi et al., 2016).

Methods	(Ours)	P.Net	ShapeNet	KD-Net	MRTNet	3DCNN	RS-Net	O-CNN
mIoU	**84.2**	83.7	81.4	77.2	83.0	79.4	84.9	85.9
Airplane	83.8	83.4	81	79.9	81.0	75.1	82.7	85.5
Bag	88.9	78.7	78.4	71.2	76.7	72.8	86.4	87.1
Cap	91.9	82.5	77.7	80.9	87.0	73.3	84.1	84.7
Car	72	74.9	75.7	68.8	73.8	70.0	78.2	77.0
Chair	88	89.6	87.6	88.0	89.1	87.2	90.4	91.1
Earphone	47.0	73.0	61.9	72.4	67.6	63.5	69.3	85.1
Guitar	86.8	91.5	92	88.9	90.6	88.4	91.4	91.9
Knife	86.7	85.9	85.4	86.4	85.4	79.6	87.0	87.4
Lamp	89.8	80.8	82.5	79.8	80.6	74.4	83.5	83.3
Laptop	60.8	95.3	95.7	94.9	95.1	93.9	95.4	95.4
Motorbike	93.7	65.2	70.6	55.8	64.4	58.7	66.0	56.9
Mug	94.4	93.0	91.9	86.5	91.8	91.8	92.6	96.2
Pistol	80	81.2	85.9	79.3	79.7	76.4	81.8	81.6
Rocket	86.1	57.9	53.1	50.4	57.0	51.2	56.1	53.5
Skateboard	70.1	72.8	69.8	71.1	69.1	65.3	75.8	74.1
Table	74.1	80.6	75.3	80.2	80.6	77.1	82.2	84.4

The bold values used to differentiate our results from the rest of the other methods.

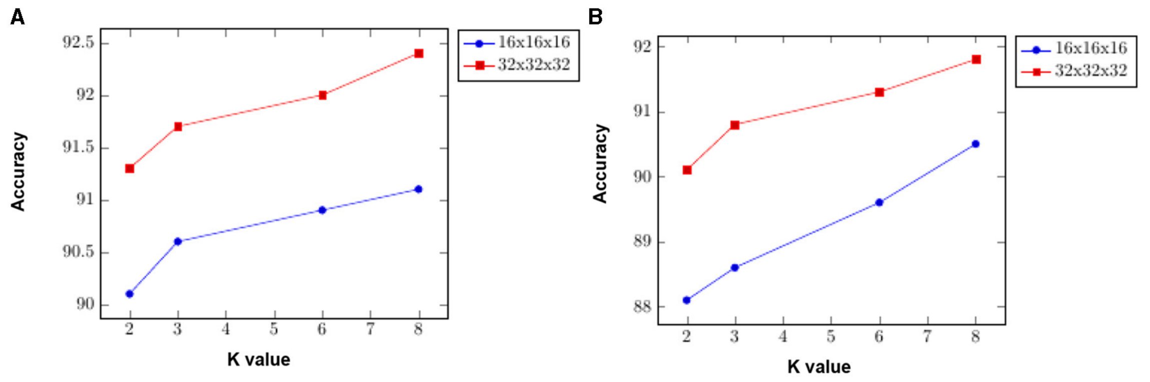


FIGURE 9
To highlight the influence of both the DGFE module and the feature encoding network, a ModelNet10 classification ablation test is presented. We experimented with some variables including different grid sizes and K values. **(A)** Shows how the feature encoding network performs with 16 × 16 × 16 and 32 × 32 × 32 grid sizes and different values of K; **(B)** demonstrates the performance of the DGFE module's effects of extracted features on 16 × 16 × 16 and 32 × 32 × 32 grid sizes with 2, 3, 6, and 8 K values.

residual network using the modules that have been proposed, which achieve a notable balance of accuracy and speed. In both ModelNet10 and ModelNet40 datasets, the proposed improved fused feature residual network offers a significant advantage over the bulk of voxel and point cloud-based approaches, as shown in Tables 1, 2. Due to its scalability and efficiency, the proposed method can be used in extracting large-scale features of high-resolution inputs.

Although our method performs well with normal datasets, we note that when noise is added to the datasets, the performance drops, for example, when Gaussian noise is added to the 3D models, the performance decreases despite applying different parameters. In future, instead of directly sampling points, we will use sparse convolutions to convert them to a small number of voxels and sample non-empty voxels to ensure that precise point positions are retained.

In addition, numerous mechanisms for attention employed in transformer approaches are adaptable and offer a high potential for future advances. We think cutting-edge outcomes can be attained by extending generic point cloud processing innovation

TABLE 6 Different combinations of feature encoding units on ModelNet10.

FEB unit	Acc (%)	Logloss
ABAB	95.6	2.22
BABA	93.8	2.38
AABB	94.54	2.25
BBAA	93.94	2.32

TABLE 7 Average testing time of our method with others on ModelNet40.

Method	Classification (ms)	Segmentation (ms)
PointNet++ (Qi et al., 2017)	163	-
3DCNN (Leng et al., 2016)	49	137
SpecGCN (Wang et al., 2018)	11254	-
DGCNN (Wang Y. et al., 2019)	52	87.8
3D-UNet (Cicek et al., 2016)	-	682.1
RSNet (Huang et al., 2018)	-	74.6
(Ours)	**28**	**19**

The bold values used to differentiate our results from the rest of the other methods.

TABLE 8 Effects of neighborhood query on ModelNet10 classification.

Sift query		Ball query		KNN
$r = 0.1$	$r = 0.2$	$r = 0.1$	$r = 0.2$	
90.8%	91.0%	92.6%	93.0%	95.6%

to transformer techniques. For instance, one possible option we are looking at is by swapping out the feature extraction module in our network design for one that is transformer/attention-based. Instead of just depending on transformers to extract features, we can conduct local feature extraction using non-transformer-based approaches and then couple it with a transformer for global feature interaction which will lead to the extraction of more fine grain features.

Data availability statement

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

Author contributions

AG: conceptualization of this study, methodology, writing—original draft preparation, and software. CL: conceptualization, software, supervision, resources, project administration, and funding acquisition. HJ, YN, and MA: data curation, writing—reviewing and editing, and software. HC: data curation, software, and supervision. All authors contributed to the article and approved the submitted version.

Funding

This study was supported by the Fujian Province University Key Lab for the Analysis and Application of Industry Big Data, Fujian Key Lab of Agriculture IOT Application, and IOT Application Engineering Research Center of Fujian Province Colleges and Universities.

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

Arshad, S., Shahzad, M., Riaz, Q., and Fraz, M. (2019). DPRNet: deep 3D point based residual network for semantic segmentation and classification of 3D point clouds. *IEEE Access* 7, 68892–68904. doi: 10.1109/ACCESS.2019.2918862

Atzmon, M., Maron, H., and Lipman, Y. (2018). Point convolutional neural networks by extension operators. *ACM Trans. Graph.* 37, 1–12. doi: 10.1145/3197517.3201301

Bai, S., Bai, X., Zhou, Z., Zhang, Z., and Latecki, L. J. (2016)."GIFT: A real-time and scalable 3D shape search engine," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, NV: IEEE), 5023–5032. doi: 10.1109/CVPR.2016.543

Bello, S. A., Wang, C., Wambugu, N. M., and Adam, J. M. (2021). FFpointNet: local and global fused feature for 3D point clouds analysis. *Neurocomputing* 461, 55–62. doi: 10.1016/j.neucom.2021.07.044

Bello, Saifullahi, A., Yu, S., Wang, C., Adam, Jibril, M., and Li, J. (2020). Review: deep learning on 3D point clouds. *Remot. Sens.* 12, 11. doi: 10.3390/rs12111729

Brock, A., Lim, T., Ritchie, J., and Weston, N. (2016). Generative and discriminative voxel modeling with convolutional neural networks. *ArXiv.* doi: 10.48550/arXiv.1608.04236

Charles, R., Su, H., Kaichun, M., and Guibas, L. (2017). "PointNet: Deep learning on point sets for 3D classification and segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI: IEEE), 77–85. doi: 10.1109/CVPR.2017.16

Chen, Z., Jing, L., Liang, Y., Tian, Y., and Li, B. (2021). Multimodal semi-supervised learning for 3D objects. *ArXiv.* doi: 10.48550/arXiv.2110.11601

Chiotellis, I., Triebel, R., Windheuser, T., and Cremers, D. (2016). "Non-rigid 3D shape retrieval via large margin nearest neighbor embedding," in *European Conference on Computer Vision (ECCV)* (Amsterdam). doi: 10.1007/978-3-319-46475-6_21

Choy, C., Danfei, X., JunYoung, G., Kevin, C., and Savarese, S. (2016). "3D-R2N2: A unified approach for single and multi-view 3D object reconstruction," in *European Conference on Computer Vision (ECCV)* (Amsterdam).

Cicek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., and Ronneberger, O. (2016). 3D U-Net: Learning dense volumetric segmentation from sparse annotation. *ArXiv*. doi: 10.48550/arXiv.1606.06650

Dominguez, M., Dhamdhere, R., Petkar, A., Jain, S., Sah, S., and Ptucha, R. (2018). "General-purpose deep point cloud feature extractor," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)* (Lake Tahoe, NV: IEEE), 1972–1981. doi: 10.1109/WACV.2018.00218

Eldar, Y., Lindenbaum, M., Porat, M., and Zeevi, Y. (1997). The farthest point strategy for progressive image sampling. *IEEE Trans. Image Process.* 9, 1305–1315. doi: 10.1109/83.623193

Elhassan, M. A., Huang, C., Yang, C., and Munea, T. L. (2021). DSANet: dilated spatial attention for real-time semantic segmentation in urban street scenes. *Expert Syst. Appl.* 183, 115090. doi: 10.1016/j.eswa.2021.115090

Ergün, O., and Sahillioglu, Y. (2023). 3D point cloud classification with ACGAN-3D and VACWGAN-GP. *Turk. J. Electr. Eng. Comput. Sci.* 31, 381–395. doi: 10.55730/1300-0632.3990

Gao, M., Ruan, N., Shi, J., and Zhou, W. (2022). Deep neural network for 3D shape classification based on mesh feature. *Sensors* 22, 187040. doi: 10.3390/s22187040

Gezawa, A. S., Bello, Z. A., Wang, Q., and Yunqi, L. (2021). A voxelized point clouds representation for object classification and segmentation on 3D data. *J. Supercomput.* 21, 1–22. doi: 10.1007/s11227-021-03899-x

Gezawa, Sulaiman, A., Zhang, Y., Wang, Q., and Lei, Y. (2020). A review on deep learning approaches for 3D data representations in retrieval and classifications. *IEEE Access* 8, 57566–57593. doi: 10.1109/ACCESS.2020.2982196

Han, Z., Shang, M., Liu, Z., Vong, C.-M., Liu, Y.-S., Zwicker, M., et al. (2019). SeqViews2SeqLabels: learning 3D global features via aggregating sequential views by RNN with attention. *IEEE Trans. Image Process.* 28, 658–672. doi: 10.1109/TIP.2018.2868426

He, K., Zhang, X., Ren, S., and Sun, J. (2016a). "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, NV: IEEE), 770–778. doi: 10.1109/CVPR.2016.90

He, K., Zhang, X., Ren, S., and Sun, J. (2016b). Identity mappings in deep residual networks. *ArXiv*. doi: 10.48550/arXiv.1603.05027

Hua, B.-S., Tran, M.-K., and Yueng, S.-K. (2018). "Pointwise convolutional neural networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT: IEEE), 984–993. doi: 10.1109/CVPR.2018.00109

Huang, F., Xu, C., Tu, X., and Li, S. (2019). Weight loss for point clouds classification. *J. Phys.* 1229, e012045. doi: 10.1088/1742-6596/1229/1/012045

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. (2017). "Densely connected convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI: IEEE), 2261–2269. doi: 10.1109/CVPR.2017.243

Huang, Q., Wang, W., and Neumann, U. (2018). "Recurrent slice networks for 3D segmentation of point clouds," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT: IEEE), 2626–2635. doi: 10.1109/CVPR.2018.00278

Ioffe, S., and Szegedy, C. (2015). Batch normalization: accelerating deep network training by reducing internal covariate shift. *ArXiv*. doi: 10.48550/arXiv.1502.03167

Kasaei, H. (2019). OrthographicNet: a deep learning approach for 3d object recognition in open-ended domains. *ArXiv*. doi: 10.48550/arXiv.1902.03057

Kingma, D. P., and Ba, J. (2015). Adam: a method for stochastic optimization. *CoRR*. doi: 10.48550/arXiv.1412.6980

Klokov, R., and Lempitsky, V. (2017). "Escape from cells: Deep Kd-networks for the recognition of 3D point cloud models," in *2017 IEEE International Conference on Computer Vision (ICCV)* (Venice: IEEE), 863–872. doi: 10.1109/ICCV.2017.99

Kohonen, T. (1998). The self-organizing map. *Neurocomputing* 21, 1–6. doi: 10.1016/S0925-2312(98)00030-7

Kuangen, Z., Ming, H., Wang, J., de Silva, C. W., and Fu, C. (2019). Linked dynamic graph CNN: learning on point cloud via linking hierarchical features. *ArXiv*. doi: 10.48550/arXiv.1904.10014

Landrieu, L., and Simonovsky, M. (2018). "Large-scale point cloud semantic segmentation with superpoint graphs," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Salt Lake City, UT: IEEE), 4558–4567. doi: 10.1109/CVPR.2018.00479

Le, T., and Duan, Y. (2018). "PointGrid: A deep network for 3D shape understanding," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT: IEEE), 9204–9214. doi: 10.1109/CVPR.2018.00959

Leng, B., Liu, Y., Yu, K., Zhang, X., and Xiong, Z. (2016). 3D object understanding with 3D convolutional neural networks. *Inf. Sci.* 366, 188–201. doi: 10.1016/j.ins.2015.08.007

Li, G., Muller, M., Qian, G., Delgadillo, I. C., Abualshour, A., Thabet, A., et al. (2023). DeepGCNs: making GCNs go as deep as CNNs. *IEEE Trans. Pattern Anal. Mach. Intell.* 45, 6923–6939. doi: 10.1109/TPAMI.2021.3074057

Li, J., Chen, B. M., and Lee, G. H. (2018). "SO-Net: Self-organizing network for point cloud analysis," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT: IEEE), 9397–9406. doi: 10.1109/CVPR.2018.00979

Li, Y., Bu, R., Sun, M., Wu, W., Di, X., and Chen, B. (2018). "PointCNN: convolution on x-transformed points," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NIPS'18)* (Red Hook, NY: Curran Associates Inc), 828–838.

Li, Y., Pirk, S., Su, H., Qi, C., and Guibas, L. (2016). FPNN: field probing neural networks for 3D data. *ArXiv*. doi: 10.48550/arXiv.1605.06240

Liu, S., Giles, L., and Ororbia, A. (2018). "Learning a hierarchical latent-variable model of 3D shapes," in *2018 International Conference on 3D Vision* (Verona), 542–551. doi: 10.1109/3DV.2018.00068

Liu, Y., Wang, B., Lv, Y., Li, L., and Wang, F. (2023). Point cloud classification using content-based transformer via clustering in feature space. *ArXiv*. doi: 10.48550/arXiv.2303.04599

Long, H., Lee, S.-H., and Kwon, K.-R. (2021). A deep learning method for 3D object classification and retrieval using the global point signature plus and deep wide residual network. *Sensors* 21, 82644. doi: 10.3390/s21082644

Lu, D., Xie, Q., Gao, K., Xu, L., and Li, J. (2022). 3DCTN: 3D convolution-transformer network for point cloud classification. *IEEE Trans. Intell. Transport. Syst.* 23, 24854–24865. doi: 10.1109/TITS.2022.3198836

Ma, C., An, W., Lei, Y., and Guo, Y. (2017). "BV-CNNS: binary volumetric convolutional networks for 3D object recognition," in *British Machine Vision Conference 2017, BMVC 2017* (London: BMVA Press).

Maturana, D., and Scherer, S. (2015). "VoxNet: A 3D Convolutional Neural Network for real-time object recognition," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Hamburg: IEEE), 922–928. doi: 10.1109/IROS.2015.7353481

Nair, V., and Hinton, G. E. (2010). "Rectified linear units improve restricted Boltzmann machines," in *Proceedings of the 27th International Conference on International Conference on Machine Learning* (Madison, WI: Omnipress), 807–814.

Qi, C., Yi, L., Hao, S., and Guibas, L. (2017). "Pointnet^{++}: deep hierarchical feature learning on point sets in a metric space," in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)* (Red Hook, NY: Curran Associates Inc), 5105–5114.

Qi, Z., Dong, R., Fan, G., Ge, Z., Zhang, X., Ma, K., et al. (2023). Contrast with reconstruct: contrastive 3D representation learning guided by generative pretraining. *ArXiv*. doi: 10.48550/arXiv.2302.02318

Qiangeng, X., Weiyue, W., Duygu, C., Mech, R., and Neumann, U. (2019). "DISN: deep implicit surface network for high-quality single-view 3D reconstruction," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems* (Red Hook, NY: Curran Associates Inc), 492–502.

Riegler, G., Ulusoy, A. O., and Geiger, A. (2017). "OctNet: Learning deep 3D representations at high resolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI: IEEE), 6620–6629. doi: 10.1109/CVPR.2017.701

Sfikas, K., Theoharis, T., Pratikakis, I. (2017). "Exploiting the PANORAMA representation for convolutional neural network classification and retrieval," in *Proceedings of the Workshop on 3D Object Retrieval (3Dor '17)* (Goslar: Eurographics Association), 1-7. doi: 10.2312/3dor.20171045

Shi, B., Bai, S., Zhou, Z., and Bai, X. (2015). DeepPano: deep panoramic representation for 3-D shape recognition. *IEEE Sign. Process. Lett.* 22, 2339–2343. doi: 10.1109/LSP.2015.2480802

Simonovsky, M., and Komodakis, N. (2017). "Dynamic edge-conditioned filters in convolutional neural networks on graphs," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI: IEEE), 29–38. doi: 10.1109/CVPR.2017.11

Sinha, A., Bai, J., and Ramani, K. (2016). "Deep learning 3D shape surfaces using geometry images," in *European Conference on Computer Vision (ECCV)* (Amsterdam).

Song, Y., Gao, L., Li, X., and Shen, W. (2020). A novel point cloud encoding method based on local information for 3D classification and segmentation. *Sensors* 20, 92501. doi: 10.3390/s20092501

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2015). "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Boston, MA: IEEE), 1–9. doi: 10.1109/CVPR.2015.7298594

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., and Guibas, L. (2019). "KPConv: Flexible and deformable convolution for point clouds," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (Seoul: IEEE), 6410–6419. doi: 10.1109/ICCV.2019.00651

Tian, Y., Chen, L., Song, W., Sung, Y., and Woo, S. (2020). DGCB-Net: dynamic graph convolutional broad network for 3D object recognition in point cloud. *Remote. Sens.* 13, 66. doi: 10.3390/rs13010066

Varga, M., Jadlovský, J., and Jadlovska, S. (2020). Generative enhancement of 3D image classifiers. *Appl. Sci.* 2020, 10217433. doi: 10.3390/app1021 7433

Wang, C., Cheng, M., Sohel, F., Bennamoun, M., and Li, J. (2019a). NormalNet: a voxel-based CNN for 3D object classification and retrieval. *Neurocomputing* 323, 139–147. doi: 10.1016/j.neucom.2018.09.075

Wang, C., Samari, B., and Siddiqi, K. (2018). "Local spectral graph convolution for point set feature learning," in *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part IV* (Berlin; Heidelberg: Springer-Verlag), 56–71. doi: 10.1007/978-3-030-01225-0_4

Wang, D. Z., and Posner, I. (2015). "Voting for voting in online point cloud object detection," in *Robotics: Science and Systems* (Rome), 10–15607.

Wang, H., Shi, C., Shi, S., Lei, M., Wang, S., He, D., et al. (2023). DSVT: dynamic sparse voxel transformer with rotated sets. *ArXiv.* doi: 10.48550/arXiv.2301.06051

Wang, L., Huang, Y., Hou, Y., Zhang, S., and Shan, J. (2019). "Graph attention convolution for point cloud semantic segmentation," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Long Beach, CA: IEEE), 10288–10297. doi: 10.1109/CVPR.2019.01054

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., and Solomon, J. M. (2019). Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph.* 38, 1–12. doi: 10.1145/3326362

Wei, X., Yu, R., and Sun, J. (2020). "View-GCN: View-based graph convolutional network for 3D shape analysis," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Seattle, WA: IEEE), 1847–1856. doi: 10.1109/CVPR42600.2020.00192

Wu, J., Zhang, C., Xue, T., Freeman, B., and Tenenbaum, J. (2016). "Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling," in *Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16)* (Red Hook, NY: Curran Associates Inc), 82–90.

Wu, W., Qi, Z., and Fuxin, L. (2019). "PointConv: Deep convolutional networks on 3D point clouds," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Long Beach, CA: IEEE), 9613–9622. doi: 10.1109/CVPR.2019.00985

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., et al. (2015). "3D ShapeNets: A deep representation for volumetric shapes," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Boston, MA: IEEE), 1912–1920. doi: 10.1109/CVPR.2015.7298801

Yang, H., Wang, W., Chen, M., Lin, B., He, T., Chen, H., et al. (2023). PVT-SSD: single-stage 3d object detector with point-voxel transformer. *ArXiv.* doi: 10.48550/arXiv.2305.06621

Yang, J., Zhang, Q., Ni, B., Li, L., Liu, J., Zhou, M., et al. (2019). "Modeling point clouds with self-attention and gumbel subset sampling," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Long Beach, CA: IEEE), 3318–3327, doi: 10.1109/CVPR.2019.00344

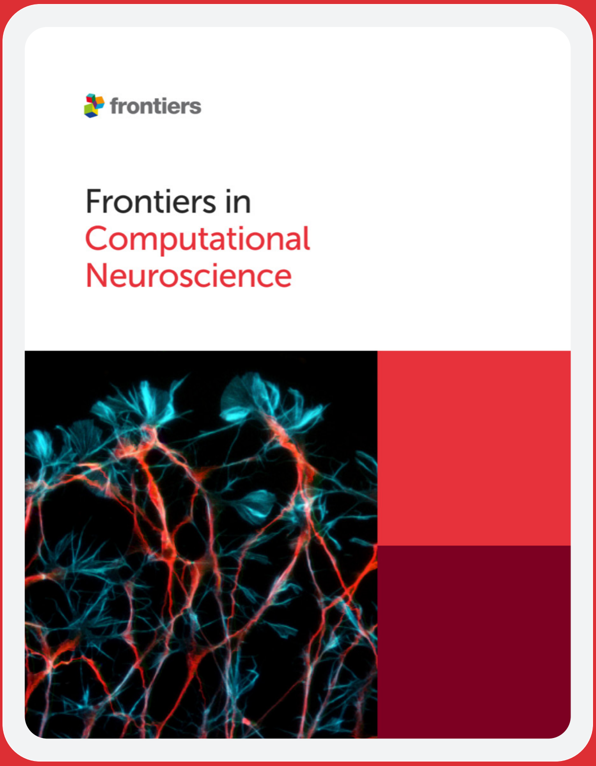
Yavartanoo, M., Hung, S.-H., Neshatavar, R., Zhang, Y., and Lee, K. M. (2021). "PolyNet: Polynomial neural network for 3D shape recognition with polyshape representation," in *2021 International Conference on 3D Vision (3DV)* (London), 1014–1023, doi: 10.1109/3DV53792.2021.00109

Yi, L., Kim, V. G., Ceylan, D., Shen, I.-C., Yan, M., Su, H., et al. (2016). A scalable active framework for region annotation in 3D shape collections. *ACM Trans. Graph.* 35, 1–12. doi: 10.1145/2980179.2980238

Yifan, X., Tianqi, F., Mingye, X., Long, Z., and Qiao, Y. (2018). "SpiderCNN: deep learning on point sets with parameterized convolutional filters," in *European Conference on Computer Vision (ECCV)* (Munich).

Zhijian, L., Haotian, T., Yujun, L., and Song, H. (2019). "Point-voxel CNN for efficient 3D deep learning," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems* (Red Hook, NY: Curran Associates Inc), 965–975.

Zhou, Y., and Tuzel, O. (2018). "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Salt Lake City, UT: IEEE), 4490–4499. doi: 10.1109/CVPR.2018.00472



frontiers

Frontiers in
Computational
Neuroscience

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact