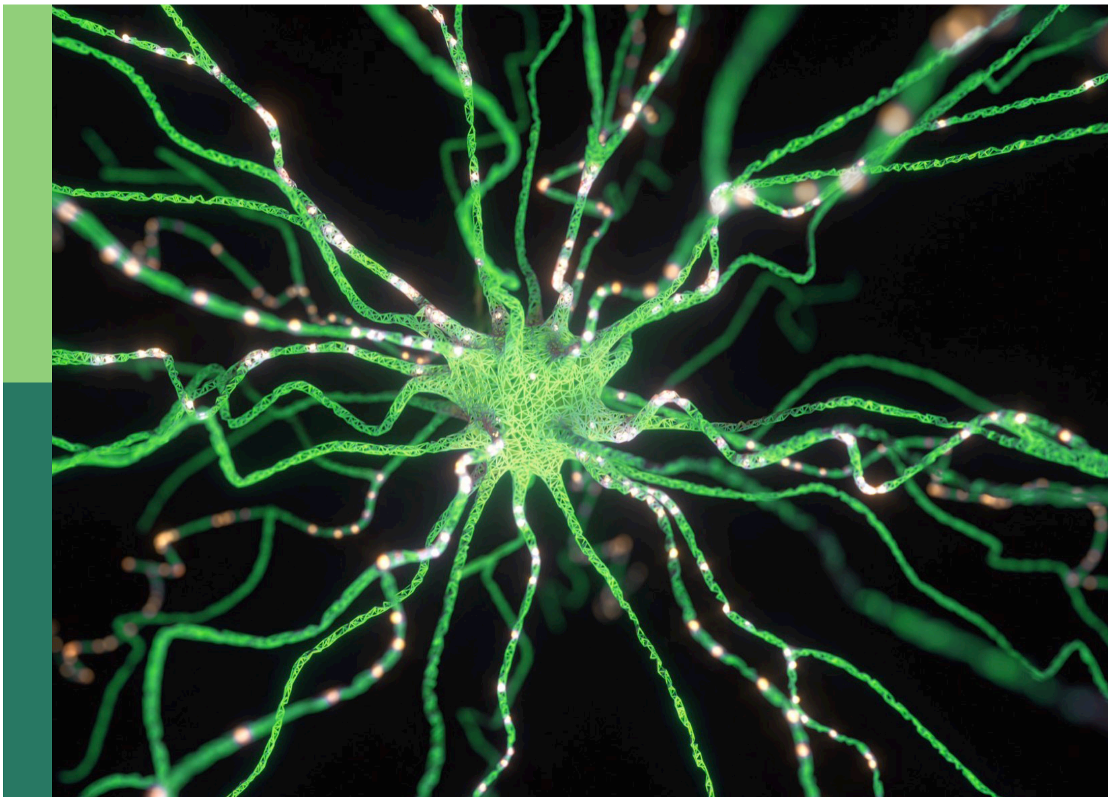
frontiers | Research Topics

Recent advances in image fusion and quality improvement for cyber-physical systems

Edited by
Xin Jin, Jingyu Hou, Zhou Wei and Shin-Jye Lee

Published in
Frontiers in Neurorobotics



ISSN 1664-8714
ISBN 978-2-8325-2459-6
DOI 10.3389/978-2-8325-2459-6

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

Recent advances in image fusion and quality improvement for cyber-physical systems

Topic editors

Xin Jin — Yunnan University, China

Jingyu Hou — Deakin University, Australia

Zhou Wei — Yunnan University, China

Shin-Jye Lee — National Chiao Tung University, Taiwan

Citation

Jin, X., Hou, J., Wei, Z., Lee, S.-J., eds. (2023). *Recent advances in image fusion and quality improvement for cyber-physical systems*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-8325-2459-6

Table of contents

frontiers | Frontiers in Neurorobotics

TYPE Editorial
PUBLISHED 04 May 2023
DOI 10.3389/fnbot.2023.1201266

Check for updates

OPEN ACCESS

EDITED AND REVIEWED BY
Florian Röhrbein,
Technische Universität Chemnitz, Germany

*CORRESPONDENCE
Xin Jin
✉ xinxin_jin@163.com

RECEIVED 06 April 2023
ACCEPTED 14 April 2023
PUBLISHED 04 May 2023

CITATION
Jin X, Hou J, Zhou W and Lee S-J (2023) Editorial: Recent advances in image fusion and quality improvement for cyber-physical systems. *Front. Neurorobot.* 17:1201266. doi: 10.3389/fnbot.2023.1201266

Editorial: Recent advances in image fusion and quality improvement for cyber-physical systems

Xin Jin[1]*, Jingyu Hou[2], Wei Zhou[1] and Shin-Jye Lee[3]

[1]School of Software, Yunnan University, Kunming, China, [2]School of Information Technology, Deakin University, Geelong, VIC, Australia, [3]Institute of Technology Management, National Chiao Tung University, Hsinchu, Taiwan

KEYWORDS

artificial neural networks, embedded learning system, feature extraction, image quality improvement, image fusion, robot vision

Editorial on the Research Topic
Recent advances in image fusion and quality improvement for cyber-physical systems

Multi-source visual information fusion and quality improvement can help the robotic system to perceive the real world, and image fusion is a computational technique fusing the multi-source images from multiple sensors into a synthesized image that provides either comprehensive or reliable description, and quality improvement technique can be used to address the challenge of low-quality image analysis task (Jin et al., 2017, 2021, 2023; Chen et al., 2021; Wang et al., 2022; Jiang et al., 2023). At present, a lot of brain-inspired algorithms methods (or models) are aggressively proposed to accomplish these two tasks, and the artificial neural network has become one of the most popular techniques in processing image fusion and quality improvement techniques in this decade, especially deep convolutional neural networks (Chen et al., 2021; Jin et al., 2021, 2023). This is an exciting research field for the research community of image fusion and there are many interesting issues remain to be explored, such as deep few-shot learning, unsupervised learning, application of embodied neural systems, and industrial applications.

How to develop a sound biological neural network and embedded system to extract the multiple features of source images are basically two key questions that need to be addressed in the fields of image fusion and quality improvement. Hence, studies in this field can be divided into two aspects: first, new end-to-end neural network models for merging constituent parts during the image fusion process; Second, the embodiment of artificial neural networks for image processing systems. In addition, current booming techniques, including deep neural systems and embodied artificial intelligence systems, are considered as potential future trends for reinforcing the performance of image fusion and quality improvement.

In the first work entitled "*Multi-focus image fusion dataset and algorithm test in real environment*," Liu S et al. proposed a multi-focus image fusion dataset named HBU-CVMDSP. The dataset can truly reflect the real-world scene, which included 66 groups of images captured by smartphones. Five image fusion algorithms were performed on the HBU-CVMDSP dataset, which revealed that the HBU-CVMDSP dataset could better promote the research of multi-focus image fusion.

Due to insufficient view refinement feature extraction and poor generalization ability of the network model affecting the classification accuracy, Wang et al. proposed a multi-view SoftPool attention convolutional network for 3D model classification tasks. The multi-view features were extracted through ResNest and adaptive pooling modules, and then processed by SoftPool, which enabled the subsequent refinement extraction. The experimental results showed that the proposed model is effective.

In the third paper, Kong et al. proposed the model of convolutional extreme learning machine (CELM) for the fusion of multimodal medical images. In this method, CELM served as an important tool to extract and capture the features of source images from a variety of different angles, and the final fused image can be obtained by integrating the significant features. Experiments showed that the proposed method has obvious superiorities in gray image fusion and color image fusion.

The visual quality of images will be seriously affected by bad weather conditions, especially on foggy days. Yang et al. proposed a new transformer-based progressive residual network (PRnet) to achieve the quality improvement and obtain a fog-free image. In this work, the swin transformer block encoded the feature representation of the decomposed block and continuously reduced the feature mapping resolution. The decoder was used to recursively select and fuse image features. Experiments showed that the performance of the proposed method was better than other state-of-the-art methods.

Zhang C et al. proposed a lightweight multi-dimensional dynamic convolutional network (LMDCNet) for real-time semantic segmentation with an ideal trade-off between model parameters, segmentation accuracy and inference speed. In this work, the encoder was a depth-wise asymmetric bottleneck module with multi-dimensional dynamic convolution and shuffling operations (MS-DAB), which increased the utilization of local and contextual information of features. Finally, a feature pyramid module (SC-FP) based on spatial and channel attention can perform the multi-scale fusion of features accompanied by feature selection.

Ye et al. proposed a dual branch CNN network (BD-CNN) for the fusion and classification of multi-source remote sensing data. Comparing with ELM algorithm and SVM algorithm, the proposed BD-CNN model can effectively fuse and classify multi-source remote sensing data.

Electricity transmission line monitoring in hazy weather will face some problems, such as reduced contrast and chromatic aberration. Therefore, Zhang M et al. proposed an image defogging algorithm for the electricity transmission line monitoring system. In this research, an optimized quadtree segmentation method for calculating global atmospheric light was proposed. Moreover, the detail sharpening post-processing based on visibility and air light level was introduced to enhance the detail level of electricity transmission lines in the defogging image. Experiments proved that the algorithm performs well in improving image quality.

Chen et al. proposed an improved multi-exposure fusion method based on the exposure fusion framework and the color dissimilarity feature to solve the problem of ghosting artifacts. First, an improved exposure fusion framework based on the camera response model was applied to preprocess the input image sequence. Then, an improved color dissimilarity feature was used to detect the object motion features in dynamic scenes. Finally, the improved pyramid model was adopted to retain detailed information about the poor exposure areas.

To preserve more local details and with few artifacts in panoramas, Tang et al. presented an improved mesh-based joint optimization image stitching model. An improved energy function containing a color similarity term and a regularization parameter strategy of combining the proposed method with an as-projective-as-possible (APAP) warp was performed. Moreover, calculating the distance between the vertex and the nearest matched feature point to the vertex ensured a more natural stitching effect in non-overlapping areas.

The 1D convolution is not limited by the input size and has the advantage of fewer parameters. Thus, Zhang C et al. designed a lightweight semantic segmentation network (LSNet) composed of full 1D convolution. Moreover, increasing the depth of network in the decoder can effectively solve the misalignment of upsampling and improve the accuracy of network segmentation. Experiments demonstrated that the proposed method can achieved better performance in accuracy and parameters.

As most attack methods rely on a relatively loose noise budget in image, Liu R et al. proposed a novel framework named Dual-Flow for generating adversarial examples by disturbing the latent representation of the clean examples. The spatial transform techniques were applied to the latent value to preserve the details of original images and guarantee the adversarial images' quality. Experiments revealed the superiority of the proposed method in synthesizing adversarial examples.

Mi et al. proposed a deep learning algorithm based on the modified YOLOv4 network to improve the accuracy of railway defects detection. In this mehod, the rail region extraction, improved Retinex image enhancement, background modeling difference and threshold segmentation were performed sequentially to obtain the segmentation map of defects. For the classification of defects, Res2Net and Convolutional Block Attention Module (CBAM) were introduced to improve the receptive field and small target position weights.

Shi et al. proposed an evaluation system based on image quality indexes, resource occupancy and energy consumption metrics, which verified the performances of different near-infrared image colorization methods on low-power NVIDIA Jetson embedded systems. Eleven infrared image colorization methods were tested on three different configurations of NVIDIA Jetson boards. The experimental results indicated that the CICZ had the smallest energy consumption per unit of time. Pix2Pix and TIC-CGAN showed superiority in image quality and latency metrics. Moreover, the RecycleGAN, PearlGAN and I2V-GAN had smaller memory usage than other methods on edge devices.

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 National Natural Science Foundation of China (Nos. 62101481, 62002313, and 62261060), Basic Research Project of Yunnan Province (Nos. 202201AU070033 and 202201AT070112), and Key Laboratory in Software Engineering of Yunnan Province (No. 2020SE408).

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

Chen, L., Li, X., Luo, L., and Ma, J. (2021). Multi-focus image fusion based on multi-scale gradients and image matting. *IEEE Trans. Multimedia* 24, 655–667. doi: 10.1109/TMM.2021.3057493

Jiang, Q., Jin, X., Cui, X., Yao, S., Li, K., Zhou, W., et al. (2023). A lightweight multimode medical image fusion method using similarity measure between intuitionistic fuzzy sets joint laplacian pyramid. *IEEE Trans. Emerg. Topics Comput. Intellig.* 1–17. doi: 10.1109/TETCI.2022.3231657

Jin, X., Huang, S., Jiang, Q., Li, S.-J., Wu, L., Yao, S., et al. (2021). Semi-supervised remote sensing image fusion using multi-scale conditional generative adversarial network with siamese structure. *IEEE J. Selected Topics Appl. Earth Observ. Remote Sens.* 14, 7066–7084. doi: 10.1109/JSTARS.2021.3090958

Jin, X., Jiang, Q., Yao, S., Zhou, D., Nie, R., Hai, J., et al. (2017). A survey of infrared and visual image fusion methods. *Infrared Phys. Technol.* 85, 478–501. doi: 10.1016/j.infrared.2017.07.010

Jin, X., Xi, X., Zhou, D., Ren, X., Yang, J., and Jiang, Q. (2023). An unsupervised multi-focus image fusion method based on transformer and U-Net. *IET Image Process.* 17, 733–746. doi: 10.1049/ipr2.12668

Wang, G., Li, W., Du, J., Xiao, B., and Gao, X. (2022). Medical image fusion and denoising algorithm based on a decomposition model of hybrid variation-sparse representation. *IEEE J. Biomed. Health Inform.* 26, 5584–5595. doi: 10.1109/JBHI.2022.3196710

TYPE Data Report
PUBLISHED 18 October 2022
DOI 10.3389/fnbot.2022.1024742

Multi-focus image fusion dataset and algorithm test in real environment

Shuaiqi Liu[1,2,3], Weijian Peng[1,2], Wenjing Jiang[1,2], Yang Yang[1,2], Jie Zhao[1,2] and Yonggang Su[1,2]*

[1]College of Electronic and Information Engineering, Hebei University, Baoding, China, [2]Machine Vision Technological Innovation Center of Hebei, Baoding, China, [3]National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences, Beijing, China

Check for updates

OPEN ACCESS

EDITED BY
Xin Jin,
Yunnan University, China

REVIEWED BY
JInjiang Li,
Shandong Institute of Business and Technology, China
Zhiqin Zhu,
Chongqing University of Posts and Telecommunications, China

*CORRESPONDENCE
Yonggang Su
ygsu0726@163.com

RECEIVED 22 August 2022
ACCEPTED 09 September 2022
PUBLISHED 18 October 2022

CITATION
Liu S, Peng W, Jiang W, Yang Y, Zhao J and Su Y (2022) Multi-focus image fusion dataset and algorithm test in real environment.
Front. Neurorobot. 16:1024742.
doi: 10.3389/fnbot.2022.1024742

KEYWORDS

image fusion, multi-focus image fusion dataset, image preprocessing, multi-focus image fusion algorithm test, real environment

Introduction

In the past three decades, not only some classical MIF datasets have appeared, but also MIF technology has developed rapidly (Zheng et al., 2020; Zhu et al., 2021). The existing MIF datasets can be divided into two categories, namely, the simulated image dataset obtained by applying Gaussian blur to the existing image dataset and the benchmark image dataset captured by the professional camera. The source image after Gaussian blurring in the multi-focus simulated image dataset are difficult to reflect the information of focused and unfocused objects in the real environment. The benchmark image dataset also has imaging equipment limited to professional cameras. Both of them are difficult to achieve the application of MIF technology in the real environment.

MIF algorithms can be classified into three categories i.e., spatial domain fusion algorithms, transform domain fusion algorithms, and fusion algorithms based on deep learning (Liu et al., 2021). The spatial domain fusion algorithms mainly take pixel-level gradient information or image blocks for fusion. Bouzos et al. (2019) presented a MIF algorithm based on conditional random field optimization. Xiao et al. (2020) presented a MIF algorithm based on Hessian matrix. The transform domain fusion algorithms consist of three processes: image transformation, coefficient fusion and inverse transformation. Liu et al. (2019) proposed a MIF algorithm based on an adaptive dual-channel impulse cortical model and differential images in non-subsampled Shearlet transform (NSST) domain. In recent years, the fusion algorithms based on deep learning have become a research hotspot in the field of multi-focused image fusion. Zhang et al. (2020) proposed an image fusion framework based on convolutional neural network, which utilizes two convolutional layers to extract salient features from source images. Liu et al. (2022) proposed a MIF algorithm based on low vision image reconstruction and focus feature extraction. Although these MIF algorithms have achieved good image fusion results among these public datasets, the image fusion databases used by these algorithms are all data taken by professional cameras or synthetic data, which cannot reflect the fusion performance of the fusion algorithm in the real environment.

As mentioned above, in the past few years, a series of MIF algorithms have been developed by scholars from various countries. To test the performance of these algorithms, some classic public MIF datasets have occurred. Currently, the commonly used datasets include Multi Focus-Photography Contest dataset (http://www.pxleyes.com/photography-contest/19726), Lytro color multi-focus image dataset (Nejati et al., 2015), Savic dataset (http://dsp.etfbl.net/mif/) and Aymaz dataset (https://github.com/sametymaz/Multi-focus-Image-Fusion-Dataset), etc. Some of these datasets were captured by professional cameras, and others were obtained by applying Gaussian blur to existing image datasets. The Multi Focus-Photography Contest dataset is an image photography competition held by the Photography Contest website. It contains 27 pairs of multi-focus images. Images in Lytro multi-focus dataset were acquired by the Lytro camera which is an all-optical camera whose imaging system employs a microlens array focused on the focal plane of the camera's main lens. The Lytro multi-focus dataset includes 20 groups of color multi-focus images and four sets of multi-source focus images. The image resolution is and the image format is jpg. The Savic dataset is collected by Nikon D5000 camera and contains 27 pairs of images. In Savic dataset, 21 pairs of images with format jpg are taken indoors, and 6 pairs of images with format bmp are used for MIF algorithm testing. In Aymaz dataset, the 150 multi-focus images are obtained by using the Gaussian blur function to locally blur some common image datasets. This dataset also contains some multiple source images of the same scene with different focal points. In addition to color multi-focus datasets, there are also some grayscale multi-focus datasets, and some images in grayscale multi-focus datasets.

The above-mentioned datasets can well reflect the performance of the fusion algorithms to some extent. However, these datasets can hardly reflect the application of MIF techniques in real environment. At present, the most commonly used camera device in daily life is the smartphone. With the continuous development of the imaging technology, the smartphone photography is more and more recognized by people. Therefore, it is necessary to try to construct a real-environment dataset by using different smartphones. In order to better build the database and collect images of the real environment more widely, we selected five mobile phones that were among the top ten in sales nationwide at that time for data collection such as HUAWEI Mate 30, OPPO Reno Z, Honor30 Pro+, Honor V30 Pro and iPhone XR to collect the multi-focus images in HBU-CVMDSP dataset. There are some unavoidable problems in collecting images with mobile phones, such as jitter, not completely overlapped and brightness. To address these issues, the proposed dataset is pre-processed after acquisition with image cropping, standardization of basic image attributes and image alignment. The contributions of this paper are as follows: In this paper, we construct a real-environment dataset named as HBU-CVMDSP, which includes 66 groups of multi-focus images. we give the detail of how to pre-process the raw data of the real-environment dataset, and the experiments prove that it is effectively for testing the fusion algorithms. We also test the performance of some existing image fusion algorithms on the HBU-CVMDSP dataset.

TABLE 1 Acquisition equipment.

Smartphone model	Camera description
HUAWEI Mate 30	Rear triple camera layout: 40-megapixel (MP) camera, 16 MP super-wide-angle camera and 8 MP telephoto camera
OPPO Reno Z	Rear dual-camera layout: 48 MP camera and 5 MP depth-of-field lens
Honor 30 Pro+	Rear three-camera layout: 50 MP super-sensitive camera, 16 MP super-wide-angle camera and 8 MP telephoto camera
Honor V30 Pro	Rear triple camera layout: 40 MP main camera, 12 MP super-wide-angle camera and 8 MP telephoto camera
iPhone XR	Rear single-camera layout: 12 MP wide-angle camera

Collection and construction of the dataset

Due to the variability of image effects from different smartphones, five different models of smartphones shown in Table 1 are used for image collection in this paper.

In this paper, the constructed real-environment multi-focus image dataset is named as HBU-CVMDSP. There are two kinds of sceneries i.e., natural scenery and artificial scenery in HBU-CVMDSP dataset, and these sceneries are selected from the laboratory, campus, gymnasium, and shopping mall, respectively. The HBU-CVMDSP dataset contains 66 groups of multi-focus images with jpg format. The image size is uniformly cropped to 512 × 512 to ensure the efficient execution of the experiment. Figure 1 shows some images in HBU-CVMDSP dataset.

Image preprocessing

In order to solve these unavoidable problems when capturing images with mobile phones, the proposed dataset is preprocessed by image clipping, standardization of basic image attributes and image registration after acquisition, such as Figure 2.

To further illustrate the necessity of image preprocessing, before the dataset is preprocessed, we use dense scale-invariant feature transform (DSIFT) (Liu et al., 2015) and CNN (Liu et al., 2017) based image fusion algorithms to examine the dataset. The partial fusion results of the DSIFT and CNN can be found in https://www.researchgate.net/publication/359468841.

It can be seen from the above results that the fusion effects are not visually satisfactory. The ghosting at the image edges is

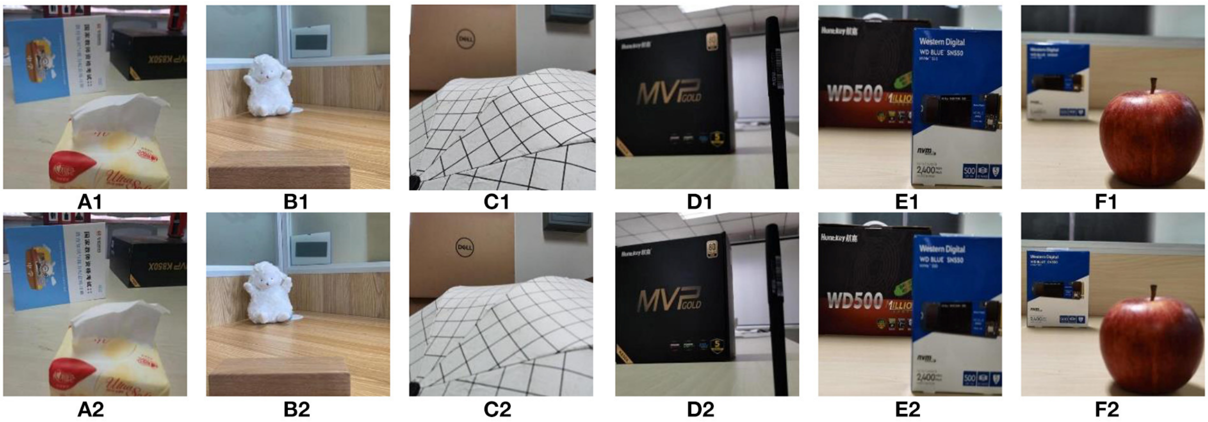


FIGURE 1
Some images selected from the HBU-CVMDSP dataset. **(A1–F1)** are the foreground focused image in a group of multi focus images. **(A2–F2)** are the background focused image in a group of multi focus images.

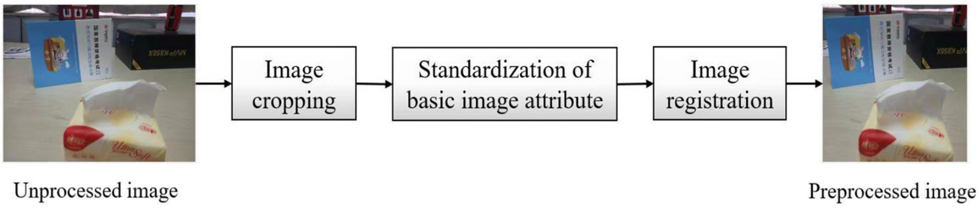


FIGURE 2
Schematic diagram of Image preprocessing.

mainly due to the misregistration of the images in the dataset, while the blocking and distortion in the images are due to the inconsistency of the brightness between the two source images in the dataset. Therefore, we conduct image cropping, standardization of basic attributes and registration processing on the dataset to ameliorate the quality of the fused images. If the mobile phone device shoots scenes with different focus areas, the obtained image field of view will be different. When the image background information is clear, the field of view is wider, and when the image near field information is clear, the view is narrower. Therefore, if two images with different focal points have the same size, the field of view of the two images will be different, and the ghosting will appear during the fusion process. In addition, the slight jitter when taking pictures will also lead to a slight gap in the field of view of two images. The images in HBU-CVMDSP dataset are cropped using the nearest neighbor interpolation algorithm. The details be found in https://www.researchgate.net/publication/359468841.

When smartphones collect a foreground and background focused image, due to the different depth of field, the attributes such as brightness and contrast of the image will be different. A group of images with different attributes will affect the matching of feature points in the image registration process, and the fusion image will appear block effect, resulting in unsatisfactory fusion result. In this paper, we standardize the basic attributes of color images using the SHINE_color toolbox (Willenbockel et al., 2010). When standardizing the basic attributes of images, we designate one image in the image group as the source image and the other image as the target image. Firstly, the source image and target image are transformed from RGB space to HSV space. Then the chroma, saturation and luminance are separated, the standardization of the basic attributes of the images is accomplished by adjusting the luminance channel of the source image and the target image to be equal in spatial frequency and direction. In this paper, the SIFT algorithm is used for image registration.

Experimental results and analysis

Experiment and analysis

In this experiment, we use the following nine metrics to quantitatively evaluate the performance of the image fusion

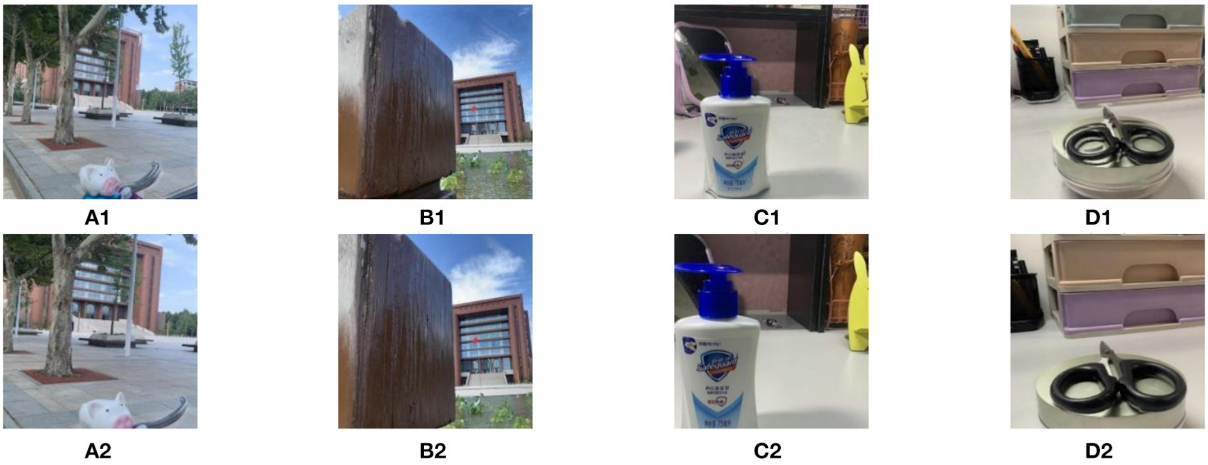


FIGURE 3
The fusion results before and after image registration. **(A1–D1)** are the fused images for the multi focus image pair without registration processing. **(A2–D2)** are the fused images by registration images.

algorithms: (1) Normalized mutual information (NMI), which can effectively improve the stability of the MI (Liu et al., 2020). (2) Nonlinear correlation information entropy (NCIE), which is a metric used to evaluate the quality of the fusion image (Su et al., 2022). (3) Gradient-based evaluation metric Q_G (Liu et al., 2020), which is used to evaluate the gradient information of the source image retained in the fused image. (4) Phase consistency based evaluation metric was proposed in Liu et al. (2020). (5) Structural similarity based evaluation metric Q_S, which is an image quality evaluation metric based on the universal quality index (Liu et al., 2020). (6) Structural similarity based evaluation metric Q_Y (Liu et al., 2020). (7) Human perception based evaluation metric Q_{CB}, which can be used to evaluate the contrast information between images (Liu et al., 2020). (8) Human perception based evaluation metric Q_{CV}, which is an image fusion evaluation metric based on human visual perception (Liu et al., 2020). (9) Tsallis entropy is a generalization of Shannon entropy, which can be used to evaluate the retentive information between the source image and the fusion image. For Q_{MI}, Q_{NCIE}, Q_G, Q_P, Q_S, Q_Y, Q_{CB}, and Q_{TE}, the higher the value of them is, the better the fusion result will be. And for the Q_{CV}, the smaller the value is, the better the fusion result will be.

Ablation experiment

To validate the importance of the pre-processing of the dataset, we use DSIFT and CNN fusion algorithms to conduct the fusion experiments on the dataset before and after image registration, and compare the subjective and objective fusion results of the two fusion algorithms. The experiments are completed by a PC with Intel core i5-10500, 3.10 GHz CPU, 8GB RAM memory, and NVIDIA GeForce GTX 1660 SUPER GPU. Due to space limitation, we only give the experimental results of the DSIFT algorithm. The experimental results of the CNN algorithm are shown in https://www.researchgate.net/publication/359468841.

The fusion results of the DSIFT algorithm are shown in Figure 3. The first row and second row in Figure 3 are the fusion results corresponding to the dataset before image registration and the dataset after image registration, respectively. Obviously, after image registration, the visual effects of the fused images in the second row have been significantly improved.

In addition, we calculate the values of Q_{MI}, Q_{TE}, Q_{NCIE}, Q_G, Q_P, Q_S, Q_Y, Q_{CV}, and Q_{CB} of the fused images obtained by DSIFT algorithm on the dataset before and after image registration, respectively. The values of the nine metrics are shown in Table 2, respectively, from which one can find that in addition to the decrease of the Q_{CV} value, the Q_{MI}, Q_{TE}, Q_{NCIE}, Q_G, Q_P, Q_S, Q_Y, and Q_{CB} values of the fused images obtained by the DSIFT on the dataset after image registration are all increased. Therefore, conducting the image registration process on the dataset can effectively improve the performance of the fusion algorithms both in subjective vision and objective evaluation.

The fusion results of the DSIFT algorithm on the dataset before and after standardizing the basic attributes of images are shown in the first row and second row of the Figure 4, respectively. After the standardization of the image basic attribute, the visual effects of the fused images shown in the second row of the Figure 4 have been significantly improved.

Furthermore, we also calculate the values of Q_{MI}, Q_{TE}, Q_{NCIE}, Q_G, Q_P, Q_S, Q_Y, Q_{CV}, and Q_{CB} of the fused images obtained by DSIFT algorithm on the dataset before and after

TABLE 2 The nine metrics' values of the fused images before and after image registration.

Test image	Preprocessing	Q_{MI}	Q_{TE}	Q_{NCIE}	Q_G	Q_P	Q_S	Q_Y	Q_{CV}	Q_{CB}
Piggy image	Before registration	1.029	0.3761	0.8445	0.6564	0.5465	0.854	0.9568	165.3	0.6904
	After registration	1.2	0.4389	0.8572	0.6861	0.6957	0.9339	0.9709	51.29	0.7578
Wood pile image	Before registration	1.021	0.3657	0.833	0.6702	0.6836	0.9043	0.9321	77.48	0.7011
	After registration	1.135	0.4046	0.8393	0.7185	0.7571	0.9478	0.9736	57.23	0.7788
Handwashing fluid	Before registration	1.192	0.4137	0.8474	0.6458	0.6512	0.9187	0.9105	145.9	0.6569
image	After registration	1.31	0.4443	0.8562	0.6771	0.7512	0.9659	0.9287	13.14	0.6974
Scissors image	Before registration	0.9387	0.3545	0.8279	0.6109	0.4073	0.8662	0.852	92.77	0.5971
	After registration	1.208	0.4276	0.8479	0.641	0.7335	0.9425	0.9208	7.673	0.6936

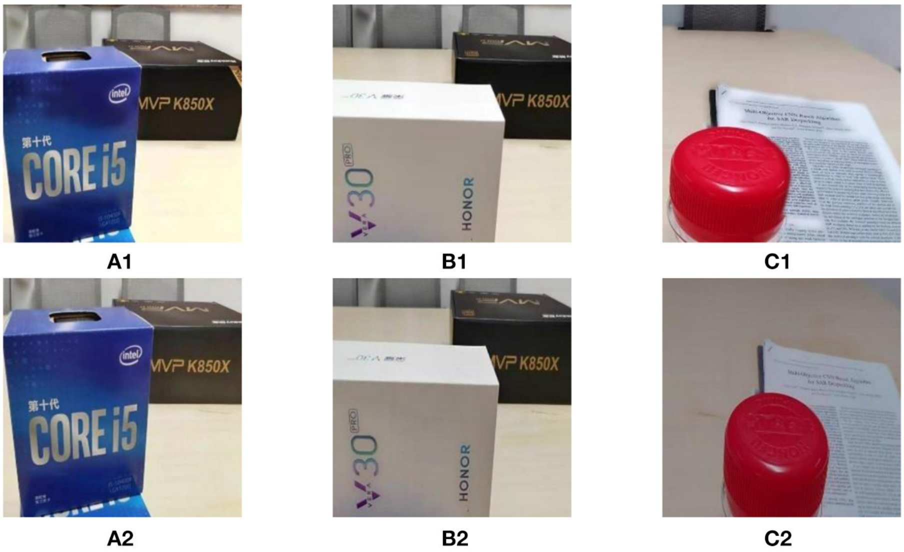


FIGURE 4
The fusion results before and after standardization of the image basic attribute. **(A1–C1)** are the fused images for the multi focus image pair without image basic attribute standardization processing. **(A2–C2)** are the fused images by image basic attribute standardization processing.

standardization of the image basic attribute. The calculated results of the nine metrics are shown in Table 3, respectively. From which one can find that in addition to the decrease of the value, the values of Q_{MI}, Q_{TE}, Q_{NCIE}, Q_G, Q_P, Q_S, Q_Y, and Q_{CB} of the fused images obtained by the DSIFT algorithm on the dataset after the standardization of the image basic attribute are all increased. Therefore, after the dataset is standardized by the image basic attribute, both the subjective vision and the objective evaluation are all improved.

Test of existing image fusion algorithms

In this subsection, we test the performance of some existing image fusion algorithms on the HBU-CVMDSP dataset. The multi-focus image fusion algorithms used in the test include multi-scale guided filtering algorithm (MGF) (Bavirisetti et al., 2019), dense scale-invariant feature transformation algorithm (DSIFT) (Liu et al., 2015), a general image fusion algorithm based on convolutional neural network (IFCNN) (Zhang et al., 2020), MIF algorithm based on convolutional neural network (CNN) (Liu et al., 2017), and unsupervised depth model for MIF (SESF) (Ma et al., 2020). We select six pairs of images from the HBU-CVMDSP dataset to test the above algorithms, and the selected images are shown in Figure 1. Figure 5 shows the fusion results of different algorithms on the selected images. In order to better show the visual effects of different fusion algorithms, the image of the red rectangular area in the figure is enlarged in this paper. From the Figure 5, it can be found that the fused image obtained all the fusion methods are all kinds of problems, such as block effect, unfocused pixels on the edge, blurred edges, the detailed information lost, the boundary too smooth, artificial artifacts, misclassification of focused pixels, distorted, and poor spatial consistency.

The nine metrics' values of the fused images in Figure 5 are shown in Table 4, in which the best result of each group

TABLE 3 The nine metrics' values of the fused images before and after standardization of the image basic attribute.

Test image	Preprocessing	Q_{MI}	Q_{TE}	Q_{NCIE}	Q_G	Q_P	Q_S	Q_Y	Q_{CV}	Q_{CB}
Blue black box	Before normalization	1.049	0.4295	0.8333	0.597	0.6621	0.8845	0.8211	156.4	0.6423
image	After normalization	1.151	0.4546	0.8372	0.6216	0.738	0.9515	0.9012	13.72	0.6541
White black box	Before normalization	1.199	0.4643	0.8445	0.6224	0.5284	0.9153	0 8392	43.79	0.6182
image	After normalization	1.233	0.4664	0.8454	0.6274	0.5985	0.9559	0.8899	24.99	0.6196
Bottle cap image	Before normalization	1.037	0.4542	0.8219	0.6639	0.5057	0.8898	0.8412	398.5	0.5374
	After normalization	1.196	0.4557	0.8303	0.6966	0.5568	0.9808	0.9081	25.64	0.657

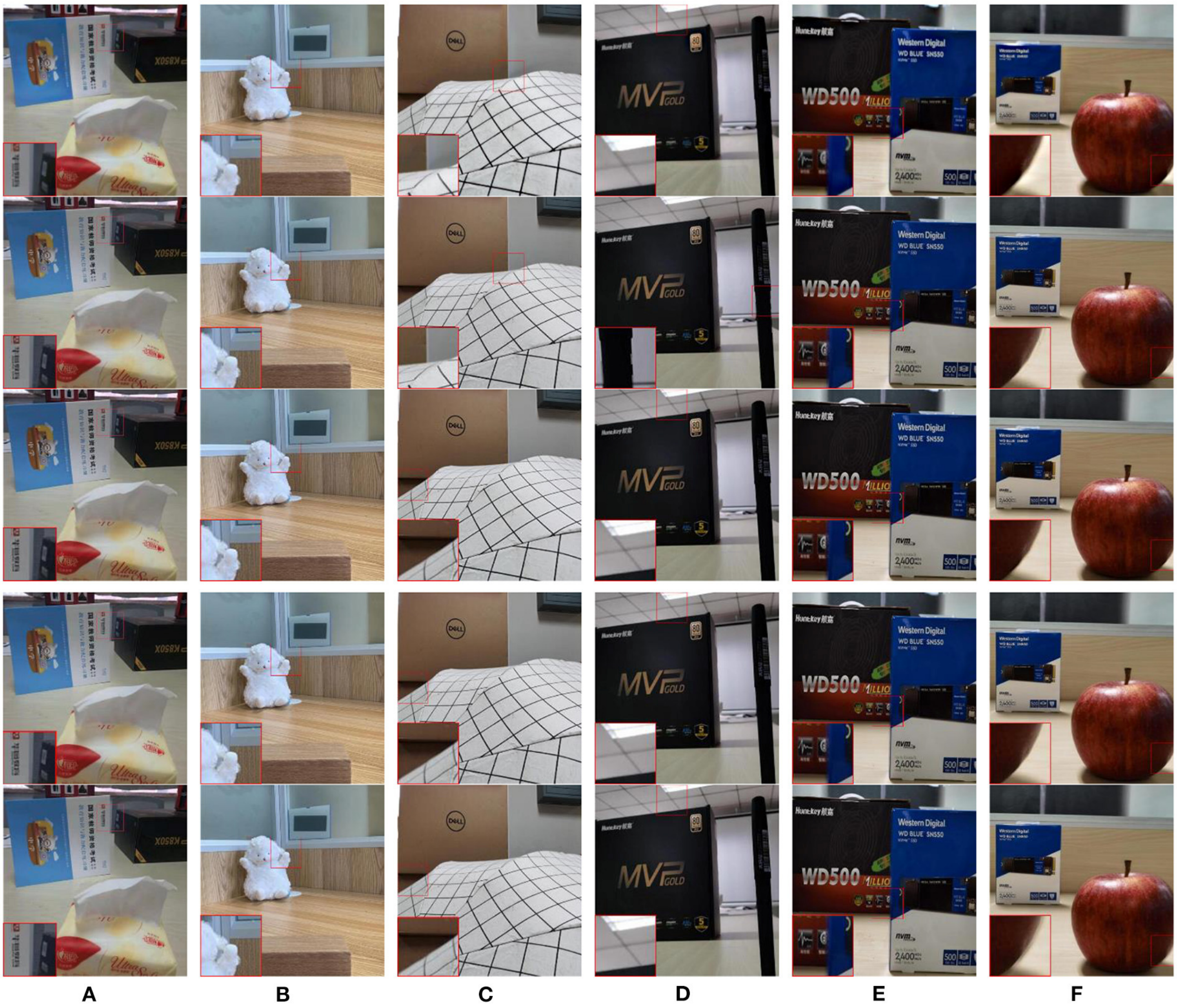


FIGURE 5
Fusion results of different algorithms. **(A–F)** are the fusion results of MGF, DSIFT, IFCNN, CNN, and SESF, respectively.

of fused images is bolded. As can be seen from the Table 4, in the objective evaluation of the fusion results of MGF, DSIFT, IFCNN, CNN and SESF in the real environment, no fusion algorithm has competitive performance compared with other comparison algorithms, which indicates that the multi focus image dataset in the real environment can reflect that the existing fusion algorithms cannot meet the application of MIF technology in the real environment. In addition, due to the limited generalization ability, these existing fusion algorithms all transfer specific prior knowledge to the model, and then

TABLE 4 The nine metrics' values of the fused images obtained by different fusion algorithms.

Fused images	Fusion algorithms	Q_{MI}	Q_{TE}	Q_{NCIE}	Q_G	Q_P	Q_S	Q_Y	Q_{CV}	Q_{CB}
Figure 5A	MGF	0.9503	0.4118	0.8337	0.5144	0.6205	0.9703	0.8081	171.61	0.6218
	DSIFT	**1.331**	**0.4474**	**0.8591**	**0.6593**	0.7285	0.9713	0.9365	52.805	0.7087
	IFCNN	1.1692	0.4364	0.8449	0.5663	0.6786	**0.9751**	0.86	**36.691**	0.6591
	CNN	1.2944	0.4449	0.8548	0.6515	**0.7599**	0.9736	**0.9478**	52.857	**0.7094**
	SESF	1.2839	0.4412	0.8544	0.641	0.7201	0.9696	0.9438	52.823	0.7079
Figure 5B	MGF	1.0355	**0.4622**	0.8336	0.6567	0.7942	**0.9697**	0.9066	**7.5085**	0.7040
	DSIFT	1.259	0.4419	0.8457	0.7262	0.9103	0.9673	0.9595	8.991	0.6965
	IFCNN	1.0766	0.4535	0.8356	0.6711	0.8433	0.967	0.9241	9.2736	0.6853
	CNN	1.2673	0.4456	0.8471	0.7364	**0.9132**	0.9688	**0.9707**	9.4967	0.7087
	SESF	**1.2852**	0.4477	**0.8491**	**0.9319**	0.9118	0.9681	0.9705	9.3427	**0.7196**
Figure 5C	MGF	0.8845	0.3979	0.8284	0.5109	0.6787	0.9507	0.8332	131.73	0.6279
	DSIFT	**1.3773**	**0.455**	**0.8593**	0.696	0.7722	0.9646	0.9793	119.89	0.7871
	IFCNN	1.0918	0.4162	0.8381	0.5677	0.7317	0.9642	0.8996	**35.251**	0.6962
	CNN	1.3699	0.4519	0.8582	**0.6993**	**0.7732**	**0.9649**	**0.9898**	119.86	**0.7949**
	SESF	1.3498	0.4481	0.857	0.6913	0.7717	**0.9649**	0.9813	35.31	0.7902
Figure 5D	MGF	1.0282	0.4094	0.8352	0.5355	0.6556	0.9598	0.7767	41.591	0.6395
	DSIFT	**1.4076**	**0.4498**	**0.8602**	0.6732	0.792	0.9714	0.9287	46.321	0.7394
	IFCNN	1.2627	0.4372	0.8474	0.5987	0.7460	**0.9727**	0.8541	**36.335**	0.6912
	CNN	1.3808	0.4464	0.8561	**0.6804**	**0.8007**	0.9722	**0.9562**	44.702	**0.7575**
	SESF	1.3962	0.4472	0.859	0.673	0.7884	0.9707	0.9454	44.88	0.7504
Figure 5E	MGF	0.9027	0.412	0.8293	0.526	0.6821	0.9508	0.7965	72.736	0.6049
	DSIFT	**1.3144**	0.438	**0.8497**	**0.6772**	0.8211	0.9654	0.9442	**23.078**	0.7069
	IFCNN	1.1571	**0.4413**	0.8406	0.5843	0.7547	**0.9691**	0.8659	26.5	0.6623
	CNN	1.2906	0.4363	0.848	0.6771	**0.8369**	0.9675	**0.9659**	23.374	**0.7234**
	SESF	1.2784	0.4334	0.8474	0.6656	0.8083	0.9641	0.9424	29.576	0.7165
Figure 5F	MGF	0.9452	0.4101	0.8323	0.5279	0.5967	0.9611	0.8309	96.451	0.6142
	DSIFT	**1.32**	**0.4479**	**0.8531**	0.6927	0.7965	0.9666	0.9684	39.348	0.7202
	IFCNN	1.174	0.4388	0.8436	0.5964	0.7127	**0.9691**	0.9	**19.685**	0.6651
	CNN	1.3072	0.447	0.8522	**0.6949**	**0.8299**	0.9678	0.9776	38.345	**0.7232**
	SESF	1.297	0.4451	0.8521	0.6915	0.8053	0.9669	**0.9784**	45.9072	0.7208

The bold values represent optimal values.

perform image fusion. However, images in the real world are very complex, and cannot be achieved only through the prior knowledge of inherent images. Therefore, the HBU-CVMDSP dataset can be used as a new test set to promote the development of the field of MIF and narrow the gap between the theoretical and real environmental data of image fusion algorithms.

Conclusion

Due to the existing MIF datasets cannot reflect the image registration caused by physical movement or camera shake, and the brightness differences caused by illumination in real life, we proposed a new MIF dataset i.e., the HBU-CVMDSP dataset. Images in this dataset are captured by smartphone, and can truly reflect the real-world scene. In addition, we test the performance of some existing fusion algorithms on the proposed dataset. The results indicate that the performance of these algorithms on the proposed dataset has much room for improvement. Therefore, the HBU-CVMDSP dataset can better promote the research of the MIF algorithms.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: https://www.researchgate.net/publication/359468841.

Author contributions

WP and YY performed the computer simulations. SL, WJ, and JZ analyzed the data. SL, WP, and YY wrote the original draft. WJ, YS, and JZ revised and edited the manuscript. YS polished the manuscript. All authors confirmed the submitted version.

Funding

This work was supported in part by National Natural Science Foundation of China under Grant No. 62172139, Natural Science Foundation of Hebei Province under Grant No. F2022201055, Project Funded by China Postdoctoral under Grant No. 2022M713361, Science Research Project of Hebei Province under Grant No. BJ2020030, Natural Science Interdisciplinary Research Program of Hebei University under Grant No. DXK202102, Open Project Program of the National Laboratory of Pattern Recognition (NLPR) under Grant No. 202200007, Foundation of President of Hebei University under Grant No. XZJJ201909, Research Project of Hebei University Intelligent Financial Application Technology R&D Center under Grant No. XGZJ2022022, and Open Foundation of Guangdong Key Laboratory of Digital Signal and Image Processing Technology under Grant No. 2020GDDSIPL-04.

Acknowledgments

Thanks for supporting by the High-Performance Computing Center of Hebei University.

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

Bavirisetti, D. P., Xiao, G., Zhao, J., Dhuli, R., and Liu, G. (2019). Multi-scale Guided image and video fusion: a fast and efficient approach. *Circ. Syst. Signal Proc.* 38, 5576–5605. doi: 10.1007/s00034-019-01131-z

Bouzos, O., Andreadis, I., and Mitianoudis, N. (2019). Conditional random field model for robust multi-focus image fusion. *IEEE Trans. Image Proc.* 28, 5636–5648. doi: 10.1109/TIP.2019.2922097

Liu, S., Ma, J., Yang, Y., Qiu, T., Li, H., Hu, S., et al. (2022). A multi-focus color image fusion algorithm based on low vision image reconstruction and focused feature extraction. *Signal Proc. Image Commun.* 100, 116533. doi: 10.1016/j.image.2021.116533

Liu, S., Ma, J., Yin, L., Li, H., Cong, S., Ma, X., et al. (2020). Multi-focus color image fusion algorithm based on super-resolution reconstruction and focused area detection. *IEEE Access* 8, 90760–90778. doi: 10.1109/ACCESS.2020.2993404

Liu, S., Miao, S., and Su, J. (2021). UMAG-Net: A new unsupervised multiattention-guided network for hyperspectral and multispectral image fusion. *IEEE J. Select. Top. Appl. Earth Observat. Remote Sens.* 14, 7373–7385. doi: 10.1109/JSTARS.2021.3097178

Liu, S., Wang, J., Lu, Y., Li, H., Zhao, J., and Zhu, Z. (2019). Multi-focus image fusion based on adaptive dual-channel spiking cortical model in non-subsampled shearlet domain. *IEEE Access* 7, 56367–56388. doi: 10.1109/ACCESS.2019.2900376

Liu, Y., Chen, X., Peng, H., and Wang, Z. (2017). Multi-focus image fusion with a deep convolutional neural network. *Informat. Fusion* 36, 191–207. doi: 10.1016/j.inffus.2016.12.001

Liu, Y., Liu, S., and Wang, Z. (2015). Multi-focus image fusion with dense SIFT. *Inf. Fusion* 23, 139–155. doi: 10.1016/j.inffus.2014.05.004

Ma, B., Zhu, Y., Yin, X., Ban, X., Huang, H., and Mukeshimana, M. (2020). SESF-Fuse: an unsupervised deep model for multi-focus image fusion. *Neural Comput. Appl.* 33, 5793–5804. doi: 10.1007/s00521-020-05358-9

Nejati, M., Samavi, S., and Shirani, S. (2015). Multi-focus image fusion using dictionary-based sparse representation. *Inf. Fusion* 25, 72–84. doi: 10.1016/j.inffus.2014.10.004

Su, X., Li, J., and Hua, Z. (2022). Transformer-based regression network for pansharpening remote sensing images. *IEEE Trans. Geosci. Remote Sens.* 60, 5407423. doi: 10.1109/TGRS.2022.3152425

Willenbockel, V., Sadr, J., Fiset, D., Horne, G. O., Gosselin, F., and Tanaka, J. W. (2010). Controlling low-level image properties: the SHINE toolbox. *Behav. Res. Methods* 42, 671–684. doi: 10.3758/BRM.42.3.671

Xiao, B., Ou, G., Tang, H., Bi, X., and Li, W. (2020). Multi-focus image fusion by hessian matrix based decomposition. *IEEE Trans. Multimedia* 22, 285–297. doi: 10.1109/TMM.2019.2928516

Zhang, Y., Liu, Y., Sun, P., Yan, H., Zhao, X., and Zhang, L. (2020). IFCNN: a general image fusion framework based on convolutional neural network. *Inf. Fusion* 54, 99–118. doi: 10.1016/j.inffus.2019.07.011

Zheng, M., Qi, G., Zhu, Z., Li, Y., Wei, H., and Liu, Y. (2020). Image dehazing by an artificial image fusion method based on adaptive structure decomposition. *IEEE Sens. J.* 20, 8062–8072. doi: 10.1109/JSEN.2020.2981719

Zhu, Z., Wei, H., Hu, G., Li, Y., Qi, G., and Mazur, N. (2021). A novel fast single image dehazing algorithm based on artificial multiexposure image fusion. *IEEE Trans. Instrum. Meas.* 70, 1–23. doi: 10.1109/TIM.2020.30 24335

frontiers | Frontiers in Neurorobotics

TYPE Original Research
PUBLISHED 16 November 2022
DOI 10.3389/fnbot.2022.1029968

Check for updates

OPEN ACCESS

EDITED BY
Zhou Wei,
Yunnan University, China

REVIEWED BY
Haisheng LI,
Beijing Technology and Business University, China
Ly Ngoc,
Ho Chi Minh City University of Science, Vietnam
Weizhi Nie,
Tianjin University, China

*CORRESPONDENCE
Xiaolin Wang
203602864@st.usst.edu.cn

RECEIVED 28 August 2022
ACCEPTED 01 November 2022
PUBLISHED 16 November 2022

CITATION
Wang W, Wang X, Chen G and Zhou H (2022) Multi-view SoftPool attention convolutional networks for 3D model classification.
Front. Neurorobot. 16:1029968.
doi: 10.3389/fnbot.2022.1029968

Multi-view SoftPool attention convolutional networks for 3D model classification

Wenju Wang, Xiaolin Wang*, Gang Chen and Haoran Zhou

College of Communication and Art Design, University of Shanghai for Science and Technology, Shanghai, China

Introduction: Existing multi-view-based 3D model classification methods have the problems of insufficient view refinement feature extraction and poor generalization ability of the network model, which makes it difficult to further improve the classification accuracy. To this end, this paper proposes a multi-view SoftPool attention convolutional network for 3D model classification tasks.

Methods: This method extracts multi-view features through ResNest and adaptive pooling modules, and the extracted features can better represent 3D models. Then, the results of the multi-view feature extraction processed using SoftPool are used as the Query for the self-attentive calculation, which enables the subsequent refinement extraction. We then input the attention scores calculated by Query and Key in the self-attention calculation into the mobile inverted bottleneck convolution, which effectively improves the generalization of the network model. Based on our proposed method, a compact 3D global descriptor is finally generated, achieving a high-accuracy 3D model classification performance.

Results: Experimental results showed that our method achieves 96.96% OA and 95.68% AA on ModelNet40 and 98.57% OA and 98.42% AA on ModelNet10.

Discussion: Compared with a multitude of popular methods, our algorithm model achieves the state-of-the-art classification accuracy.

KEYWORDS

3D model classification, multi-view, attention, SoftPool, convolutional

1. Introduction

With the rapid development of 3D acquisition technology, various types of sensor devices (e.g., 3D scanners, LIDAR, and RGB-D cameras) can collect 3D data conveniently and quickly (Grenzdörffer et al., 2020). 3D data are abundant in geometry, shape, and scale information and simple in expression, so are well suited for 3D scene perception and understanding. 3D model-based classification is an important fundamental task in 3D visual perception tasks such as target segmentation, recognition and tracking, and matching. 3D model classification methods are currently extensively applied in the fields of robotics (Kästner et al., 2020), autonomous driving (Yu et al., 2021), 3D scene reconstruction (Pontes et al., 2017), augmented reality (Adikari et al., 2020), and medicine (Liu et al., 2020); hence, 3D model classification methods have become a research hotspot.

3D model classification methods can be divided into two fields: traditional and recent deep learning. Early 3D model classification tasks focused on hand-designed feature extraction followed by machine learning methods for classification (e.g., extreme learning machines and support vector learning). Lalonde et al. (2005) investigated the automatic data-driven scale selection problem using an approach driven by the Gaussian mixture model geometry. The method does not consider the relationship between neighbors, and the results are affected by noise, leading to poor classification accuracy. To solve these problems, Niemeyer et al. (2014) combined the contextual information and then embedded the random forest classifier into the conditional random field (CRE), which improved the classification accuracy to some extent. However, optimization is still essential in terms of feature extraction and the graph structure, as well as research on reducing the amount of data and the training time.

Traditional methods generally have several deficiencies, including limited manual feature extraction and low classification accuracy. Deep learning technology has achieved considerably good performance in computer vision, natural language processing, speech recognition, and other fields. In recent years, ModelNet (Wu et al., 2015), ShapeNet (Yang et al., 2021), ScanNet (Zou et al., 2021), and other publicly available datasets have also driven research in 3D model classification based on deep learning. 3D model classification methods based on deep learning can be divided into three categories based on the representation of the input data: voxel-based, point cloud-based, and multi-view-based.

1.1. Voxel-based methods

The voxel-based model method aims to voxelize the point cloud first, then employ a 3D convolutional neural network (CNN) to extract features, and finally complete the classification task. Maturana and Scherer (2015) proposed VoxelNet based on the idea of voxels, which is the voxelization of unstructured point cloud data into regular grid data for classification. The method corresponds each grid to a voxel, and the values in the grid cells are normalized and input to the convolutional layer in the network for feature extraction and classification. However, this method consumes a large amount of memory because of the large number of zero-valued voxels that appear in the process. Wu et al. (2015) proposed a convolutional deep belief network (3DShapeNet) for the classification of 3D models of different kinds and different poses. Both VoxNet and 3DShapeNet have the problems of prohibitive memory overhead in the computation and low accuracy of model classification. To reduce the memory consumption and running time, Riegler et al. (2017) proposed OctNet, a sparse 3D data representation method. The spatial stratification is represented as a series of unbalanced octree structures with pooled features stored on the leaf nodes in the octree. This method allows CNNs to handle high resolutions with reduced memory consumption, yet the problem of losing local geometric information has not been solved. Aiming to solve the problem, Wang et al. (2018) divided the whole space into voxels of different scales and employed the proposed multi-scale convolutional network (MSNet) to learn local features adaptively and fuse the local features to predict the class probability of the model. The network allows for improved classification accuracy and the ability to retain a large amount of information, but the training time of a voxelized grid can be exceedingly long. To reduce the time consumption, Le and Duan (2018) proposed the 3D convolutional grid PointGrid. It belongs to the regular embedded voxel grid, and the network can extract a large number of local features for 3D model classification.

In summary, the voxel-based method converts 3D point clouds into voxel meshes, solving the problem of unstructured 3D point clouds. However, as the voxelization requires the input voxel format to be regular for a convolution operation, a large amount of information is lost when the voxel resolution is low, which causes the problem of low classification accuracy. Moreover, it has the problem of high computational cost when the resolution is high.

1.2. Point cloud-based methods

The point cloud-based method aims to directly classify point cloud data obtained by 3D scanners, LIDAR, and RGB-D cameras using the corresponding approaches. Qi et al. (2017a) considered the direct processing of point cloud data and proposed the PointNet network, which transforms the input point cloud through the T-Net matrix and applies the multilayer perceptron (MLP) to learn the features of the points and aggregate them into global features. Their experiment and analysis showed that PointNet made a great breakthrough in point cloud classification and segmentation, but it could not capture local information and had poor generalization ability. PointNet++ (Qi et al., 2017b) was proposed based on the shortcomings of PointNet in recognizing fine-grained patterns. By introducing a hierarchical neural network and metric spatial distance, the context ratio can be increased, and thus the network can better learn local features. The introduced ensemble learning layer can adaptively combine multiple scale features for classification. Nevertheless, this method lacks some structural information between points. Ma et al. (2018) proposed the 3DMAX-Net architecture influenced by the contextual information mechanism. This network can obtain the contextual features in 3D point cloud space through the introduced multi-scale feature learning block, while the features learned by the network are aggregated through a local-global feature aggregation block. Qiu et al. (2021) proposed a density resolution network by introducing an adaptive extended point algorithm; an error minimization module in the network is utilized to extract multi-resolution information, and local features are fused to achieve the point cloud classification task. The classification accuracy of the model was shown to be higher than that in the PointNet network. Additionally, both 3DMAX-Net network and density-resolution network are not applicable to large-scale point clouds; they are also especially insufficient in the case of many object classes.

To address the problem that most networks cannot adapt to large-scale point clouds, Hu et al. (2020) proposed RandLA-Net, which is based on a complex sampling technique that devises random point sampling to reduce computation and memory, while the introduced local feature aggregation blocks retain important information among neighbors. RandLA-Net can directly handle large-scale point clouds, and using a lightweight network can improve classification accuracy while greatly reducing the computational memory and time overhead. However, because the RandLA-Net network chooses random sampling, there is a loss of useful information. Liang et al. (2019) proposed a deep graph CNN for local geometric feature extraction, which obtains a large amount of useful information and has a smaller memory consumption compared to previous graph convolution methods. Zhang et al. (2020) proposed an omnidirectional graph neural network for further improving the performance of the network and reducing the complexity of the model. The method proposes LKPO-GNN for obtaining local and global spatial information, learning the local topology of the point cloud using the omnidirectional local KNNs pattern, and aggregating the local information spatial structure to obtain the global map using GNN. In contrast, the KNN pattern still has defects in neighborhood search. Feng et al. (2020) considered the lack of performance in neighborhood search and constructed local graphs based on searching neighborhood points in multiple directions while assigning attention coefficients to each edge of the graph and aggregating centroid features as a weighted sum of its neighboring points to obtain local features. Moreover, the point-by-point spatial attention module is used to generate the interdependency matrix of points so that local features and contextual information can be obtained simultaneously. The performance of this method is enhanced in point cloud classification and segmentation. Wen et al. (2020) proposed a novel deep learning network of Point2SpatialCapsule based on aggregating local features and spatial relationships of point clouds. This network consists of two modules, geometric feature aggregation, and spatial relationship aggregation, which are capable of aggregating local features to clustering centers and aggregating their spatial relationships in the feature space using spatially aware capsules. This method has greatly elevated the accuracy of tasks (e.g., point cloud classification retrieval).

However, owing to the disorderly and unstructured nature of 3D point clouds, as well as the fact that scanned models in real scenes can be obscured and result in partial data loss and complex scenes, direct methods of processing point clouds are often more complex and take longer to train.

1.3. Multi-view-based methods

The multi-view-based method aims to project the 3D model from multiple virtual cameras into the 2D plane and then perform convolutional feature extraction and fusion on the obtained multi-views to accomplish the task of 3D model classification. The earliest rendering of 3D point clouds into multi-views and applying them to model classification is the MVCNN network proposed by Su et al. (2015). The classification accuracy and performance of MVCNN represent a remarkable breakthrough in point cloud classification, but because of the maximum pooling, keeping only the largest elements in these views can lead to a large amount of information loss. To reduce the loss of effective information, Wang et al. proposed RCPCNN (Wang C. et al., 2019) to perform dominant set clustering from the views of the same cluster. RCPCNN is updated iteratively in the pooling layer in a round-robin fashion. This method improves the classification performance but ignores the relationships among views. Feng et al. (2018) introduced a hierarchical view-group-shape framework, called GVCNN, which is based on MVCNN to better utilize the connection between multiple views. It can find more discriminative features among views and offers a significant improvement in classification accuracy. Yet, this method relies too much on the choice of the viewpoint angle and is not applicable to the case of a small number of views. Yu et al. (2018) proposed MHBN using the relationship between the polynomial kernel and bilinear pool and considered that local complementary information exists among different views. Bilinear pooling aggregates local features to measure similar pairs of related patch pairs and coordinates the merging of bilinear features to generate a more discriminative 3D object representation. MHBN offers an improvement in classification accuracy and storage efficiency, and also effectively suppresses irrelevant matching pairs. Ma et al. (2019) combined CNNs with long short-term memory (LSTM) based on the sequential nature among views and used LSTM and sequential voting layers to aggregate multi-view features into shape descriptors for object recognition.

Han et al. (2019b) proposed the SeqViews2SeqLabels network considering the spatial relationship of views. It is composed of an encoder for aggregating sequence views and a decoder for global feature prediction sequence labels. An attention mechanism is incorporated in this decoder, and specific views are assigned more weights to improve the discriminative ability. Moreover, better classification accuracy is obtained. For this reason, they further proposed the 3D2SeqViews network (Han et al., 2019a), which has more novel hierarchical attention to efficiently aggregate the content information of views and spatially related information between views. It affords great progress in global feature aggregation. However, CNN and LSTM combined with SeqViews2SeqLabels networks can only aggregate ordered views, not unordered views. Based on this problem, Yang and Wang (2019) proposed a relational network from the perspective of relationships among different view regions and views. The training methods effectively connect the corresponding regions through the self-attention module, combining the inter-view relationships to highlight the salient information more, which can enhance the information of single-view images. In contrast, there are still shortcomings in the selection of relationships among views, and selecting views that do not overlap and just complement each other still needs to be studied further. To improve the generalization ability and performance of the model, Sun et al. (2021) proposed a dynamically routed CNN. The method is based on a dynamic routing algorithm for adaptive selection of features for transformation, which does not ignore the inconspicuous information in the pooling layer and effectively fuses the features of all views. Wei et al. (2020) proposed view-GCN from the perspective of graph convolution. It is a hierarchical network based on view-graph representation, which is a viewgraph constructed by using multiple views as graph nodes and sampling representative views by the introduced view selection mechanism. The local and non-local convolution of this network performs feature transformation, which can obtain 3D object descriptors with different levels of feature combinations. Yet, this network is less flexible and scalable for shallow GCNs, and cannot pass the labels with little training data to the whole graph structure. On this basis, Liu et al. (2021) proposed a hierarchical multi-view context modeling approach, which consists of four main components: view-level context learning, the multi-view grouping module, the primitive group level, and the group fusion module. The method can fuse group-by-group contextual features into compact 3D object descriptors for object classification according to their importance.

So far, the view-based approach has achieved the best results on 3D model classification tasks. Compared to the direct point cloud and voxel processing approach, it can capture the features of the view more easily and learn the view features to synthesize true global feature descriptors with the help of a proven CNN. However, the method still has shortcomings in feature extraction, because the traditional pooled downsampling method cannot treat each view equally and only retains the information considered important. This leads to the problem of the insufficient extraction of view refinement feature information and the loss of a large amount of view feature information. However, different convolutional models learn different classification rules through a given dataset, so the classification accuracy predicted by the network model for unknown datasets varies greatly. Therefore, different convolutional models do not have the same degree of generalization. Both insufficient extractions of view refinement feature information and weak model generalization affect the further improvement of 3D model classification accuracy. Based on the above analysis, we propose a multi-view SoftPool attention convolutional network framework (MVMSAN) for 3D model classification tasks. Compared with traditional methods, our method employs a SoftPool attention convolution framework that can extract refined view feature information, effectively solving the problem of feature information loss and insufficient detail feature extraction during downsampling while enhancing the generalization ability of the model. Thus, the framework improves the accuracy of 3D model classification.

This study made the following contributions:

(1) We propose the MVMSAN network framework. It employs ResNest with the adaptive pooling method, SoftPool attention method, and self-attention convolution method to generate discriminative global descriptors for 3D model classification. Compared with a multitude of popular methods, our network framework achieves the state-of-the-art classification accuracy.

(2) ResNest with the adaptive pooling method removes the last fully connected layer and adds an adaptive pooling layer. This method can be applied to the extraction of view feature information, which focuses more on the feature information among view channels, reinforces the representation of feature maps, and better obtains real 3D features from 2D views.

(3) The SoftPool attention method can obtain finer view feature information, emphasize the importance of detailed features, and obtain more distinguishing features with model categories, because SoftPool uses the processed view feature value as the Query value of the self-attention. The self-attention-based convolution method can also improve the generalization ability of the model and focus on the learning ability of the algorithmic framework to increase the accuracy of 3D model classification, because Mobile inverted Bottleneck Convolution (MBConv) is used to process the Query and Key of self-attention.

(4) Our extensive experiments on the ModelNet40 and ModelNet10 datasets demonstrate the effectiveness of the proposed method. The experimental results show that, compared with existing state-of-the-art classification methods, the overall classification accuracy of our method on the two datasets reaches 96.96 and 98.57%, respectively.

2. Methods

The framework diagram Multi-view SoftPool Attention Convolution (MVMSAN) proposed by us is divided into three modules (Figure 1): the 3D model multi-view acquisition module, multi-view refinement feature extraction module, and feature fusion classification module. The multi-view acquisition module presents the 3D model in multiple views. The multi-view refinement feature extraction module employs ResNest with an adaptive pooling method to extract the feature information of the view. Then it uses our proposed SoftPool attention convolution method for view feature refinement extraction, which enables the subsequent fusion to generate more compact global descriptors. The feature fusion classification module aggregates refined features through pooling layers to generate global representation and completes 3D model classification by 1 × 1 convolution. The MVMSAN network framework will obtain a trained classification network model in the training phase, which uses datasets including ModelNet40 and ModelNet10 as training data. Any 3D mesh model can be input into the MVMSAN classification model trained for classification prediction in the testing phase.

2.1. 3D model multi-view acquisition

Our input is a mesh, point cloud representation of the 3D model. Then, a set of images from different angles $V = \{v_1, ...v_i..., v_{20}\}$ are used instead of the virtual 3D model, where V_i denotes the 2D images generated from 1 to 20 different viewpoint angles for any 3D model. The process applies the viewpoint selection method proposed by Kanezaki et al. (2018), which involves placing the 3D model at the center of the ortho dodecahedron and 20 virtual cameras on 20 vertices of the ortho dodecahedron. The dodecahedron is chosen because it has the highest number of vertices among the ortho polyhedra, and all viewpoints are evenly distributed in the 3D space where the 3D model is located.

2.2. Multi-view refinement local feature extraction

2.2.1. Extraction of view features based on ResNest with the adaptive pooling method

For the 20 views $V = \{v_1, ...v_i..., v_{20}\}$ obtained from the 3D model rendering, we use ResNest (Zhang et al., 2022) to extract the view features. ResNest is based on ResNet with the addition of split-attention blocks, which can exploit the interrelationship among view channels. Thus, it increases the perceptual field of feature extraction, strengthens the representation of feature maps, and reduces information loss. The view feature information extracted by ResNest is denoted as$\{m_1, ...m_i..., m_{20}\}$. See Equation (1):

$$\begin{cases} m_1 = ResNest(v_1) \\ \vdots \\ m_i = ResNest(v_i) \\ \vdots \\ m_{20} = ResNest(v_{20}) \end{cases} \tag{1}$$

where $\{m_1, ...m_i..., m_{20}\}$ denotes the 20 extracted view features.

All the view features are stitched together to obtain the following Equation:

$$M = \sum_{i=1}^{i=20} ResNest(v_i) \tag{2}$$

To satisfy the data input requirements for the subsequent SoftPool attention convolution processing (Section 2.2.2), we propose a combination of ResNest and adaptive pooling for view feature extraction. In this method, ResNest removes the final fully-connected layer and adds an AdaptiveAvgPool2d process. This is because adaptive pooling can obtain the output of a specified size based on an input, and the number of features in the input and output does not change. Therefore, the output of ResNest after adaptive pooling ensures that the view feature information extracted by the network remains unchanged and also satisfies the input requirements for the subsequent SoftPool attention convolution.

The view features extracted by ResNest are processed by the adaptive pooling layer to obtain F, as shown in Equation (3):

$$F = AAP(M) \tag{3}$$

2.2.2. Refined feature extraction based on SoftPool attention convolution

There is also some unnecessary information in the view features (F) extracted using ResNest with the adaptive pooling method. This information is redundant for aggregation into a global descriptor. For this purpose, we propose a SoftPool attention convolution method to accomplish refined feature extraction. This method mainly relies on the self-attention mechanism (Zhang et al., 2019). As self-attention can process the entire input view feature information globally, its strong global perception capability enables global feature extraction of view features. However, it is deficient in the refinement extraction of local features of the view. Moreover, it lacks the inductive bias property, so it has poor generalization. Also, our proposed SoftPool attention convolution method solves these problems and can achieve fine-grained extraction of view features. It contains the following two modules: Refinement feature extraction based on the SoftPool self-attention method; and Model generalization enhancement based on self-attention convolution (Figure 2).

2.2.3. Refined feature extraction based on softPool self-attention method

The pooling layer used in most neural networks is either max pooling or average pooling. Max pooling selects only the max activation values in the region, resulting in a large amount of information loss. In contrast, average pooling averages all activation values, which reduces the overall region

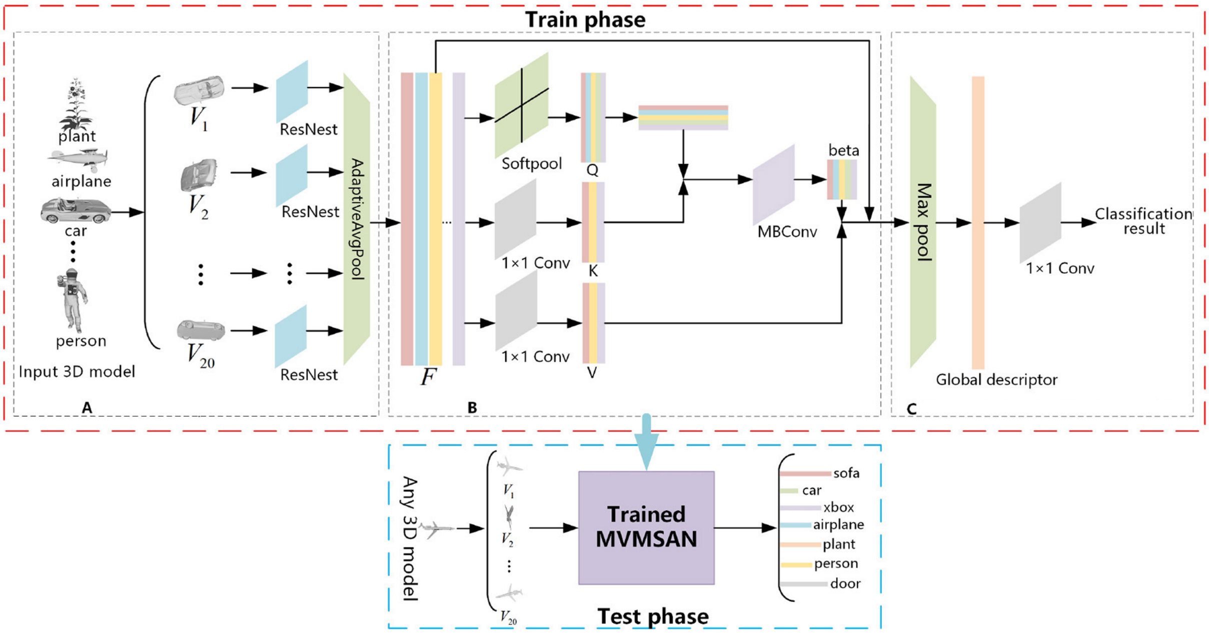


FIGURE 1
Multi-view SoftPool attention convolution (MVMSAN) network framework. **(A)** Multi-view acquisition module. **(B)** Multi-view refined feature extraction module. **(C)** Feature fusion classification module.

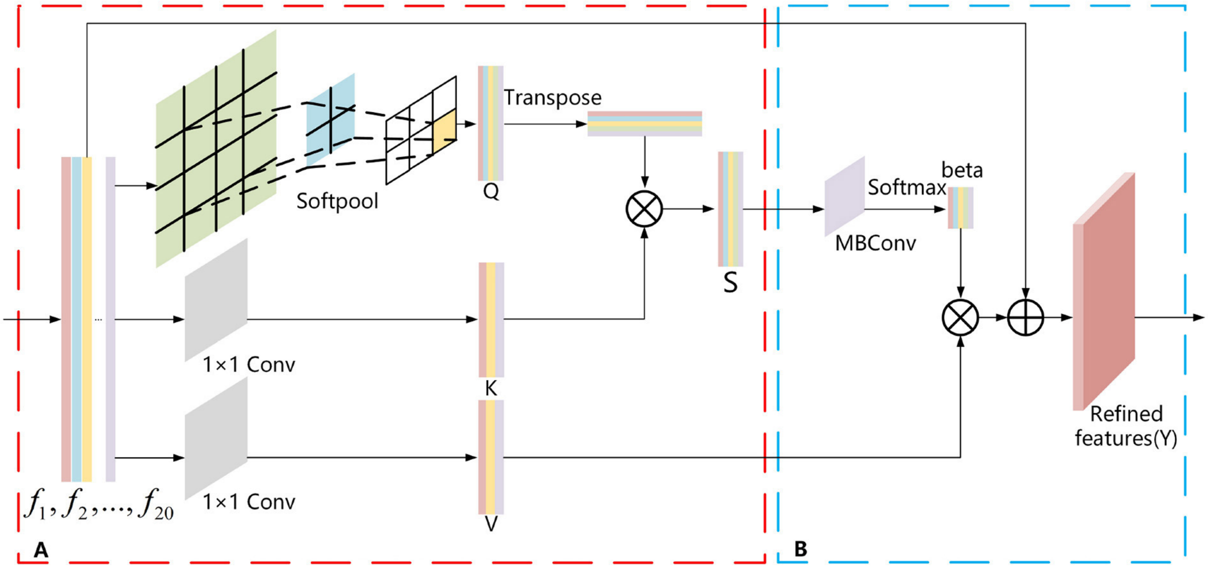


FIGURE 2
SoftPool attention convolution method. **(A)** Refinement feature extraction based on SoftPool. **(B)** Model generalized enhancement based on self-attention convolution.

characteristics. Therefore, it is not appropriate to choose either max pooling or average pooling for view feature extraction. The SoftPool method (Stergiou et al., 2021) first selects the activation graph, divides the individual activation values in the activation graph by the sum of the natural exponents of all activation values to obtain the corresponding weight values, multiplies all the weights by the corresponding activation values, and sums them to obtain the output. This makes all activation values of the feature map act on the final output, which is the greatest difference between SoftPool and max and average pooling. To this end, this paper proposes the SoftPool self-attention method, which makes full use of the strong global perception capability of self-attention and preserves the detailed information of multi-view features by using SoftPool. The self-attention mechanism obtains the corresponding V-value after calculating the similarity between Q and K vectors, and then the V-value is weighted and summed to obtain the value of the self-attention method. In this method, SoftPool uses the processed view feature F-value as the Q value of self-attention, which can refine the multi-view feature downsampling process and retain more multi-view feature detail information to achieve refined feature extraction (Figure 2A). It effectively overcomes the shortage of the self-attention mechanism in viewing the local feature refinement extraction and helps to generate ultimate global descriptors with discriminative ability.

The process is divided into two steps:

(1) For the $F = \left\{f_1, ...f_i..., f_{20}\right\}$ view features extracted by ResNest with the adaptive pooling method, f_i denotes the feature of the i-th view. We take the view feature (F) as input and generate a feature map (Q) by SoftPool (Stergiou et al., 2021) processing. Two 1 × 1 convolutions are also used to generate the feature maps K and V. See Equations (4), (5), and (6):

$$Q = SoftPool(F) \tag{4}$$

$$K = Conv_{1\times1}(F) \tag{5}$$

$$V = Conv_{1\times1}(F) \tag{6}$$

where F denotes the feature vector of size m × n, $Conv_{1\times1}$ is a 1 × 1 convolution kernel, K and V are the feature vectors obtained by the 1 × 1 convolution operation, and Q is the feature vector obtained by the output of the SoftPool operation.

(2) The vector S is obtained by multiplying the vector K with the transpose vector Q^T, as shown in Equation (7):

$$S = K \times Q^T \tag{7}$$

where T is the transpose operation, × is the product operation between two vectors, and S denotes the matrix vector of the multiplication of K and Q^T.

2.2.4. Model generalization enhancement based on self-attention convolution

The self-attention mechanism has weak generalization owing to the lack of inductive bias (Dai et al., 2021). In contrast, convolution has good generalization ability owing to its convolution kernel, which is static and possesses translational invariance. To this end, we introduce the mobile inverted bottleneck convolution (MBConv) (Sandler et al., 2018), which is currently the most advanced convolution, in the self-attention mechanism to enhance the generalization (Figure 2B). The main principle of MBConv is that the input features are first up-dimensioned using 1 × 1 convolution, and then the information between their length and width is extracted by depth-separable convolution. The dimensionalized input feature information is downscaled by point convolution to obtain information across channels. A linear activation function is adopted in the dimensionality reduction process to prevent information loss. To prevent network degradation, a reversal residual block is added at the end to sum the reduced-dimensional features with the input features, which significantly improves the generalization performance of the model.

The process is divided into two steps.

(1) Input the vector S into *MBConv* (Sandler et al., 2018) and use the *SoftMax* function for scaling and normalization to obtain the attention weight values, as follows:

$$beta = Softmax\left(\frac{MBConv(S)}{\sqrt{d_k}}\right) \tag{8}$$

(2) Take this attention weight value and multiply it with the V vector to obtain the result of the self-attention calculation O :

$$O = beta \times V \tag{9}$$

where *beta* denotes the attention weights obtained by passing the S matrix through the *SoftMax* function, *SoftMax* is the activation function, and $\sqrt{d_k}$ is used to prevent the S value from being too large when the dimensionality is large.

We combine ResNest with the multi-view features (F) obtained by the adaptive pooling method with the result of the self-attention calculation (O) to finally obtain the refined features (Y) extracted by the SoftPool attention convolution method:

$$Y = F + gamma * O \tag{10}$$

where *gamma* is the parameter, and Y denotes the refined features.

2.3. Feature fusion classification

In this section, we describe the multi-view feature fusion classification module. It is shown in Figure 3. For the refined features (Y) obtained from the above equation, *Maxpooling* is utilized to aggregate the features and thus generate a compact global descriptor (*Global*), as shown in Equation (11). The 1 × 1 convolution allows the number of channels to be reduced by controlling the number of convolution kernels, and it does not limit the size of the input features. Therefore, we input the generated global descriptor (*Global*) to the 1 × 1 convolution to obtain the result of the 3D model classification, as shown in Equation (12).

$$Global = Max(Y) \tag{11}$$

$$Z = Conv_{1\times1}(Global) \tag{12}$$

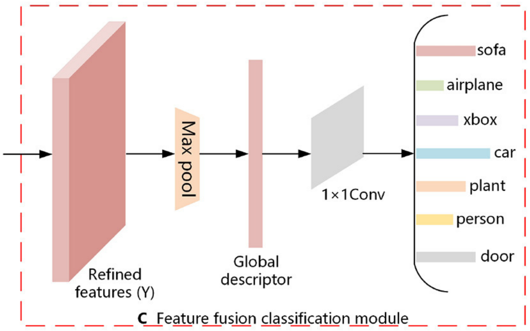


FIGURE 3
Feature fusion classification.

where Z denotes the result of the determination of N classes of objects, Max denotes the pooling aggregation operation, $Global$ denotes the resulting global descriptor, and $Conv_{1\times1}$ denotes the convolution operation with a 1 × 1 convolution kernel.

3. Experiment

3.1. Datasets

To evaluate the performance of our proposed MVMSAN network, we conducted extensive classification comparison experiments using the ModelNet40 and ModelNet10 datasets. ModelNet40 includes 3D CAD models in 40 common grid forms, including 9,843 training models and 2,468 testing models. ModelNet10 contains 10 categories of 3D CAD models, with 3,991 training models and 908 testing models.Since the number of models varies across categories, we chose the overall accuracy OA (Uy et al., 2019; Equation 13) for each sample and the average accuracy AA (Zhai et al., 2020) (Equation 14) for each category as metrics to evaluate the classification performance. It is noteworthy that OA is the ratio of the number of correctly classified samples to the total number of samples, and AA is the average of the ratio of the number of correct predictions to the total number of predictions for each category. See Equations (13) and (14) for details.

$$OA = \frac{1}{N}\sum_{i=1}^{c} x_{ii} \tag{13}$$

$$AA = \frac{sum(recall)}{C} \tag{14}$$

where N is the total number of samples, x_{ii} is the number of correct classifications, and C denotes the category of the dataset, and recall denotes the ratio of predictions to samples.

3.2. Experimental setup and analysis

We conducted our experiments using a computer with Windows 10, Inter 8700K CPU, 64 GB RAM, and the RTX2080 graphics card. In all experiments, our environment was set to PyTorch 1.2 (Paszke et al., 2017) and Cuda 10.0. The experiment was divided into two training phases. The first phase classified only a single view to enable fine-tuning of the model while removing the SoftPool attention convolution module. The second stage added SoftPool attention convolutional blocks to train all views of the 3D model, which was used to train the whole classification framework. We only performed test experiments in the second stage and set 20 epochs. We optimized the entire network architecture using the Adam (Zhang, 2018) optimizer. The initial learning rate and L2 regularization weight decay parameters were set to 0.0001 and 0.001, respectively, to accelerate model convergence and reduce model overfitting.

3.3. Impact of CNN on classification performance

A pretrained CNN is used as a backbone model to improve the performance of various tasks, e.g., classification and segmentation. To extract view feature information more quickly and effectively, we connected the SoftPool attention convolution module to the encoders, such as ResNet18 (He et al., 2016), Densenet121 (Huang et al., 2017), ResNest50d, ResNest26d, and ResNest14d, in the ModelNet40 and ModelNet10 datasets. The experimental results are shown in Table 1. On the ModelNet40, the whole network had the shortest training time when using ResNet18, while the network deepened and the training time prolonged when using DenseNet121 and ResNest50d. In particular, the training process of the ResNest50d network model took 809 min (312 min more than ResNest14d). Employing ResNest14d as the backbone model, the OA and AA metrics of the MVMSAN network reached 96.96% and 95.68%, respectively, achieving the best classification performance. Hence, we chose ResNest14d as the backbone model for extracting multi-view features.

3.4. The effect of different number of views on classification performance

To more intuitively observe the view feature information in different angles, we selected 2D views of seven different categories of 3D models for display. As shown in Figure 4, the view V in the piano category ignores the key feature information of the keys; therefore, if a single view is used for experiments, the loss of feature information will affect the classification accuracy. Multiple views can fuse the feature information of different

TABLE 1 Effects of different backbone models on classification performance.

Network	ModelNet40			ModelNet10		
	Tim(min)	OA(%)	AA(%)	Tim(min)	OA(%)	AA(%)
Resnet18	**366**	96.31	94.43	**147**	98.45	98.22
Densenet121	748	96.59	94.81	290	98.23	97.98
ResNest50d	809	96.31	94.22	327	98.24	98.07
ResNest26d	599	96.72	95.33	239	**98.67**	**98.45**
ResNest14d	497	**96.96**	**95.68**	200	98.57	98.42

The bold values represent the best performance.

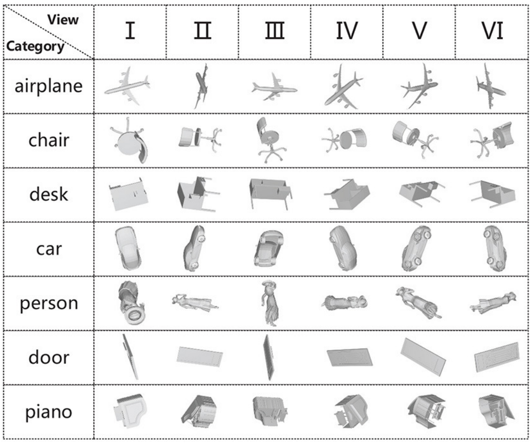


FIGURE 4
Six views of different models.

views to make up for the loss of single view feature information. To further investigate the effect of the number of views on the model classification performance, we randomly selected 3, 6, and 12 views from the 20 views obtained from 20 viewpoint angles for each 3D model in experiments. At the same time, the classification performance of MVMSAN was also compared with other advanced methods [such as MVCNN (Su et al., 2015), RCPCNN (Wang C. et al., 2019), 3D2SeqViews (Han et al., 2019a), VERAM (Chen et al., 2019), MHBN (Yu et al., 2018), and RN (Yang and Wang, 2019)] under 3, 6, and 12 number of views. The experimental results are shown in Table 2.

On the ModelNet40 dataset, MVMSAN network outperformed other methods (such as MVCNN, RCPCNN, 3D2SeqViews, VERAM, MHBN, and RN). Compared with the RN network, our network improved OA by 3.0, 2.8, and 2.6% in each view configuration. In comparison with the classic MVCNN network, it improved by 5.2, 5.0, and 5.5%, respectively. From Table 2, we can see that the classification accuracy did not increase with the number of views; for example, our method achieved the best experimental results in six views. Meanwhile, it can be seen from the Table 2 that OA of our MVMSAN model can still reached 96.35, 96.84, and 96.80% in 3, 6, and 12 views. This experiment shows that our network has high robustness.

TABLE 2 The effect of the number of views on classification performance.

Methods	ModelNet40		
	3 views	6 views	12 views
MVCNN	91.33	92.01	91.49
RCPNN	92.10	92.22	92.18
3D2SeqViews	92.10	93.07	93.40
VERAM	92.40	93.30	93.70
MHBN	93.78	94.12	93.42
RN	93.50	94.10	94.30
MVMSAN(Ours)	**96.35**	**96.84**	**96.80**

The bold values represent the best performance.

TABLE 3 Ablation study (ModelNet40).

ATT	Soft	MBConv	OA(%)	AA(%)
✓			96.43	94.70
✓	✓		96.11	94.47
✓		✓	96.40	94.62
✓	✓	✓	**96.96**	**95.68**

ATT represents attention calculation, Soft represents the SoftPool method, and MBConv represents the mobile inverted bottleneck convolution. The bold values represent the best performance.

TABLE 4 Ablation study (ModelNet10).

ATT	Soft	MBConv	OA(%)	AA(%)
✓			98.34	98.20
✓	✓		98.23	97.98
✓		✓	98.24	97.99
✓	✓	✓	**98.57**	**98.42**

ATT represents attention calculation, Soft represents the SoftPool method, and MBConv represents the mobile inverted bottleneck convolution. The bold values represent the best performance.

The high robustness achieved by the MVMSAN model is mainly attributed to our proposed SoftPool attention convolution method. SoftPool uses the processed view feature value as the Query value of the self-attended to obtain refined view feature information. Under any number of 1–20 views, these fine-grained view features can hold salient features related to model categories. Subsequent Mobile inverted bottleneck convolution (MBConv) can process the Query and Key of self-attentive, which significantly improve the generalization performance of MVMSAN model. The learning ability for our model also becomes stronger, so that it can achieve high classification accuracy with any number of 1–20 views.

TABLE 5 Comparison of the effect of 1 × 1 convolution on classification performance.

Network	OA(%)	AA(%)	Time(min)
FC	96.79	95.20	524
1 × 1Conv	**96.96**	**95.68**	**497**

The bold values represent the best performance.

3.5. Ablation experiments

We supplement a set of ablation experiments to demonstrate the generalization performance of SoftPool attentional convolution method proposed by us (see Tables 3, 4). The experimental results on the ModelNet40 dataset show that our proposed SoftPool attentional convolution method achieved the best classification performance on ModelNet40 (96.96% for OA and 95.68% for AA). The OA and AA obtained by applying only the output of SoftPool as the Query vector of attention were 96.11 and 94.47%, respectively, which were lower than those of the SoftPool attention convolution method. This is because the network model at this point is less generalizable, i.e., the classification ability learned by this network from the training set performs poorly. Adopting only MBConv to process the computational results of Query and Key of attention led to an insufficient feature extraction capability of the network. The loss

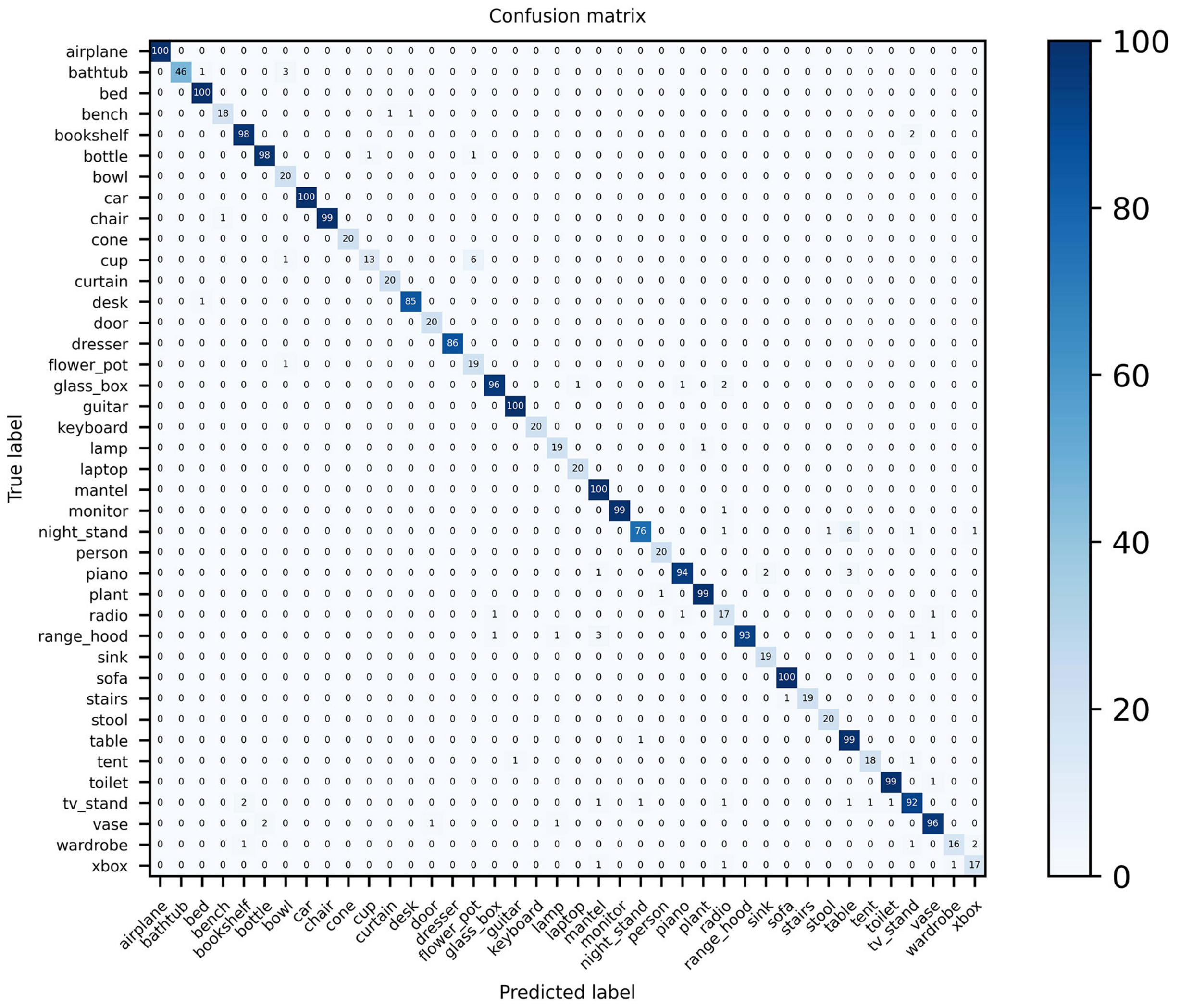
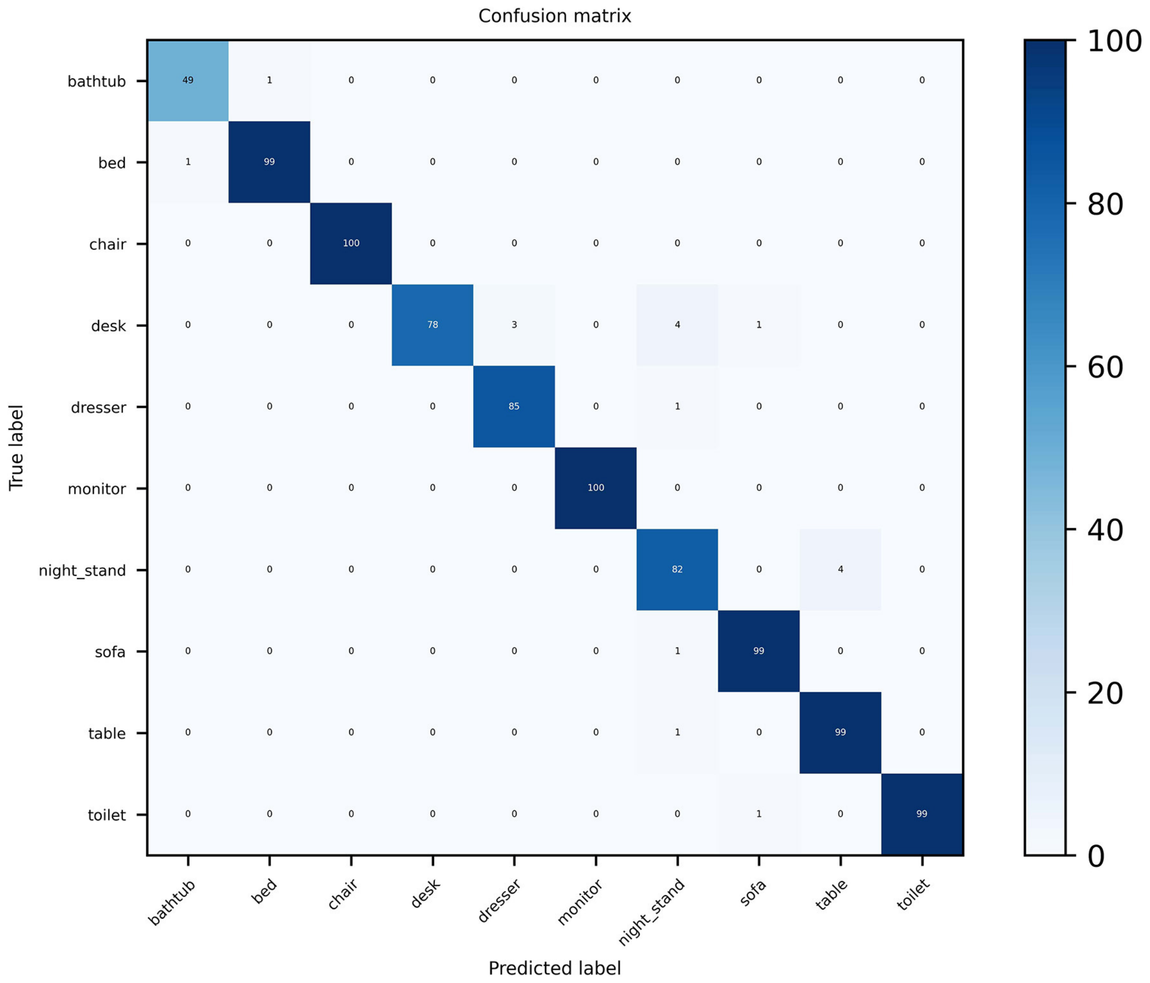


FIGURE 5
Confusion matrix visualization of MVMSAN on ModelNet40.



True label \ Predicted label	bathtub	bed	chair	desk	dresser	monitor	night_stand	sofa	table	toilet
bathtub	49	1	0	0	0	0	0	0	0	0
bed	1	99	0	0	0	0	0	0	0	0
chair	0	0	100	0	0	0	0	0	0	0
desk	0	0	0	78	3	0	4	1	0	0
dresser	0	0	0	0	85	0	1	0	0	0
monitor	0	0	0	0	0	100	0	0	0	0
night_stand	0	0	0	0	0	0	82	0	4	0
sofa	0	0	0	0	0	0	1	99	0	0
table	0	0	0	0	0	0	1	0	99	0
toilet	0	0	0	0	0	0	0	1	0	99

FIGURE 6
Confusion matrix visualization of MVMSAN on ModelNet10.

of this feature information further reduced the classification accuracy (96.40 and 94.62% for OA and AA, respectively). We also obtained consistent experimental results on the ModelNet10 dataset (see Table 4).

It further proves that the best performance of the entire model can be achieved with the output result of SoftPool as the Query value of attention and MBConv to process the computational results of Query and Key of attention. It is worth noting that our algorithm can achieve 96.96% on OA and 95.68% on AA. The result is closely related to the refined feature extraction of SoftPool self-attention method and the model generalization enhancement of self-attention convolution method. The above two factors are indispensable.

We also employed a 1 × 1 convolution alternative to the fully connected layer that the network ends up using for classification. As shown in Table 5, the OA and AA using 1 × 1 convolution reached 96.96 and 95.68%, respectively, which is 0.17 and 0.48% improvement compared with fully connected layers. By using 1 × 1 convolution with fewer parameters, the training time in the same environment was also reduced by 27 min.

3.6. Confusion matrix visualization

Confusion matrix visualization can intuitively demonstrate the advanced performance of the MVMSAN method on the 3D model classification task. Especially in the case that some view features have high similarity, our method still has high classification prediction performance. We plot the confusion matrix on the ModelNet40 and ModelNet10 datasets. On ModelNet40, it can be seen from Figure 5 that MVMSAN achieved 100% classification accuracy on categories such as airplane, bed, sofa, and guitar. In some harder categories, such as night stand, table, and xbox, some views have high similarity.

TABLE 6 Classification performance comparison with other methods.

Network	Modality	ModelNet40		ModelNet10	
		OA(%)	AA(%)	OA(%)	AA(%)
3D ShapeNets	Voxel	84.70	77.30	-	83.54
VoxNet	Voxel	85.90	83.00	-	92.00
Pointgrid	Voxel	92.0	88.90	-	-
PointNet	Point Cloud	89.20	86.20	-	-
PointNet++	Point Cloud	91.90	-	-	-
Mo-Net	Point Cloud	92.40	90.30	-	-
DGCNN	Point Cloud	93.50	90.70	-	-
MVCNN	12-Views	92.10	89.90	-	-
GVCNN	12-Views	92.6	-	-	-
MHBN	6-Views	94.12	92.20	95.00	95.00
	12-Views	93.42	-	-	-
RN	6-Views	94.10	-	-	-
	12-Views	94.30	92.30	95.30	95.10
HMVCM	12-Views	94.57	-	95.7	-
MVMSAN (Ours)	3-Views	96.35	94.62	97.80	97.65
	6-Views	96.84	95.65	98.56	**98.50**
	12-Views	96.80	95.31	98.57	98.37
	20-Views	**96.96**	**95.68**	**98.57**	98.42

The bold values represent the best performance.

In this case, our MVMSAN model can also classify correctly. It can be seen from Figure 5 that 76 samples are correctly classified among the 86 the night stand models.

For the ModelNet10 dataset, it can be seen from Figure 6 that our MVMSAN also achieved 100% classification accuracy on the chair and monitor categories. In some views, desk, dresser, sofa and other 3D models have high similarity. The existing networks will confuse the feature information of 3D models and cause classification errors. However, our MVMSAN model still has high classification performance for this situation. For example, 78 samples are correctly classified among the 86 the desk models in Figure 6.

The data in the figure is enough to demonstrate the superiority of our approach on the model classification task. Especially for view features with high similarity, our network model is still able to achieve high classification prediction performance.

3.7. Comparison with other methods

We compared the classification performance of voxel-based methods [3DShapeNets (Wu et al., 2015), VoxNet (Maturana and Scherer, 2015), and Pointgrid (Le and Duan, 2018)], point cloud-based methods [PointNet (Qi et al., 2017a), PointNet++ (Qi et al., 2017b), MO-Net (Joseph-Rivlin et al., 2019) and DGCNN (Wang Y. et al., 2019) and view-based methods [MVCNN (Su et al., 2015)], GVCNN (Feng et al., 2018), MHBN (Yu et al., 2018), RN (Yang and Wang, 2019), and HMVCN (Liu et al., 2021)]

As shown in Table 6, the proposed MVMSAN outperformed other deep learning methods. Compared with the most classical multi-view-based model classification method (MVCNN), MVMSAN improved OA and AA by 5 and 6%, respectively. Compared with the GVCNN, MHBN, and RN methods, MVMSAN showed considerable improvement. HMVCN is a recently proposed model classification method based on bidirectional LSTM, and its OA reached 94.57%. Our method achieved 2.5% higher OA compared to HMVCN. On the ModelNet10 dataset, the MVMSAN method also achieved the best classification performance (98.57% for OA and 98.42% for AA).

The excellent performance of our MVMSAN method on the two ModelNet datasets is attributed to three factors: (1) ResNest removes the last fully connected layer and adds an adaptive pooling layer. It can prove that the relationship between view channels can increase the receptive field of view feature extraction, so that the network obtains more detailed features from the input data related to the output. (2) Using the output result of SoftPool as the Query vector of attention can realize the refined down-sampling processing of view feature information, and effectively solve the problem of insufficient extraction and loss of detailed information in the process of view feature extraction. (3) MBConv is employed to process the calculation results of Query and Key of attention. It can enhance the generalization of the model, thereby improving the classification accuracy.

4. Conclusion

In this paper, we proposed a multi-view SoftPool attention convolutional network framework, MVMSAN, for 3D model classification. The traditional method does not treat each view equally in the view feature extraction process, and only extracts the feature information that is considered important. This causes the problem of insufficient extraction of the view refinement feature information and loss. Our proposed SoftPool attention convolution framework could achieve refined down-sampling processing for all view features equally, thereby obtaining more useful information from the input data related to the output results, improving the generalization of the model, and achieving high-precision 3D model classification. To better evaluate our network framework, we conducted several experiments to validate the impact of each component of the framework. The experimental results demonstrate that our framework has achieved better classification accuracy on the ModelNet40 and ModelNet10 datasets compared to other advanced methods.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: http://modelnet.cs.princeton.edu/.

Author contributions

WW and XW brought up the core concept and architecture of this manuscript. XW wrote the paper. GC and HZ corrected the sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

Funding

This study was supported by the Natural Science Foundation of Shanghai under Grant No. 19ZR1435900.

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

Adikari, S. B., Ganegoda, N. C., Meegama, R. G., and Wanniarachchi, I. L. (2020). Applicability of a single depth sensor in real-time 3d clothes simulation: augmented reality virtual dressing room using kinect sensor. *Adv. Hum. Comput. Interact*. 2020, 1314598. doi: 10.1155/2020/1314598

Chen, S., Zheng, L., Zhang, Y., Sun, Z., and Xu, K. (2019). Veram: view-enhanced recurrent attention model for 3D shape classification. *IEEE Trans. Vis. Comput. Graph*. 25, 3244–3257. doi: 10.1109/TVCG.2018.2866793

Dai, Z., Liu, H., Le, Q. V., and Tan, M. (2021). "Coatnet: marrying convolution and attention for all data sizes," in *Advances in Neural Information Processing Systems, Vol. 34*, eds M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan (Montreal: Curran Associates, Inc.), 3965–3977

Feng, M., Zhang, L., Lin, X., Gilani, S. Z., and Mian, A. (2020). Point attention network for semantic segmentation of 3D point clouds. *Pattern Recognit*. 107, 107446. doi: 10.1016/j.patcog.2020.107446

Feng, Y., Zhang, Z., Zhao, X., Ji, R., and Gao, Y. (2018). "Gvcnn: group-view convolutional neural networks for 3D shape recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Salt Lake City, UT: IEEE).

Grenzdörffer, T., Günther, M., and Hertzberg, J. (2020). "Ycb-m: a multi-camera rgb-d dataset for object recognition and 6D of pose estimation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)* (Paris: IEEE), 3650–3656.

Han, Z., Lu, H., Liu, Z., Vong, C.-M., Liu, Y.-S., Zwicker, M., et al. (2019a). 3d2seqviews: aggregating sequential views for 3d global feature learning by cnn with hierarchical attention aggregation. *IEEE Trans. Image Process*. 28, 3986–3999. doi: 10.1109/TIP.2019.2904460

Han, Z., Shang, M., Liu, Z., Vong, C.-M., Liu, Y.-S., Zwicker, M., et al. (2019b). 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. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, NV: IEEE).

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., et al. (2020). "Randla-net: efficient semantic segmentation of large-scale point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Seattle, WA: IEEE).

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. (2017). "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI: IEEE).

Joseph-Rivlin, M., Zvirin, A., and Kimmel, R. (2019). "Momen(e)t: Flavor the moments in learning to classify shapes," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops* (Seoul: IEEE).

Kanezaki, A., Matsushita, Y., and Nishida, Y. (2018). "Rotationnet: joint object categorization and pose estimation using multiviews from unsupervised viewpoints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Salt Lake City, UT: IEEE).

Kästner, L., Frasineanu, V. C., and Lambrecht, J. (2020). "A 3D-deep-learning-based augmented reality calibration method for robotic environments using depth sensor data," in *2020 IEEE International Conference on Robotics and Automation (ICRA)* (Paris: IEEE), 1135–1141.

Lalonde, J., Unnikrishnan, R., Vandapel, N., and Hebert, M. (2005). "Scale selection for classification of point-sampled 3D surfaces," in *Fifth International Conference on 3-D Digital Imaging and Modeling (3DIM'05)* (Ottawa), 285–292.

Le, T., and Duan, Y. (2018). "Pointgrid: a deep network for 3D shape understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Salt Lake City, UT: IEEE).

Liang, Z., Yang, M., Deng, L., Wang, C., and Wang, B. (2019). "Hierarchical depthwise graph convolutional neural network for 3D semantic segmentation of point clouds," in *2019 International Conference on Robotics and Automation (ICRA)* (Montreal), 8152–8158.

Liu, A.-A., Zhou, H., Nie, W., Liu, Z., Liu, W., Xie, H., et al. (2021). Hierarchical multi-view context modelling for 3D object classification and retrieval. *Inf. Sci.* 547, 984–995. doi: 10.1016/j.ins.2020.09.057

Liu, C.-H., Lee, P., Chen, Y.-L., Yen, C.-W., and Yu, C.-W. (2020). Study of postural stability features by using kinect depth sensors to assess body joint coordination patterns. *Sensors* 20, 1291. doi: 10.3390/s20051291

Ma, C., Guo, Y., Yang, J., and An, W. (2019). Learning multi-view representation with lstm for 3D shape recognition and retrieval. *IEEE Trans. Multimedia* 21, 1169–1182. doi: 10.1109/TMM.2018.2875512

Ma, Y., Guo, Y., Lei, Y., Lu, M., and Zhang, J. (2018). "3dmax-net: a multi-scale spatial contextual network for 3D point cloud semantic segmentation," in *2018 24th International Conference on Pattern Recognition (ICPR)* (Beijing), 1560–1566.

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.

Niemeyer, J., Rottensteiner, F., and Soergel, U. (2014). Contextual classification of lidar data and building object detection in urban areas. *ISPRS J. Photogram. Remote Sens.* 87, 152–165. doi: 10.1016/j.isprsjprs.2013.11.001

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., et al. (2017). Automatic differentiation in pytorch.

Pontes, J. K., Kong, C., Eriksson, A. P., Fookes, C., Sridharan, S., and Lucey, S. (2017). Compact model representation for 3D reconstruction. *CoRR*, abs/1707.07360. doi: 10.1109/3DV.2017.00020

Qi, C. R., Su, H., Mo, K., and Guibas, L. J. (2017a). "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu: IEEE).

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017b). "Pointnet++: deep hierarchical feature learning on point sets in a metric space," in *Advances in Neural Information Processing Systems, Vol. 3*, eds I. Guyon, U. Luxburg, V. S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Long Beach, CA: Curran Associates, Inc.).

Qiu, S., Anwar, S., and Barnes, N. (2021). "Dense-resolution network for point cloud classification and segmentation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (Waikoloa: IEEE), 3813–3822.

Riegler, G., Osman Ulusoy, A., and Geiger, A. (2017). "Octnet: learning deep 3d representations at high resolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI: IEEE).

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). "Mobilenetv2: inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Salt Lake City, UT: IEEE).

Stergiou, A., Poppe, R., and Kalliatakis, G. (2021). "Refining activation downsampling with softpool," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (Montreal: IEEE), 10357–10366.

Su, H., Maji, S., Kalogerakis, E., and Learned-Miller, E. (2015). "Multi-view convolutional neural networks for 3D shape recognition," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (Santiago: IEEE).

Sun, K., Zhang, J., Liu, J., Yu, R., and Song, Z. (2021). Drcnn: dynamic routing convolutional neural network for multi-view 3D object recognition. *IEEE Trans. Image Process.* 30, 868–877. doi: 10.1109/TIP.2020.3039378

Uy, M. A., Pham, Q.-H., Hua, B.-S., Nguyen, T., and Yeung, S.-K. (2019). "Revisiting point cloud classification: a new benchmark dataset and classification model on real-world data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (Seoul: IEEE).

Wang, C., Pelillo, M., and Siddiqi, K. (2019). Dominant set clustering and pooling for multi-view 3D object recognition. *CoRR*, abs/1906.01592. doi: 10.48550/arXiv.1906.01592

Wang, L., Huang, Y., Shan, J., and He, L. (2018). Msnet: multi-scale convolutional network for point cloud classification. *Remote Sens.* 10, 612. doi: 10.3390/rs10040612

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Seattle, WA: IEEE).

Wen, X., Han, Z., Liu, X., and Liu, Y.-S. (2020). Point2spatialcapsule: aggregating features and spatial relationships of local regions on point clouds using spatial-aware capsules. *IEEE Trans. Image Process.* 29, 8855–8869. doi: 10.1109/TIP.2020.3019925

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., et al. (2015). "3D shapenets: a deep representation for volumetric shapes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Boston: IEEE).

Yang, S., Xu, M., Xie, H., Perry, S., and Xia, J. (2021). "Single-view 3D object reconstruction from shape priors in memory," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Nashville: IEEE), 3152–3161.

Yang, Z., and Wang, L. (2019). "Learning relationships for multi-view 3D object recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (Seoul: IEEE).

Yu, D., Ji, S., Liu, J., and Wei, S. (2021). Automatic 3D building reconstruction from multi-view aerial images with deep learning. *ISPRS J. Photogram. Remote Sens.* 171, 155–170. doi: 10.1016/j.isprsjprs.2020.11.011

Yu, T., Meng, J., and Yuan, J. (2018). "Multi-view harmonized bilinear network for 3D object recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Salt Lake City, UT: IEEE).

Zhai, R., Li, X., Wang, Z., Guo, S., Hou, S., Hou, Y., et al. (2020). Point cloud classification model based on a dual-input deep network framework. *IEEE Access* 8, 55991–55999. doi: 10.1109/ACCESS.2020.2981357

Zhang, H., Goodfellow, I., Metaxas, D., and Odena, A. (2019). "Self-attention generative adversarial networks," in *Proceedings of the 36th International Conference on Machine Learning, volume 97 of Proceedings of Machine Learning Research*, eds K. Chaudhuri and R. Salakhutdinov (Long Beach, CA: PMLR), 7354–7363.

Zhang, H., Wu, C., Zhang, Z., Zhu, Y., Lin, H., Zhang, Z., et al. (2022). "Resnest: split-attention networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops* (New Orleans, LA: IEEE), 2736–2746.

Zhang, W., Su, S., Wang, B., Hong, Q., and Sun, L. (2020). Local K-NNS pattern in omni-direction graph convolution neural network for 3D point clouds. *Neurocomputing* 413, 487–498. doi: 10.1016/j.neucom.2020.06.095

Zhang, Z. (2018). "Improved adam optimizer for deep neural networks," in *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)* (Banff, AB: IEEE), 1–2.

Zou, W., Wu, D., Tian, S., Xiang, C., Li, X., and Zhang, L. (2021). End-to-end 6dof pose estimation from monocular rgb images. *IEEE Trans. Consum. Electron.* 67, 87–96. doi: 10.1109/TCE.2021.3057137

frontiers | Frontiers in Neurorobotics

TYPE Original Research
PUBLISHED 16 November 2022
DOI 10.3389/fnbot.2022.1050981

OPEN ACCESS

EDITED BY
Xin Jin,
Yunnan University, China

REVIEWED BY
Jiao Du,
Guangzhou University, China
B. Rajalingam,
Marian Engineering College, India

*CORRESPONDENCE
Weiwei Kong
kwwking@163.com

RECEIVED 22 September 2022
ACCEPTED 28 October 2022
PUBLISHED 16 November 2022

CITATION
Kong W, Li C and Lei Y (2022) Multimodal medical image fusion using convolutional neural network and extreme learning machine. *Front. Neurorobot.* 16:1050981. doi: 10.3389/fnbot.2022.1050981

Multimodal medical image fusion using convolutional neural network and extreme learning machine

Weiwei Kong[1,2,3]*, Chi Li[1,2,3] and Yang Lei[4]

[1]School of Computer Science and Technology, Xi'an University of Posts and Telecommunications, Xi'an, China, [2]Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processing, Xi'an, China, [3]Xi'an Key Laboratory of Big Data and Intelligent Computing, Xi'an, China, [4]College of Cryptography Engineering, Engineering University of PAP, Xi'an, China

The emergence of multimodal medical imaging technology greatly increases the accuracy of clinical diagnosis and etiological analysis. Nevertheless, each medical imaging modal unavoidably has its own limitations, so the fusion of multimodal medical images may become an effective solution. In this paper, a novel fusion method on the multimodal medical images exploiting convolutional neural network (CNN) and extreme learning machine (ELM) is proposed. As a typical representative in deep learning, CNN has been gaining more and more popularity in the field of image processing. However, CNN often suffers from several drawbacks, such as high computational costs and intensive human interventions. To this end, the model of convolutional extreme learning machine (CELM) is constructed by incorporating ELM into the traditional CNN model. CELM serves as an important tool to extract and capture the features of the source images from a variety of different angles. The final fused image can be obtained by integrating the significant features together. Experimental results indicate that, the proposed method is not only helpful to enhance the accuracy of the lesion detection and localization, but also superior to the current state-of-the-art ones in terms of both subjective visual performance and objective criteria.

KEYWORDS

image fusion, modality, multimodal medical image, convolutional neural network, extreme learning machine

Introduction

As is well known, the accuracy of lesion detection and localization is crucial during the whole clinical diagnosis and treatment. So far, the rapid growth of medical imaging technologies such as computed tomography (CT), magnetic resonance imaging (MRI), positron emission tomography (PET) and single-photon emission computed tomography (SPECT) has provided us much richer information on the physical condition. CT can accurately detect the slight differences of the bone density in a transection plane, which is regarded as an ideal way to observe the lesions of the bone. Nevertheless, its capacity of the tissue characterization is weak. The information of the soft tissue can be better visualized in MRI images, but the movement information such as the body metabolism cannot be found. Unlike MRI, PET images can reflect the activity of the life metabolism through the accumulation of certain substance so as to achieve the purpose of diagnosis, but they are often with lower resolution. The main advantage of SPECT is to demonstrate the changes in blood flow, function and the metabolism of organs or diseases, which is beneficial to the early and specific diagnosis of the disease. Obviously, due to the respective different mechanism, each imaging modality unavoidably has its characteristics and inherent drawbacks. To this end, the fusion of the medical images with multiple different modalities may be an effective solution, because it can not only combine the advantages together to accurately implement the localization and description of the lesion, but also reduce the storage cost of the patient information database.

Recently, a variety of fusion methods on multimodal medical images have been proposed during the past decades. Basically, these methods can be mainly grouped into the following categories, namely spatial domain-based methods, transform domain-based methods, soft computing-based methods, and deep learning-based ones.

The representative spatial domain-based methods include simple averaging, maximum choosing, principal component analysis (PCA) (He et al., 2010) and so on. Although most of the above methods have comparatively high operating speed and simple framework, they often tend to suffer from contrast reduction and spectrum distortion in the final fused image. Therefore, the pure spatial domain-based methods are rarely used at present.

Unlike spatial domain-based methods, the core scheme of transform domain-based methods usually consists of three steps. Firstly, the source image is converted to the frequency domain to get several sub-images which commonly contain one approximation image with low-pass coefficients and several detail images with high-pass coefficients. Secondly, certain rules are adopted to complete the fusion of sub-images at corresponding stages. Finally, the final fused image is reconstructed. The classical methods include, but are not limited to, Laplacian pyramid transform, discrete wavelet transform (DWT), contourlet transform, shearlet transform and so on, which have pioneered the use of transform domain-based concept. However, with further in-depth research on the medical image fusion, the defects of the above classical methods are gradually revealed. Under this background, a series of improved versions have been presented in the past decade. Du et al. (2016) introduced union Laplacian pyramid to complete the fusion of medical images. Some improved versions of DWT such as dual tree complex wavelet transform (DT-CWT) (Yu et al., 2016), non-subsampled rotated complex wavelet transform (NSRCxWT) (Chavan et al., 2017), discrete stationary wavelet transform (DSWT) (Ganasala and Prasad, 2020a; Chao et al., 2022) were presented to complete the fusion of medical images. Compared with DWT, these three new versions share both the redundancy feature and the shift-invariance property, which effectively avoid the Gibbs phenomenon in DWT. Similarly, in order to overcome the absence of shift-invariance in the original contourlet transform and shearlet transform, the corresponding improved versions namely non-subsampled contourlet transform (NSCT) and non-subsampled shearlet transform (NSST) were proposed successively. In comparison to the aforementioned transform domain-based methods, NSCT and NSST have both manifested competitive fusion performance due to their flexible structures. Zhu et al. (2019) combined NSCT, phase congruency and local Laplacian energy together to present a novel fusion method for multi-modality medical images. Liu X. et al. (2017), Liu et al. (2018) proposed two NSST-based methods to fuse the CT and MRI images.

In addition to spatial domain-based methods and transform domain-based methods, extensive work has also been conducted with soft computing-based methods dedicated to multimodal medical image fusion. A great many representative models, including dictionary learning model (Zhu et al., 2016; Li et al., 2018), gray wolf optimization (Daniel, 2018), fuzzy theory (Yang et al., 2019), pulse coupled neural network (Liu X. et al., 2016; Xu et al., 2016), sparse representation (Liu and Wang, 2015; Liu Y. et al., 2016), total variation (Zhao and Lu, 2017), guided filter (Li et al., 2019; Zhang et al., 2021), genetic algorithm (Kavitha and Thyagharajan, 2017; Arif and Wang, 2020), compressed sensing (Ding et al., 2019), structure tensor (Du et al., 2020c), local extrema (Du et al., 2020b), Otsu's method (Du et al., 2020a) and so on, were successfully used to fuse the medical images.

Since the transform domain-based methods and soft computing-based methods have both manifested to be promising in the field of medical image fusion, some novel hybrid methods have also been presented in recent years. Jiang et al. (2018) combined interval type-2 fuzzy sets with NSST to complete the fusion task of multi-sensor images. Gao et al. (2021) proposed a fusion method based on particle swarm optimization optimized fuzzy logic in NSST domain. Asha et al. (2019) constructed a novel fusion scheme based on NSST and gray wolf optimization. Singh and Anand (2020) employed NSST to decompose the source images, and then used sparse representation and dictionary learning model to fuse the sub-images. Yin et al. (2019) and Zhang et al. (2020) each proposed a NSST-PCNN based fusion method for medical images. The guided filter was combined with NSST to deal with the issue of multi sensor image fusion (Ganasala and Prasad, 2020b). Zhu et al. (2022) combined the advantages of both spatial domain and transform domain methods to construct an efficient hybrid image fusion method. Besides, the collective view of the applicability and progress of information fusion techniques in medical imaging were reviewed respectively in Hermessi et al. (2021) and Azam et al. (2022).

In recent years, the deep learning-based methods play significant roles in the field of medical image fusion, and have been gaining more and more popularity in both the academic and industry community. In 2017, convolutional neural network (CNN) was firstly introduced into the area of image fusion by Liu Y. et al. (2017). Fan et al. (2019) deeply researched the semantic information of the medical image with different modalities, and proposed a semantic-based fusion method for medical images. Aside from CNN, another representative deep learning model namely generative adversarial network (GAN) was used to deal with the issue of image fusion in 2019 (Ma et al., 2019). The unsupervised deep networks for medical image fusion were presented in references (Jung et al., 2020; Fu et al., 2021; Xu and Ma, 2021; Shi et al., 2022). Goyal et al. (2022) combined transform domain-based methods and deep learning-based methods together to present a composite method for image fusion and denoising.

After consulting a great deal of literature, we found that how much information from the original source medical images is retained in the final fused image greatly determines the image quality, which is crucial to the further clinical diagnosis and treatment. So far, the single transformed domain-based methods and relevant hybrid ones have been widely employed to deal with the fusion issue of medical images. However, the transformed domain-based methods may introduce the frequency distortion into the fused image. With the rapid development of the deep learning theory and its reasonable biological background, more and more attention is being paid to the deep learning-based methods such as CNN. Therefore, we desire to develop a novel fusion method based on CNN to fuse the medical images. It is noteworthy that each single theory always has its advantages and disadvantages and deep learning is no exception, which is usually accompanied by a huge amount of computational costs. To this end, we need to construct or adopt some model to reduce the computational complexity as much as possible.

In this paper, a novel fusion method on the multimodal medical images exploiting CNN and extreme learning machine (ELM) (Huang et al., 2006, 2012; Feng et al., 2009) is proposed. On the one hand, since the nature of the medical image fusion can be regarded as the classification problem, the existing successful experiences of CNN can be fully applied. On the other hand, due to a great many parameters, the computational cost of CNN is high. ELM is a single hidden layer feed-forward network, and its algorithm complexity is very low. Besides, since ELM belongs to a convex optimization problem, it will not fall into the local optimum. Therefore, ELM is utilized to improve the traditional CNN model in this paper.

The main contributions of this paper can be summarized as follows.

- A novel method based on CNN and ELM is proposed to deal with the fusion issue of multimodal medical images.
- It is proved that, apart from the area of multi-focus image fusion, the CNN model can also be used in the field of multimodal medical image fusion.
- The traditional CNN model is integrated with ELM to be a modified version called convolutional extreme learning machine (CELM) which has not only much better performance, but also much faster running speed.
- Experimental results demonstrate that the proposed method has obvious superiorities over the current typical ones in terms of both gray image fusion and color image fusion, which is beneficial to obviously enhancing the precision of disease detection and diagnosis directly.

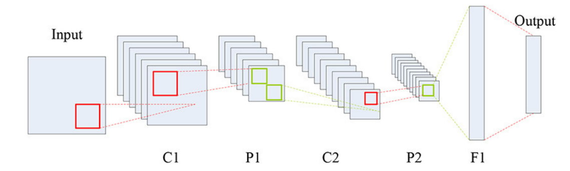


FIGURE 1
Typical CNN structure.

The rest of this paper is organized as follows. The involved theories of CNN and ELM are reviewed in Related work section followed by the proposed multimodal medical image fusion framework in Proposed method section. Experimental results with relevant analysis are reported in fourth section. In Conclusions section, the concluding remarks are given in the end.

Related work

The models relevant to the proposed method are introduced in this section. The two important concepts, namely CNN and ELM are briefly reviewed as follows.

Convolutional neural network

As a representative neural network in the field of deep learning, CNN aims to learn a multistage feature representation of the input data, and each stage usually consists of a series of feature maps connected *via* different types of calculations such as convolution, pooling and full connection. As shown in Figure 1, a typical CNN structure is composed of five types of components including the input layer, convolution layers, pooling layers, full connection layer, and the output layer.

In Figure 1, C, P and F denote the convolution, max-pooling and full connection operations, respectively, which can generate a series of feature maps. Each coefficient in the feature maps is known as a neuron. Clearly, CNN is an end-to-end system. The roles of the three types of layers, namely convolution, pooling and full connection, can be summarized as feature extraction, feature selection, and the classifier.

Here, the input data is a two-dimensional image. The neurons between the adjacent stages are connected by the operations of convolution and pooling, so that the number of the parameters to be learned declines a lot. The mathematical expression of the convolution layer can be described as:

$$y^j = b^j + \sum_i k^{ij} * x^i \tag{1}$$

where k^{ij} and b^j are the convolution kernel and the bias, respectively. The symbol * denotes the 2D convolution. x^i is the ith input feature map and y^j is the jth output one.

In fact, during the convolution course, the non-linear activation is also conducted. The common activation functions include sigmoid function, rectified linear units (ReLU), and so on. Here, ReLU is adopted whose mathematical expression can be written as:

$$y^j = max\left(0, b^j + \sum_i k^{ij} * x^i\right) \tag{2}$$

In CNN, the convolution layer is usually followed by the pooling layer. The common pooling rules include max-pooling and average-pooling, which can select the maximum or the average value of a certain region to form new feature maps. Due to the special mechanism of the pooling layer, it can bring some desirable invariance such as translation and rotation. Moreover, it can also decrease the dimension of the feature maps which is favorable for reducing the computational costs as well as the memory consumption.

Through the alternation of multiple convolution and pooling layers, CNN relies on the full connection layer to classify the extracted features to obtain the probability distribution Y based on the input. In fact, CNN can be viewed as a converter where the original matrix X can be mapped into a new feature expression Y after multiple stages of data transformation and dimension reduction. The mathematical expression can be written as:

$$Y(i) = P(L = l_i | H_0; (k, b)) \tag{3}$$

where H_0 is the original matrix, and the training objective of CNN is to minimize the loss function $L(k, b)$. k and b are the convolution kernel and the bias, respectively, which can be updated layer by layer *via* the following equations.

$$k_i = k_i - \eta \frac{\partial E(k, b)}{\partial k_i} \tag{4}$$

$$b_i = b_i - \eta \frac{\partial E(k, b)}{\partial b_i} \tag{5}$$

$$E(k, b) = L(k, b) + \frac{\lambda}{2} k^T k \tag{6}$$

where λ and η denote the weight decay parameter and the learning rate, respectively.

According to the mechanism of CNN mentioned above, the important features of the image can be classified. Some fused methods for multi-focus images based on CNN have been published in recent years. Although CNN-based fusion methods have been gaining more and more popularity, their inherent problems such as being prone to local minima, intensive manual intervention and the waste of the computing resources still cannot be ignored.

Extreme learning machine

Different from the conventional neural networks, ELM is a single hidden layer feed-forward neural network. It is generally known that most current neural networks have many knotty drawbacks. (a) The training speed is slow. (b) It is easy for them to be trapped into the local optimum. (c) The learning rate is very sensitive to the parameters selection. Fortunately, ELM is able to generate randomly the weights between the input and the hidden layer as well as the threshold of the neuron in the hidden layer, and the weights adjustment is totally unnecessary. In other words, the optimum solution can be obtained, provided the neuron number in the hidden layer is given.

Suppose N training samples $(\mathbf{x}_i, \mathbf{t}_i)$ and a single layer feed-forward neural network with L neurons in the hidden layers and M ones in the output layers. The concrete steps of the learning *via* ELM are as follows.

Step 1: The node parameters are allocated randomly, which is independent of the input data.
Step 2: Computing the output matrix $\mathbf{h}(\boldsymbol{x}) = [g_1(\boldsymbol{x}), \ldots, g_L(\boldsymbol{x})]^T$ of the hidden layers for $\boldsymbol{x}$. Obviously, the size of $\mathbf{h}(\boldsymbol{x})$ is $N \times M$, which is the mapping result from N input data to L neurons in essence.
Step 3: Computing the output weights matrix $\boldsymbol{\beta} = [\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_L]^T$. $\boldsymbol{\beta} = \mathbf{H}^T\mathbf{T}$. $\mathbf{H} = [\mathbf{h}^T(\boldsymbol{x}_1), \ldots, \mathbf{h}^T(\boldsymbol{x}_N)]^T$, and $\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_N]^T$ is the training objective. The output weights matrix $\boldsymbol{\beta}$ can be obtained by using the regularized least squares method as follows.

$$\beta = \left(\frac{I}{C} + H^T H\right)^{-1} H^T T \tag{7}$$

where C is the regularization coefficient.

Besides, a hidden neuron of ELM can be a sub-network of several neurons. The scheme of the ELM feature mapping is shown in Figure 2.

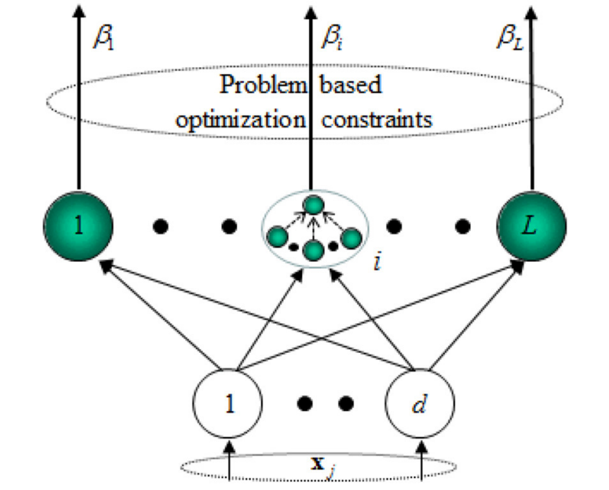


FIGURE 2
Scheme of the ELM feature mapping.

Proposed method

In this section, the proposed fusion method for multimodal medical images based on CNN and ELM is presented in detail. The concrete content can be divided into three subsections, including the structure of convolutional extreme learning machine (CELM), network design, and the fusion schemes.

Structure of CELM

As described in Related work section, we can reach several conclusions as follows.

- It is feasible to utilize CNN to deal with the issue of image fusion.
- There are still inherent drawbacks in the traditional CNN model, so it has large development potentiality.
- ELM not only owns many superiorities over other current neural networks, but also shares great similarities with CNN in structure.

Therefore, it is sensible to integrate CNN with ELM to combine the both advantages together, which may also introduce a novel and more effective solution to the fusion of multimodal medical images. To this end, the CELM model is proposed in this paper, whose structure is shown in Figure 3.

As shown in Figure 3, C and P denote the convolution and pooling operations, respectively, and the mechanism of ELM has been added into the CNN structure. CELM is composed of an input layer, an output layer, and several hidden layers where the convolution layers and the pooling layers alternately appear.

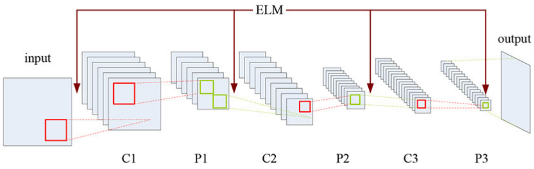


FIGURE 3
Structure of CELM.

The convolution layer consists of several maps recording the features of the previous layer *via* several different convolution kernels. The pooling layer introduces the translation invariance into the network, and the dimension of the feature map in the previous layer will also decrease. Meanwhile, the number of the feature maps in the pooling layer always equals to the one in the previous convolution layer. It is noteworthy that, except for the first convolution layer, the neurons of the feature map in the convolution layer are all connected to all the feature maps in the previous pooling layer, while the ones in the pooling layer are only connected to the corresponding feature maps in the previous convolution layer. As for the original full connection layer in the original CNN model, it has been replaced by the global average pooling layer (Lin et al., 2014), which is favorable for sharply cutting down the number of parameters.

With regard to the feature extraction, ELM can randomly generate the weights between the input layer and the first convolution layer as well as the ones between the pooling layer and the following convolution layer, as shown in Figure 3. Here, we suppose that there are two original multimodal medical images denoted by *A* and *B*, respectively. If the source images are color ones, we can convert them into gray ones or deal with them in different color spaces, which will be involved in a later section.

In CELM, the weights are viewed to be agreeing with the normal distribution, and the weight matrix can be obtained as follows.

$$P=\left[\hat{p}^{1},\hat{p}^{2},\ldots,\hat{p}^{i},\ldots\hat{p}^{N}\right],1\leq i\leq N \tag{8}$$

where **P** is the initial weight matrix, *N* is the number of convolution kernels, and the size of each element in Equation (8) is *r* × *r*. Therefore, if the size of the previous layer is *k* × *k*, the size of the corresponding feature map would be $(k - r + 1) \times (k - r + 1)$.

The convolution node on the point at (x, y) on the *i*th feature map can be obtained as

$$c_{x,y,i}\left(\Theta\right)=\sum\nolimits_{m=1}^{r}\sum\nolimits_{n=1}^{r}\Theta_{x+m-1,y+n-1}\cdot p^{i}_{m,n} \tag{9}$$

where "Θ" denotes the source image *A* or *B*.

As for the pooling layer, the max-pooling strategy is adopted except the last layer. The pooling node on the point at (u, v) on

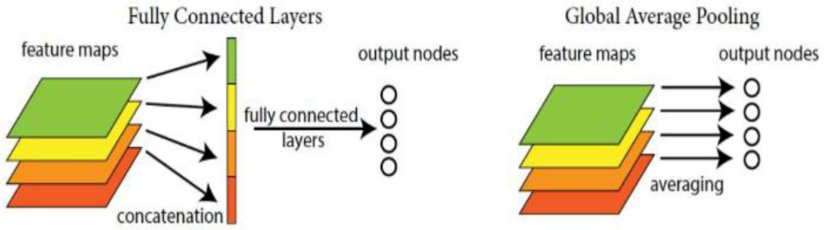


FIGURE 4
Diagram of the global average pooling layer.

the *j*th pooling map can be obtained as:

$$c_{u,v,j}(\Theta) = max\left[c_{x,y,i}\right], x, y = u - z, \ldots, u + z \qquad (10)$$

where *z* denotes the pooling size.

Due to involving a large number of parameters, the original full connection layer in CNN is substituted for the global average pooling one here, so that we can directly treat the feature maps as the category confidence ones, and save the computational costs and storage space. The diagram of the global average pooling layer is shown in Figure 4.

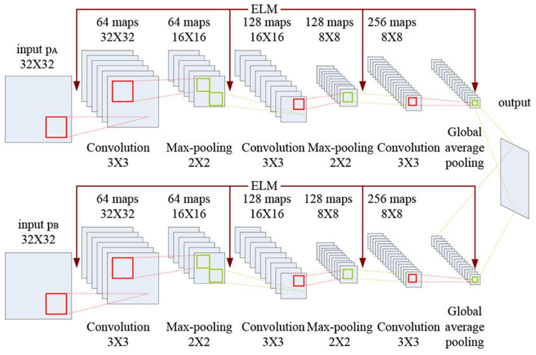


FIGURE 5
CELM diagram used for multimodal medical image fusion.

Network design

In this work, multimodal medical image fusion is regarded as a classification problem. CELM is able to provide the output ranging from 0 to 1 according to a series of image patches {p_A, p_B}. As is known, the essence of image fusion is to extract the important information from the source images and then fuse it into a single one. Fortunately, CELM can just lead us to find the representative information *via* classification. Specifically, the output near to 1 indicates the information in p_A has better reference value, while the information in p_B seems more typical if the output is close to 0. Therefore, the pair of the patches {p_A, p_B} from the same scene can be used as the training samples in CELM. For example, if the information in p_A is more valuable than that in p_B, the corresponding label is set to 1, otherwise the label is set to 0. For sake of maintaining the image information integrity, the whole source medical images are input into the CELM as a whole rather than dividing them into a series of patches. The results in the output layer can provide the scores reflecting the information importance in the source images.

As for the details of the network, two important points need to be made. (a) The network framework can be mainly categorized into three types according to the reference (Zagoruyko and Komodakis, 2015), namely siamese, pseudo-siamese and two-channel. The last type just has a trunk rather than branches. The difference between siamese and pseudo-siamese lies in whether the weights of the branches of them are the same or not. Here, the siamese type is chosen as the network framework in this paper, the reason for which can be summarized as follows. Firstly, due to the weight sharing, the network training course is easy and timesaving. Secondly, take the fusion course of two source images for example, two branches with the same weights indicate the same schemes of feature extracting are used for these two images, which is just consistent with the process of image fusion. (b) The final fusion performance has something to do with the size of the input patch. For example, when the patch size is set to 64 × 64, the classification ability of the network is relatively high since much more image information is taken into consideration. According to Farfade et al. (2015), there is the 2-power law relation between the kernel stride and the number of the max-pooling layer. In other words, if there are four max-pooling layers, the corresponding stride is $2^4 = 16$ pixels. Obviously, the final fused image will suffer from blocky effects. Therefore, in order to guarantee the classification ability and remove the blocky effects as much as possible, the patch size is set to 32 × 32 in this paper.

The CELM diagram used for multimodal medical image fusion is shown in Figure 5.

As indicated in Figure 5, each branch consists of three convolution layers, two max-pooling layers and a global average pooling layer. The kernel size and the stride of the convolution layer are set to 3 × 3 and 1, while the corresponding values of the max-pooling layer are set to 2 × 2 and 2. Here, the global average pooling is used for realizing the function of the original full connection layer in CNN, and the 256 feature maps are obtained for classification.

Fusion schemes

In this paper, the training datasets of CELM are from the website www.ctisus.com, which is the premier radiological website dedicated to multimodal scanning. This website has an incredible library of content ranging from multimodal scan protocols, lectures, case studies, medical illustrations, and a monthly quiz. CTisus.com provides the latest in radiology technology and 3D imaging information, and uploads new content daily.

After constructing the CELM, the fusion issue of the multimodal medical images can be achieved. The specific implementation process consists of two stages, namely 1-stage and 2-stage. Here, we only take the fusion of two images into consideration, and the method can be extended to the case of the fusion of more than two images.

During the 1-stage, the concrete steps are as follows.

Input: Patches of the multimodal medical images to be fused.

Output: The 1-stage fused image.

Initialization: The CELM depicted in Figure 5.

Step 1.1: The patch of 32 × 32 pixels are fed into the CELM.

Step 1.2: By using the two convolution layers, we can obtain 64 and 128 feature maps, respectively. The kernel sizes of the two convolution layers are set to 3 × 3, and the strides of the convolution layers are set to 1.

Step 1.3: The kernel sizes of the two max-pooling layers are both set to 2 × 2, and the strides of the convolution layers are set to 2. And 128 feature maps can be obtained.

Step 1.4: The 128 feature maps are fed into another third convolution layer with the size of 3 × 3 to generate 256 feature maps.

Step 1.5: The global average pooling layer is used to deal with the 256 feature maps in Step 1.4.

Step 1.6: Guarantee that all the pixels of the source images are performed by CELM, and the output can be obtained as:

$$label(i,j)=\begin{cases}1,\ if\ A\left(i,j\right)\ is\ better\ than\ B\left(i,j\right)\\ 0,\ otherwise\end{cases}\qquad(11)$$

$$F(i,j)=\begin{cases}A\left(i,j\right),\ if\ label\left(i,j\right)=1\\ B\left(i,j\right),\ if\ label\left(i,j\right)\neq 1\end{cases}\qquad(12)$$

where "*label*" is the classification result of CELM. *A*, *B* and *F* denote the two source images and the final fused one, respectively. (*i*, *j*) is the coordinate of the pixel in the image.

It should be noted that there will be inconsistency during the fused image, namely a pixel from the source image *A* may be surrounded by a great many counterparts from *B*.

In order to overcome the problem mentioned above, a consistency matrix denoted by *C* is defined here to describe the ownership of the pixels. If the pixel *F*(*i*, *j*) is from *A*, the value of the corresponding element *C*(*i*, *j*) is set to 1, otherwise the value is 0. Then, a filter whose size and stride are 3 × 3 and 1 respectively is used. In the 3 × 3 window, three cases may appear. (a) If the sum of the surrounding eight elements in *C* is greater than or equal to five, the corresponding pixel in *A* will be selected as the counterpart in *F*. (b) If the sum of the surrounding eight elements in *C* is less than or equal to three, the corresponding pixel in *B* will be selected as the counterpart in *F*. (c) If the sum of the surrounding eight elements in *C* is four, the original value in *F* will remain unchanged.

After the 1-stage, the initial fused image can be obtained. However, unlike the fusion of other types of images, higher requirements and standards are needed in the fusion course of multimodal medical images to enhance the precision of lesion detection and diagnosis. In the 2-stage, the connection between the two source images and the initial fused one is analyzed and discussed further. The diagram of the 2-stage is shown in Figure 6.

As shown in Figure 6, *A*, *B*, *F* and *FF* denote the two source images, the initial fused one and the final fused one, respectively. "sub" is the subtraction operator. "*F-A*" stands for the subtraction result between *F* and *A*. Similarly, "*F-B*" stands for the subtraction result between *F* and *B*. *MF* and *MFF* denote the binary mapping of the images *F* and *FF*. MM is the abbreviation of mathematical morphology.

In this paper, the simple subtraction operator is used to measure the similarity between the initial fused image and the source one. The concrete steps of the 2-stage are as follows.

Input: Two source images denoted by *A* and *B*, and the initial fused image *F*.

Output: The 2-stage fusion result *FF*.

Initialization: The two source images and the initial fused one are given.

Step 2.1: The subtraction operation is conducted between *A* and *F* to generate the image *F-A*. Similarly, the image *F-B* can be also obtained.

"*F-A*" and "*F-B*" can describe the extent of feature extracting from the other original source image.

Step 2.2: Compute the value of root mean square error (RMSE) between "*F-A*" and "*B*" to obtain $RMSE_{F-A,B}$. Meanwhile, $RMSE_{F-B,A}$ can also be computed. Here, the size of the window used to compute RMSE is 5 × 5.

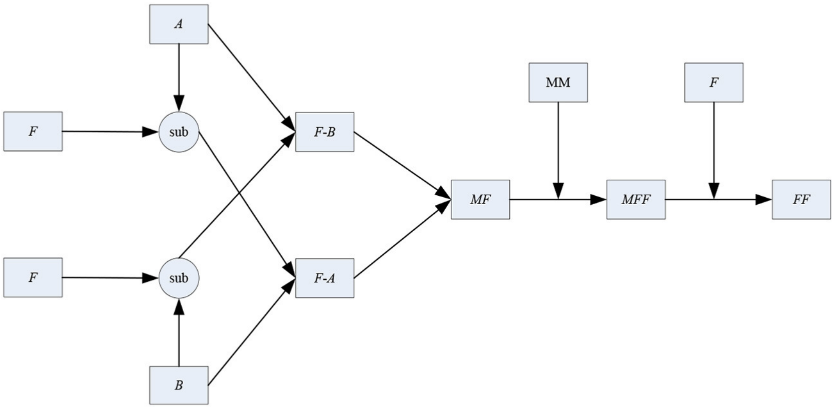


FIGURE 6
Diagram of the two-stage.

Step 2.3: Construct a new matrix *MF* with the same size as *F*. The elements of *MF* can be determined as:

$$\boldsymbol{MF}(\boldsymbol{i},\boldsymbol{j}) = \begin{cases} \boldsymbol{1,\ if\ RMSE_{F-B,A} > RMSE_{F-A,B}} \\ \boldsymbol{0,\ otherwise} \end{cases} \tag{13}$$

where $MF(i,\ j) = 1$ indicates that the difference between *F-B* and *A* is more obvious than that between *F-A* and *B*. In other words, more information should be fused in *A* than that in *B*, otherwise we may should place more emphasis on *B* rather than *A*.

Step 2.4: With the help of MM, *MF* is optimized by a series of opening and closing operators to smooth over the object outlines and the connection between each other. Here, the structure element is a square identity matrix of the size 5 × 5. The modified mapping denoted by *MFF* can be obtained.

Step 2.5: *MFF* and *F* are both taken into account to determine the final fused image *FF*. Please note that compared with the requirements in the 1-stage, the modification condition is more rigorous here. The reason for it lies in that the initial fused image have been already obtained in the 1-stage, while the main objective of the 2-stage aims to further optimization. The elements of *FF* can be optimized as:

$$\boldsymbol{FF}(\boldsymbol{i},\boldsymbol{j}) = \begin{cases} \boldsymbol{1,\ if\ MFF\left(i,j\right) = 1\ and\ sum\left(i,j\right) = 8} \\ \boldsymbol{0,\ if\ MFF\left(i,j\right) = 0\ and\ sum\left(i,j\right) = 0} \\ \boldsymbol{F(i,j),\ otherwise} \end{cases} \tag{14}$$

where "sum" denotes the sum of the elements surrounding $(i,\ j)$ in *MFF*. The window is of size 3 × 3. As Equation (14), if and only if the elements in the window are all from the same source image, the corresponding value in the initial fused image may be modified. Otherwise, the element will still remain unchanged.

It is also noteworthy that if the source images are color ones, we need to convert them into gray ones or deal with them in different color spaces. The color is usually characterized by three independent attributes, which interact on each other to form a spatial coordinate called color space. The color space can be divided into two categories including primary color space, and color brightness separation color space according to the basic structure. RGB and YUV are the typical representatives of the above categories respectively.

RGB mode is an additive one with luminescent screen, while CMYK mode is a printing subtractive one with reflective color. IHS mode suffers from spectral information distortion, which easily leads to medical accidents. Unlike the above three common modes, YUV mode can deal with brightness or color without mutual influences, so it depends on neither light nor pigment. Moreover, YUV includes all color modes the human can see in theory, and it is able to make up for the drawbacks of RGB, CMYK and IHS. Therefore, YUV mode is chosen as the color space in this paper.

During the fusion course of medical source images, we may encounter color images, such as SPECT-TI and SPECT-Tc based ones. Under the circumstances, the RGB source image is converted into the YUV version first. Three components including Y, U and V can be obtained. The Y channel describes the brightness information of the image whereas the other two channels cover the color information. The Y component is fused using the proposed scheme followed by the conversion from YUV to RGB to get the final fused image *F*.

Experimental results with relevant analysis

In order to verify the effectiveness and the superiorities of the proposed method, a series of simulation experiments are performed. Concretely, the section is composed of six parts. The information on the source images to be fused, the methods which are used to be compared with the proposed one, and the experiment settings are given in detail in Experimental

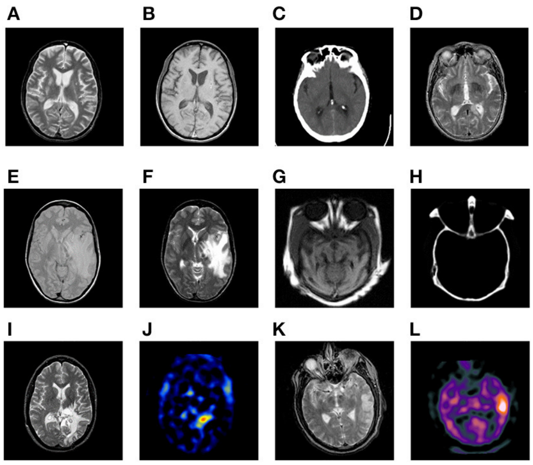


FIGURE 7
Six pairs of multimodal medical source images. Pair I **(A,B)**. Pair II **(C,D)**. Pair III **(E,F)**. Pair IV **(G,H)**. Pair V **(I,J)**. Pair VI **(K,L)**.

setups section. Objective evaluation metrics section lists the objective quantity metrics used in the following experiments. In Experiments on gray and color source images section, the comparisons on the gray images and color ones are conducted in terms of both subjective visual performance and objective quantity results. As the extensive research, the application of the proposed method in other types of source images is also investigated in Applications of the proposed method in other types of source images section followed by the average running time of the proposed method in Average running time of the proposed method section. In the end, the discussions on the potential research directions of the proposed method are given in Discussions on the potential research directions of the proposed method section.

Experimental setups

Six pairs of multimodal medical images are used in the following experiments, which are shown in Figure 7. There are several points requiring to be noted. (a) For simplicity, the corresponding pairs of source images are named as Pair I–VI. (b) All the images share the same size of 256 × 256 pixels, and can be downloaded from the Harvard university site[1] or the Netherland TNO site[2] (c) From the color perspective, the images in pair I–IV are gray ones covering 256-level gray scale, while the images in pair V–VI such as SPECT ones are in pseudo-color. (d) The images with different modalities own a great deal of complementary information, which is beneficial for increasing the accuracy of the lesion detection and localization.

The proposed method is compared with seven representative and recently published ones, which are the adaptive sparse representation (ASR)-based (Liu and Wang, 2015) one, the convolutional sparse representation (CSR)-based one (Liu Y. et al., 2016), the non-subsampled rotated complex wavelet transform (NSRCxWT)-based one (Chavan et al., 2017), the guided filtering fusion (GFF)-based one (Li et al., 2013), the cross bilateral filter (CBF)-based one (Kumar, 2015), CNN-based one (Liu Y. et al., 2017) and gradient transfer and total variation (GTTV)-based one (Ma et al., 2016). Generally speaking, ASR, CSR and NSRCxWT belong to the scope of TDB, while the other four methods are SCB ones. In order to guarantee the objectivity during the whole process of simulation experiments, the free parameters of the seven methods used to be compared are all set as the original references reported.

Objective evaluation metrics

As is well known, it is one-sided for us to evaluate the fusion performance only by subjective inspection. The objective quantity evaluation also plays a significant part during the whole process of image fusion. In Liu et al. (2012), the 12 metrics which are recently proposed and typical are fully analyzed and discussed. On the whole, they can be categorized as four types, namely information theory-based metrics, image feature-based metrics, image structural similarity-based metrics, and human perception inspired fusion metrics. In this paper, four metrics each of which is from four different types above respectively are selected to perform the objective evaluation on the final fused results, including spatial frequency (Q_{SF}) (Zheng et al., 2007), Piella's metric (Q_{Piella}) (Piella and Heijmans, 2003), mutual information (Q_{MI}) (Hossny et al., 2008), and Chen-Varshney metric (Q_{CV}) (Chen and Varshney, 2007).

Experiments on gray and color source images

From the modality perspective, the source images are of six different combinations as follows.

- Pair I (MR-T2 and MR-T1)
- Pair II (CT and MR-T2)
- Pair III (MR-PD and MR-T2)
- Pair IV (CT and MR)
- Pair V (MR-T2 and SPECT-TI)
- Pair VI (MR-T2 and SPECT-Tc)

The fusion results based on the eight different methods are shown in Figure 8.

[1] http://www.med.harvard.edu/AANLIB/home.htm

[2] http://www.imagefusion.org

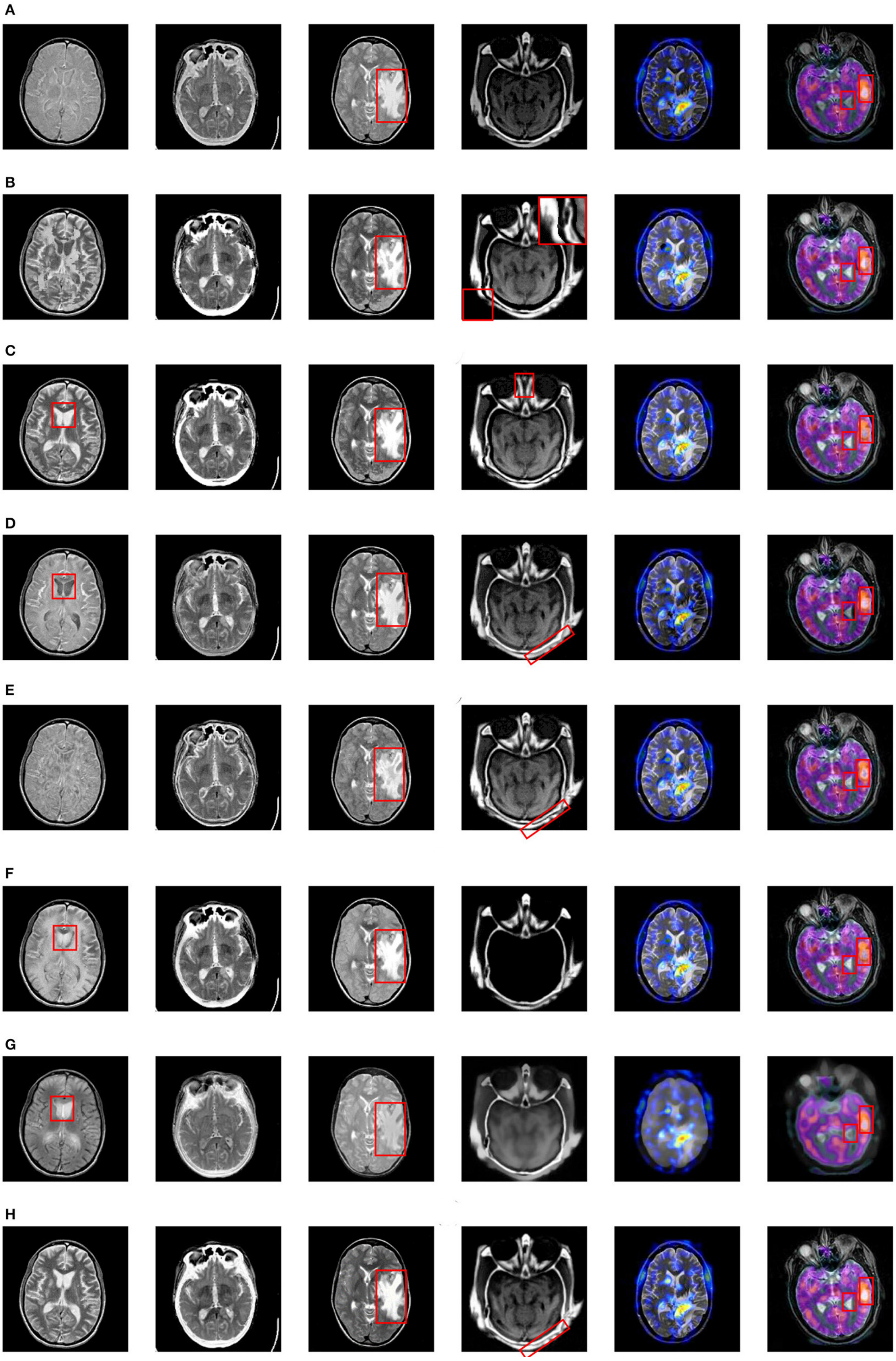


FIGURE 8
Fusion results based on eight different methods. **(A)** ASR, **(B)** CSR, **(C)** NSRCxWT, **(D)** GFF, **(E)** CBF, **(F)** CNN, **(G)** GTTV, **(H)** Proposed.

As for the fused results on Pair I, the ASR-based and CBF-based methods suffers from poor contrast. A great deal of artifacts can be easily found in the fused image based on CSR. Besides, the information of the source images doesn't obtain a fully expression in the fused images based on GFF, CNN and GTTV (please see the red rectangles), which is very unfavorable to the lesion detection and localization. In comparison, the fused images based on NSRCxWT and the proposed one have much better visual performance. In Pair II, a striking comparison can be easily observed that the outline information in Figure 7C is not adequately described by the other seven methods except the proposed one. In other words, the bright white outline is supposed to appear continuously and obviously in the fused image. As to Pair III, the center-right region can be used as a reference (see the red rectangles). The fused images based on ASR, GFF and GTTV have a relatively low contrast level. What is worse, some artifacts even appear in the fused results based on CSR and CBF. Compared with the above five methods, NSRCxWT, CNN and the proposed one all have satisfactory visual performance. However, through careful observation, it can be found that the proposed method has more superiorities over other two ones in terms of the image texture and the information representation. In Pair IV, the original information of the source CT image is almost lost in the fused images based on ASR, CNN and GTTV. In the fused image based on NSRCxWT, there is also an obvious lack of the source MRI information (see the red rectangles). Similarly, the information locating at the bottom right corner in the CBF-based result is also missing. A terrible indented edge can be noticed in the fused result based on CSR (see the magnified region in the upper right corner). Compared with the other six methods, GFF and the proposed method have much better visual performance, but the latter owns much clearer contours than the former, which can be found in the red rectangles. The experiments on Pair V and Pair VI involve the fusion between the gray image and the color one, and their fused results are also in color. Compared with the gray counterparts, color images are able to offer much more information with no doubt. Pair V describes the case of anaplastic astrocytoma. The significant lesion regions obtain better descriptions in the fused image based on the proposed method than other ones. Pair VI addresses another case. Here, for sake of distinguishing the differences among the eight methods, two regions are selected as the references to evaluate the fusion performance (see the red rectangles). Based

TABLE 1 Objective evaluation on the fused images based on different methods.

		ASR	CSR	NSRCxWT	GFF	CBF	CNN	GTTV	Proposed
Pair I	Q_{SF}	34.8118	44.1029	42.4388	35.9596	36.8943	36.2532	34.5648	**45.2897**
	Q_{Piella}	0.7094	0.7219	0.7299	0.7224	0.7302	0.7001	0.5910	**0.7520**
	Q_{MI}	0.7083	0.8813	1.1378	0.6984	0.7198	0.7799	0.6727	**1.1507**
	Q_{CV}	400.0300	367.5945	375.6842	402.3830	414.0351	302.5264	**830.0512**	423.3613
Pair II	Q_{SF}	40.8550	**50.0756**	49.7400	39.9966	47.7477	44.3366	32.0796	49.9253
	Q_{Piella}	0.7373	0.6991	0.7465	0.6587	0.7377	0.7431	0.5075	**0.7783**
	Q_{MI}	0.6974	0.8798	1.0025	0.6704	0.7735	0.9054	0.6418	**1.0780**
	Q_{CV}	1,145.383	1,290.245	716.1920	2,142.597	2237.970	971.9320	**3,762.081**	2535.860
Pair III	Q_{SF}	39.0054	41.8544	40.3306	38.8861	38.1430	40.5021	27.7984	**42.5274**
	Q_{Piella}	0.8974	0.9014	0.9009	0.9053	0.9012	0.8998	0.6221	**0.9193**
	Q_{MI}	0.9498	1.0634	0.9922	0.9013	0.8901	0.9977	0.8141	**1.0675**
	Q_{CV}	169.2490	161.1503	179.7873	150.0123	187.5749	139.1230	**1575.770**	177.0231
Pair IV	Q_{SF}	28.4958	35.3432	36.7455	28.4490	32.4930	28.5946	24.0985	**36.9254**
	Q_{Piella}	0.7667	0.8350	0.8295	**0.8408**	0.8612	0.7688	0.6847	0.8407
	Q_{MI}	0.5002	0.7131	1.0356	0.5855	0.8597	0.5167	0.4761	**1.0553**
	Q_{CV}	1,449.801	2,126.931	2,525.826	1,436.559	2,481.416	1,187.209	1,486.342	**2638.738**
Pair V	Q_{SF}	28.1230	31.2896	31.5508	31.0568	31.5580	31.5670	11.9156	**32.4373**
	Q_{Piella}	0.8102	0.9198	0.9118	0.9246	0.9213	0.9109	0.3985	**0.9265**
	Q_{MI}	0.5846	0.9318	**1.0688**	0.8039	0.9030	1.0590	0.5759	1.0548
	Q_{CV}	228.7973	18.3058	16.3897	25.1995	46.7478	16.3897	**985.4931**	231.8284
Pair VI	Q_{SF}	27.2272	30.8178	31.3711	30.6399	30.6681	30.6681	11.9758	**31.6360**
	Q_{Piella}	0.8237	0.9154	0.9133	0.9226	0.9189	0.9189	0.3655	**0.9249**
	Q_{MI}	0.5806	0.9026	**1.0588**	0.7958	0.8377	0.8377	0.4874	1.0177
	Q_{CV}	106.9227	8.0261	7.7041	28.8184	37.1281	7.7035	**822.7828**	127.5496

The bold values indicate the optimal results.

on the eight fused images, the information of the corresponding regions is not fully described by ASR, GFF and GTTV. What is worse, in the right red rectangles, the artifacts can be observed in the fused images based on CSR, NSRCxWT, CBF and CNN. In comparison with other seven methods, the two regions in the fused image based on the proposed method are much better described.

Of course, there may be individual divergences during the evaluating process. To this end, the four metrics mentioned in subsection *B* are used to evaluate the fusion effects from more balanced and objective perspectives, and the numerical results are reported in Table 1, in which the value shown in bold in each row indicate the best result among the eight methods. Obviously, as for the first three metrics Q_{SF}, Q_{Piella} and Q_{MI}, the proposed method is almost always ranked the first. Owing to the special mechanism of GTTV, its Q_{CV} value is abnormal.

Applications of the proposed method in other types of source images

Different types of images often have diverse characteristics. In order to verify and evaluate the comprehensive performance of the proposed method, extensive investigations on its usage in other types of source images are conducted in this subsection. Here, another two types of source images are selected, namely a pair of multi-focus source images and a pair of visible and infrared source ones, which are denoted by Pair VII and Pair VIII, respectively. These two pairs of source images are shown in Figure 9.

Apart from multimodal medical images, multi-focus images, gray and infrared images are also research hotspots in the field of image fusion. Therefore, these typical types of images are selected as the source images, and the corresponding fusion results are shown in Figure 10. In addition, the objective evaluation results are reported in Table 2. As can be observed, the fused images based on the proposed method are of satisfactory quality.

Average running time of the proposed method

Typically, the visual effect as well as the metric values seems to be the focus of our attention. However, in the practical situations, the computational cost especially the average running time is also a very important factor we are interested in. In this subsection, the experimental results on Pair I are taken into consideration.

The hardware platform concerning the experiments above is as follows. A computer is equipped with an IntelCore i7-7700 3.60 GHz CPU and 16 GB memory. Besides, a GPU module GTX1060 is also employed here. All the simulation experiments are performed with matlab 2014b. In order to guarantee the objectivity of the experimental results, the same experiments are performed thrice *via* the proposed method, and then the average running time is calculated to be the final result. The statistics show that it only takes 1.32 s to achieve the final fused image *via* the proposed method, which is perfectly acceptable to the applications of the lesion detection and localization.

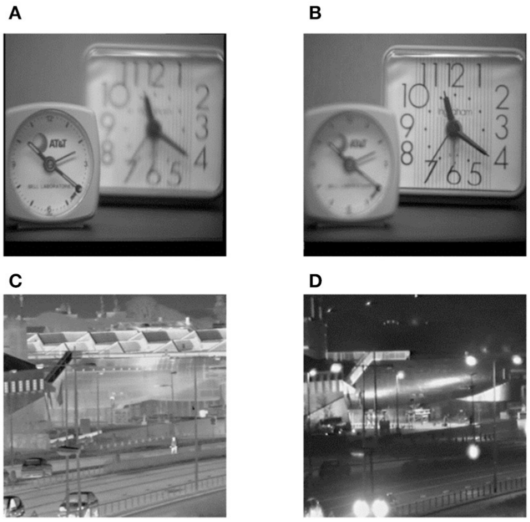


FIGURE 9
Another two types of source images. **(A)** Left-focus source image, **(B)** Right-focus source image, **(C)** Infrared source image, **(D)** Visible light source image.

Discussions on the potential research directions of the proposed method

Although the proposed method is proved to be effective to deal with the fusion issue of the multimodal medical images, it doesn't mean that there is no room for development of CNN theory. On the contrary, lots of researches and investigations are still required to be done in the future. To the best of our knowledge, the following several points are worth researching.

- Optimization of CNN architecture. It is well known that the birth of CNN is of epoch-making significance of the milestone for the area of image processing. However, the traditional CNN architecture has its own inherent drawbacks, which has been mentioned in Related work section. Therefore, the further researches on the optimization of CNN architecture are very necessary. On the one hand, CNN is a representative model in the deep learning field. The relation between the network depth of

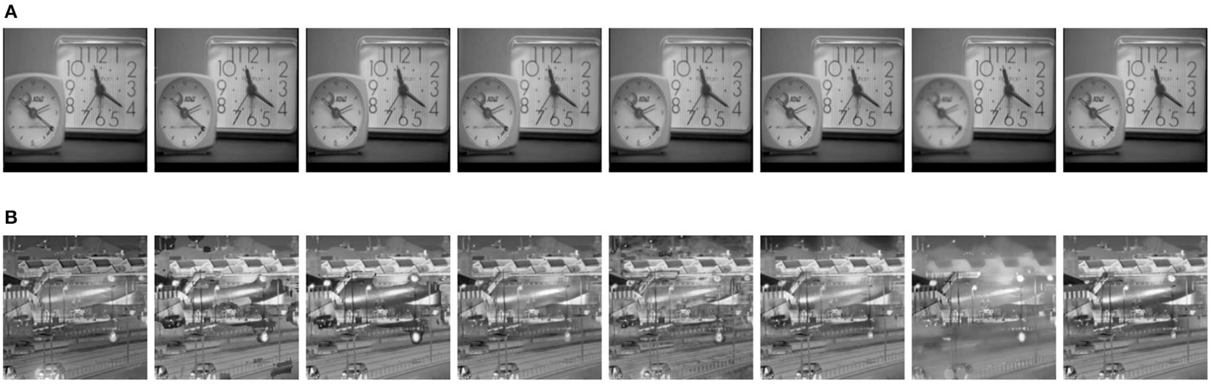


FIGURE 10
Fusion results on two pairs of source images with eight different methods. **(A)** Fusion results on Pair VII (from left to right: ASR, CSR, NSRCxWT, GFF, CBF, CNN, GTTV, Proposed). **(B)** Fusion results on Pair VIII (from left to right: ASR, CSR, NSRCxWT, GFF, CBF, CNN, GTTV, Proposed).

TABLE 2 Objective evaluation on the fused images based on different methods.

		ASR	CSR	NSRCxWT	GFF	CBF	CNN	GTTV	Proposed
Pair VII	Q_{SF}	23.9632	24.1251	24.1067	24.4411	23.1825	24.3852	22.2385	**24.8856**
	Q_{Piella}	0.9377	0.9328	0.9311	0.9325	0.9388	0.9323	0.9060	**0.9421**
	Q_{MI}	1.0345	1.0905	1.1473	1.1031	1.0791	1.2059	1.1002	**1.2634**
	Q_{CV}	54.6205	63.0834	64.7638	64.8442	64.2539	64.6459	**93.4914**	63.2364
Pair VIII	Q_{SF}	30.2317	35.6011	35.5563	30.5951	33.6958	30.1220	22.2862	**35.9478**
	Q_{Piella}	0.8227	0.8178	0.8045	0.8270	0.8341	0.7967	0.5837	**0.8345**
	Q_{MI}	0.3698	0.6208	0.6356	0.3808	0.3833	**0.6467**	0.3255	0.6033
	Q_{CV}	837.8217	1,298.0269	1,317.6476	1,209.4535	1,101.0403	1,325.8068	1,245.9047	**1,390.4678**

The bold values indicate the optimal results.

CNN and the final performance is always an interesting and meaningful topic. On the other hand, in this paper, the introduction of another theory is proved to be effective to overcome the above drawbacks of CNN to a certain extent, so the combination between CNN and other theories could be the future direction of development.

- As other typical fusion methods, the main structure is commonly composed of fusion models and fusion schemes. These two parts both play an instructive role in the whole process of image fusion. As for the fusion models, it has been involved in (a). Similarly, the investigations on the fusion schemes should also be emphasized in the future.

Limitations of the proposed method

Despite its effectiveness, the proposed method also has its inherent limitations as follows.

Firstly, due to the nature of deep learning, the size of the training datasets determines the performance of the proposed method to a large extent. However, compared with the current well-known image datasets, the size of the medical image datasets suitable for training is usually small, so that the learning ability of the proposed network is limited. To solve this problem, the deep cooperation with domestic and foreign well-known medical institutions is necessary, and the construction of large medical image database is expectable.

Secondly, as the important component, ELM can significantly improve the execution efficiency of the proposed method, but its nonlinear representation ability is not well. Therefore, how to improve the classical ELM to optimize the representation ability of nonlinear features becomes a research direction in the future.

Conclusions

In this paper, a novel fusion method called CELM is proposed to deal with the fusion issue of multimodal medical images. CELM combines the advantages of both CNN and ELM. Compared with other typical fusion methods, the proposed one has obvious superiorities in terms of both subjective visual quality and objective metric values. In addition, the potential research directions of the proposed method are also given and discussed, the contents of which will be the emphasis of our next work in future.

Data availability statement

The data generated during the current study are not publicly available due to funding restrictions.

Author contributions

Conceptualization: WK and CL. Methodology: WK. Software and validation: CL. Writing: CL and YL. All authors contributed to the article and approved the submitted version.

Funding

This research was partially supported by the National Natural Science Foundation of China (Grant Number 61902296) and the Natural Science Foundation of Shaanxi Province of China (Grant Number 2022JM-369).

Acknowledgments

The authors would like to thank the reviewers and editors for their invaluable suggestions.

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

Arif, M., and Wang, G. J. (2020). Fast curvelet transform through genetic algorithm for multimodal medical image fusion. *Soft Comput.* 24, 1815–1836. doi: 10.1007/s00500-019-04011-5

Asha, C. S., Lal, S., Gurupur, V. P., and Saxena, P. U. P. (2019). Multi-modal medical image fusion with adaptive weighted combination of NSST bands using chaotic grey wolf optimization. *IEEE Access* 7, 40782–40796. doi: 10.1109/ACCESS.2019.2908076

Azam, M. A., Khan, K. B., Salahuddin, S., and Rehman, E. (2022). A review on multimodal medical image fusion: Compendious analysis of medical modalities, multimodal databases, fusion techniques and quality metrics. *Comput. Biol. Med.* 144, 105253. doi: 10.1016/j.compbiomed.2022.105253

Chao, Z., Duan, X. G., Jia, S. F., Guo, X. J., Liu, H., Jia, F. C., et al. (2022). Medical image fusion via discrete stationary wavelet transform and an enhanced radial basis function neural network. *Appl. Soft Comput.* 118, 108542. doi: 10.1016/j.asoc.2022.108542

Chavan, S. S., Mahajan, A., Talbar, S. N., Desai, S., and Thakur, M., and D'cruz, A. (2017). Nonsubsampled rotated complex wavelet transform (NSRCxWT) for medical image fusion related to clinical aspects in neurocysticercosis. *Comput. Biol. Med.* 81, 64–78. doi: 10.1016/j.compbiomed.2016.12.006

Chen, H., and Varshney, P. K. (2007). A human perception inspired quality metric for image fusion based on regional information. *Inform. Fusion* 8, 193–207. doi: 10.1016/j.inffus.2005.10.001

Daniel, E. (2018). Optimum wavelet-based homomorphic medical image fusion using hybrid genetic-grey wolf optimization algorithm. *IEEE Sens. J.* 18, 6804–6811. doi: 10.1109/JSEN.2018.2822712

Ding, S. F., Du, P., Zhao, X. Y., Zhu, Q. B., and Xue, Y. (2019). BEMD image fusion based on PCNN and compressed sensing. *Soft Comput.* 23, 10045–10054. doi: 10.1007/s00500-018-3560-8

Du, J., Fang, M. E., Yu, Y. F., and Lu, G. (2020a). An adaptive two-scale biomedical image fusion method with statistical comparisons. *Comput. Meth. Prog. Biol.* 196, 105603. doi: 10.1016/j.cmpb.2020.105603

Du, J., Li, W. S., and Tan, H. L. (2020b). Three-layer image representation by an enhanced illumination-based image fusion method. *IEEE J. Biomed Health.* 24, 1169–1179. doi: 10.1109/JBHI.2019.2930978

Du, J., Li, W. S., and Tan, H. L. (2020c). Three-layer medical image fusion with tensor-based features. *Inform. Sci.* 525, 93–108. doi: 10.1016/j.ins.2020.03.051

Du, J., Li, W. S., Xiao, B., and Nawaz, Q. (2016). Union Laplacian pyramid with multiple features for medical image fusion. *Neurocomputing* 194, 326–339. doi: 10.1016/j.neucom.2016.02.047

Fan, F. D., Huang, Y. Y., Wang, L., Xiong, X. W., Jiang, Z. H., Zhang, Z. F., et al. (2019). A semantic-based medical image fusion approach. *arXiv* [preprint] arXiv:1906.00225.

Farfade, S. S., Saberian, M. J., and Li, L. J. (2015). "Multi-view face detection using deep convolutional neural networks," in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*. doi: 10.1145/2671188.2749408

Feng, G., Huang, G. B., and Lin, Q. (2009). Error minimized extreme learning machine with growth of hidden nodes and incremental learning. *IEEE Trans. Neural Netw.* 20, 1352–1357. doi: 10.1109/TNN.2009.2024147

Fu, J., Li, W. S., Du, J., and Xu, L. M. (2021). DSAGAN: A generative adversarial network based on dual-stream attention mechanism for anatomical and functional image fusion. *Inform. Sci.* 576, 484–506. doi: 10.1016/j.ins.2021.06.083

Ganasala, P., and Prasad, A. D. (2020a). Contrast enhanced multi sensor image fusion based on guided image filter and NSST. *IEEE Sens. J.* 20, 939–946. doi: 10.1109/JSEN.2019.2944249

Ganasala, P., and Prasad, A. D. (2020b). Medical image fusion based on laws of texture energy measures in stationary wavelet transform domain. *Int. J. Imag. Syst. Tech.* 30, 544–557. doi: 10.1002/ima.22393

Gao, Y., Ma, S. W., Liu, J. J., Liu, Y. Y., and Zhang, X. X. (2021). Fusion of medical images based on salient features extraction by PSO optimized fuzzy logic in NSST domain. *Biomed. Signal Process.* 69, 102852. doi: 10.1016/j.bspc.2021.102852

Goyal, S., Singh, V., Rani, A., and Yadav, N. (2022). Multimodal image fusion and denoising in NSCT domain using CNN and FOTGV. *Biomed. Signal Process.* 71, 103214. doi: 10.1016/j.bspc.2021.103214

He, C. T., Liu, Q. X., Li, H. L., and Wang, H. X. (2010). Multimodal medical image fusion based on IHS and PCA. *Proc. Eng.* 7, 280–285. doi: 10.1016/j.proeng.2010.11.045

Hermessi, H., Mourali, O., and Zagrouba, E. (2021). Multimodal medical image fusion review: theoretical background and recent advances. *Signal Process*. 183, 108036. doi: 10.1016/j.sigpro.2021.108036

Hossny, M., Nahavandi, S., and Creighton, D. (2008). Comments on 'information measure for performance of image fusion. *Electron. Lett*. 44, 1066–1067. doi: 10.1049/el:20081754

Huang, G. B., Zhou, H., Ding, X., and Zhang, R. (2012). Extreme learning machine for regression and multiclass classification. *IEEE Trans. Syst*. 42, 513–519. doi: 10.1109/TSMCB.2011.2168604

Huang, G. B., Zhu, Q. Y., and Siew, C. K. (2006). Extreme learning machine: theory and applications. *Neurocomputing* 70, 489–501. doi: 10.1016/j.neucom.2005.12.126

Jiang, Q., Jin, X., Hou, J. Y., Lee, S., and Yao, S. W. (2018). Multi-sensor image fusion based on interval type-2 fuzzy sets and regional features in nonsubsampled shearlet transform domain. *IEEE Sens. J.* 18, 2494–2505. doi: 10.1109/JSEN.2018.2791642

Jung, H., Kim, Y., Jang, H., Ha, N., and Sohn, K. (2020). Unsupervised deep image fusion with structure tensor representations. *IEEE Trans. Image Process*. 29, 3845–3858. doi: 10.1109/TIP.2020.2966075

Kavitha, S., and Thyagharajan, K. K. (2017). Efficient DWT-based fusion techniques using genetic algorithm for optimal parameter estimation. *Soft Comput*. 21, 3307–3316. doi: 10.1007/s00500-015-2009-6

Kumar, B. K. S. (2015). Image fusion based on pixel significance using cross bilateral filter. *Signal Image Video Process*. 9, 1193–1204. doi: 10.1007/s11760-013-0556-9

Li, H., He, X., Tao, D., Tang, Y., and Wang, R. (2018). Joint medical image fusion, denoising and enhancement via discriminative low-rank sparse dictionaries learning. *Pattern Recogn*. 79, 130–146. doi: 10.1016/j.patcog.2018.02.005

Li, S. T., Kang, X. D., and Hu, J. W. (2013). Image fusion with guided filtering. *IEEE Trans. Image Process*. 22, 2864–2875. doi: 10.1109/TIP.2013.2244222

Li, W. S., Jia, L. H., and Du, J. (2019). Multi-modal sensor medical image fusion based on multiple salient features with guided image filter. *IEEE Access* 7, 173019–173033. doi: 10.1109/ACCESS.2019.2953786

Lin, M., Chen, Q., and Yan, S. C. (2014). Network in network. *arXiv* [preprint] arXiv:1312.4400.

Liu, X. B., Mei, W. B., and Du, H. Q. (2016). Multimodality medical image fusion algorithm based on gradient minimization smoothing filter and pulse coupled neural network. *Biomed. Signal Process*. 30, 140–148. doi: 10.1016/j.bspc.2016.06.013

Liu, X. B., Mei, W. B., and Du, H. Q. (2017). Structure tensor and nonsubsampled shearlet transform based algorithm for CT and MRI image fusion. *Neurocomputing* 235, 131–139. doi: 10.1016/j.neucom.2017.01.006

Liu, X. B., Mei, W. B., and Du, H. Q. (2018). Multi-modality medical image fusion based on image decomposition framework and nonsubsampled shearlet transform. *Biomed. Signal Process*. 40, 343–350. doi: 10.1016/j.bspc.2017.10.001

Liu, Y., Chen, X., Peng, H., and Wang, Z. F. (2017). Multi-focus image fusion with a deep convolutional neural network. *Inform. Fusion* 36, 191–207. doi: 10.1016/j.inffus.2016.12.001

Liu, Y., Chen, X., Ward, R., and Wang, Z. J. (2016). Image fusion with convolutional sparse representation. *IEEE Signal Process. Lett*. 23, 1882–1886. doi: 10.1109/LSP.2016.2618776

Liu, Y., and Wang, Z. F. (2015). Simultaneous image fusion and denosing with adaptive sparse representation. *IET Image Process*. 9, 347–357. doi: 10.1049/iet-ipr.2014.0311

Liu, Z., Blasch, E., Xue, Z. Y., Zhao, J. Y., Laganiere, R., Wu, W., et al. (2012). Fusion algorithms for context enhancement in night vision: a comparative study. *IEEE Trans. Pattern Anal. Mach. Intell*. 34, 94–109. doi: 10.1109/TPAMI.2011.109

Ma, J. Y., Chen, C., Li, C., and Huang, J. (2016). Infrared and visible image fusion via gradient transfer and total variation minimization. *Inform. Fusion* 31, 100–109. doi: 10.1016/j.inffus.2016.02.001

Ma, J. Y., Yu, W., Liang, P. W., Li, C., and Jiang, J. J. (2019). FusionGAN: A generative adversarial network for infrared and visible image fusion. *Inform. Fusion* 48, 11–26. doi: 10.1016/j.inffus.2018.09.004

Piella, G., and Heijmans, H. (2003). "A new quality metric for image fusion," in *Proceedings 2003 International Conference on Image Processing, Barcelona, Spain, 14-17 September*. doi: 10.1109/ICIP.2003.1247209

Shi, Z. H., Zhang, C. W., Ye, D., Qin, P. L., Zhou, R., Lei, L., et al. (2022). MMI-Fuse: multimodal brain image fusion with multiattention module. *IEEE Access* 10, 37200–37214. doi: 10.1109/ACCESS.2022.3163260

Singh, S., and Anand, R. S. (2020). Multimodal medical image sensor fusion model using sparse K-SVD dictionary learning in nonsubsampled shearlet domain. *IEEE Trans. Instrum. Meas*. 69, 593–607. doi: 10.1109/TIM.2019.2902808

Xu, H., and Ma, J. Y. (2021). EMFusion: an unsupervised enhanced medical image fusion network. *Inform. Fusion* 76, 177–186. doi: 10.1016/j.inffus.2021.06.001

Xu, X. Z., Shan, D., Wang, G. Y., and Jiang, X. Y. (2016). Multimodal medical image fusion using PCNN optimized by the QPSO algorithm. *Appl. Soft Comput*. 46, 588–595. doi: 10.1016/j.asoc.2016.03.028

Yang, Y., Wu, J. H., Huang, S. Y., Fang, Y. M., Lin, P., Que, Y., et al. (2019). Multimodal medical image fusion based on fuzzy discrimination with structural patch decomposition. *IEEE J. Biomed Health*. 23, 1647–1660. doi: 10.1109/JBHI.2018.2869096

Yin, M., Liu, X. N., Liu, Y., and Chen, X. (2019). Medical image fusion with parameter-adaptive pulse coupled neural network in nonsubsampled shearlet transform domain. *IEEE Trans. Instrum. Meas*. 68, 49–64. doi: 10.1109/TIM.2018.2838778

Yu, B. T., Jia, B., Ding, L., Cai, Z. X., Wu, Q., Law, R., et al. (2016). Hybrid dual-tree complex wavelet transform and support vector machine for digital multi-focus image fusion. *Neurocomputing*. 182, 1–9. doi: 10.1016/j.neucom.2015.10.084

Zagoruyko, S., and Komodakis, N. (2015). "Learning to compare image patches via convolutional neural networks," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Boston, MA). doi: 10.1109/CVPR.2015.7299064

Zhang, L. X., Zeng, G. P., Wei, J. J., and Xuan, Z. C. (2020). Multi-modality image fusion in adaptive-parameters SPCNN based on inherent characteristics of image. *IEEE Sens. J*. 20, 11820–11827. doi: 10.1109/JSEN.2019.2948783

Zhang, S., Huang, F. Y., Liu, B. Q., Li, G., Chen, Y. C., Chen, Y. D., et al. (2021). A multi-modal image fusion framework based on guided filter and sparse representation. *Opt. Laser Eng*. 137, 106354. doi: 10.1016/j.optlaseng.2020.106354

Zhao, W. D., and Lu, H. C. (2017). Medical image fusion and denoising with alternating sequential filter and adaptive fractional order total variation. *IEEE Trans. Instrum. Meas*. 66, 2283–2294. doi: 10.1109/TIM.2017.2700198

Zheng, Y., Essock, E. A., Hansen, B. C., and Haun, A. M. (2007). A new metric based on extended spatial frequency and its application to DWT based fusion algorithms. *Inform. Fusion* 8, 177–192. doi: 10.1016/j.inffus.2005.04.003

Zhu, R., Li, X. F., Zhang, X. L., and Wang, J. (2022). HID: the hybrid image decomposition model for MRI and CT fusion. *IEEE J. Biomed Health*. 26, 727–739. doi: 10.1109/JBHI.2021.3097374

Zhu, Z. Q., Chai, Y., Yin, H. P., Li, Y. X., and Liu, Z. D. (2016). A novel dictionary learning approach for multi-modality medical image fusion. *Neurocomputing* 214, 471–482. doi: 10.1016/j.neucom.2016.06.036

Zhu, Z. Q., Zheng, M. Y., Qi, G. Q., Wang, D., and Xiang, Y. (2019). A phase congruency and local Laplacian energy based multi-modality medical image fusion method in NSCT domain. *IEEE Access* 7, 20811–20824. doi: 10.1109/ACCESS.2019.2898111

frontiers | Frontiers in Neurorobotics

TYPE Original Research
PUBLISHED 06 December 2022
DOI 10.3389/fnbot.2022.1084543

Transformer-based progressive residual network for single image dehazing

Zhe Yang[1], Xiaoling Li[1,2] and Jinjiang Li[1,3]*

[1]School of Computer Science and Technology, Intgrow Education Technology, Qingdao Vocational and Technical College of Hotel Management, Shandong Technology and Business University, Yantai, China, [2]Institute of Artificial Intelligence, University of Science and Technology Beijing, Beijing, China, [3]Co-Innovation Center of Shandong Colleges and Universities, Future Intelligent Computing, Shandong Technology and Business University, Yantai, China

Check for updates

OPEN ACCESS

EDITED BY
Xin Jin,
Yunnan University, China

REVIEWED BY
Yin Gao,
Quanzhou Institute of Equipment Manufacturing, China
Qiang Guo,
Shandong University of Finance and Economics, China

*CORRESPONDENCE
Jinjiang Li
lijinjiang@sdtbu.edu.cn

RECEIVED 30 October 2022
ACCEPTED 17 November 2022
PUBLISHED 06 December 2022

CITATION
Yang Z, Li X and Li J (2022) Transformer-based progressive residual network for single image dehazing.
Front. Neurorobot. 16:1084543.
doi: 10.3389/fnbot.2022.1084543

Introduction: The seriously degraded fogging image affects the further visual tasks. How to obtain a fog-free image is not only challenging, but also important in computer vision. Recently, the vision transformer (ViT) architecture has achieved very efficient performance in several vision areas.

Methods: In this paper, we propose a new transformer-based progressive residual network. Different from the existing single-stage ViT architecture, we recursively call the progressive residual network with the introduction of swin transformer. Specifically, our progressive residual network consists of three main components: the recurrent block, the transformer codecs and the supervise fusion module. First, the recursive block learns the features of the input image, while connecting the original image features of the original iteration. Then, the encoder introduces the swin transformer block to encode the feature representation of the decomposed block, and continuously reduces the feature mapping resolution to extract remote context features. The decoder recursively selects and fuses image features by combining attention mechanism and dense residual blocks. In addition, we add a channel attention mechanism between codecs to focus on the importance of different features.

Results and discussion: The experimental results show that the performance of this method outperforms state-of-the-art handcrafted and learning-based methods.

KEYWORDS

transformer, residual network, image dehazing, progressive recurrent, multiple self-attention

1. Introduction

Due to the color distortion, blurring and other quality problems of haze images that affect further information capture, single image deblurring has always been a challenging and highly concerned problem. The deblurring method originates from the classical atmospheric scattering model, and the imaging formula is as follows:

$$\begin{aligned} I(x) &= J(x)\,t(x) + A\,(1 - t(x))\,, \\ t(x) &= e^{-\beta(\lambda)d(x)}, \end{aligned} \tag{1}$$

where $I(x)$ is the degraded image, $J(x)$ is the brightness of the scene when it does not propagate through the water, $t(x)$ is the transmissivity of the propagation medium, $\beta(\lambda)$ is the attenuation coefficient of different wavelengths of light, λ represents different color channels, $d(x)$ is the distance between the camera and objects, and A is the ambient atmospheric light of the scene. Many deblurring methods based on imaging models (He et al., 2010; Zhu et al., 2015; Berman et al., 2016, 2018; Middleton, 2019) restore clean images by reversing the blurring process, in which the atmospheric channel A (x) and the medium transmission map t(x) need to be estimated by manual prior. Although the quality of the blurred image is improved to some extent, these physical priors are not always reliable, and without priors and constraints, the blurring performance will be further reduced, resulting in artifacts and color distortion.

With the development of deep learning in recent years, convolutional neural network has become the backbone of various visual tasks due to its robustness and accuracy. The progress of CNN architecture improves network performance and promotes the progress of single image defogging (Qin et al., 2020) and other hierarchical visual tasks (Afshar et al., 2020; El Helou and Süsstrunk, 2020; Akbari et al., 2021). Although the method based on CNN has special representational ability. It is unable to learn global and remote semantic information interaction well due to the localization of convolution operation. To overcome these problems, some methods add self-attention mechanism (Wang et al., 2020). While others use full attention structure to replace traditional RNN modeling, and propose transformer model to solve Seq2Seq problem (Vaswani et al., 2017). Compared to CNN, Transformer does not increase to distance from the number of operations required to calculate the association between two positions, and can not only do parallel calculations, but also efficiently process global information and encode longer sequences. Due to its powerful presentation capabilities, researchers have applied Transformer to computer vision tasks such as image representation (Wu et al., 2020), image segmentation (Zheng et al., 2021), object detection (Carion et al., 2020; Zhu et al., 2020), pose estimation (Huang et al., 2020a,b; Lin et al., 2021b) and pre-training (Chen et al., 2021a). There are still some problems that can not be ignored when the model is transferred to the visual task, such as the large scale change of the visual target and the high resolution pixel of CV.

Recently, researchers have improved Vit, and swin transformer (Liu et al., 2021) has solved these problems and proved its effectiveness and superiority in target detection, instance segmentation, semantic segmentation and other task fields. Therefore, some methods uses it as the backbone for image classification, image restoration and medical image segmentation. For example, Chen et al. (2021b) introduces a transformer to encode image features and extract contextual input sequences. Cao et al. (2021) proposes a pure transformer similar to u-net for medical image segmentation. Input tokenized image patches to a transformer-based u-shaped encoder-decoder architecture with skip-connections for local-global semantic feature learning. Liang et al. (2021) uses several swin Transformer layers and a residual swin transformer block with a residual connection for image restoration. In order to obtain image features from multi-scale, Gao et al. (2021) proposes a method combining swin transformer trunk and traditional multi-stage network, which effectively improved the ability of feature extraction. Yue et al. (2021) proposes an iterative and progressive sampling strategy and combined with the transformer to classify images.

Inspired by the above process, we proposed an progressive residual network (PRnet) based on swin transformer. PRnet consists of recurrent block, transformer codecs and supervised fusion modules. First, we have a recurrent block that learns shallow features of input images and introduces a long short-term memory (LSTM) network to connect different iterations, ensuring that more of the original image features can be retained over multiple iterations of the model. The transformer codec then learns the sequence representation of the input image through the u-net structure, and effectively extracts the remote context features from multiple scales of the image. The encoder introduces swin transformer block to encode feature representation from the decomposed patch, and continuously reduces the resolution of feature map for local relationship modeling. Decoder decodes hidden features through convolution and upsampling and realizes dimensional transformation to further predict the semantic output of the global context representation. In addition, we connect the encoders through skip connection and add channel attention. this design can effectively avoid the loss of original features and improve the quality of the output image. Finally, the supervised fusion module combines the attention mechanism and dense residual blocks to recursively select and fuse the image features and transfer the attention-guided features to the next stage, which can effectively preserve the original features of the image and prevent the model from over-fitting. In addition, the whole recursive process under the supervision of the original input image can effectively retain the original resolution characteristics of the image, improve the learning efficiency and defogging performance of the network.

To validate our approach, we tested it on different data sets. A large number of experiments and qualitative and quantitative evaluations show that our iterative strategy is beneficial to image restoration and is superior to other state-of-the art methods (see Figure 1). In short, our contribution is:

- We introduce the swin transformer into the iterative progressive residual network (PRnet), which obtains sufficient contextual semantic information and spatial features by learning multi-scale feature information of the input image.

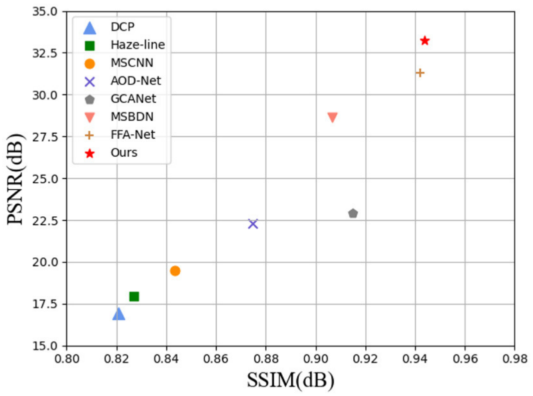


FIGURE 1
Image dehazing on the RESIDE dataset (Li et al., 2018). Under different evaluation indexes, the performance of our method is the most advanced (SSIM on x-axis and PSNR on y-axis) when compared with several advanced methods.

- We introduce channel attention between the encoder and decoder, which makes the module focus on extracting significant useful features related to clean image in the input image.
- We design a supervised fusion module, which combined the dense residual block with attention to conduct recursive supervised fusion of image features under the supervision of ground-truth.

2. Related work

In this section, we will conduct a comprehensive review of fog removal methods and vision transformer relevant to our work. We will conduct a comprehensive review of single image defogging algorithms, including traditional image defogging and deep learning-based image defogging methods.

2.1. Model-based method

By observing and analyzing the imaging process of fog image and its relationship with clean image, the physical model of atmospheric scattering for fog imaging is established. The model-based method tries to estimate the atmospheric light and medium transmission map using the handmade prior knowledge, and then restore the blurred image. Dark channel prior (DCP) is one of the outstanding representatives of priority-based methods. He et al. (2010) assumed that each pixel with a value close to zero has at least one color channel, and combined it with haze imaging model to recover high-quality fog-free images. Zhu et al. (2015) proposed a method of restoring image color attenuation by establishing a linear model to estimate the depth of field information. Berman et al. (2016, 2018) propose an algorithm based on non-local prior to predicting atmospheric light by identifying haze lines and estimating transmission per pixel. Although these methods have achieved some success, they are still constrained by prior knowledge, which may lead to insufficient demisting effect and more serious artifacts and blurriness.

2.2. Deep-learning method

In recent years, a large number of methods based on deep learning have flooded with the field of dehazing. Some deep learning methods still combine physical models or prior knowledge to improve the accuracy of fog removal. Kar et al. (2020) takes the atmospheric light and transmission diagrams estimated by convolutional architecture as a prior condition, and uses an iterative mechanism to gradually update the estimated value to the more appropriate estimated value of fuzzy conditions. Yan et al. (2020) uses multi-scale convolutional neural network combined with atmospheric scattering model to extract features of different scales from global to local. By learning the mapping relationship between hazy images and their transmission images, Ren et al. (2020) predicts projected images at multiple scales and refined the results of defogging. Different from the above methods, Anvari and Athitsos (2020), Liu et al. (2020b), Wang et al. (2021), and Zhang et al. (2022) directly restores blurred images end-to-end by learning the mapping between blurred and clear images. Anvari and Athitsos (2020) combines encoder-decoder structure and residual block to restore fog-free scenes. Through local residual learning and feature attention mechanism, Qin et al. (2020) designs an end-to-end feature fusion attention network to directly restore fog-free images. Liu et al. (2020b) uses residual blocks in fine-grained and coarse-grained networks to generate clean images directly from input fuzzy images. These methods use residual learning to enable network residual links to bypass unimportant information and enable the network architecture to focus on more effective information.

In addition, some methods take into account the morphological differences of fuzzy images at different scales to extract, transfer and fuse multi-scale image features. For example, Yeh et al. (2019) relies on multi-scale residual learning and image decomposition to remove haze from a single image, and feature transmission benefited from the basic components of remnant CNN architecture and simplified u-net structure. Liu et al. (2019) performs multi-scale estimation based on attention, alleviates the bottleneck problem of traditional multi-scale methods and reduces the output image artifacts. Li et al. (2021) designed a dual attention to extract global features and guide subsequent recursive units. Through the strengthen-operate-subtract boosting strategy, Dong et al. (2020) proposes a multi-scale enhanced defogging network with dense feature fusion based on u-net architecture. Despite its success, the limitations of the convolution layer, the main building block of CNN networks, limit the ability to learn remote spatial relevance in such networks. To solve these problems, we have introduced the swin transformer block in this paper.

2.3. Vision transformer

Transformer was first proposed for machine translation Vaswani et al. (2017) and is widely used in many natural languages processing tasks. Because of its powerful representation ability, it has recently been applied to computer vision tasks. To adapt transformer for visual tasks, the researchers have modified it. For example, Transformer model does not have translation invariance and locality like CNN. Parmar et al. (2018) applies self-attention to local fields and solves the problem that it cannot be well generalized to new tasks when data is insufficient. In addition, location information is very important for Transformer. Dosovitskiy et al. (2020) adds position embedding to feature vector and proposes a visual transformer (ViT), which directly applies pure transformer to image patch sequence to complete image classification task. In addition, Transformer model does not have translation invariance and locality like CNN. So it cannot be generalized to new tasks when data is insufficient. Liu et al. (2021) improves ViT by limiting self-attention computation to non-overlapping local windows and allowing cross-window connections to improve efficiency. This layered architecture has the flexibility to model at a variety of scales, which can be well generalized to new tasks. For example, with Swin Transformer as its backbone, Xie et al. (2021) uses self-supervised learning methods to handle object detection and semantic segmentation tasks. Cao et al. (2021) proposes a pure Transformer similar to u-net for medical image segmentation based on u-encoder-decoder architecture and learning local and global semantic features by skipping connections. Huang et al. (2022) has designed an adaptive group attention for Swin Transformer, which reduces the model parameters while taking into account the network performance. Lin et al. (2021a) tries to incorporate the advantages of layered

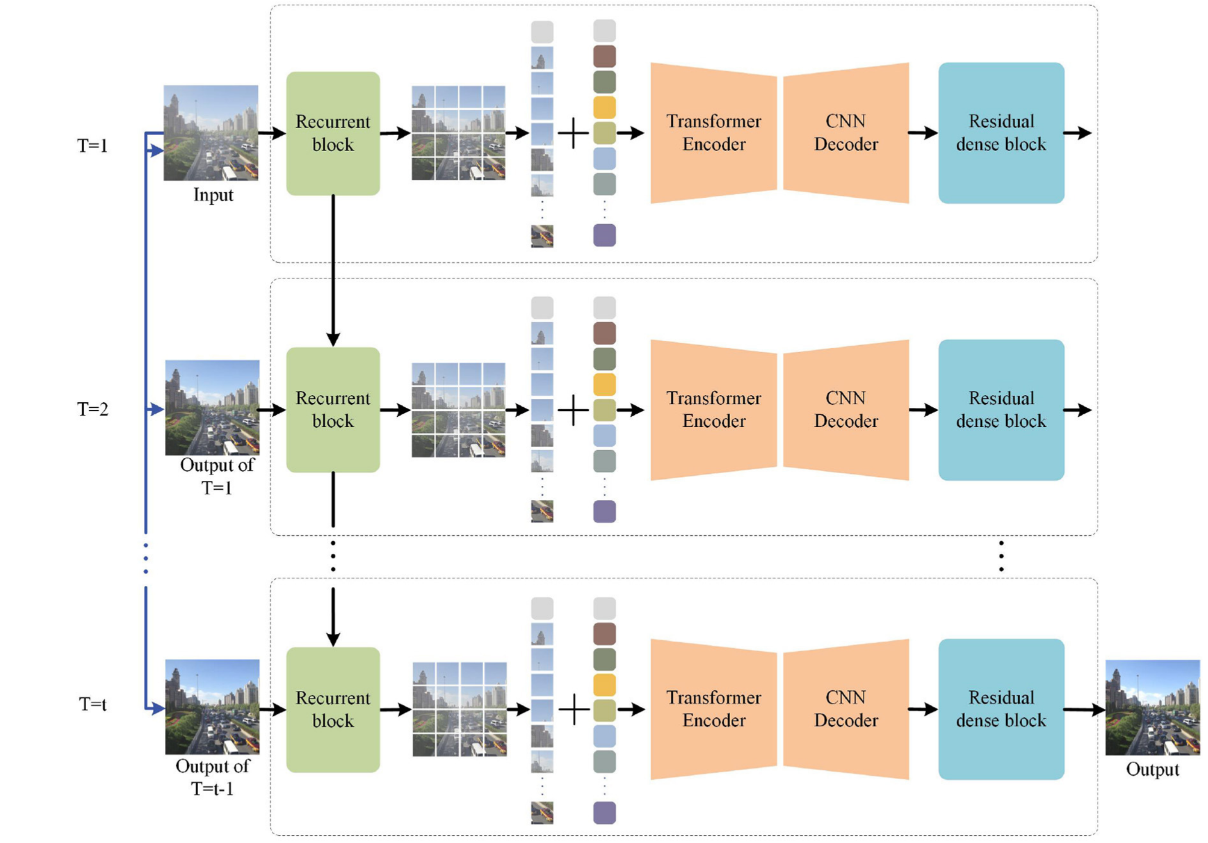


FIGURE 2
The proposed framework for PRnet. PRnet extracts early features through recurrent blocks, then extracts multi-scale features through transformer codec, and finally integrates the features into the supervised fusion module. The blue line represents concatenate operation, and the black line represents forward.

Swin Transformer into the standard encoder and decoder U-shaped architecture at the same time, so as to improve the semantic segmentation quality of different medical images. It designes a strong baseline model for image recovery based on Swin Transformer, and combined Swin Transformer layer with residual connection for depth feature extraction. The success of Swin Transformer in these visual tasks proves that it is superior in some respects to the full convolution approach.

3. Progressive image dehazing networks

In this section, we first introduce the cross scales supervisory integration mechanism (CSSI) and then introduce our overall architecture of progressive residual networks. As shown in Figure 2, it is made up of recurrent block, a Transformer encoder-decoder module based on the u-net architecture, and a supervised fusion module. Finally, we will describe the details of each module and the loss function in detail.

3.1. Cross scales supervisory integration mechanism

Our analysis shows that if the encoder and decoder are independent from each other, multi-scale features cannot interact with each other, which will greatly reduce the performance of the model (Figure 3). If features are fused through simple transfer, convolution or addition microstructures, and these features are treated equally, it is easy to cause redundancy and bring great burden to the network. To solve this problem, we added cross scales supervisory integration (CSSI) between encoders, which can improve the learning efficiency of U-codecs, make full use of features of different scales, and ensure the connectivity of the model. CSSI converts the output feature of encoder layer through 1×1 convolution. Then, the convolution features are paid attention to the information useful to the current output features through the channel attention block (CAB). The channel attention mechanism aggregates spatial dimension features using operations such as convolution, activation function, global average pooling and maximum pooling. Subsequently, the above features are fused through the following skip connection:

$$F_i = C_i \oplus E_i = CAB[\text{conv}(E_i)] \oplus E_i, \tag{2}$$

where E_i and C_i represent the output of the encoder layer and channel attention mechanism respectively. Next, the output feature of the encoder layer is fused with the up-sampling and convolution operation results of the previous decoder layer to obtain the input feature of the next decoder layer:

$$D_i = CSSI[F_i, \text{conv}(\uparrow D_{i-1})], \tag{3}$$

where D_{i-1} and D_i represent the features of the previous and next decoder layers.

CSSI explores the relationship between feature maps of different channels through channel attention, adjusts and aggregates different feature maps in the process of feature interaction, and finally transfers them to the decoder layer. On the one hand, CSSI makes the network pay more attention to find the significant useful information related to the current output in the input data, which can effectively avoid the loss of original features and improve the quality of the output image. On the other hand, CSSI can improve the efficiency of feature fusion and interaction between codecs with different resolutions, effectively reducing the network burden.

3.2. Progressive networks

Swin Transformer interacts with the global information of the image, without considering the importance of the content of the image area and the overall structure of the object, and cannot pay better attention to the structure and details of the image. In order to make up for the above defects, we propose a new progressive residual network (PRnet), which solves the problem of fog removal through multiple stages. At the same time, u-transformer encoder-decoder is used in each stage to learn the morphological features of foggy images at different scales. To avoid the increase and over-fitting of network parameters, different from the previous multi-stage, we do not pile up several sub-networks, but use the recursive calculation between stages to share the same network parameters in multiple stages. In addition, while swin transformer avoids the segmentation edge loss problem, the Transformer image is smaller than the original image resolution. Therefore, ground truth is used to supervise the network, which can suppress features with less information in the current stage and only allow useful features to be transmitted to the next stage.

3.2.1. Progressive recurrent block

We designed a Recurrent block in PRnet to learn the shallow features of the input image, and introduced the Long Short-Term Memory (LSTM) (Yamak et al., 2019) networks to connect different iterations to ensure the propagation of features across multiple stages of the model. In the process of feature dependence, more original image features can be retained. As shown in Figure 4, taking the t iteration as an example, we input the original foggy image and the predicted image output by the iteration into the network together, go through the convolutional layer 3 × 3 × 64 with a step size of 1, and then go through the activation function(ReLU) performs nonlinear correction. In the subsequent convolution, we did not perform batch normalization, but added an LSTM layer. LSTM introduces and splices the feature map output x_{t-1} from

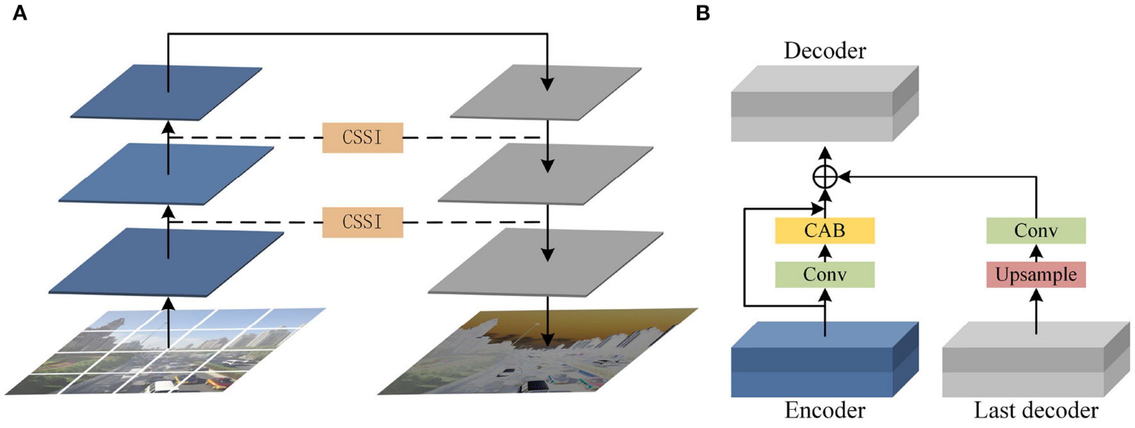


FIGURE 3
(A) Encoder-decoder block. **(B)** Cross scale supervisory integration mechanism between encoder decoder and the last decoder.

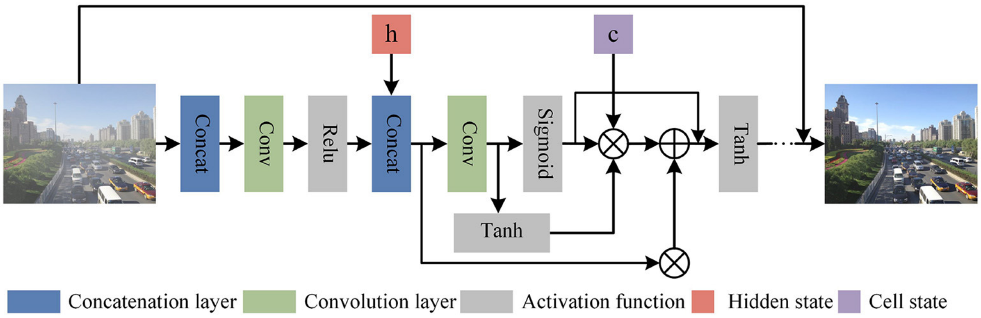


FIGURE 4
Progressive recurrent block structure. ⊗ Represents Hadamard Product, and the corresponding elements in the matrix are multiplied. ⊕ Represents matrix addition operation.

the $t-1$ iteration and the previous hidden state h_{t-1}. The feature graph i_t is obtained by convolution, which is used to determine which information is important and needs to be retained. Then feature graphs f_t and o_t controlling forgotten data were obtained through sigmoid activation function, and then forgetting and remembering were carried out according to the following formula:

$$C_t = f_t * C_{t-1} + i_t * \underset{t}{\tilde{C}}, \qquad (4)$$

Among them, $\underset{t}{\tilde{C}}$ represents the cell state, which is a feature map obtained by passing h_{t-1} and x_{t-1} to the Tanh function. Next, multiply o_t with C_t after Tanh activation to obtain h_t to determine the information carried in the hidden state, namely:

$$h_t = o_t * \mathrm{Tanh}(C_t), \qquad (5)$$

where h_t is output as the current cell, which is passed to the next time period with the new cell state C_t. The output of the entire asymptotic recursive process can be expressed as:

$$f_{res} = LSTM(x_{t-1}, h_{t-1}) \qquad (6)$$

3.2.2. Transformer encoder-decoder

As we all know, multi-scale networks can not only extract low-level high-resolution features and texture detail information, but also extract high-level feature semantic information, and fully extract and utilize image features at different scales. Therefore, we combine the advantages of swin transformer and cnn to design encoder-decoder based on u-net architecture. By learning the sequence representation of the input image, we can ensure that sufficient contextual semantic information and spatial features are acquired during the long-distance transmission.

Swin transformer introduces the locality idea in the Multiple Self-Attention (MSA) module to perform self-attention computation in the window region without overlap. Because of its hierarchical design and generalization, it has proven its effectiveness in several fields such as object detection, semantic segmentation and image denoising. Therefore, we apply Swin transformer directly in encoder to encode the feature representation from the decomposed patch.

Our encoder generates different number tokens through three layers of encoder layer. The first, second and third layers generate $\frac{H}{4} \times \frac{W}{4}$, $\frac{H}{8} \times \frac{W}{8}$, and $\frac{H}{16} \times \frac{W}{16}$ tokens respectively. Each stage consists of Patch Merging and some Swin Transformer Blocks. We merged the image resolution by a sliding window operation for Patch Merging, and divided the image with a given size of $H \times W$ into RGB image patches, and marked them as the original pixel Mosaic vector with a size of 4×4. It is then mapped to a vector of dimension 4C using linear embedding. At this time, the output dimension is set to 2C and the feature size is set to $\frac{H}{8} \times \frac{W}{8}$ from the original $\frac{H}{4} \times \frac{W}{4}$. Next, the output feature z^{l-1} enters two consecutive Swin Transformer Blocks for feature transformation. Unlike MSA in ViT, Swin Transformer Block computes self-attention by adding a relative position bias B to the corresponding head, then the output feature z^{l-1} of layer l can be written as follows:

$$\begin{aligned} z_m^l &= SW - MSA(LN(z^{l-1})) + z^{l-1}, \\ z^l &= MLP(LN(z_m^l)) + z^l, l = 1, 2, 3, \end{aligned} \tag{7}$$

where z_m^l represent the output of multi-head self-attention, z^l represent the output of MLP.

Corresponding to the encoder, a symmetric decoder is constructed based on the swin transformer, forming an encoder-decoder based on the u-net architecture.To recover the spatial order, we use a convolution module and upsampling to form a Decoder layer. In the first layer the hidden features are first decoded by bilinear upsampling of the input features ($\frac{H}{16} \times \frac{W}{16} \times 4C$).And then implement dimension transformation in the convolution module. A linear layer is applied to map the dimensions to 2C, then the resolution is extended to $\frac{H}{8} \times \frac{W}{8}$, and finally the output feature ($\frac{H}{8} \times \frac{W}{8} \times 2C$) is fed into the next Decoder layer. Bilinear up-sampling operation can ensure the same dimensions before and after the fusion, so that the fusion and feature mapping under the same dimension can be carried out again. In addition, Decoder decodes hidden features while further predicting the semantic output of the global context representation.

3.2.3. Supervise fusion module

First, the output features of Swin transformer decoder are supervised by ground-truth and attention maps are generated by Supervised Attention (Zamir et al., 2021) to assist the delivery of useful features and effectively preserve the original features of the image. Next, we introduce residual blocks to learn deeper features. Inspired by Kim et al. (2016), we use recursion to unfold the residual block by calling the residual block 5 times, with both input and output channels of 64 and a convolution kernel size of 3×3. In addition, a skip connection is used in the residual block to connect the input and output, which is then passed to the next residual block as input. The calculation formula is as follows:

$$x_i = x_{i-1} + ReLU(x_{i-1}, w_{i-1}) \tag{8}$$

where x_i is the output of the current residual block, x_{i-1} is the output of the last residual block, ReLU is the activation function, which can effectively improve the accuracy of the model.

3.3. Loss function

The aim of our training is to recover clear images with low-level and high-level features from fogged images. In order to obtain high quality images, we use a combined loss function for optimization during the training process. Therefore, given a training dataset $\left\{ R_T^n, G^n \right\}_n^N$ for T-stage, we solve

$$L = \sum_{T=1}^{t} \{\alpha L_C(R_T^n, G^n) + \beta L_S(R_T^n, G^n)\}, \tag{9}$$

where R_T^n is the outputs of stage T, and G^n represents the ground-truth images. The loss coefficients of α and β are set to 0.2 and 4. And L_C is the Charbonnier loss (Charbonnier et al., 1994), used to calculate the pixel loss between the predicted image and the ground truth. In addition, $L_S(R_T^n, G^n)$ is the structural similarity loss (Wang et al., 2004), which is used to evaluate the structural similarity of the content of the two images. To avoid images suffering from distortion and low peak signal-to-noise ratio (PSNR), Ren et al. (2019) uses negative SSIM loss in an image recovery task and demonstrates the effectiveness of this loss on PSNR, SSIM and visual.

4. Experimental results

In this section, we first present the training details and evaluation metrics. Then, our method is compared qualitatively and quantitatively with advanced methods on multiple datasets. Finally, we conduct ablation experiments.

4.1. Experimental setup

The RESIDE dataset (Li et al., 2018) is a large-scale benchmark including synthetic images and real-world blurred images. The RESIDE is composed of five sub-data sets: Indoor Training Set (ITS), Outdoor Training Set (OTS), Synthetic

Objective Testing Set (SOTS), Real-world Task-driven Testing Set (RTTS) and Hybrid Subjective Testing Set (HSTS) constitute. We selected 20,000 pairs and 500 pairs from SOTS as outdoor scene training set and outdoor scene test set respectively, and 2,000 pairs of real blurred images from RTTS for testing. In addition to the RESIDE dataset, we also conducted experiments on another publicly available dataset. O-HAZE (Ancuti et al., 2018) is an outdoor scene dataset proposed by NTIRE2018

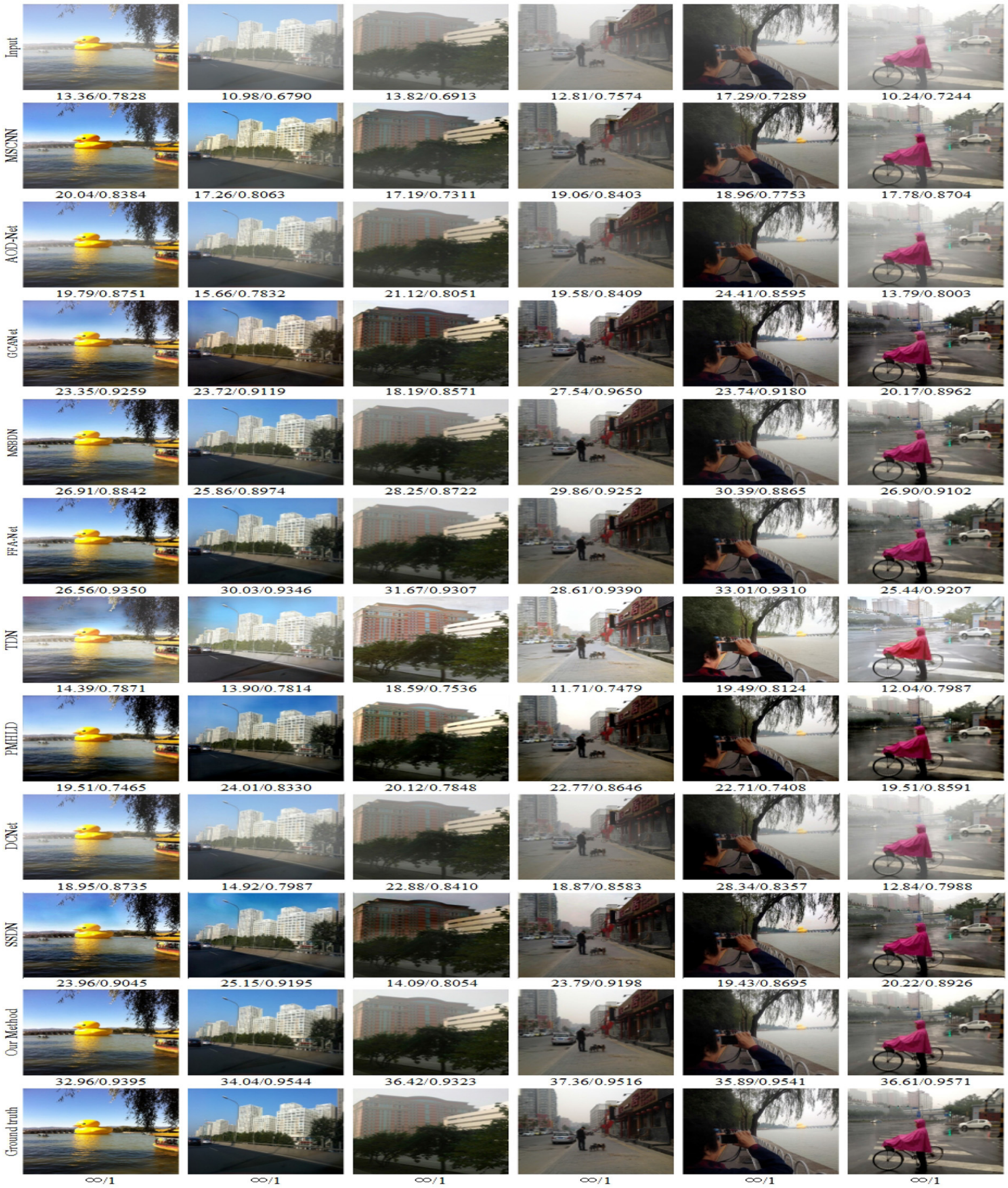


FIGURE 5
Visual results on the SOTS dataset. Best viewed on a high-resolution display.

Image Dehazing Challenge, including 45 pairs of real foggy images and corresponding fog-free images. These fogged images are taken by professional haze instruments, which can well record the same visual content under fog-free and fogged conditions. We choose 35 pairs as the training set, 5 pairs as the validation set, and 5 pairs as the test set.

Our network was trained on an Ubuntu environment, using the ADAM (Kingma and Ba, 2014) optimizer and on an NVIDIA RTX2080ti GPUs. The training was performed using the Pytorch framework. The initial learning rate was set to 3×10^{-5} and gradually decreases to 1×10^{-6}. The network was trained for 50 epochs, and the input image size was $512 \times 512 \times 3$.

In order to evaluate the image quality of single image defogging and compare it with other methods. We used the two most commonly used evaluation metrics in defogging methods: Peak Signal to Noise Ratio (PSNR) and structural similarity (SSIM). PSNR is a pixel-level image quality evaluation method used to measure the difference of gray values between two images. The higher the PSNR value,the lower the distortion

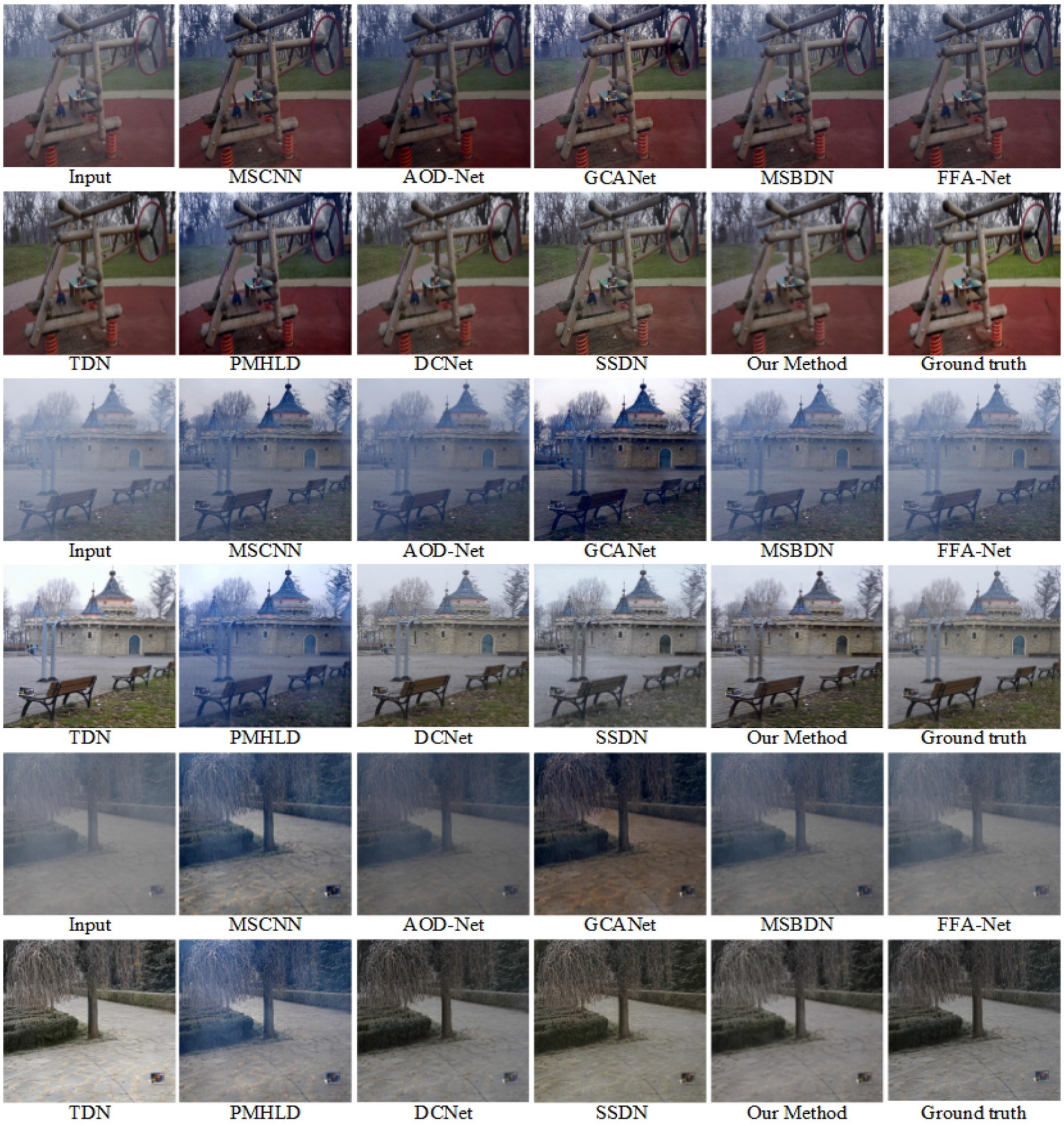


FIGURE 6
Visual results on the O-HAZE dataset. Best viewed on a high-resolution display.

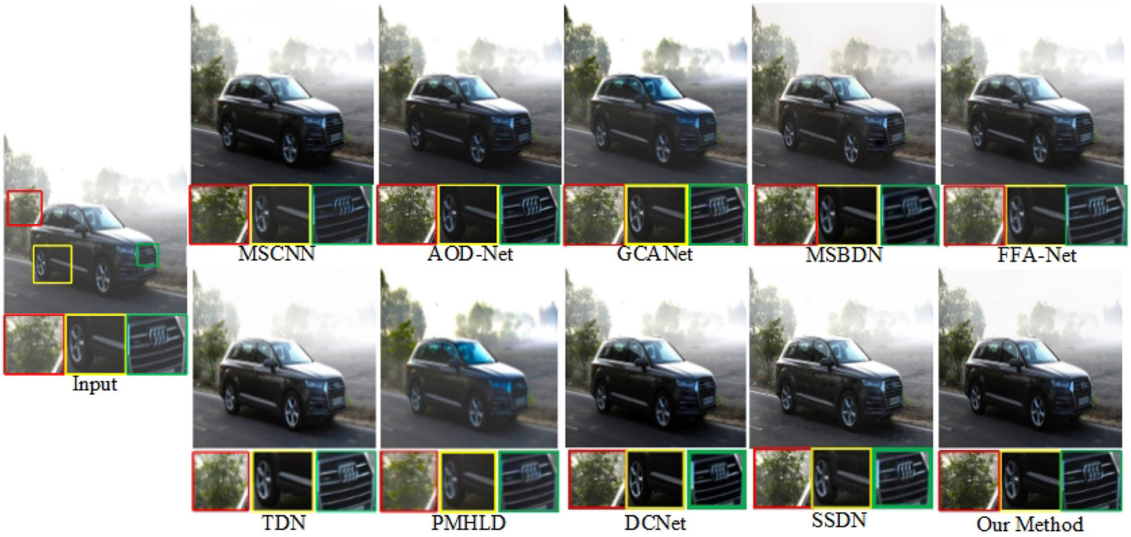


FIGURE 7
Visual quality comparison on real mist images.

between the evaluated image and the ground-truth image, and the better the quality; on the contrary, the poorer the quality. SSIM is a measure of covariance to determine the degree of structural similarity between images according to the degree of correlation between image pixels. The higher SSIM value, the more structure or color information the image retains, and the better the effect of the resulting image. What's more, we use the scikit-image library of python to calculation them. In addition, since there is no ground-truth image in real-world datasets, we use Fog Aware Density Evaluator (FADE) (Choi et al., 2015) to evaluate the haze density of the restored image. We also adopted the non-reference blind image quality evaluation indicators, NIQE (Mittal et al., 2012). NIQE is used to normalize the image contrast into blocks, and determine the image quality by calculating the average value of the local contrast of each block.

4.2. Image dehazing results

We evaluated the defogging results objectively and subjectively on different datasets, and compared the proposed defogging method with seven state-of-the-art methods, namely, MSCNN, AOD-Net, GCANet (Chen et al., 2019), MSBDN, FFA-Net, TDN (Chen et al., 2020), PMHLD (Liu et al., 2020a), DCNet (Bhola et al., 2021), and SSDN (Huang et al., 2021).

4.2.1. Subjective evaluation

We selected outdoor synthetic and real fogged images from the RESIDE dataset for testing, and combined our method with seven advanced methods. In addition, to verify the effectiveness of our network, we also selected real fog images from the O-HAZE dataset for testing, and selected three of them for comparison and presentation. The original fogged images, ground truth and the defogging results using 8 methods are shown in Figures 5–8.

In Figure 5, the top row shows the input fog image. It can be seen that MSCNN, AOD-Net and DCNet are not ideal in a slightly complex environment, and the restored colors are not bright enough. The GCA, TDN and SSDN methods have the problems of color difference, color spot and color oversaturation. MSBDN, FFA-Net, PMHLD and our methods are relatively close to the real ground images, but MSBDN and FFA-Net are not satisfactory in restoring remote scenes, while PMHLD produces color differences in the sky of column 1 and column 2. In contrast, our method performs better in color and detail in complex environments. For example, our method removes the haze around people in the fourth and fifth columns more thoroughly.

Figure 6 shows the demisting effects of different methods in the O-HAZE dataset. In the first two layers, the fog removal effect under the mist is displayed. MSCNN, AOD-Net, MSBDN, and FFA-Net not only did not remove the influence of haze, but also deepened the blurriness of the scene and made the overall color darker. Although GCANet and PMHLD reduce the fogging effect, the color of the image itself is affected, and the overall brightness of the output image is low. TDN, SSDN and our method generate more visible results with more significant demisting effect and clearer texture details.

Figures 7, 8 show the demisting effect of real scenes at different shooting distances. In these two images, the overall brightness of the images restored by MSCNN, GCA-Net, and

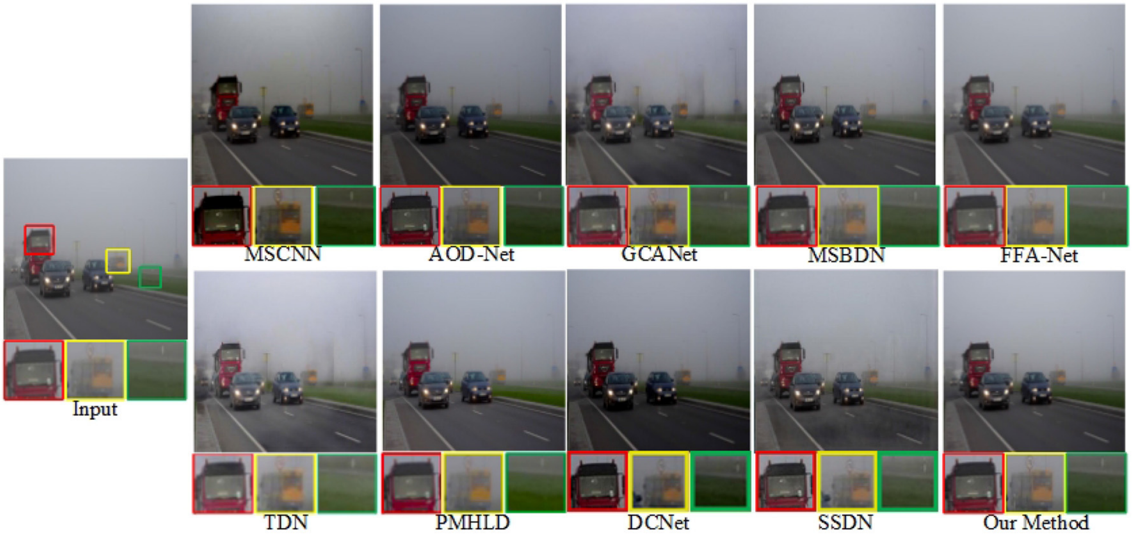


FIGURE 8
Visual quality comparison on real dense fog images.

TABLE 1 Quantitatively compare the dehazing results with SOTA methods on the RESIDE and O-HAZE datasets.

Method	SOTS		O-HAZE	
	SSIM	PSNR	SSIM	PSNR
MSCNN	0.8436	19.49	0.7359	18.93
AOD-Net	0.8747	22.31	0.6724	18.19
GCANet	0.9151	22.89	0.6633	15.77
MSBDN	0.9068	28.64	0.6378	18.46
FFA-Net	0.9422	31.31	0.6792	18.07
TDN	0.7857	17.38	0.7286	19.41
PMHLD	0.8276	23.81	0.4839	14.40
DCNet	0.8343	19.47	0.7028	20.74
SSDN	0.8852	21.11	0.7789	25.71
Ours	0.9439	33.25	0.8758	24.19

Best and second best scores are red and blue. The table shows the average of the data.

TABLE 2 Quantitative and efficiency comparison in RTTS dataset.

Method	NIQE	FADE	Runtimes
MSCNN	3.2499	1.1716	2.3356
AOD-Net	3.4439	1.4342	0.1904
GCANet	3.2615	1.0135	0.0821
MSBDN	3.4248	1.5211	0.0394
FFA-Net	3.4515	2.0205	0.6561
TDN	3.3356	0.9217	0.8767
PMHLD	3.2254	0.7240	0.3321
DCNet	3.4188	1.2886	0.1725
SSDN	3.3756	1.8476	0.3357
Ours	3.1752	0.7873	0.4436

Color numbers indicate the best indicator value.

DCNet is low, such as a large area of dark areas in the sky. The overall color of TDN, MSBDN, and FFA Net is not bright enough, and the distant scenes are not well recovered. SSDN and our method restore relatively complete details, but in the first scene, SSDN is blurred in the vegetation (red box area), and our details processing is more prominent. Compared with these advanced methods, PMHLD and our methods have more realistic details and better visibility in the restored images.

In summary, our method is visually outstanding in both synthetic and real scenes, and the recovered images are more thoroughly defogged and have clearer details such as color textures.

4.2.2. Objective evaluation

In the previous section we evaluated the images after defogging through visual effects.In this section, we provide an objective analysis of several methods using two different quality evaluation metrics, PSNR and SSIM. We count the data metrics averaged over the RESIDE dataset and the O-HAZE dataset for each method and visualize them. In addition, we also show the values of SSIM and PSNR metrics for several images in Figure 5. It can be found that the PSNR values of our method are much higher than the other methods, which indicates that the less distortion and better quality between the images processed by our method and the ground-truth images. As can be observed in Table 1: our method outperforms all SOTA methods with SSIM and PSNR of 0.9438 and 33.2523 dB on the RESIDE dataset. It is intuitively seen in Figure 1 that our method significantly outperforms other methods in

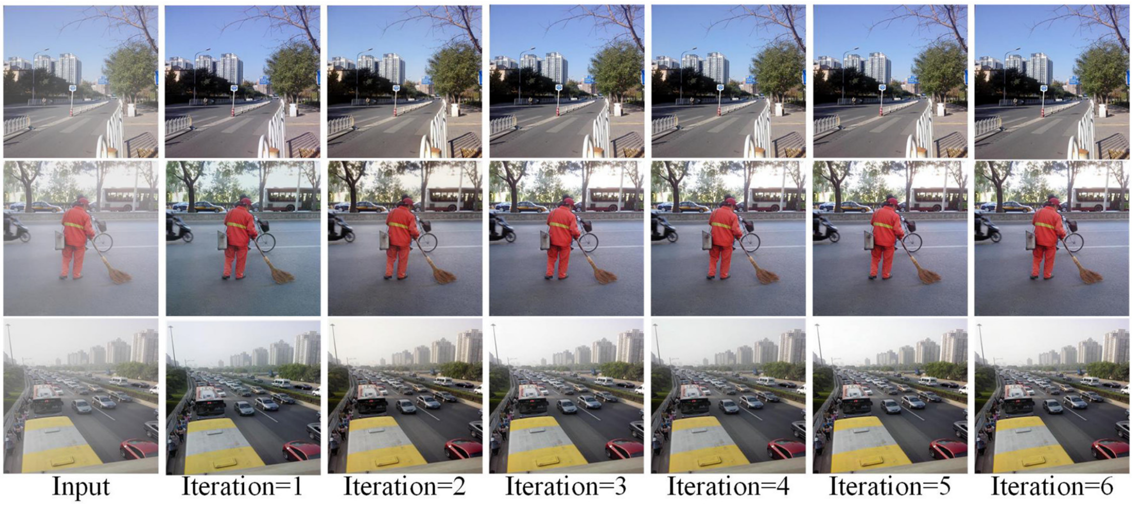


FIGURE 9
Single image defogging image obtained in different iterations.

TABLE 3 Use outdoor synthetic images to test models with different iteration times, use PSNR, SSIM, and TIME for comparison.

	SSIM	PSNR	TIME
Iteration=3	0.9289	32.71	0.3354
Iteration=4	0.9356	33.14	0.3863
Iteration=5	0.9438	33.25	0.4436
Iteration=6	0.9438	33.28	0.4986

The value in the table is the average of all images.

two metrics. In addition, the O-HAZE dataset outperforms the other methods with 0.8758dB and 24.1986dB. Compared with the RESIDE dataset, the haze in this dataset is more dense, the image quality degrades more seriously, and the defogging is more difficult, which further confirms the effectiveness of our method in a dense fog environment.

Table 2 shows the objective indicators and time comparison of all methods on RTTS. NIQE, and BRISQUE evaluated the overall quality of the image. Our method obtained the best results of NIQE, indicating that the results in this paper have excellent colors and details. In terms of FADE metric, our method obtained suboptimal, while PMHLD obtained the optimal FADE value. This is inseparable from the effective haze removal of PMHLD. In terms of time, our method has only achieved the fourth place, not outstanding in efficiency.

4.3. Ablation study

Our approach shares the same network parameters across multiple stages through the iterative idea of using recursive computation between stages. We speculate that the defogging effect of the model will change with the increase of the number of iterations, so it is crucial to determine the optimal number of iterations.We hypothesize that the defogging effect of the model varies with the number of iterations, so it is crucial to determine the optimal number of iterations. We trained the model using iterations 1–6 under the RESIDE dataset, and Figure 9 shows the effect of image defogging under different iterations. The visual effects were similar from the 3rd to the 5th iteration, so we made an objective evaluation of these iterations. According to the comparison of PSNR and SSIM in Table 3, we found that the metrics of the third iteration and the fourth iteration were slightly lower, while the metrics of the fifth iteration and the sixth iteration were similar. By comparing the time, we choose the fifth iteration as the optimal number of iterations.

5. Conclusion

In this paper, we propose a new transformer-based progressive residual network (PRnet). Our method recursively invokes the residual network to gradually recover clean images under ground-truth supervision. First of all, PRnet learns the features of the input images through recurrent block, while taking care of connecting the different stages to ensure that more original image features are retained during the multi-stage feature transfer of the model. We design a codec with u-net structure in combination with swin-transformer, which can ensure that sufficient contextual semantic information and spatial features are obtained during long-distance transmission. In addition, CSSI, which can ensure the synergy and connectivity of the transformer codec. Finally, the supervised fusion module can adaptively select and fuse the image features, and transfer the attention-guided features to the next stage.In addition, we demonstrate the effectiveness of the progressive network through experiments, and our model provides high-quality defogging on multiple data sets. Nonhomogeneous de-hazing is the next topic we would like to explore with our approach, as it is crucial to study complex foggy environments in real scenarios.

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

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

Funding

This research was supported by the National Natural Science Foundation of China (61772319, 62002200, 62202268, and 61972235) and Shandong Natural Science Foundation of China (ZR2021MF107 and ZR2022MA076).

Conflict of interest

Author ZY was employed by Intgrow Education Technology.

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

Afshar, P., Heidarian, S., Naderkhani, F., Oikonomou, A., Plataniotis, K. N., and Mohammadi, A. (2020). COVID-caps: a capsule network-based framework for identification of COVID-19 cases from x-ray images. *Pattern Recognit Lett.* 138, 638–643. doi: 10.1016/j.patrec.2020.09.010

Akbari, H., Yuan, L., Qian, R., Chuang, W.-H., Chang, S.-F., Cui, Y., et al. (2021). Vatt: transformers for multimodal self-supervised learning from raw video, audio and text. *arXiv preprint arXiv:2104.11178*. doi: 10.48550/arXiv.2104.11178

Ancuti, C. O., Ancuti, C., Timofte, R., and De Vleeschouwer, C. (2018). "O-haze: a dehazing benchmark with real hazy and haze-free outdoor images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (Salt Lake City, UT: IEEE), 754–762.

Anvari, Z., and Athitsos, V. (2020). Dehaze-glcgan: unpaired single image de-hazing via adversarial training. *arXiv preprint arXiv:2008.06632*. doi: 10.48550/arXiv.2008.06632

Berman, D., Avidan, S., and Avidan, S. (2016). "Non-local image dehazing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV: IEEE), 1674–1682.

Berman, D., Treibitz, T., and Avidan, S. (2018). Single image dehazing using haze-lines. *IEEE Trans. Pattern Anal. Mach. Intell.* 42, 720–734. doi: 10.1109/TPAMI.2018.2882478

Bhola, A., Sharma, T., and Verma, N. K. (2021). Dcnet: dark channel network for single-image dehazing. *Mach. Vis. Appl.* 32, 1–11. doi: 10.1007/s00138-021-01173-x

Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., et al. (2021). Swin-unet: Unet-like pure transformer for medical image segmentation. *arXiv preprint arXiv:2105.05537*. doi: 10.48550/arXiv.2105.05537

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). "End-to-end object detection with transformers," in *European Conference on Computer Vision* (Glasgow: Springer), 213–229.

Charbonnier, P., Blanc-Feraud, L., Aubert, G., and Barlaud, M. (1994). "Two deterministic half-quadratic regularization algorithms for computed imaging," in *Proceedings of 1st International Conference on Image Processing, Vol. 2* (Austin, TX: IEEE), 168–172.

Chen, D., He, M., Fan, Q., Liao, J., Zhang, L., Hou, D., et al. (2019). "Gated context aggregation network for image dehazing and deraining," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)* (Waikoloa, HI: IEEE), 1375–1383.

Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., et al. (2021a). "Pre-trained image processing transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Nashville, TN: IEEE), 12299–12310.

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., et al. (2021b). Transunet: transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv:2102.04306*. doi: 10.48550/arXiv.2102.04306

Chen, W.-T., Fang, H.-Y., Ding, J.-J., and Kuo, S.-Y. (2020). Pmhld: patch map-based hybrid learning dehazenet for single image haze removal. *IEEE Trans. Image Process.* 29, 6773–6788. doi: 10.1109/TIP.2020.2993407

Choi, L. K., You, J., and Bovik, A. C. (2015). Referenceless prediction of perceptual fog density and perceptual image defogging. *IEEE Trans. Image Process.* 24, 3888–3901. doi: 10.1109/TIP.2015.2456502

Dong, H., Pan, J., Xiang, L., Hu, Z., Zhang, X., Wang, F., et al. (2020). "Multi-scale boosted dehazing network with dense feature fusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Seattle, WA: IEEE), 2157–2167.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2020). An image is worth 16x16 words: transformers for image recognition at scale. *arXiv preprint arXiv*:2010.*11929*. doi: 10.48550/arXiv.2010.11929

El Helou, M., and Süsstrunk, S. (2020). Blind universal bayesian image denoising with gaussian noise level learning. *IEEE Trans. Image Process.* 29, 4885–4897. doi: 10.1109/TIP.2020.2976814

Gao, J., Gong, M., and Li, X. (2021). Congested crowd instance localization with dilated convolutional swin transformer. *arXiv preprint arXiv:2108.00584*. doi: 10.1016/j.neucom.2022.09.113

He, K., Sun, J., and Tang, X. (2010). Single image haze removal using dark channel prior. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 2341–2353. doi: 10.1109/TPAMI.2010.168

Huang, L., Tan, J., Liu, J., and Yuan, J. (2020a). "Hand-transformer: non-autoregressive structured modeling for 3D hand pose estimation," in *European Conference on Computer Vision* (Glasgow: Springer), 17–33.

Huang, L., Tan, J., Meng, J., Liu, J., and Yuan, J. (2020b). "Hot-net: non-autoregressive transformer for 3D hand-object pose estimation," in *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle), 3136–3145.

Huang, P., Zhao, L., Jiang, R., Wang, T., and Zhang, X. (2021). Self-filtering image dehazing with self-supporting module. *Neurocomputing* 432, 57–69. doi: 10.1016/j.neucom.2020.11.039

Huang, Z., Li, J., Hua, Z., and Fan, L. (2022). Underwater image enhancement via adaptive group attention-based multiscale cascade transformer. *IEEE Trans. Instrum. Meas.* 71, 1–18. doi: 10.1109/TIM.2022.3189630

Kar, A., Dhara, S. K., Sen, D., and Biswas, P. K. (2020). Transmission map and atmospheric light guided iterative updater network for single image dehazing. *arXiv preprint arXiv:2008.01701*. doi: 10.48550/arXiv.2008.01701

Kim, J., Lee, J. K., and Lee, K. M. (2016). "Deeply-recursive convolutional network for image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV: IEEE), 1637–1645.

Kingma, D. P., and Ba, J. (2014). Adam: a method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. doi: 10.48550/arXiv.1412.6980

Li, B., Ren, W., Fu, D., Tao, D., Feng, D., Zeng, W., et al. (2018). Benchmarking single-image dehazing and beyond. *IEEE Trans. Image Process.* 28, 492–505. doi: 10.1109/TIP.2018.2867951

Li, J., Feng, X., and Hua, Z. (2021). Low-light image enhancement via progressive-recursive network. *IEEE Trans. Circ. Syst. Video Technol.* 31, 4227–4240. doi: 10.1109/TCSVT.2021.3049940

Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., and Timofte, R. (2021). Swinir: image restoration using swin transformer. *arXiv preprint arXiv:2108.10257*. doi: 10.1109/ICCVW54120.2021.00210

Lin, A., Chen, B., Xu, J., Zhang, Z., and Lu, G. (2021a). Ds-transunet: dual swin transformer u-net for medical image segmentation. *arXiv preprint arXiv:2106.06716*. doi: 10.1109/TIM.2022.3178991

Lin, K., Wang, L., and Liu, Z. (2021b). "End-to-end human pose and mesh reconstruction with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Nashville, TN: IEEE), 1954–1963.

Liu, J., Wu, H., Xie, Y., Qu, Y., and Ma, L. (2020a). "Trident dehazing network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (Seattle, WA: IEEE), 430–431.

Liu, Q., Qin, Y., Xie, Z., Cao, Z., and Jia, L. (2020b). An efficient residual-based method for railway image dehazing. *Sensors* 20, 6204. doi: 10.3390/s20216204

Liu, X., Ma, Y., Shi, Z., and Chen, J. (2019). "Griddehazenet: Attention-based multi-scale network for image dehazing," in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (Seoul: IEEE), 7314–7323.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., et al. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*. doi: 10.1109/ICCV48922.2021.00986

Middleton, W. E. K. (2019). *Vision Through the Atmosphere*. Toronto: University of Toronto Press.

Mittal, A., Soundararajan, R., and Bovik, A. C. (2012). Making a "completely blind" image quality analyzer. *IEEE Signal Process Lett.* 20, 209–212. doi: 10.1109/LSP.2012.2227726

Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, L., Shazeer, N., Ku, A., et al. (2018). "Image transformer," in *International Conference on Machine Learning* (Macao), 4055–4064.

Qin, X., Wang, Z., Bai, Y., Xie, X., and Jia, H. (2020). "Ffa-net: Feature fusion attention network for single image dehazing," in *Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34* (New York, NY), 11908–11915.

Ren, D., Zuo, W., Hu, Q., Zhu, P., and Meng, D. (2019). "Progressive image deraining networks: a better and simpler baseline," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Long Beach, CA: IEEE), 3937–3946.

Ren, W., Pan, J., Zhang, H., Cao, X., and Yang, M.-H. (2020). Single image dehazing via multi-scale convolutional neural networks with holistic edges. *Int. J. Comput. Vis.* 128, 240–259. doi: 10.1007/s11263-019-01235-8

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). "Attention is all you need," in *Advances in Neural Information Processing Systems* (Long Beach, CA), 5998–6008.

Wang, C., Wu, Y., Su, Z., and Chen, J. (2020). "Joint self-attention and scale-aggregation for self-calibrated deraining network," in *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle), 2517–2525.

Wang, N., Cui, Z., Su, Y., and Li, A. (2021). Rgnam: recurrent grid network with an attention mechanism for single-image dehazing. *J. Electron. Imaging* 30, 033026. doi: 10.1117/1.JEI.30.3.033026

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.* 13, 600–612. doi: 10.1109/TIP.2003.819861

Wu, B., Xu, C., Dai, X., Wan, A., Zhang, P., Yan, Z., et al. (2020). Visual transformers: token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*. doi: 10.48550/arXiv.2006.03677

Xie, Z., Lin, Y., Yao, Z., Zhang, Z., Dai, Q., Cao, Y., et al. (2021). Self-supervised learning with swin transformers. *arXiv preprint arXiv:2105.04553*. doi: 10.48550/arXiv.2105.04553

Yamak, P. T., Yujian, L., and Gadosey, P. K. (2019). "A comparison between arima, lstm, and gru for time series forecasting," in *Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence* (Sanya), 49–55.

Yan, J., Li, C., Zheng, Y., Xu, S., and Yan, X. (2020). Mmp-net: a multi-scale feature multiple parallel fusion network for single image haze removal. *IEEE Access* 8, 25431–25441. doi: 10.1109/ACCESS.2020.2971092

Yeh, C.-H., Huang, C.-H., and Kang, L.-W. (2019). Multi-scale deep residual learning-based single image haze removal via image decomposition. *IEEE Trans. Image Process.* 29, 3153–3167. doi: 10.1109/TIP.2019.2957929

Yue, X., Sun, S., Kuang, Z., Wei, M., Torr, P., Zhang, W., et al. (2021). Vision transformer with progressive sampling. *arXiv preprint arXiv:2108.01684*. doi: 10.1109/ICCV48922.2021.00044

Zamir, S. W., Arora, A., Khan, S., Hayat, M., Khan, F. S., Yang, M.-H., et al. (2021). "Multi-stage progressive image restoration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Nashville, TN: IEEE), 14821–14831.

Zhang, T., Li, J., and Fan, H. (2022). Progressive edge-sensing dynamic scene deblurring. *Comput. Vis. Media* 8, 495–508. doi: 10.1007/s41095-021-0246-4

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., et al. (2021). "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Nashville, TN: IEEE), 6881–6890.

Zhu, Q., Mai, J., and Shao, L. (2015). A fast single image haze removal algorithm using color attenuation prior. *IEEE Trans. Image Process.* 24, 3522–3533. doi: 10.1109/TIP.2015.2446191

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., and Dai, J. (2020). Deformable detr: deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*. doi: 10.48550/arXiv.2010.04159

TYPE Original Research
PUBLISHED 15 December 2022
DOI 10.3389/fnbot.2022.1075520

A lightweight multi-dimension dynamic convolutional network for real-time semantic segmentation

Chunyu Zhang[1]*, Fang Xu[2], Chengdong Wu[1] and Chenglong Xu[3]

[1]Faculty of Robot Science and Engineering, Northeastern University, Shenyang, China, [2]Shenyang Siasun Robot & Automation Company Ltd., Shenyang, China, [3]College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin, China

OPEN ACCESS

EDITED BY
Xin Jin,
Yunnan University, China

REVIEWED BY
Yihan Lin,
Tsinghua University, China
Morteza Shariatee,
Iowa State University, United States
Rohit Shukla,
University of Wisconsin-Madison,
United States

*CORRESPONDENCE
Chunyu Zhang
1910654@stu.neu.edu.cn

RECEIVED 20 October 2022
ACCEPTED 25 November 2022
PUBLISHED 15 December 2022

CITATION
Zhang C, Xu F, Wu C and Xu C (2022) A lightweight multi-dimension dynamic convolutional network for real-time semantic segmentation. *Front. Neurorobot.* 16:1075520. doi: 10.3389/fnbot.2022.1075520

Semantic segmentation can address the perceived needs of autonomous driving and micro-robots and is one of the challenging tasks in computer vision. From the application point of view, the difficulty faced by semantic segmentation is how to satisfy inference speed, network parameters, and segmentation accuracy at the same time. This paper proposes a lightweight multi-dimensional dynamic convolutional network (LMDCNet) for real-time semantic segmentation to address this problem. At the core of our architecture is Multidimensional Dynamic Convolution (MDy-Conv), which uses an attention mechanism and factorial convolution to remain efficient while maintaining remarkable accuracy. Specifically, LMDCNet belongs to an asymmetric network architecture. Therefore, we design an encoder module containing MDy-Conv convolution: MS-DAB. The success of this module is attributed to the use of MDy-Conv convolution, which increases the utilization of local and contextual information of features. Furthermore, we design a decoder module containing a feature pyramid and attention: SC-FP, which performs a multi-scale fusion of features accompanied by feature selection. On the Cityscapes and CamVid datasets, LMDCNet achieves accuracies of 73.8 mIoU and 69.6 mIoU at 71.2 FPS and 92.4 FPS, respectively, without pre-training or post-processing. Our designed LMDCNet is trained and inferred only on one 1080Ti GPU. Our experiments show that LMDCNet achieves a good balance between segmentation accuracy and network parameters with only 1.05 M.

KEYWORDS

semantic segmentation, lightweight network, dynamic convolution, encoder-decoder, multi-dimension convolution

1 Introduction

Semantic segmentation, widely used in the real world, classifies every pixel of a visual image. Semantic segmentation visualization uses different colors to distinguish different classes of objects effectively. Semantic segmentation is mainly used in scene analysis, including medical imaging, autonomous driving, and satellite maps. Semantic segmentation has become one of the most critical tasks in computer vision.

Fully convolutional networks (FCN) (Long et al., 2015) pioneered the end-to-end training of neural networks, and many semantic segmentation networks use a full convolution approach to network construction. U-Net (Ronneberger et al., 2015) adopts a symmetric network structure and fuses high-level and low-level semantic information in decoding. SegNet (Badrinarayanan et al., 2017) introduces a pooling operation with pixel indices to optimize segmentation details at the decoder stage. In order to achieve higher segmentation accuracy, high-precision networks such as DeepLab series (Chen et al., 2017a,b, 2018), APCNet (He et al., 2019), and CANet (Zhang et al., 2019) have been proposed one after another. In practical application scenarios, slow inference speed and many parameters are the main reasons semantic segmentation cannot be applied. On the Cityscape dataset (Cordts et al., 2016), networks that meet the 80% accuracy requirement have inference speeds below 10 FPS or model parameters over 100 M. Lightweight real-time semantic segmentation research is imminent.

Lightweight real-time semantic segmentation requires a neural network that perfectly balances segmentation accuracy and parameter quantity. Typical lightweight real-time semantic segmentation networks are SegNet, ENet (Paszke et al., 2016), ICNet (Zhao et al., 2018), ERFNet (Romera Carmena et al., 2018), CGNet (Wu et al., 2020), BiSeNet (Yu et al., 2018), EDANet (Mehta et al., 2018), ESPNetV2 (Mehta et al., 2019), ESNet (Wang et al., 2019b), DABNet (Li G. et al., 2019), LEDNet (Wang et al., 2019a), DFANet (Li H. et al., 2019), FDDWNet (Liu et al., 2020), LRNNet (Jiang et al., 2020), LRDNet (Zhuang et al., 2021), JPANet (Hu et al., 2022), LEANet (Zhang et al., 2022) and our LMDCNet, As shown in **Figure 1**. When applying semantic segmentation, our first consideration is segmentation accuracy. PSPNet (Lv et al., 2021) pursues the fusion of multi-scale information, and SFNet (Lo et al., 2019) performs scale alignment of different features. The accuracy of these networks meets practical requirements, but the device's computing power is too demanding. To overcome the memory requirement of the algorithm, ESPNetV2 proposes dilated convolutions for semantic segmentation, mainly to increase the receptive field. BiSeNetV2 adds a spatial branch to compensate for the loss of details in semantic segmentation. STDC-Seg designs the coding backbone network to reduce the number of parameters. These algorithms are less demanding on equipment but have poor segmentation accuracy.

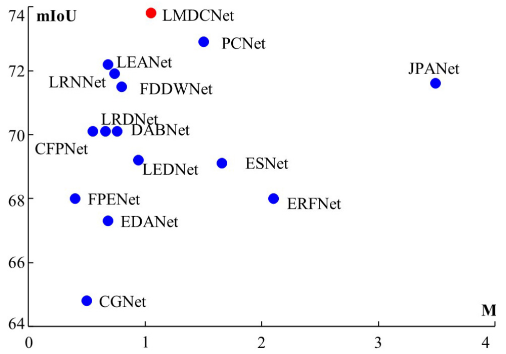


FIGURE 1
Accuracy of segmentation (mIoU) and network parameters (M) derived from Cityscapes test set. Clearly, our LMDCNet achieves the optimal balance between segmentation accuracy and parameters.

This paper proposes a lightweight multi-dimensional dynamic convolutional network (LMDCNet) to solve the problem of unbalanced accuracy and parameters. The network adopts an asymmetric structure; the relevant details are shown in **Figure 2**. We design a new multi-dimensional dynamic convolution (MDy-Conv), which uses an attention mechanism for convolution and linearly combines multiple factorial convolutions to find a convolution kernel suitable for the current feature. Specifically, the operation flow is shown in **Figure 3**. We design the MS-DAB module to include MDy-Conv, residual connections, and channel shuffling operations. The encoder structure performs channel separation to reduce computational complexity. MDy-Conv is used to improve the coding performance, and channel shuffling improves the robustness of the network. Residual connections are used to reuse features and reduce the difficulty of training. The overall structure of the encoder is designed to achieve a perfect balance of encoding performance and parameters. We design a decoder with a feature pyramid structure, spatial attention, and channel attention: SC-FP. Feature Pyramid Module (FP) obtains multi-scale contextual information of features. Combining spatial and channel attention for efficient feature selection improves computational efficiency. To improve segmentation accuracy, SC-FP achieves a good balance between feature space details and computational network cost.

In brief, we have the following contributions:

1. A multi-dimensional dynamic convolution (MDy-Conv) is proposed. It adopts an attention mechanism for convolution and linearly combines multiple convolution kernels to find the best convolution kernel that conforms to the current feature encoding, thereby improving the encoding ability;
2. We propose a depth-asymmetric bottleneck module with multi-dimensional dynamic convolution and shuffling

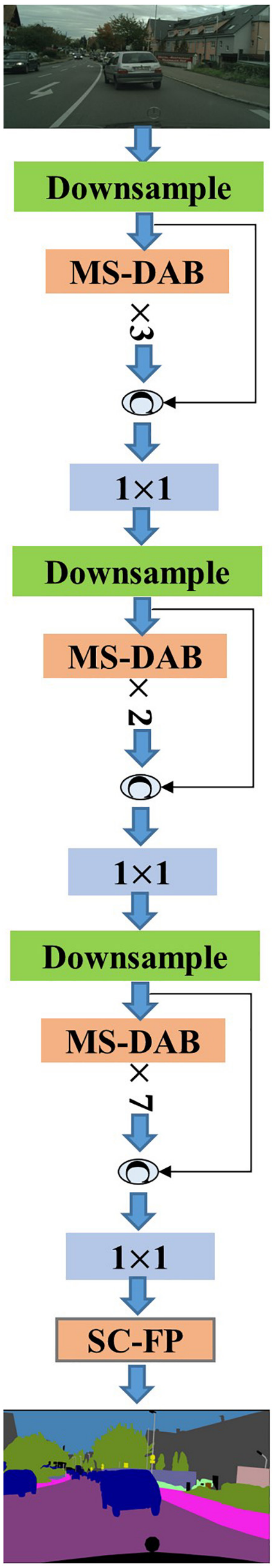


FIGURE 2
Overview architecture of the proposed LMDCNet.

operations (MS-DAB module). It can effectively extract local and contextual information about features and fuse them. The MS-DAB module is far superior to similar modules in segmentation accuracy and parameters;

3. A feature pyramid (SC-FP module) with spatial and channel attention is proposed. The simplified feature pyramid incorporates multi-scale contextual information and uses spatial and channel attention for feature selection. Combining the two algorithms can extract more effective information during decoding and improve segmentation accuracy;
4. Using MS-DAB and SC-FP modules, create a Lightweight Multi-dimensional Dynamic Convolutional Network (LMDCNet). Evaluation results on the Cityscape dataset show that LMDCNet outperforms state-of-the-art networks, achieving the best balance between segmentation accuracy and parameters. On the CamVid dataset, the segmentation accuracy surpasses the current algorithms and reaches the top level.

2 Related work

In this section, we introduce algorithms related to lightweight real-time semantic segmentation, including the following: Dilated convolution, Attention mechanism, and Lightweight semantic segmentation network.

2.1 Dilated convolution

Dilated convolution is one of the standard methods for lightweight real-time semantic segmentation to reduce the number of parameters. This convolution has an additional hyper-parameter, called dilated rate, to represent the number of intervals in the kernel (e.g., the standard convolution is dilated rate 1). Yu and Koltun (2015) first applied dilated convolution to semantic segmentation algorithms. Later, the DeepLab series and DABNet, among others, borrowed the method further to improve the segmentation accuracy of semantic segmentation networks. Dilated convolution increases the convolutional receptive field and acquires contextual information. However, the dilated convolution produces grid effects due to adding 0 elements. Wang et al. (2018) proposed a hybrid dilated convolution that uses different dilated rates for each layer of the network so that the receptive field covers the entire region.

2.2 Attention mechanism

The role of the attention mechanism is to select features, highlight important information, and suppress unnecessary information. In order to make full use of limited visual information processing resources, attention is required to select features during information processing. SENet (Hu et al., 2018) (Squeeze and Excitation Network) is typical channel attention, and its purpose is to select feature channels. ECANet (Wang et al., 2020) is an enhanced version of SENet with a detailed explanation of channel attention. Convolutional block attention module (CBAM) (Woo et al., 2018) connects channel attention and spatial attention to form a hybrid attention mechanism.

2.3 Lightweight semantic segmentation network

Lightweight semantic segmentation network can accomplish on-device semantic segmentation tasks. Low computation, real-time reasoning, and accurate segmentation require lightweight semantic segmentation for practical tasks. At this stage, the devices that implement lightweight semantic segmentation are 1080Ti, 2080Ti, Titan, and 3080. Their processing power is 1080Ti < 2080Ti < Titan < 3080. We summarize three principles for designing lightweight semantic segmentation at this stage: (1) Improvement of the existing lightweight network backbone. For example, DFANet aims to use a lightweight classification network to encode semantic segmentation. Shuffle-Seg is an application of the lightweight classification network ShuffleNet in the direction of semantic segmentation. (2) Create a lightweight coding module as the coding base. For example, LEDNet uses only decomposed convolutional methods to design coding units. (3) Reduce the loss of segmentation details and increase the network coding branch. For example, BiSeNet designed a semantic segmentation network with spatial and context branches.

3 Materials and methods

In this section, we propose the LMDCNet network to balance the accuracy and the number of parameters for semantic segmentation. In Section "3.1 Multi-dimension dynamic convolution," we propose multi-dimension dynamic convolution (MDy-Conv). We propose a depth-asymmetric bottleneck module with multi-dimension dynamic convolution and shuffling operations (MS-DAB module) and describe it in detail in Section "3.2 MS-DAB module." In Section "3.3 SC-FP module," we propose a feature pyramid module with spatial and channel attention (SC-FP module). Finally, we design the architecture of the whole network in Section "3.4 Network architecture".

3.1 Multi-dimension dynamic convolution

Dynamic convolution has become the focus of attention in recent years. The output y of ordinary convolution is equal to the convolution operation performed by the convolution kernel $conv$ and the input x, and * represents the convolution operation, as shown in Equation 1. Dynamic convolution is a convolution obtained by linearly combining multiple convolution kernels. The current feature obtains the weight in the combination process through correlation processing. As the input features change, the combined weight of the convolution also changes, so it is a dynamic convolution. CondConv (Yang et al., 2019) and DyConv (Chen et al., 2020) are typical dynamic convolutions whose structure is shown in **Figure 3B**. CondConv and DyConv use a modified SE (Squeeze-and-Excitation) attention structure to calculate convolution weights. The convolution kernel obtained by multiplying the weight with multiple convolutions and then adding them is dynamic convolution. The specific operation process is shown in **Figure 3B**. Then the output of a typical dynamic convolution follows Equation 2, where $\odot$ represents the multiply add operation, and a_C represents the convolution combination weight vector obtained by processing in the channel-wise direction. *Conv* represents the list of convolution kernels. The combined weight of this dynamic convolution is derived from the channel direction of the feature, the information obtained is limited, the convolution kernel cannot be linearly combined, and the generated dynamic convolution could be more optimal.

$$y = \boldsymbol{conv} * \boldsymbol{x} \tag{1}$$

$$y = (\alpha_C \odot \boldsymbol{Conv}) * \boldsymbol{x} \tag{2}$$

$$y = (\{\alpha_W + \alpha_H + \alpha_C\} \odot \boldsymbol{Conv}) * \boldsymbol{x} \tag{3}$$

As shown in **Figure 3A**, the feature map contains three dimensions, namely height (H), width (W), and channel (C). Locating a point in the feature map requires three dimensions to work together, and a single dimension cannot lock a point. Similarly, a single feature channel direction cannot determine the optimal convolution combination (i.e., weight), and three directions must work together. Based on the above arguments, we design a multi-dimensional dynamic convolution (MDy-Conv), and the detailed operation flow is shown in **Figure 3C**. The dynamic convolution we designed contains the information on the feature map's three directions (H, W, and C), and the resulting convolution kernel combined weight is optimal. The specific description of the dynamic

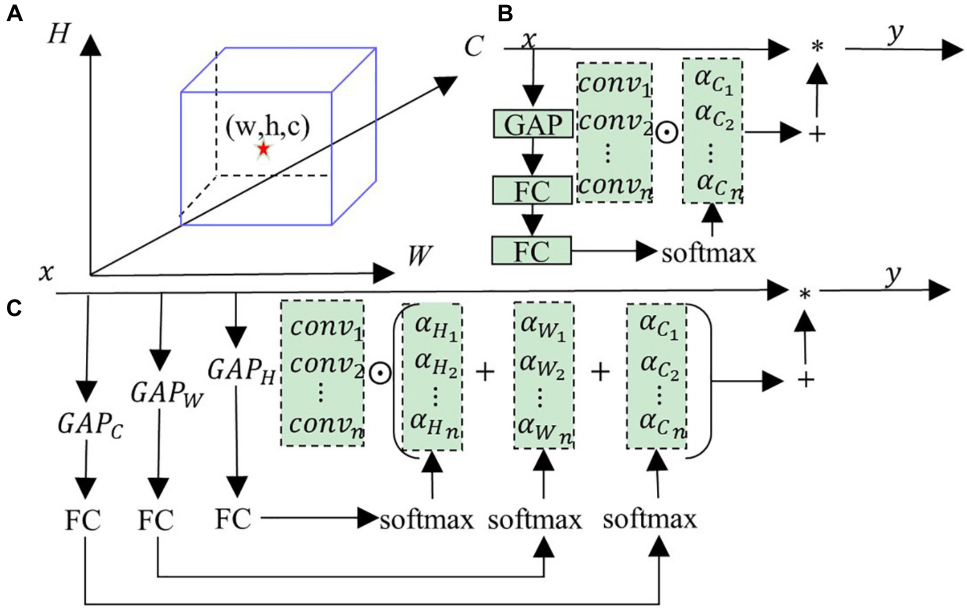


FIGURE 3
(A) Shows the feature map, **(B)** shows the typical dynamic convolution structure, **(C)** shows the multi-dimensional dynamic convolution structure (MDy-Conv).

convolution generated by the feature x is as follows: (1) Perform global average pooling (GAP) on the three directions (height, width, and channel) of the feature x to obtain three tensors ($c \times 1 \times 1$, $h \times 1 \times 1$, $w \times 1 \times 1$), where (c, w, h) represent the channel, width, and height values, respectively; (2) They are sent to 3 fully connected layers (FC) and softmax, respectively, to obtain the exact size tensor ($r \times 1 \times 1$), where the size of r represents the number of convolutions participating in the calculation; (3) Add the three tensors to get the final convolution weight, and its size is also ($r \times 1 \times 1$); and (4) Multi-dimensional dynamic convolution (MDy-Conv) that performs multiplication and addition operations on r convolutions and convolution weights. The mathematical expression of multi-dimensional dynamic convolution is shown in Equation 3, (a_C, a_H, a_W) represents the tensor obtained by feature x after global average pooling, fully connected layer, and softmax.

As shown in Figure 3, the differences between our multi-dimensional dynamic convolution and others are: First, we entirely use the information in the feature map to find the optimal solution for the combination of convolutions. In contrast, ordinary dynamic convolution only considers channel direction. Second, we use a single-layer fully connected layer, traditional dynamic convolution uses two layers, and we have fewer parameters. Third, the performance of our designed dynamic convolutional encoding is stronger than other dynamic convolutions, which we verified in comparative experiments.

3.2 MS-DAB module

The coding module of the lightweight real-time semantic segmentation network design pays more attention to the coding ability and the number of parameters. Most of the encoding modules adopt the structure of ResNet's residual module. As shown in Figure 4, ERFNet designs a non-bottleneck-1D module using decomposed convolutions. ShuffleNet designs a lightweight real-time encoding model using group and depth-wise separable convolution. The DAB module uses asymmetric depth-wise separable convolution and asymmetric depth-wise dilated separable convolution.

Based on the above observations, our MS-DAB module design is shown in Figure 4D. First, we use a channel separation technique to segment the input features in the channel direction, thereby reducing the computational complexity. The depth-wise separable dilated convolution and dynamic convolution can improve the expressiveness of the model without increasing the network width and depth. Therefore, we replace the 3 × 3 convolutions in the first branch with 3 × 1 convolutions and 1 × 3 convolutions. We replace the standard 3 × 3 convolution in the second branch with 3 × 1 and 1 × 3 depth-wise multi-dimensional dynamic convolution. In order to achieve a better encoding effect, the feature maps of the two branches are spliced together, and 1 × 1 convolution is used to perform information fusion between feature map channels. In order to increase the receptive field of the module and obtain the contextual information of the feature, we add a 3 × 1 and

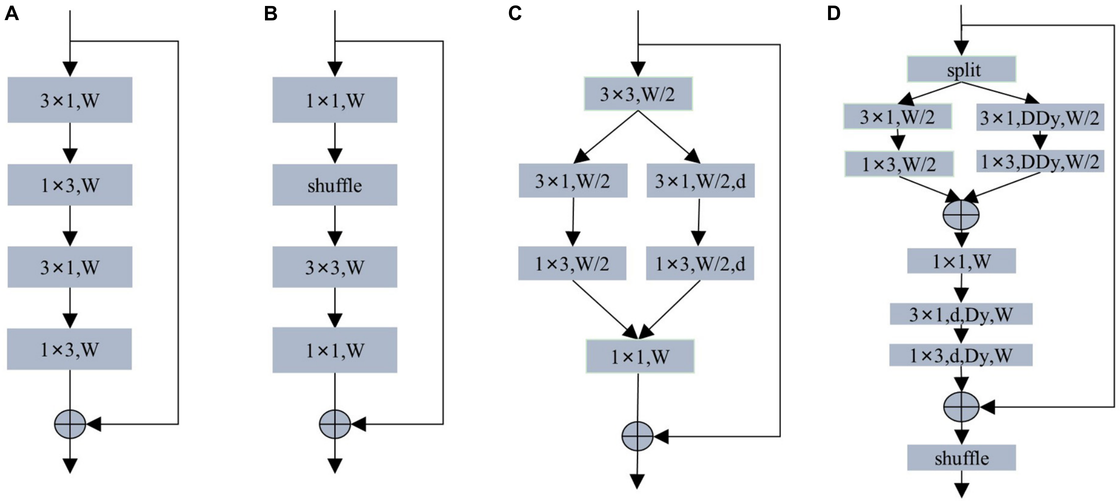


FIGURE 4
(A) Non-bottleneck-1D module. **(B)** ShuffleNet module. **(C)** DAB module. **(D)** Our MS-DAB module. W denotes the number of input channels. d denotes dilated convolution. DDy denotes depth-separable dynamic convolution. Dy denotes dilated dynamic convolution. For brevity, the batch normalization and activation functions are not marked.

a 1 × 3 multi-dimensional dilated dynamic convolution. Afterward, residual connections are used to improve feature utilization and simplify training. Finally, we use the shuffle operation in Figure 4B to enhance the robustness of the encoder.

Compared with the residual module of the same type, our MS-DAB module has the following advantages: First, we introduce MDy-Conv convolution in the residual module, which improves the encoding ability of the module; Second, the module adopts feature channel separation. The operation is separated from the convolution depth to reduce the computational complexity; Thirdly, the hollow multi-dimensional dynamic convolution is introduced to increase the receptive field of the encoder and improve the segmentation accuracy; Finally, channel shuffling and residual connection are used to improve the robustness of the network, reduce the difficulty of training.

3.3 SC-FP module

The image segmentation scene is complex and changeable, and simple upsampling will lose details. Moreover, most lightweight real-time semantic segmentation adopts three coding stages, resulting in a too-small receptive field. Lightweight real-time semantic segmentation requires the decoding part to increase the receptive field, improve multi-scale information fusion, and reduce the loss of details. Therefore, we design a decoding module feature pyramid with spatial and channel attention (SC-FP module) that includes feature pyramid structure, spatial attention, and channel attention mechanisms. Feature pyramid structure can fuse multi-scale context information while increasing the receptive field of the network and reducing the loss of details. FPN proposes a feature pyramid structure, as shown in Figure 5A. FPN works well for multi-scale object recognition. However, too many layers exist in each encoding stage, resulting in an enormous computational burden. Channel Attention (CA) and Spatial Attention (SA) can perform feature selection on both channels and spaces, and the specific structures are shown in Figures 5B, C.

Based on the above observations, we designed the SC-FP module, as shown in Figure 5D. It integrates feature pyramid, channel attention, and spatial attention, effectively enhancing the ability to capture multi-scale contextual information and reducing the loss of image details. The decoder contains four branches: feature pyramid branch, channel attention branch, spatial attention branch, and channel compression branch. The feature pyramid branch comprises 3 × 3, 5 × 5, and 7 × 7 convolutions. Due to the smaller resolution of the features, using larger convolution kernels brings little computational burden. To further improve the performance, a channel attention branch is introduced. Channel attention consists of global max-pooling, global average-pooling, and two fully connected layers. Unlike other channel attention, we adopt a double pooling operation, which can obtain more channel information. The third branch is the feature channel compression branch, where the 1 × 1 convolution fuses the information between different channels to make the output channel equal to the segmentation category. Considering that the loss of details in lightweight real-time semantic segmentation seriously affects the segmentation accuracy, a spatial attention branch is introduced to integrate the global

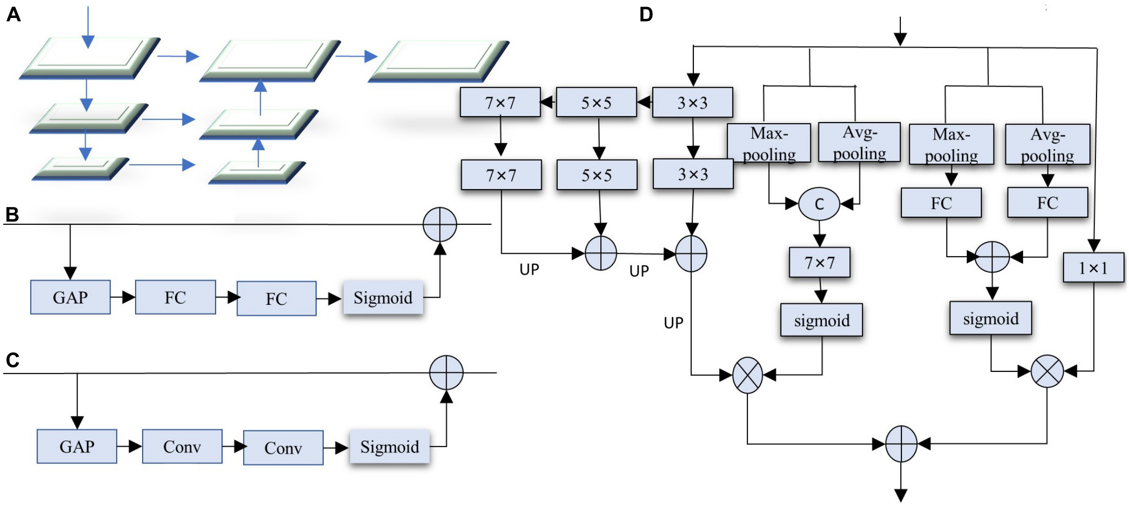


FIGURE 5
(A) Feature pyramid network (FPN). **(B)** Channel attention (CA). **(C)** Spatial attention (SA). **(D)** Our SC-FP module. For brevity, the batch normalization and activation functions are not marked.

context. Spatial attention includes global average-pooling, global max-pooling, and 7 × 7 convolutions. Channel attention performs channel selection on the result of the 1 × 1 convolution, while spatial attention acts on the output of the pyramid to highlight detailed information. Finally, the two results are added point by point to generate the decoded feature map.

Our SC-FP module has the following advantages: First, it adopts a feature pyramid structure to increase the receptive field of the network, capture multi-scale context information, reduce the loss of details, and improve network performance. Second, it introduces a dual attention mechanism to integrate context information further, increase attention to detail information, and improve segmentation accuracy; Third, to reduce the computational burden, point-by-point multiplication or addition is used for feature fusion. Although a larger convolution kernel is used, the feature map resolution is lower and does not increase the computational complexity.

3.4 Network architecture

Our main objective in this work is to create a compact model that can strike the best balance between segmentation accuracy and network parameters. We propose the LMDCNet depicted in Figure 2 utilizing the SC-FP and MS-DAB modules to achieve this. The specific architecture of our LMDCNet, which has an asymmetric encoder-decoder, is displayed in Table 1.

In the encoder section of LMDCNet, we created three downsampling blocks and three encoder stages. The initial block in ENet, a cascaded output of 3 × 3 convolution with step 2 and a 2 × 2 pooling, serves as the downsampling block. The downsampling operation produces thumbnails of the corresponding images, enabling deeper networks to gather more contextual data while requiring less computational work. Downsampling, however, lowers spatial resolution, which typically results in a loss of spatial information and impacts the predictions' outcomes. Therefore, to maintain a good balance, only three downsampling operations—for a total downsampling rate of eight—are carried out in our LMDCNet. The three, two, and seven MS-DAB modules comprise LMDCNet's three encoder stages. We introduce dilated convolution in the MS-DAB module. To solve the grid problem, we follow the design

TABLE 1 The detailed architecture of lightweight multi-dimensional dynamic convolutional network (LMDCNet).

Stage	Type	Channel	Output size
Encoder	Downsampling	32	512 × 256
	MS-DAB × 3	32	512 × 256
	Downsampling	64	256 × 128
	MS-DAB × 2	64	256 × 128
	Downsampling	128	128 × 64
	MS-DAB ($r = 1$)	128	128 × 64
	MS-DAB ($r = 2$)	128	128 × 64
	MS-DAB ($r = 5$)	128	128 × 64
	MS-DAB ($r = 2$)	128	128 × 64
	MS-DAB ($r = 5$)	128	128 × 64
	MS-DAB ($r = 9$)	128	128 × 64
	MS-DAB ($r = 17$)	128	128 × 64
Decoder	SC-FP	C	128 × 64
	Upsampling	C	1024 × 512

"Channel" denotes the number of output feature maps, and "C" is the number of classes. "Output size" denotes the output size with an input size of 1024 × 512.

concept of HDC (hybrid dilated convolution) when designing the dilation rates: First, the adjacent dilation rates cannot be greater than the common divisor of 1; Second, the dilation rates can be designed as a zigzag structure, such as (1, 2, 5, 2, 5, 7); Third, the final dilation rates should cover the maximum segmentation target. The specific design of the network is as follows: the dilation rates of the first stage and the second stage are set to 1, and the dilation rates of the third stage is set to (1, 2, 5, 2, 5, 9, 17).

Many lightweight real-time networks remove the decoder part, and proper decoding can improve network accuracy. The decoder includes the SC-FP module and the upsampling module; obviously, our network architecture is asymmetric. The SC-FP module contains feature pyramids and attention, which can refine the detailed information on segmentation and the selection of features. The feature map size does not match the input image size, and a bilinear interpolation algorithm is needed to recover the feature map resolution. The parameters of the decoder part are few but can effectively improve the segmentation accuracy. Our network has no complicated data processing links in the training process, and the number of parameters is only 1.05 M.

4 Experiments

In this section, we evaluate the performance of our designed LMDCNet on two challenging public datasets, the Cityscapes, and CamVid datasets. We first introduce the two datasets used in the experiments and the implementation details. The effectiveness of each LMDCNet component is then demonstrated using a series of ablation experiments on the Cityscapes validation set. Finally, we present evaluation results on the CamVid and Cityscapes test sets and comparisons with other lightweight real-time semantic segmentation networks.

4.1 Datasets

4.1.1 Cityscape dataset

Cityscape dataset is a large dataset for semantic segmentation for training. The dataset contains 5000 finely labeled images and 20,000 coarsely labeled images. Usually, fine-labeled images are used for network training, and coarse images are used for network migration for pre-training. The resolution of the images is 1024 × 2048, and the default classification label is 19 classes. We compressed the image resolution to 512 × 1024 to improve the inference speed.

4.1.2 CamVid dataset

CamVid dataset uses street scenes from video sequences as semantic segmentation training data. The dataset has 701 high-quality training images, of which 367 are the training set, 101 are the validation set, and 233 are the test set. The dataset contains 32 semantic categories, and the categories commonly used for network training are 11 categories. The resolution of the images is 720 × 960, and 360 × 480 is used in our training process.

4.2 Implementation details

4.2.1 Environment configuration

The model creation and training were based on the Pytorch platform with CUDA 9.0 and cuDNN 7, and all experiments were conducted on a machine outfitted with an Intel i7-10700K CPU and a single NVIDIA GTX 1080Ti GPU (11G).

4.2.2 Network training configuration

We did not employ any additional datasets as network preprocessing. We used small batch stochastic gradient descent (SGD) during the training process as the optimization function with a weight decay of 2e-4 and a momentum of 0.9. The batch processing size is 8 for the Cityscapes dataset and 16 for the CamVid dataset. The cross-entropy loss function is used for the loss function. The initial learning rate for the Cityscapes dataset is 4.5e-2, and the CamVid dataset is 1e-3 using the "poly" learning rate technique. The current epoch learning rate is $lr = init\_lr \times \left(1 - epoch/\max\_epoch\right)^{power}$, where power is 0.9.

4.2.3 Data augmentation

Data augmentation reduces the risk of training overfitting. Our experiments used the following methods for data augmentation: average subtraction, random level flipping, and random scaling. The scales of random scaling during training are 0.75, 1.0, 1.25, 1.5, 1.75, and 2.0.

4.2.4 Evaluating indicator

The evaluation metrics of semantic segmentation include three aspects: segmentation accuracy, inference speed, and model size. Segmentation accuracy is measured by mean Intersection over Union (mIoU); inference speed is measured by the number of frames per second (FPS) processed in the image, and model size is measured by the number of statistically learnable parameters (M).

4.2.5 Network performance balance indicator

We designed an optimal balance index to evaluate the accuracy and parameter amount of lightweight real-time semantic segmentation, and it is named as increment rate (IR). The most critical indicators of lightweight real-time semantic segmentation are segmentation accuracy, parameter amount and inference speed. Because the inference speed is related to the verification platform, the speed of comparing lightweight real-time semantic segmentation must be on a unified platform. The standard for evaluating the quality of a lightweight real-time

TABLE 2 Ablation study results of depth-asymmetric bottleneck module with multi-dimensional dynamic convolution and shuffling operations (MS-DAB module).

Type	Model	mIoU (%)	FPS	Params (M)
Baseline	LMDCNet	73.8	71.2	1.05
Ablation for residual module	LMDCNet-Non-bottleneck-1D	68.4	74.3	1.90
	LMDCNet-DAB	69.7	84.1	0.90
	LMDCNet-ShuffleNet	66.8	97.3	0.56
Ablation for dilation rates	4,4,4,4,4,4,4	72.3	70.8	1.05
	2,2,5,5,9,9,17	72.9	70.8	1.05
	2,2,4,4,8,8,16	73.2	70.9	1.05
Ablation for actiation function	Relu	73.4	70.9	1.05
Ablation for convolution	1D	70.2	73.2	1.01
	Cond-Conv	71.6	71.0	1.08
	Dy-Conv	72.8	71.5	1.04

TABLE 3 Ablation study results of feature pyramid with spatial and channel attention (SC-FP module).

Type	Model	mIoU (%)	FPS	Params (M)
Baseline	LMDCNet	73.8	71.2	1.05
Ablation for decoder depth	LMDCNet-1 × 1	72.0	76.4	1.04
Ablation for attention	LMDCNet-CA	73.3	72.9	1.05
	LMDCNet-SA	73.1	73.2	1.05
Ablation for FP kernel size	LMDCNet-K333	73.1	73.2	1.05
	LMDCNet-K235	73.4	72.7	1.05
	LMDCNet-K135	72.8	72.9	1.05

TABLE 4 Evaluation results of our lightweight multi-dimensional dynamic convolutional network (LMDCNet) and other state-of-the-art real-time semantic segmentation models on the Cityscapes test set.

Model	Input size	Pretrain	GPU	mIoU (%)	FPS	Params (M)	IR
SegNet	640 × 360	ImageNet	TitanX	57	16.7	29.5	0.69
ENet	640 × 360	No	TitanX	58.3	135.4	0.4	1.08
ICNet	1024 × 2048	ImageNet	TitanX	69.5	30.3	26.5	0.87
ERFNet	512 × 1024	No	TitanX	68	41.7	2.1	1.23
ESPNet	512 × 1024	No	TitanX	60.3	112	2.1	1.09
BiSeNet	768 × 1536	ImageNet	TitanX	68.4	72.3	5.8	1.16
Fast-SCNN	1024 × 2408	ImageNet	TitanX	68	123.5	1.11	1.25
ESPNetV2	512 × 1024	No	TitanX	66.2	67	1.25	1.21
DFANet	512 × 1024	ImageNet	TitanX	70.3	160	7.8	1.15
LEDNet	512 × 1024	No	1080Ti	69.2	71	0.94	1.27
ESNet	512 × 1024	No	1080Ti	69.1	63	1.66	1.25
DABNet	512 × 1024	No	1080Ti	70.1	104	0.76	1.29
FDDWNet	512 × 1024	No	2080Ti	71.5	60	0.8	1.32
BCPNet	512 × 1024	No	TitanX	68.4	250.4	0.61	1.27
DDPNet	768 × 1536	No	1080Ti	74.0	85.4	2.52	1.32
LEANet	512 × 1024	No	1080Ti	71.9	77.3	0.74	1.35
SFNet	1024 × 2048	No	1080Ti	78.9	26	12.87	1.19
PIDNet-S	1024 × 2048	No	3090	78.8	93.2	7.6	1.29
LMDCNet	512 × 1024	No	1080Ti	73.8	72.1	1.05	1.36

TABLE 5 Evaluation results of each class Intersection over Union (IoU) (%) and class mIoU (%) on the Cityscapes test set.

Model	Ro	Si	Bui	Wa	Fe	Po	Tl	Ts	Ve	Te	Sk	Pe	Ri	Ca	Tru	Bus	Tr	Mo	Bi	Cl
SegNet	96.4	73.2	84.0	28.4	29.0	35.7	39.8	45.1	87.0	63.8	91.8	62.8	42.8	89.3	38.1	43.1	44.1	35.8	51.9	57.0
ENet	96.3	74.2	75.0	32.2	33.2	43.4	34.1	44.0	88.6	61.4	90.6	65.5	38.4	90.6	36.9	50.5	48.1	38.8	55.4	58.3
ICNet	97.1	79.2	89.7	43.2	48.9	61.5	60.4	63.4	91.5	68.3	93.5	74.6	56.1	92.6	51.3	72.7	51.3	53.6	70.5	69.5
ERFNet	97.7	81.0	89.8	42.5	48.0	56.3	59.8	65.3	91.4	68.2	94.2	76.8	57.1	92.8	50.8	60.1	51.8	47.3	61.7	68.0
Fast-SCNN	97.9	81.6	89.7	46.4	48.6	48.3	53.0	60.5	90.7	67.2	94.3	74.0	54.6	93.0	57.4	65.5	58.2	50.0	61.2	68.0
ESPNet	97.0	77.5	76.2	35.0	36.1	45.0	35.6	46.3	90.8	63.2	92.6	67.0	40.9	92.3	38.1	52.5	50.1	41.8	47.2	60.3
ESPNetV2	97.3	78.6	88.8	43.5	42.1	49.3	52.6	60.0	90.5	66.8	93.3	72.9	53.1	91.8	53.0	65.9	53.2	44.2	59.9	66.2
LEDNet	97.1	78.3	90.4	46.5	48.1	60.9	60.4	71.1	91.2	60.0	93.2	74.3	51.8	92.3	61.0	72.4	51.0	43.3	70.2	69.2
ESNet	97.1	78.5	90.4	46.5	48.1	60.1	60.4	70.9	91.1	59.9	93.2	74.3	51.8	92.3	61.0	72.3	51.0	43.3	70.2	69.1
DABNet	97.9	82.0	90.6	45.5	50.1	59.3	63.5	67.7	91.8	70.1	92.8	78.1	57.8	93.7	52.8	63.7	56.0	51.3	66.8	70.1
FDDWNet	98.0	82.4	91.1	52.5	51.2	59.9	64.4	68.9	92.5	70.3	94.4	80.8	59.8	94.0	56.5	68.9	48.6	55.7	67.7	71.5
LEANet	98.1	82.7	91.0	51.0	53.2	58.8	65.9	70.3	92.5	70.5	94.3	81.6	59.9	94.1	52.3	68.2	57.2	55.5	69.8	71.9
LMDCNet	98.2	82.7	91.2	51.4	53.1	59.3	65.8	70.5	92.6	70.2	94.2	81.5	59.8	94.2	52.9	68.1	57.7	55.7	69.7	73.8

semantic segmentation network is that the higher the accuracy, the lower the parameters, and the better the network. It is equivalent to an inverse relationship between the segmentation accuracy and the number of parameters. We must divide the accuracy by the number of parameters. Let us take a simple example: the accuracy of ENet is 58.3, the parameters are 0.4, the ratio of accuracy to parameters is 145.75, the accuracy of DABNet is 70.1, and the parameters are 0.76, then the accuracy and parameters ratio is 92.27. We know that DABNet is recognized as a network with much better performance than ENet. However, the ratio of accuracy to parameters is higher than DABNet, which shows that the relationship between accuracy and parameters is not $y = a \times x$. There is an offset b between them. The relationship is $y = a \times \left(x + b\right)$. We have sorted out the formula:

$$\boldsymbol{a = y/(x + b)} \tag{4}$$

Among them, y represents the segmentation accuracy (mIoU), x represents the parameters (M), b represents the offset, and a represents the increment rate (IR). We bring the accuracy of PIDNet-S, 78.8 mIoU and parameter 7.6 M, and the accuracy of 70.1 mIoU and parameter 0.76 M of DABNet, which are recognized as the best lightweight real-time semantic segmentation at this stage, into the formula, and get $b = 53.4$. In this article, we take $b = 53.4$. The calculation formula of the IR is:

$$\boldsymbol{a = y/(x + 53.4)} \tag{5}$$

4.3 Ablation study

4.3.1 Ablation study for MS-DAB module

4.3.1.1 Ablation for residual module

The encoder part of LMDCNet we designed uses the MS-DAB module. To prove the effectiveness of our designed encoder module, we compare the same type's residual modules. We replace the MS-DAB module with a non-bottleneck-1D module, a ShuffleNet module, and a DAB module and test them on the Cityscapes dataset. As seen in Table 2, the LMDCNet network with ShuffleNet's coding module has the lowest number of parameters and the fastest inference speed but the lowest segmentation accuracy. The combined consideration needs to be more competent for the actual segmentation task. On the other hand, the semantic segmentation network using MS-DAB has 0.15 M higher parameters than that using DAB, and the segmentation accuracy is improved by 4.1%, which is a more reasonable performance. The MS-DAB module we designed perfectly balances segmentation accuracy and parameters.

4.3.1.2 Ablation for dynamic convolution

We gradually replaced the multi-dimension dynamic convolution in MS-DAB with the factorial and dynamic

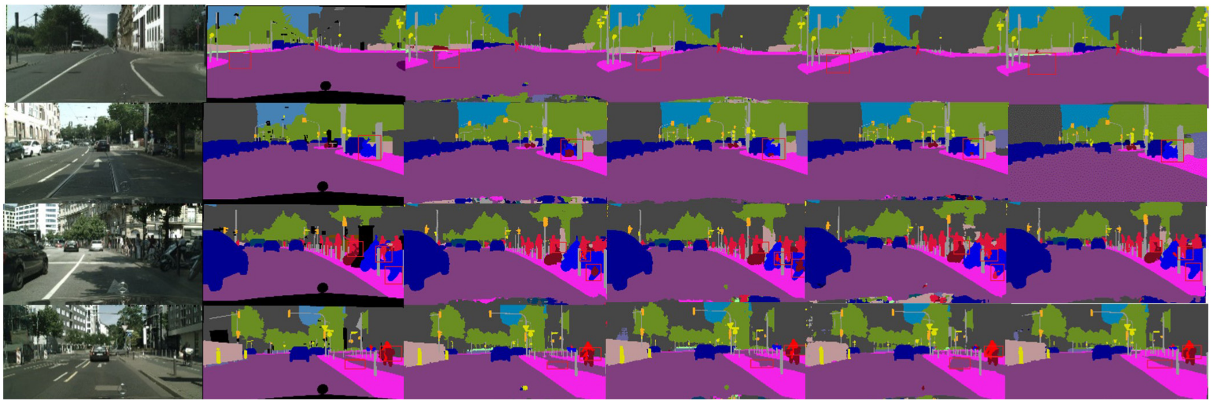


FIGURE 6
Some visual comparisons on the Cityscapes validation set. From left to right are input images, ground truth, predicted results from DABNet, FDDWNet, LEANet, and our LMDCNet.

convolution to confirm that the MDy-Conv we proposed has better experimental results than other convolutions displayed in Figure 3. Table 2 shows that the convolution with the fewest parameters and the fastest inference speed when utilizing the factorial convolution also has the least accurate segmentation. Even though there were 0.01 M more parameters with the MDy-Conv than with the Dynamic convolution module, segmentation accuracy increased by 1.0%, demonstrating the excellent efficiency of our MDy-Conv.

4.3.1.3 Ablation for dilation rates

The size of the perceptual field of the network affects the segmentation accuracy of the network, and the lightweight real-time network uses dilated convolution to improve the receptive field of the network. A reasonable dilation rate can improve the segmentation accuracy of the network while avoiding grid problems. In order to verify whether the criterion for the dilation rates we designed is correct, we designed four groups of dilation rates for tuning. The dilation rates of the first two stages of our coding part are set to 1, and the third part is set to (4,4,4,4,4,4,4), (2,2,5,5,9,9,17), (2,2,4,4,8,8,16), and (1,2,5,2,5,9,17), respectively. The results from Table 2 show that the segmentation accuracy is the lowest when the dilation rate is set to (2,2,4,4,8,8,16), and the segmentation accuracy is the largest when it is set to (1,2,5,2,5,9,17). Experiments show that the design requirements for the dilation rate of our network should follow the HDC (hybrid dilated convolution) design principle. We design the final dilation rate of the network as: (1, 2, 5, 2, 5, 9, 17).

4.3.1.4 Ablation for activation function

The introduction of nonlinear functions in the network can improve the network performance. The commonly used nonlinear functions in semantic segmentation are Relu and PRelu. We use PRelu in the baseline network and Relu in the comparison network. From the experimental results in Table 2, it is concluded that PRelu is more suitable for the LMDCNet network.

4.3.2 Ablation study for SC-FP module

4.3.2.1 Ablation for decoder module

The SC-FP module is the main component of the decoder in our LMDCNet, which is an integration of encoded features to refine the segmentation categories. However, most real-time semantic segmentation deletes the decoder to pursue inference speed. We replaced the SC-FP module in LMDCNet with 1 × 1 point convolution to justify the design of the SC-FP decoder module. Table 3 shows that 5.2 FPS improves the inference speed of the LMDCNet network with 1 × 1 convolution with 0.01 M parameter reduction, but the segmentation accuracy is decreased by 1.8%. In summary, our design of SC-FP is reasonable.

4.3.2.2 Ablation for channel attention

In designing SC-FP, we utilized the channel attention technique. To illustrate the appropriateness of choosing the channel attention branch in our SC-FP module, we removed the channel attention branch. Table 3 shows that the segmentation accuracy obtained by the decoding module without channel attention is 0.5% lower than that obtained using the SC-FP module. This experiment shows that the channel attention branch we designed can improve the segmentation accuracy of the network.

4.3.2.3 Ablation for spatial attention

We introduce the spatial attention branch in SC-FP, and spatial attention focuses more on the spatial information of segmented targets to improve segmentation accuracy. To demonstrate the role of spatial branching in the decoder, we removed the spatial attention branch for comparison.

TABLE 6 Evaluation results of our lightweight multi-dimensional dynamic convolutional network (LMDCNet) and other state-of-the-art real-time semantic segmentation models on the CamVid test set.

Model	Input size	Pretrain	GPU	mIoU (%)	FPS	Params (M)
SegNet	360 × 480	ImageNet	TitanX	55.6	–	29.5
ENet	360 × 480	No	TitanX	51.3	–	0.4
ICNet	720 × 960	ImageNet	TitanX	67.1	27.8	26.5
CGNet	360 × 480	No	2xV100	65.6	–	0.5
BiSeNet	720 × 960	ImageNet	TitanX	65.6	175	5.8
BiSeNetV2	720 × 960	ImageNet	TitanX	68.7	124.5	49.0
DFANet	720 × 960	ImageNet	TitanX	64.7	120	7.8
DABNet	360 × 480	No	1080Ti	66.2	124.4	0.76
LRNNet	360 × 480	No	1080Ti	67.6	83	0.67
DDPNet	360 × 480	No	1080Ti	67.3	–	1.1
LEANet	360 × 480	No	1080Ti	67.5	98.6	0.74
LMDCNet	360 × 480	No	1080Ti	69.6	92.4	1.04

TABLE 7 Evaluation results of each class Intersection over Union (IoU) (%) and class mIoU (%) on the CamVid test set.

Mode	Bu	Tr	Sk	Ca	Si	Ro	Pe	Fe	Po	Si	Bi	Cl
SegNet	88.8	87.3	92.4	82.1	20.5	97.2	57.1	49.3	27.5	84.4	30.7	55.6
ENet	74.7	77.8	95.1	82.4	51.0	95.1	67.2	51.7	35.4	86.7	34.1	51.3
BiSeNet	82.2	74.4	91.9	80.8	42.8	93.3	53.8	49.7	25.4	77.3	50.0	65.6
BiSeNetV2	83.0	75.8	92.0	83.7	46.5	94.6	58.8	53.6	31.9	81.4	54.0	68.7
DABNet	80.8	73.3	91.0	81.0	40.0	94.8	59.5	56.6	29.8	80.3	41.7	66.2
LEANet	82.0	75.0	91.2	83.2	44.2	94.9	63.2	55.7	30.2	81.1	41.9	67.5
LMDCNet	82.7	76.3	91.7	83.5	46.6	94.5	59.0	53.9	32.4	81.7	53.9	69.6

Table 3 shows that the segmentation accuracy obtained by the decoder module without spatial attention is 0.7% lower than that obtained using the SC-FP module. This test shows that our spatial attention branch can improve the network's ability.

4.3.2.4 Ablation for kernel size

We employ convolutions with kernel sizes of 3 × 3, 5 × 5, and 7 × 7 to obtain various context information scales in the SC-FP module's feature pyramid structure. We use a 3 × 3 kernel (K333) to replace each of the SC-FP module's three convolutions to show how effective this method is. Table 3 displays the experimental results. Additionally, we set up two convolution combinations with smaller kernel sizes: 1 × 1, 3 × 3, 5 × 5 (i.e., K135) and 2 × 2, 3 × 3, 5 × 5 (i.e., K235). Table 3 demonstrates that our SC-FP module performs well when 3 × 3, 5 × 5, and 7 × 7 convolutions are used to construct a feature pyramid structure.

4.4 Evaluation results on cityscapes

The parameters of our designed LMDCNet are 1.05 M, the inference speed on a 1080Ti is 72.1FPS, the segmentation accuracy is 73.8 mIoU, and the increase rate is 1.36. The increment rate represents the balance between the accuracy and parameters of lightweight real-time semantic segmentation, and the larger the increment rate, the better the balance. As can be seen from Table 4, our increase rate is the highest among lightweight real-time semantic segmentation at this stage. The current state-of-the-art lightweight semantic segmentation network PIDNet-S has a growth rate of 1.29, which is smaller than that of the semantic segmentation network we designed. It can be seen that our designed network outperforms PIDNet-S in the balance between accuracy and parameters. The speed of our designed network is 46.1FPS faster than that of SFNet tested on the same 1080Ti platform. The number of parameters is only 1/12 of SFNet. Among the semantic segmentation network with an input resolution of 512 × 1024, our accuracy is the highest, 1.9 mIoU higher than LEANet.

We show the results for each class IoU (%) and class mIoU (%) on the Cityscapes test set in Table 5. Overall, especially in 5 categories, our LMDCNet achieves higher segmentation accuracy, demonstrating the effectiveness of our LMDCNet. Figure 6 shows a visual comparison of the Cityscapes validation set. We can classify different objects more accurately using

LMDCNet and produce more consistent visual outputs across all categories. LMDCNet outperforms ERFNet, DABNet, and FDDWNet in the segmentation of vehicles, riders, and traffic signs.

4.5 Evaluation results on CamVid

Tables 6, 7 show the contrast between LMDCNet and other real-time semantic segmentation models for the CamVid dataset. Our LMDCNet produced effective segmentation results on the CamVid dataset. Without any prior training, our LMDCNet has a segmentation accuracy of 69.6 mIoU. Our LMDCNet can process 360 × 480 images at 92.4 FPS using a 1080Ti GPU for inference speed. In contrast to most real-time semantic segmentation models, LMDCNet has several clear advantages: fewer parameters, excellent segmentation accuracy, and quick inference speed. Our LMDCNet's performance on the CamVid dataset is the best, illustrating its superior adaptability and effectiveness.

5 Conclusion

We present a lightweight multi-dimension dynamic convolutional network (LMDCNet) with an ideal trade-off between model size, segmentation accuracy, and inference speed for real-time semantic segmentation. A multi-dimension dynamic convolution is what we suggest (MDy-Conv). In order to improve convolution presentation and maintain remarkable accuracy, it uses multi-convolutional kernel fusion. Our encoder is a depth-wise asymmetric bottleneck module with multi-dimension dynamic convolution and shuffling operations (MS-DAB module). This module can collect local and contextual information with fewer parameters and less computation. We propose a feature pyramid module (SC-FP module) based on spatial and channel attention for decoding. With minimal computational overhead, this module aggregates context data and generates pixel-level spatial and channel attention to aid in feature selection. According to experiments, our LMDCNet performs exceptionally well with the Cityscapes and CamVid datasets, making it the best option for various road scene interpretation applications.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://www.cityscapes-dataset.com/; http://mi.eng.cam.ac.uk/research/projects/VideoRec/CamVid/.

Author contributions

CZ, FX, CW, and CX were performed material preparation, data collection, and analysis. CZ wrote the first draft of the manuscript. All authors contributed to the study conception and design, commented on previous versions of the manuscript, read, and approved the final manuscript.

Acknowledgments

We would like to thank the reviewers for their constructive and valuable suggestions on the earlier drafts of this manuscript.

Conflict of interest

FX was employed by the company Shenyang Siasun Robot & Automation Company 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

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 2481–2495.

Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2017a). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* 40, 834–848. doi: 10.1109/TPAMI.2017.2699184

Chen, L. C., Papandreou, G., Schroff, F., and Adam, H. (2017b). Rethinking atrous convolution for semantic image segmentation. *arXiv* [Preprint] doi: 10.48550/arXiv.1706.05587

Chen, L. C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, (Cham: Springer), 801–818. doi: 10.1109/TCYB.2021.3085856

Chen, Y., Dai, X., Liu, M., Chen, D., Yuan, L., and Liu, Z. (2020). "Dynamic convolution: Attention over convolution kernels," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Seattle, WA, 11030–11039.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., et al. (2016). "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, 3213–3223. doi: 10.1109/TIP.2020.2976856

He, J., Deng, Z., Zhou, L., Wang, Y., and Qiao, Y. (2019). "Adaptive pyramid context network for semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Long Beach, CA, 7519–7528.

Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Salt Lake City, UT, 7132–7141.

Hu, X., Jing, L., and Sehar, U. (2022). Joint pyramid attention network for real-time semantic segmentation of urban scenes. *Appl. Intell.* 52, 580–594.

Jiang, W., Xie, Z., Li, Y., Liu, C., and Lu, H. (2020). "Lrnnet: A light-weighted network with efficient reduced non-local operation for real-time semantic segmentation," in *Proceedings of the 2020 IEEE international conference on multimedia & expo workshops (ICMEW)*, (Piscataway, NJ: IEEE), 1–6.

Li, G., Yun, I., Kim, J., and Kim, J. (2019). Dabnet: Depth-wise asymmetric bottleneck for real-time semantic segmentation. *arXiv* [Preprint] doi: 10.48550/arXiv.1907.11357

Li, H., Xiong, P., Fan, H., and Sun, J. (2019). "Dfanet: Deep feature aggregation for real-time semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, (Piscataway, NJ: IEEE), 9522–9531. doi: 10.3390/healthcare10081468

Liu, J., Zhou, Q., Qiang, Y., Kang, B., Wu, X., and Zheng, B. (2020). "FDDWNet: A lightweight convolutional neural network for real-time semantic segmentation," in *Proceedings of the ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, Barcelona, 2373–2377.

Lo, S. Y., Hang, H. M., Chan, S. W., and Lin, J.-J. (2019). "Efficient dense modules of asymmetric convolution for real-time semantic segmentation," in *Proceedings of the ACM multimedia Asia*, 1–6.

Long, J., Shelhamer, E., and Darrell, T. (2015). "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Boston, MA, 3431–3440.

Lv, Q., Sun, X., Chen, C., Dong, J., and Zhou, H. (2021). "Parallel complement network for real-time semantic segmentation of road scenes," in *Proceedings of the IEEE transactions on intelligent transportation systems*, (Piscataway, NJ: IEEE).

Mehta, S., Rastegari, M., Caspi, A., Shapiro, L., and Hajishirzi, H. (2018). "Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, (Cham: Springer), 552–568.

Mehta, S., Rastegari, M., Shapiro, L., and Hajishirzi, H. (2019). "Espnetv2: A light-weight, power efficient, and general purpose convolutional neural network," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9190–9200.

Paszke, A., Chaurasia, A., Kim, S., and Culurciello, E. (2016). Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv* [Preprint] doi: 10.48550/arXiv.1606.02147

Romera Carmena, E., Álvarez López, J. M., Bergasa Pascual, L. M., and Arroyo, R. (2018). ERFNet: Efficient residual factorized ConvNet for real-time semantic segmentation. *IEEE Trans. Intell. Transp. Syst.* 19, 263–272.

Ronneberger, O., Fischer, P., and Brox, T. (2015). "U-net: Convolutional networks for biomedical image segmentation," in *International conference on medical image computing and computer-assisted intervention*, (Cham: Springer), 234–241.

Wang, P., Chen, P., Yuan, Y., Liu, D., Huang, Z., Hou, X., et al. (2018). "Understanding convolution for semantic segmentation," in *Proceedings of the 2018 IEEE winter conference on applications of computer vision (WACV)*, (Piscataway, NJ: IEEE), 1451–1460.

Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., and Hu, Q. (2020). Eca-net: Efficient channel attention for deep convolutional neural networks. *arXiv* [Preprint] doi: 10.48550/arXiv.1910.03151

Wang, Y., Zhou, Q., Xiong, J., Wu, X., and Jin, X. (2019b). "Esnet: An efficient symmetric network for real-time semantic segmentation," in *Proceedings of the Chinese conference on pattern recognition and computer vision (PRCV)*, (Cham: Springer), 41–52.

Wang, Y., Zhou, Q., Liu, J., Xiong, J., Gao, G., Wu, X., et al. (2019a). "Lednet: A lightweight encoder-decoder network for real-time semantic segmentation," in *Proceedings of the 2019 IEEE international conference on image processing (ICIP)*, (Piscataway, NJ: IEEE), 1860–1864.

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, (Cham: Springer), 3–19.

Wu, T., Tang, S., Zhang, R., and Zhang, Y. (2020). Cgnet: A light-weight context guided network for semantic segmentation. *IEEE Trans. Image Process.* 30, 1169–1179. doi: 10.1109/TIP.2020.3042065

Yang, B., Bender, G., Le, Q. V., and Ngiam, J. (2019). "Condconv: Conditionally parameterized convolutions for efficient inference," in *Proceedings of the advances in neural information processing systems*, 32.

Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., and Sang, N. (2018). "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 325–341.

Yu, F., and Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. *arXiv* [Preprint] doi: 10.48550/arXiv.1511.07122

Zhang, C., Lin, G., Liu, F., Yao, R., and Shen, C. (2019). "Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Long Beach, CA, 5217–5226.

Zhang, X. L., Du, B. C., Luo, Z. C., and Ma, K. (2022). Lightweight and efficient asymmetric network design for real-time semantic segmentation. *Appl. Intell.* 52, 564–579. doi: 10.1155/2022/2530836

Zhao, H., Qi, X., Shen, X., Shi, J., and Jia, J. (2018). "Icnet for real-time semantic segmentation on high-resolution images," in *Proceedings of the European conference on computer vision (ECCV)*, (Cham: Springer), 405–420.

Zhuang, M., Zhong, X., Gu, D., Feng, L., Zhong, X., and Hu, H. (2021). LRDNet: A lightweight and efficient network with refined dual attention decorder for real-time semantic segmentation. *Neurocomputing* 459, 349–360.

TYPE Original Research
PUBLISHED 22 December 2022
DOI 10.3389/fnbot.2022.1095717

OPEN ACCESS

EDITED BY
Xin Jin,
Yunnan University, China

REVIEWED BY
Yonghao Xu,
Institute of Advanced Research in Artificial Intelligence (IARAI), Austria
Zhe Xu,
Institute of Geography Science and Natural Resources (CAS), China

*CORRESPONDENCE
Fanghong Ye
✉ helenye001@163.com
Zheng Zhou
✉ zhouzheng8416@163.com

RECEIVED 11 November 2022
ACCEPTED 05 December 2022
PUBLISHED 22 December 2022

CITATION
Ye F, Zhou Z, Wu Y and Enkhtur B (2022) Application of convolutional neural network in fusion and classification of multi-source remote sensing data. *Front. Neurorobot.* 16:1095717. doi: 10.3389/fnbot.2022.1095717

Application of convolutional neural network in fusion and classification of multi-source remote sensing data

Fanghong Ye[1,2]*, Zheng Zhou[3]*, Yue Wu[4] and Bayarmaa Enkhtur[5]

[1]Land Satellite Remote Sensing Application Center, Ministry of Natural Resources of People's Republic of China, Beijing, China, [2]School of Resource and Environmental Sciences, Wuhan University, Wuhan, China, [3]Ecology and Environment Monitoring and Scientific Research Center, Ministry of Ecology and Environment of the People's Republic of China, Wuhan, China, [4]Department of Natural Resources of Heilongjiang Province, Heilongjiang Provincial Institute of Land and Space Planning, Harbin, China, [5]Geospatial Information and Technology Department, Agency for Land Administration and Management, Geodesy and Cartography, Ulaanbaatar, Mongolia

Introduction: Through remote sensing images, we can understand and observe the terrain, and its application scope is relatively large, such as agriculture, military, etc.

Methods: In order to achieve more accurate and efficient multi-source remote sensing data fusion and classification, this study proposes DB-CNN algorithm, introduces SVM algorithm and ELM algorithm, and compares and verifies their performance through relevant experiments.

Results: From the results, we can find that for the dual branch CNN network structure, hyperspectral data and laser mines joint classification of data can achieve higher classification accuracy. On different data sets, the global classification accuracy of the joint classification method is 98.46%. DB-CNN model has the highest training accuracy and fastest speed in training and testing. In addition, the DB-CNN model has the lowest test error, about 0.026, 0.037 lower than the ELM model and 0.056 lower than the SVM model. The AUC value corresponding to the ROC curve of its model is about 0.922, higher than that of the other two models.

Discussion: It can be seen that the method used in this paper can significantly improve the effect of multi-source remote sensing data fusion and classification, and has certain practical value.

KEYWORDS

remote sensing image, convolutional neural network, double branch structure, hyperspectral, DB-CNN algorithm, lidar data

1. Introduction

As a depth detection technology, remote sensing is applied to space exploration, urban planning, rescue and disaster relief. It combines multi-disciplinary technologies such as earth science, space science, and computer, so it has different characteristics in terms of scope of use and technical tools (Demir and Ulke, 2020; Zhou et al., 2021c; Du et al., 2022; Lu et al., 2022). However, facing different application scenarios, remote sensing image classification needs higher accuracy, and the accuracy and performance of image classification determine the quality of the application effect. Remote sensing images usually contain a lot of spectral information, which can be used in image recognition and classification (Hu et al., 2021). In remote sensing, classification and recognition of related images is an important function, and different classification and recognition methods have different effects (Yu, 2020). The previous classification methods can not classify well, and the classification results are poor. The classification technology based on the deep learning algorithm has been studied by many scholars because of its high classification effect and performance. Convolutional neural network (CNN) has shown good performance in image feature extraction and classification. In this paper, it is applied to remote sensing image classification to improve its classification accuracy and performance.

2. Related work

In the study of remote sensing images, the main content focuses on the fusion and classification of remote sensing data. During this period, different scholars adopted different research methods. For example, Du et al. (2021) applied methods such as integrated hyperspectral images to extract and analyze remote sensing image features. After verification, it is found that the proposed method can achieve effective classification (Du et al., 2021). In the process of classifying multi-source remote sensing data, Pastorino et al. (2021) designed a hierarchical probabilistic graphical model, which combines Markov framework and decision tree method, which has certain effectiveness and feasibility (Pastorino et al., 2021). In order to improve the classification effect of remote sensing images, Luo et al. (2021) designed a combination strategy based on sorting batch mode, combined with spectral information divergence, and good classification effect can be obtained (Luo et al., 2021). Dong R. et al. (2020) proposed a fast depth-aware network that combines multiple advantages to achieve simultaneous extraction of deep and shallow features (Dong R. et al., 2020). Zhang and Han (2020) used the multi-target classification recognition model when carrying out remote sensing image segmentation and feature extraction. Through correlation verification, it can better perform correlation recognition and has strong robustness (Zhang and Han, 2020). Bazi et al. (2021) proposed a remote sensing image classification model based on the vision converter, in which the context relationship is represented through the multi head attention mechanism. After relevant verification, it is found that the classification effect of this method is better (Bazi et al., 2021). In the process of remote sensing image classification, there will be a problem of data feature distortion. Face this problem, Dong Y. et al. (2020) designed a spectral space weighted popular embedded distribution alignment method, and proved its effectiveness and practical value through experiments (Dong Y. et al., 2020). On the basis of multi-scale feature fusion, Zhang C. et al. (2020) proposed the corresponding remote sensing image classification method, which uses a new weighted eigenvalue convolutional neural network to segment images, and achieved good experimental results (Zhang C. et al., 2020). Xu Y. et al. (2019) analyzed the data fusion contest held in 2018, summarized a variety of multi-source optical remote sensing, analyzed its related land cover classification applications, and the machine vision algorithms involved. The effective combination of machine learning and observation data has become a good data analysis method (Xu Y. et al., 2019). Jin and Mountrakis (2022) classified the land cover types through the random forest algorithm, during which the remote sensing data sources were involved. The results show that the highest overall accuracy of the algorithm is 83.0%, which is much higher than the accuracy of other sensors (Jin and Mountrakis, 2022).

Ma et al. (2020) used improved CNN to classify seismic remote sensing images, and verified the method. After verification, it can have a high accuracy, and its excellent performance has an important role in earthquake prevention and disaster relief (Ma et al., 2020). Pan et al. (2020) corrected the high-resolution remote sensing classification results through end-to-end localization post-processing. This method can achieve effective correction and make the classification results have high accuracy (Pan et al., 2020). Han et al. (2020) designed a classification method combining 3D-CNN and squeeze excitation network to classify relevant sea ice remote sensing images. The practical value of this method has been proved through relevant research (Han et al., 2020). Qing et al. (2021) designed an end-to-end Transformer model and applied it to hyperspectral image classification, and the experimental results showed that it has high performance (Qing et al., 2021). Sun et al. (2021) designed a ConvCRF model with boundary constraints, which was used to improve the classification method of synthetic aperture radar images, thereby improving the classification accuracy of remote sensing images (Sun et al., 2021). Samat et al. (2020) improved the extreme gradient boosting (XGBoost) algorithm and proposed a Meta-XGBoost algorithm, which integrated the advantages of multiple methods and improved the effect of hyperspectral remote sensing image classification (Samat et al., 2020). He et al. (2020) combined a fully convolutional network with a popular graph embedding model and applied it to PolSAR image classification, which proved to have high application performance (He et al., 2020).

The above studies have used different deep learning methods to classify and identify different types of remote sensing images, and have achieved good application results. Although some methods can achieve good experimental results, the experimental process is more complicated, so there is still room for improvement in efficiency. The research adopts CNN based classification method, which can classify efficiently and has high classification accuracy.

3. Multi-source remote sensing data fusion and classification based on CNN

3.1. Build CNN model

With the continuous progress of remote sensing technology, the application scope of remote sensing image data is expanding. The application of remote sensing image data is conducive to better urban planning. Before that, it is necessary to classify multi-source remote sensing data to perform other operations. CNN algorithm has strong feature extraction ability and is widely used in data classification. Therefore, CNN is applied in multi-source remote sensing data fusion classification. As a feedforward neural network, CNN includes convolution structure and multilayer non-linearity. The algorithm can extract middle and high level abstract features from remote sensing images under the action of convolution layer and pooling layer (Deng et al., 2020; Huang et al., 2022; Zhong et al., 2022; Zhou et al., 2022). The convolutional neural network represents the target by building a multi-layer network, and its structure is shown in Figure 1.

In Figure 1, CNN includes multiple layers, such as convolution layers. At the same time, in this algorithm, features can be extracted and classified. In a convolutional neural network, each image can be represented by a matrix of pixel values. Meanwhile, in the convolution layer, the neurons are connected in a special way, and the image edges and features are extracted (Zhang et al., 2020). And the convolution operation can process image noise, and can also enhance some features. Under complex conditions, through the action of activation function, the non-linear ability of the network is strengthened. For the binary classification problem, the Sigmoid function is used, while for the image recognition classification, the ReLU function is used (Chung et al., 2020; Zhou et al., 2021a,b; Zhang et al., 2022). Finally, the model needs to be downsampled to reduce its complexity, which is done through a pooling operation. The fully connected layer belongs to the classification and recognition part, which performs weighted summation of the extracted features and performs the final output. As a key part of the convolutional neural network, the convolution layer mainly performs feature extraction and dimensionality reduction processing operations. It contains many convolution kernels, which convolve with the input and generate new feature maps. Convolution usually contains both single-channel and multi-channel types (Feng et al., 2021). Among them, the one-dimensional convolution usually plays the role of signal processing. Assuming that the input signal is listed as x_t, and $t = 1, 2, \cdots, n$, then its output expression is shown in Formula (1).

$$y_t = \sum_{k=1}^{K} w_k x_{t-k+1} \tag{1}$$

In Formula (1), w_k is the convolution kernel, and K is the length of the convolution kernel. In the processing of images and videos, two-dimensional convolution is used more frequently. Let the 2D image input be x_{ij}, where $1 \le i \le M, 1 \le j \le N$. In the same way, w_{ij} represents the convolution kernel, where $1 \le i \le m, 1 \le j \le n$. Then its output expression is shown in Formula (2).

$$y_{ij} = \sum_{u=1}^{m} \sum_{v=1}^{n} w_{uv} x_{i-u+1,j-v+1} \tag{2}$$

In Formula (2), w_{uv} is the convolution kernel, and m, n is the length of the convolution kernel. In Formula (2), we know that during the convolution operation, the filter remains stable and the entire input part is processed. At the same time, the convolution process can be trimmed by changing the step size and padding, which has a certain adjustment effect on the sliding amplitude, thereby making the boundary more complete. The pooling layer is a non-linearly connected area, located between convolution layers, and its adjacent layers are connected to each other through neurons. When extracting the main features of the image, the pooling layer has a good performance. First, the pooling layer can effectively reduce the amount of computation, thereby saving resources. Second, the pooling layer can reduce the number of parameters and the complexity of the model, thereby avoiding overfitting and ensuring scale and space invariance (Li et al., 2020). Average pooling and max pooling are the two most common methods of pooling operations, which can effectively retain the original image features. The structure diagram is shown in Figure 2.

In Figure 2, these two operations can reduce the error of feature extraction, the variance of estimated value caused by the domain, and the shift of estimated mean value caused by the error of convolution parameters. After two operations, activate the data through the activation function, which is a key step in CNN. Neural networks are generally linear calculations, and complex functions are not generated during the calculation process. The activation function can add complex models to it and effectively enhance the non-linear expression ability of the network. These functions of the activation function can play a good role in solving complex network problems, while improving the fitting ability of the model. Common activation functions are Sigmoid, Tanh, and ReLU. Among them, the definition of the sigmoid activation function is shown in Formula (3).

$$Sigmoid\,(z) = \frac{1}{1 + e^{-z}} \tag{3}$$

In Formula (3), the output value of the sigmoid activation function is between (0, 1) and has monotonicity. Its image is similar to the sigmoid, which has the advantage of stable optimization. The definition of the Tanh activation function is

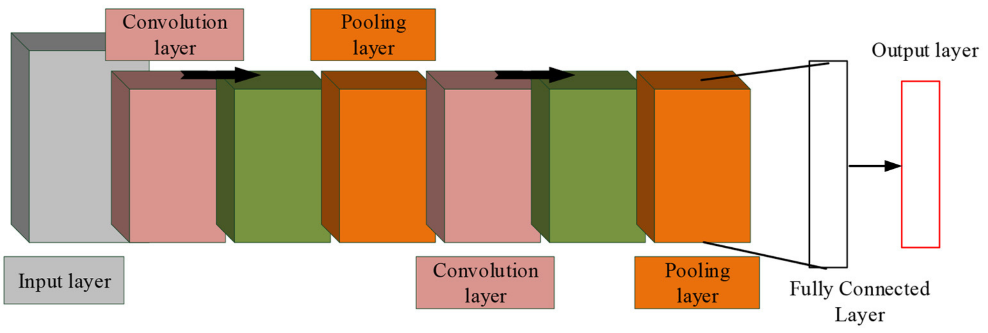


FIGURE 1
Convolutional neural network structure diagram.

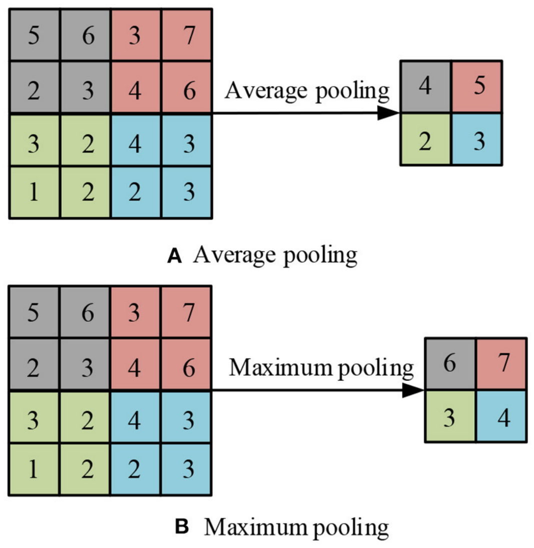


FIGURE 2
Average pool and maximum pool structure.

shown in Formula (4).

$$\text{Tanh}\,(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \tag{4}$$

In Formula (4), the output value of the Tanh activation function is between (−1, 1) and is centered at 0. At the same time, its image curve is also similar to the S-shape, and the convergence speed is faster. The relevant expression of ReLU activation function is Formula (5).

$$\text{Re}\,LU\,(z) = \max\,(0, z) \tag{5}$$

In Formula (5), when the input value is positive, the derivative of the function is always 1. Therefore, compared with the Sigmoid activation function and the Tanh activation function, it has a faster calculation speed and can effectively save resources. After the above operations are completed, the data is normalized to eliminate the influence of the index on the value. In the normalization processing operation, Faced with the problems of slow convergence speed and scattered characteristics, it is necessary to process each batch of data. For the same batch of data $X_B = \{x_1, x_2 \cdots, x_n\}$, the mean and variance expressions are shown in Formula (6) and Formula (7).

$$\mu_B = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{6}$$

$$\sigma_B^2 = \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_B)^2 \tag{7}$$

In Formula (6) and Formula (7), μ_B and σ_B^2 are the mean and variance, respectively, and a new mapping $\hat{x}_i$ can be obtained after normalization x_i, and its expression is shown in Formula (8).

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} \tag{8}$$

In Formula (8), $\varepsilon > 0$ and the value is smaller. In order to obtain the real and effective distribution of network data, scale transformation and offset processing are added after normalization, and its expression is shown in Formula (9).

$$y_i = \gamma \hat{x}_i + \beta \tag{9}$$

In Formula (9), γ and β are parameters in network training, and the update methods are shown in Formula (10) and Formula (11).

$$\nabla\gamma = \sum_{i=1}^{m} \nabla y_i \frac{\partial y_i}{\partial y} = \sum_{i=1}^{m} \nabla y_i \cdot \hat{x}_i \tag{10}$$

$$\nabla\beta = \sum_{i=1}^{m} \nabla y_i \frac{\partial y_i}{\partial \beta} = \sum_{i=1}^{m} \nabla y_i \cdot 1 = \sum_{i=1}^{m} \nabla y_i \tag{11}$$

In Formula (10) and Formula (11), the two are updated by means of derivation, and the input x_i gradient expression is shown in Formula (12).

$$\nabla x_i = \nabla \hat{x} \cdot \frac{1}{\sqrt{\sigma_B^2 + \varepsilon}} + \nabla \sigma_B^2 \cdot \frac{2\left(x_i - \mu_B\right)}{m} + \nabla \mu_B \cdot \frac{1}{m} \quad (12)$$

In Formula (12), there is a certain relationship between x_i, $\hat{X}_I$, μ_B and σ_B^2. At the same time, in the back-propagation process, calculate the gradient of $\hat{X}_I$, μ_B and σ_B^2 to x_i, as shown in Formula (13), Formula (14), and Formula (15).

$$\nabla \hat{x}_i = \nabla y_i \cdot \gamma \quad (13)$$

$$\nabla \mu_B = \sum_{i=1}^{m} \nabla \hat{x} \cdot \frac{-1}{\sqrt{\sigma_B^2 + \varepsilon}} + \nabla \sigma_B^2 \cdot \frac{1}{m} \sum_{i=1}^{m} -2\left(x_2 - \mu_B\right) \quad (14)$$

$$\nabla \sigma_B^2 = \sum_{i=1}^{m} \nabla \hat{x} \cdot \left(x_i - \mu_B\right) \cdot \frac{-1}{2} \left(\sigma_B^2 + \varepsilon\right)^{-\frac{3}{2}} \quad (15)$$

After the feature extraction and classification and recognition are completed, the results are output, thus completing the entire convolutional neural network steps.

3.2. Multi source remote sensing data fusion and classification based on CNN

Multi-source remote sensing data includes hyperspectral data (HSI) and lidar data (LiDAR), due to their different types and applicable directions, there are certain challenges in fusion and classification (Qu et al., 2021). Therefore, the research uses CNN to extract its features, and proposes a dual-branch convolutional neural network (DB-CNN), which is convenient for organically combining multiple data sources. The multi-source remote sensing data fusion and classification process based on CNN is shown in Figure 3.

In Figure 3, a dual-channel CNN network is used to extract spectral information. In HSI branch, Conv2D3 of 2-D channel is 256, Conv2D3 is 512, Max Pool is 2 * 2, Conv1D11 of 1-D channel is 256, Conv1D3 is 512, Max Pool is 2 * 1; In the HSI branch, the value of Conv2D3 is 64, the value of Cascade2D is [128, 64,128, 64], the value of Max Pool is 2 * 2, and the value of Cascade2D is [256128256128]. For hyperspectral data extraction, the spatial information is extracted by 2-D CNN, and the central pixel information is extracted by 1-D CNN. For LiDAR and Visible Light Image (VIS) data, because of their strong spatial information, the same network can be used for feature extraction. The overall network structure consists of three parts, namely spectrum, spatial channel and space-spectral fusion. The spectral channel can be divided into three parts, including convolution layer, pooling layer, etc., and batch normalization. When performing the convolution operation, a one-dimensional convolution method is adopted to process the one-dimensional vector of the spectral data. At the same time, in order to correct the data distribution, the Leaky ReLU activation function is selected to perform the correction operation. Therefore, the spectral dimension feature extraction process can be expressed as: firstly, input the spectral vector H_{ij}^{spec} into the network, then, perform correlation operation through it, and finally output the feature F_{ij}^{spec}, and expand the feature into a one-dimensional vector at the same time.

For spatial dimension feature extraction, the processing object is usually r the image block with radius around the center pixel, so the output feature F_{ij}^{spat} is the information of the center pixel and its surrounding radius r. It will also expand F_{ij}^{spat} into a one-dimensional vector and fused with F_{ij}^{spec} each other. When extracting relevant features, the consistency of the depth and structure of the dual channel network shall be ensured to make the extracted features more complete. The two kinds of features are fed into the fully connected layer after fusion, and they are reorganized and selected by learning. For the features with too little contribution, the Dropout method can be used to discard them, and the whole process can be represented by Formula (16).

$$T\left(F_{ij}^{spat}, F_{ij}^{spec}\right) = f\left(W\left(F_{ij}^{spat} \middle\| F_{ij}^{spec}\right) + b\right) \quad (16)$$

Formula (16), $.\|.$ denote feature fusion, W and b denote the weights and biases of fully connected layers. Then the above formula can be expressed as F_{hsi} and input into the softmax classifier. The classifier can predict features as corresponding probability distributions, as shown in Formula (17).

$$pred\left(i,j\right) = \frac{1}{\sum_{n=1}^{C}\left(\exp\left(\theta'_n F_{hsi}\right)\right)} \begin{bmatrix} \exp\left(\theta'_1 F_{hsi}\right) \\ \exp\left(\theta'_2 F_{hsi}\right) \\ \vdots \\ \exp\left(\theta'_C F_{hsi}\right) \end{bmatrix} \quad (17)$$

In Formula (17), $\theta_n\left(n = 1, 2, \cdots, C\right)$ represents the nth column parameter of the classifier, which $pred\left(i,j\right) \in R^C$ is a one-dimensional vector, which represents the prediction result of the pixel p_{ij}. For LiDAR or VIS data feature extraction, a cascaded CNN network is required, as shown in Figure 4.

From Figure 4, the cascade structure is mainly composed of basic cascade operations. Before entering the data into the network structure, it needs to be normalized. In the convolution operation, the convolution kernel size is set to 3 × 3. After going through the operations of all modules, expand the extracted feature through F_{LV} to obtain one-dimensional vector, and then use it as the input part of the fully connected layer. In order to improve the fusion effect of features at different levels, a Cascade block structure is designed in which different features can be

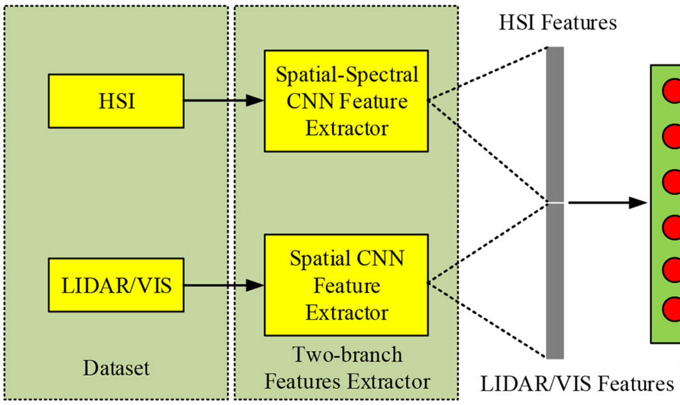

HSI
LIDAR/VIS
Dataset
Spatial-Spectral CNN Feature Extractor
Spatial CNN Feature Extractor
Two-branch Features Extractor
HSI Features
LIDAR/VIS Features
Classifier
Classes Category Labels

FIGURE 3
Related flow chart.

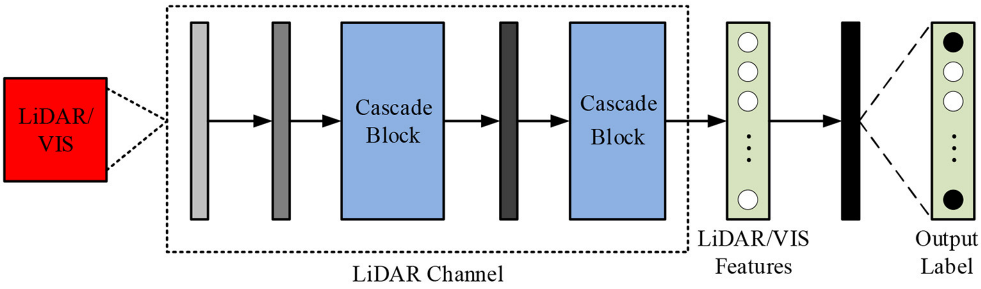


FIGURE 4
CNN network structure of cascaded modules for LiDAR/VIS feature extraction.

bridged. This structure can be represented by Formula (18).

$$\begin{cases} y_m = g_m\left(x, \{W_i, B_i\}\right) + x \\ \qquad\qquad y = g_s\left(x_s, \left\{W_j, B_j\right\}\right) + x_s \end{cases} \qquad (18)$$

In Formula (18), $g_m\left(x, \{W_i, B_i\}\right)$and $g_s\left(x_s, \left\{W_j, B_j\right\}\right)$is the operation between two channels, x and the y corresponding input and output, indicating the output of the middle layer. After the CNN network is constructed, all its parameters need to be trained and updated. For the network parameters, the feature map of each layer of the network is set to a power of 2. Since more parameters need to be trained and the distribution of these parameters is not uniform, training on two branches at the same time will have an impact on obtaining the optimal parameter solution. Therefore, it is necessary to train the parameters on the two branches separately, and then perform fine-tuning training after the two are trained. In training experiments, data and methods are the two most critical parts. Different from general deep learning training models, remote sensing image data training has a limited number of labels, and the labeling process is time-consuming and costly (Gu et al., 2022). To solve this problem, it is usually necessary to process the data in the preprocessing stage, such as rotating the image, adding Gaussian noise, etc., to expand the training set. In addition to this, all data needs to be normalized.

When performing feature extraction on HSI, 1-D CNN is responsible for extracting spectral features, while 2-D CNN is responsible for extracting spatial information (Xu et al., 2019). This dual-channel network design can reduce training update parameters, so it can save computing resources and improve training efficiency. In addition, the Cascade block structure also has certain advantages when extracting LiDAR/VIS data. This cascaded CNN network structure can transfer low-level features to high-level features, which can be reused to improve efficiency.

4. Performance analysis of multi source remote sensing data fusion and classification based on CNN

In order to effectively verify the performance of the proposed dual-channel CNN, the same type of classification models are

TABLE 1 Comparison of classification accuracy of dual-branch CNN networks on different data sets.

Data	DB-CNN(L/V)		DB-CNN(H)		DB-CNN(H+L/V)	
	OA (%)	Kappa	OA (%)	Kappa	OA (%)	Kappa
Houston	55.62	0.5168	83.21	0.8157	86.69	0.8577
Trento	84.81	0.8105	94.98	0.9285	96.83	0.9547
Pavia	92.85	0.9042	96.87	0.9593	98.46	0.9735
Salinas	91.68	0.9107	95.53	0.9487	96.58	0.9576

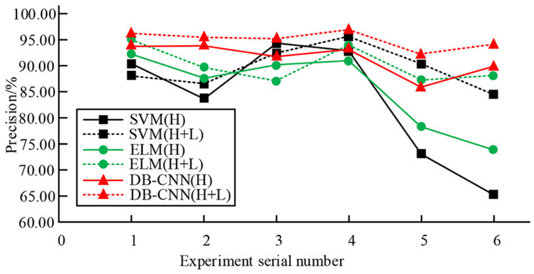


FIGURE 5
Comparison of classification accuracy of three classification models on Houston dataset.

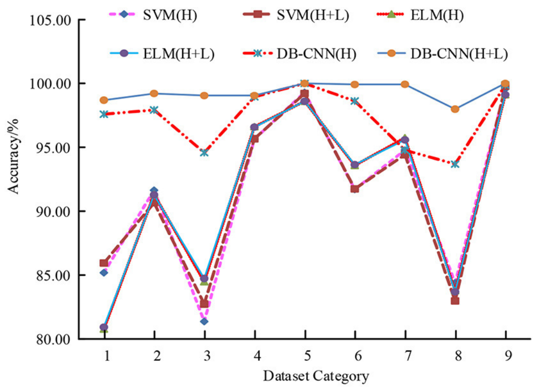


FIGURE 6
Accuracy of different classification models.

introduced: SVM algorithm and ELM algorithm. During the performance analysis, the samples used by the three methods are the same. Use (H) to represent the experiments and results of the classification model on hyperspectral, and (H+L) to represent the experimental results of the combination of hyperspectral and LiDAR. First, the experimental results of DB-CNN network using different classification methods on different datasets are analyzed. The data sets involved are Houston data set, Trento data set, Pavia data set and Salinas data set. The Houston data set consists of two parts, namely hyperspectral data and LiDAR data. The map size is 349 * 1,905; Trento dataset is shot in Trento region, Italy, with 600 * 166 pixels; The Pavia dataset was taken in Pavia, Italy, with a map size of 610 * 340; The Salinas dataset was taken in the Salinas region of Italy, and the map size is 512 * 217. The analysis results are shown in Table 1.

In Table 1, compared with a single HSI or LiDAR method, the combined method has higher global classification accuracy in different data sets. For example, on the Pavia dataset, the global classification accuracy of the three classification methods is the highest, among which the global classification accuracy of the joint classification method reaches 98.46%, which is 5.61% higher than the single LiDAR/VIS classification accuracy and 1.59% higher than the single HSI classification accuracy. At the same time, the Kappa value of the classification accuracy index of the joint classification method is 0.9735, which is 0.0693 higher than the Kappa value of the single LiDAR/VIS classification and 0.0142 higher than the Kappa value of the single HSI classification. This result shows that the classification effect of the joint classification method is better than that of the single classification method. Classification method. At the same time, the Houston data set is taken as an example to verify the classification accuracy of different classification models on this data set. The comparison results are shown in Figure 5.

As can be seen from Figure 5, for the three classification models, the fusion classification method has the best performance and the highest classification accuracy in the global classification. For example, the average accuracy of SVM model using a single HSI classification is about 82.83%, and the average accuracy of SVM model using a combination of HSI and LiDAR classification is about 89.86%. The average accuracy of the ELM model using a single HSI classification is about 85.57%, and the average accuracy of the ELM model using a combination of HSI and LiDAR classification is about 91.05%. The average accuracy of DB-CNN model using a single HSI classification is about 92.13%, and the average accuracy of DB-CNN model using a combination of HSI and LiDAR classification is about 95.08%. Therefore, in the three classification models, the average classification accuracy of the single classification method and the joint classification method corresponding to the dual branch CNN network structure is higher than that of the SVM model and ELM model, indicating that the classification effect is better.

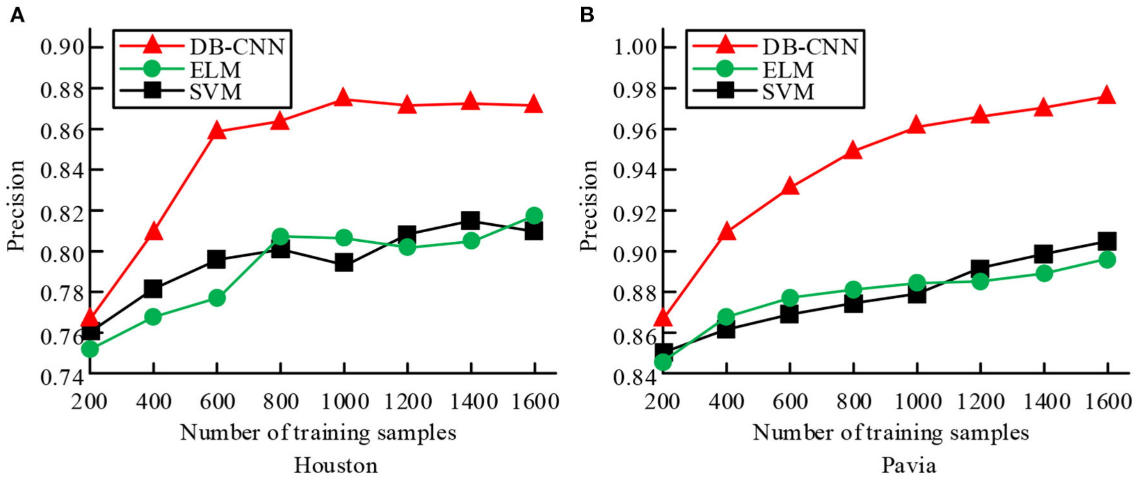


FIGURE 7
Comparison of classification performance of three classification models under different training sample numbers. **(A)** Training accuracy of the three classifications of Houston training set. **(B)** Training accuracy of the three classifications of Pavia training set.

TABLE 2 Comparison of training time of three classification models under different training sample numbers.

Number of training samples	Training time (s)			Test time (s)		
	SVM	ELM	DB-CNN	SVM	ELM	DB-CNN
200	36.4	32.7	25.3	15.7	13.4	8.1
400	68.1	61.5	49.6	28.5	25.3	15.6
600	103.9	92.4	70.6	40.9	35.8	21.5
800	135.7	119.5	91.4	51.2	43.7	26.1
1,000	160.4	142.9	113.8	60.3	49.9	29.8

The classification performance of the DB-CNN model is further analyzed through the Pavia dataset. The results are shown in Figure 6.

In Figure 6, according to the trend of the broken line chart of the accuracy rate of the six classification models, compared with the classification models corresponding to the SVM algorithm and the ELM algorithm, the accuracy rate of the classification model corresponding to the DB-CNN is higher, especially the classification accuracy rate of the two branch CNN classification model is the highest, with the highest accuracy rate of 100.00%; Moreover, the accuracy of the two branch CNN classification model is above other models, and the accuracy difference between different data sets is small, that is, the performance of the two branch CNN classification model is more stable. In addition, the classification performance of the three classification models under different training sample numbers is compared, as shown in Figure 7.

Figure 7A shows the training accuracy of the three classifications of Houston training set, and Figure 7B shows the training accuracy of the three classifications of Pavia training set. According to the trend of the graph, in the process of increasing training samples, the classification accuracy of the three classification models shows an overall upward trend. Among them, the accuracy of the dual-branch CNN network model has an obvious upward trend, and its training accuracy is higher than the other two classification models under the same number of samples. And when the number of training samples is small, the dual-branch CNN network model can also achieve better classification accuracy. In Figure 7A, when the training sample size is 800, the accuracy of DB-CNN model is 0.862, 0.062 higher than that of SVM model; In Figure 7B, when the training sample size is 1,600, the precision of ELM model and DB-CNN model is 89.73 and 97.68, respectively. The results show that the two branch CNN network model can achieve better classification accuracy when performing correlation classification. The training and test times of the three classification models under different training and test sample numbers are compared, as shown in Table 2.

In Table 2, when the number of samples becomes large, the training time and testing time of the three classification models

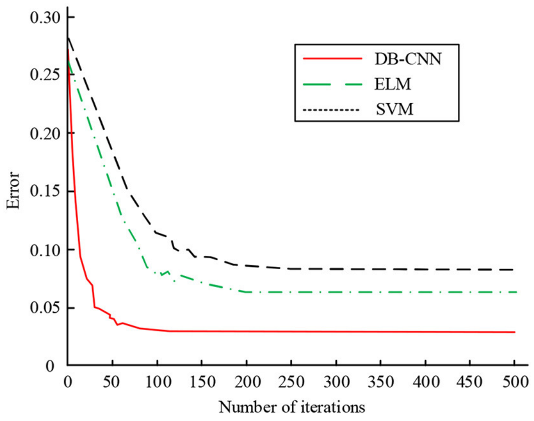


FIGURE 8
Test error comparison of three classification models on test set.

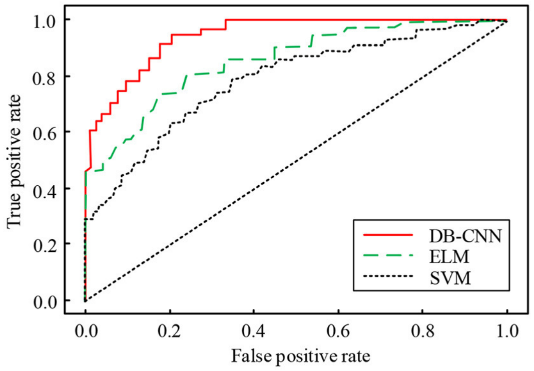


FIGURE 9
Comparison of ROC curves of three classification models.

gradually increase, and the growth trend gradually slows down. When the number of samples used for training and testing is equal, the training and testing time of the dual-branch CNN network model is the shortest, followed by the ELM model, and the SVM model with the longest training and testing time. For example, when the number of samples used for training and testing is 1,000, the training time of the dual-branch CNN model is 113.8 s, which is 29.1 s lower than the ELM model and 46.6 s lower than the SVM model; its test time is 29.8 s, which is 20.1 s lower than the ELM model, which is 30.5 s lower than the SVM model. Therefore, under the same conditions, the training efficiency and testing efficiency of the dual-branch CNN network model are higher, and it has a better effect in the fusion and classification of multi-source remote sensing data. In addition, the test errors of the three classification models on the test set are compared and analyzed, as shown in Figure 8.

In Figure 8, as the number of iterations increases, the classification errors of the three models gradually decrease and finally become stable. When the number of iterations is at a small level, the convergence speed of the dual-branch CNN network model was faster, followed by the ELM model and the SVM model. At 100 iterations, the error value of the dual-branch CNN network model is minimized and stabilized, and its error value is about 0.026. At 200 iterations, the error value of the dual-branch CNN network model is minimized and stabilized, the error value of the ELM model is minimized and stabilized, and its error value is about 0.063, the dual branch CNN network model is 0.100. When the number of iterations reaches 200, the error value of the SVM model decreases to a minimum and tends to be stable. According to the results, the two branch CNN network model has the smallest error value and the best classification effect. Finally, the ROC curves of the three classification models are compared, as shown in Figure 9.

In Figure 9, the lower area corresponding to the ROC curve of the dual-branch CNN network model is the largest, that is, the AUC value is the largest, followed by the ELM model and the SVM model. The AUC value corresponding to the dual-branch CNN network model is about 0.922. AUC value of ELM model is about 0.869, which is 0.053 lower than the dual-branch CNN network model. AUC value of SVM model is about 0.837, which is 0.032 lower than the ELM model and 0.085 lower than the dual-branch CNN network model. The ROC curve and AUC value represent the quality of the classification effect. From the above results, we can see that the classification effect of the dual-branch CNN network model is the best, and it can play a greater role in the recognition and classification of remote sensing images.

5. Conclusion

CNN can better classify and recognize, and they have been widely used in many fields. In order to realize the fusion and classification of multi-source remote sensing data, a dual branch CNN network structure model is proposed, and ELM model and SVM model are used as comparison models. According to the results obtained, it can be seen that for the dual branch CNN network, the HSI and LiDAR joint classification method has the highest global classification accuracy on different data sets. On the Pavia dataset, the global classification accuracy of the three classification methods is the highest. Among them, the global classification accuracy of the joint classification method is 98.46, 5.61% higher than that of the single LiDAR/VIS classification, and 1.59% higher than that of the single HSI classification. In the training experiment, compared with other methods, the training accuracy of BD-CNN model is higher than that of the other two classification models with the same sample number. When the number of samples used in training and testing is the same, the training time and testing time of BD-CNN model are the lowest. In the error test experiment, when the number of iterations of the DB-CNN model is 100, the test error reaches the lowest steady state, which is about 0.026, 0.037 lower than the ELM model. In addition, the ROC curve of the DB-CNN model corresponds to the largest lower area, that is, the AUC value is the largest, which is about 0.922, that is, the DB-CNN model has the best classification performance. Comprehensive analysis shows that BD-CNN model can effectively fuse and classify multi-source remote sensing data. However, there is still room for improvement. In this paper, we can discuss other depth learning methods when classifying remote sensing data to obtain better classification results.

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

FY and ZZ contributed to conception and design of the study. YW organized the database. BE performed the statistical analysis. FY wrote the first draft of the manuscript. ZZ wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

Funding

This study was supported by Space Monitoring the Environmental and Ecological Sustainable Development in the Lancang-Mekong and Cooperative Support Project of Region Pacific Region Earth Observation Cooperation.

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

Bazi, Y., Bashmal, L., Rahhal, M. M. A., Dayil, R. A., and Ajlan, N. A. (2021). Vision transformers for remote sensing image classification. *Remote Sens.* 13, 516–521. doi: 10.3390/rs13030516

Chung, Y. L., Chung, H. Y., and Tsai, W. F. (2020). Hand gesture recognition *via* image processing techniques and deep CNN. *J. Intell. Fuzzy Syst.* 39, 1–14. doi: 10.3233/JIFS-200385

Demir, V., and Ulke, A. (2020). Obtaining the manning roughness with terrestrial-remote sensing technique and flood modeling using FLO-2D: a case study Samsun from Turkey. *Geofizika* 37, 131–156. doi: 10.15233/gfz.2020.37.9

Deng, Z., Cao, Y., Zhou, X., Yi, Y., Jiang, Y., and You, I. (2020). Toward efficient image recognition in sensor-based IoT: a weight initialization optimizing method for CNN based on RGB influence proportion. *Sensors*, 20, 2866–2871. doi: 10.3390/s20102866

Dong, R., Xu, D., Jiao, L., Zhao, J., and An, J. (2020). A fast deep perception network for remote sensing scene classification. *Remote Sens.* 12, 422–439. doi: 10.3390/rs12040729

Dong, Y., Liang, T., Zhang, Y., and Du, B. (2020). Spectral-spatial weighted kernel manifold embedded distribution alignment for remote sensing image classification. *IEEE Trans. Cybernet.* 51, 3185–3197. doi: 10.1109/TCYB.2020.3004263

Du, X., Zheng, X., Lu, X., and Doudkin, A. A. (2021). Multisource remote sensing data classification with graph fusion network. *IEEE Trans. Geosci. Remote Sens.* 59, 10062–10072. doi: 10.1109/TGRS.2020.3047130

Du, Y., Qin, B., Zhao, C., Zhu, Y., Cao, J., and Ji, Y. (2022). A novel spatio-temporal synchronization method of roadside asynchronous MMW radar-camera for sensor fusion. *IEEE Trans. Intell. Transp. Syst.* 23, 22278–22289. doi: 10.1109/TITS.2021.3119079

Feng, Z., Zhu, M., Stanković, L., and Ji, H. (2021). Self-matching CAM: a novel accurate visual explanation of CNNs for SAR image interpretation. *Remote Sens.* 13, 1772–1778. doi: 10.3390/rs13091772

Gu, X., Zhang, C., Shen, Q., Han, J., Angelov, P. P., and Atkinson, P. M. (2022). A Self-training hierarchical prototype-based ensemble framework for remote sensing scene classification. *Inform. Fusion* 80, 179–204. doi: 10.1016/j.inffus.2021.11.014

Han, Y., Wei, C., Zhou, R., Hong, Z., Zhang, Y., and Yang, S. (2020). Combining 3D-CNN and squeeze-and-excitation networks for remote sensing sea ice image classification. *Math. Probl. Eng.* 2020, 8065396. doi: 10.1155/2020/8065396

He, C., He, B., Tu, M., Wang, Y., Qu, T., Wang, D., et al. (2020). Fully convolutional networks and a manifold graph embedding-based algorithm for PolSAR image classification. *Remote Sens.* 12, 1467–1473. doi: 10.3390/rs12091467

Hu, A., Chen, S., Wu, L., Xie, Z., Qiu, Q., and Xu, Y. (2021). WSGAN: an improved generative adversarial network for remote sensing image road network extraction by weakly supervised processing. *Remote Sens.* 13, 2506–2511. doi: 10.3390/rs13132506

Huang, C. Q., Jiang, F., Huang, Q. H., Wang, X. Z., Han, Z. M., and Huang, W. Y. (2022). Dual-graph attention convolution network for 3-d point cloud classification. *IEEE Trans. Neural Netw. Learn. Syst.* 2022, 1–13. doi: 10.1109/TNNLS.2022.3162301

Jin, H., and Mountrakis, G. (2022). Fusion of optical, radar and waveform LiDAR observations for land cover classification. *ISPRS J. Photogram. Remote Sens.* 187, 171–190. doi: 10.1016/j.isprsjprs.2022.03.010

Li, Z., Zhou, A., and Shen, Y. (2020). An end-to-end trainable multi-column CNN for scene recognition in extremely changing environment. *Sensors* 20, 1556–1562. doi: 10.3390/s20061556

Lu, H., Zhu, Y., Yin, M., Yin, G., and Xie, L. (2022). Multimodal fusion convolutional neural network with cross-attention mechanism for internal defect detection of magnetic tile. *IEEE Access* 10, 60876–60886. doi: 10.1109/ACCESS.2022.3180725

Luo, X., Du, H., Zhou, G., Li, X., Mao, F., Zhu, D. E., et al. (2021). A novel query strategy-based rank batch-mode active learning method for high-resolution remote sensing image classification. *Remote Sens.* 13, 2234–2256. doi: 10.3390/rs13112234

Ma, H., Liu, Y., Ren, Y., Wang, D., Yu, L., and Yu, J. (2020). Improved CNN classification method for groups of buildings damaged by earthquake, based on high resolution remote sensing images. *Remote Sens.* 12, 260. doi: 10.3390/rs12020260

Pan, X., Zhao, J., and Xu, J. (2020). An end-to-end and localized post-processing method for correcting high-resolution remote sensing classification result images. *Remote Sens.* 12, 852–856. doi: 10.3390/rs12050852

Pastorino, M., Montaldo, A., Fronda, L., Hedhli, I., Moser, G., Serpico, S. B., et al. (2021). Multisensor and multiresolution remote sensing image classification through a causal hierarchical markov framework and decision tree ensembles. *Remote Sens.* 13, 849–874. doi: 10.3390/rs130 50849

Qing, Y., Liu, W., Feng, L., and Gao, W. (2021). Improved transformer net for hyperspectral image classification. *Remote Sens.* 13, 2216–2220. doi: 10.3390/rs13112216

Qu, L., Zhu, X., Zheng, J., and Zou, L. (2021). Triple-attention-based parallel network for hyperspectral image classification. *Remote Sens.* 13, 324–329. doi: 10.3390/rs13020324

Samat, A., Li, E., Wang, W., Liu, S., Lin, C., and Abuduwaili, J. (2020). Meta-XGBoost for hyperspectral image classification using extended MSER-guided morphological profiles. *Remote Sens.* 12, 1973–1978. doi: 10.3390/rs121 21973

Sun, Z., Liu, M., Liu, P., Li, J., Yu, T., Gu, X., et al. (2021). SAR image classification using fully connected conditional random fields combined with deep learning and superpixel boundary constraint. *Remote Sens.* 13, 271–278. doi: 10.3390/rs13020271

Xu, Y., Du, B., Zhang, L., Cerra, D., Pato, M., Carmona, E., et al. (2019). Advanced multi-sensor optical remote sensing for urban land use and land cover classification: outcome of the 2018 IEEE GRSS data fusion contest. *IEEE J. Select. Topics Appl. Earth Observ. Remote Sens.* 12, 1709–1724. doi: 10.1109/JSTARS.2019.2911113

Xu, Z., Zhao, X., Guo, X., and Guo, J. (2019). Deep learning application for predicting soil organic matter content by VIS-NIR spectroscopy. *Comput. Intell. Neurosci.* 2019, 1–11. doi: 10.1155/2019/3563761

Yu, D, Xu, Q, Guo, H, Zhao, C, Lin, Y, Li, D. (2020). An efficient and lightweight convolutional neural network for remote sensing image scene classification. *Sensors* 20, 1999–2005. doi: 10.3390/s20071999

Zhang, C., Chen, Y., Yang, X., Gao, S., Li, F., Kong, A., et al. (2020). Improved remote sensing image classification based on multi-scale feature fusion. *Remote Sens.* 12, 213–219. doi: 10.3390/rs12020213

Zhang, H., and Han, J. (2020). Mathematical models for information classification and recognition of multi-target optical remote sensing images. *Open Phys.* 18, 951–960. doi: 10.1515/phys-2020-0123

Zhang, Q., Ge, L., Hensley, S., Metternicht, G. I., Liu, C., and Zhang, R (2022). PolGAN: a deep-learning-based unsupervised forest height estimation based n the synergy of PolInSAR and LiDAR data. *ISPRS J. Photogram. Remote Sens.* 186, 123–139. doi: 10.1016/j.isprsjprs.2022.02.008

Zhang, W., He, X., and Lu, W. (2020). Exploring discriminative representations for image emotion recognition with CNNs. *IEEE Trans. Multimedia*, 22, 515–523. doi: 10.1109/TMM.2019.2928998

Zhong, T., Cheng, M., Lu, S., Dong, X., and Li, Y. (2022). RCEN: a deep-learning-based background noise suppression method for DAS-VSP records. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2021.3127637

Zhou, G., Li, C., Zhang, D., Liu, D., Zhou, X., and Zhan, J. (2021a). Overview of underwater transmission characteristics of oceanic LiDAR. *IEEE J. Select. Topics Appl. Earth Observ. Remote Sens.* 14, 8144–8159. doi: 10.1109/JSTARS.2021.3100395

Zhou, G., Li, W., Zhou, X., Tan, Y., Lin, G., Li, X., et al. (2021b). An innovative echo detection system with STM32 gated and PMT adjustable gain for airborne LiDAR. *Int. J. Remote Sens.* 42, 9187–9211. doi: 10.1080/01431161.2021.1975844

Zhou, G., Zhou, X., Song, Y., Xie, D., Wang, L., Yan, G, et al. (2021c). Design of supercontinuum laser hyperspectral light detection and ranging (LiDAR) (SCLaHS LiDAR). *Int. J. Remote Sens.* 42, 3731–3755. doi: 10.1080/01431161.2021.1880662

Zhou, W., Wang, H., and Wan, Z. (2022). Ore image classification based on improved CNN. *Comput. Electrical Eng.* 99, 107819. doi: 10.1016/j.compeleceng.2022.107819

frontiers | Frontiers in Neurorobotics

TYPE Original Research
PUBLISHED 06 January 2023
DOI 10.3389/fnbot.2022.1104559

OPEN ACCESS

EDITED BY
Xin Jin,
Yunnan University, China

REVIEWED BY
Zhibin Qiu,
Nanchang University, China
Ruohan Gong,
Université de Lille, France
Zhixiang Chen,
Minnan Normal University, China

*CORRESPONDENCE
Yuanchao Hu
✉ huyuanchao3211@126.com

RECEIVED 21 November 2022
ACCEPTED 12 December 2022
PUBLISHED 06 January 2023

CITATION
Zhang M, Song Z, Yang J, Gao M, Hu Y, Yuan C, Jiang Z and Cheng W (2023) Study on the enhancement method of online monitoring image of dense fog environment with power lines in smart city. *Front. Neurorobot.* 16:1104559. doi: 10.3389/fnbot.2022.1104559

Study on the enhancement method of online monitoring image of dense fog environment with power lines in smart city

Meng Zhang[1], Zhitao Song[1], Jianfei Yang[2], Mingliang Gao[2], Yuanchao Hu[2]*, Chi Yuan[1], Zhipeng Jiang[1] and Wei Cheng[1]

[1]State Grid Hubei Power Supply Limited Company Ezhou Power Supply Company, Ezhou, China, [2]School of Electrical and Electronic Engineering, Shandong University of Technology, Zibo, China

In this research, an image defogging algorithm is proposed for the electricity transmission line monitoring system in the smart city. The electricity transmission line image is typically situated in the top part of the image which is rather thin in size. Because the electricity transmission line is situated outside, there is frequently a sizable amount of sky in the backdrop. Firstly, an optimized quadtree segmentation method for calculating global atmospheric light is proposed, which gives higher weight to the upper part of the image with the sky region. This prevents interference from bright objects on the ground and guarantees that the global atmospheric light is computed in the top section of the image with the sky region. Secondly, a method of transmission calculation based on dark pixels is introduced. Finally, a detail sharpening post-processing based on visibility level and air light level is introduced to enhance the detail level of electricity transmission lines in the defogging image. Experimental results indicate that the algorithm performs well in enhancing the image details, preventing image distortion and avoiding image oversaturation.

KEYWORDS

smart city, power lines, atmospheric scattering model, global atmospheric light, dark pixels

1. Introduction

With the development of the Internet of Things sensor and image processing technology, the monitoring requirements of the power system for the transmission line are gradually improved. The transmission line can be equipped with image sensors to observe its running status in real time, which leads to potential risks of prefabrication. As an important part of power system, electricity transmission line is an important way of power resource transmission. Its operation stability will have a direct impact on power quality. People are always concerned with monitoring electricity transmission lines in order to guarantee the secure and reliable operation of those lines. Electricity transmission lines are installed outdoors and exposed to fog, rain, dew and other weather conditions for a long time, and are greatly affected by the environment. Faults such as insulator defects may occur, which will seriously affect the normal use of the electricity transmission line and reduce the service life of the line. Once the electricity transmission line fails, accidents such as tripping and power outages may occur, resulting in human and economic losses. Therefore, regular inspection of the electricity transmission line is of great significance to ensure the reliable, safe, and efficient operation of the electricity transmission line. The inspection of electricity transmission lines has been dominated by manual inspection for a long time, but manual inspection requires staff to work in the outdoor environment for a long time, which not only has poor monitoring efficiency and accuracy, but also has potential safety hazards for staff. Therefore, in recent years, the method of monitoring electricity transmission lines through a video monitor system has been widely used, which is of importance to improve the monitoring efficiency of electricity transmission lines and speed up the construction of smart cities.

In recent years, constant fog has become one of the terrible weather situations damaging the power grid's atmospheric environment as a result of the rapid development of the economic scale and the acceleration of urbanization. Fog is a common occurrence in the atmosphere. In foggy circumstances, the air is dense with atmospheric particles that not only absorb and scatter the reflected light from the scene, but also disseminate some of it into the observation equipment (Xu et al., 2015). Therefore, in haze weather, the images obtained by the monitoring system and the vision system will be seriously degraded, such as image color offset, reduced visibility, loss of details, and other problems, which seriously affect image detection, tracking, recognition and the use of the monitoring system (Su et al., 2020). Electricity transmission line monitoring in hazy weather will face some problems, such as reduced contrast, chromatic aberration, and unclear details, which will significantly impact the visual impact of monitoring power transmission lines, adversely affect transmission line monitoring, and even cause misjudgment. Therefore, it is necessary to conduct defogging research for transmission line monitoring.

The two kinds of defogging algorithms that are now most often utilized are image enhancement and image restoration. Since computer hardware has improved quickly in recent years, image defogging algorithms based on machine learning have also been proposed (Sharma et al., 2021). The image enhancement-based defogging algorithm merely improves the image contrast and other characteristics using image enhancement technology to achieve the defogging goal. It does not take into account the physical process of fog generation. Traditional image contrast enhancement methods include histogram redistribution (Zhou et al., 2016), intensity transformation (Sangeetha and Anusudha, 2017), homomorphic filtering (Seow and Asari, 2006), wavelet transform (Jun and Rong, 2013), and Retinex algorithm (Jobson et al., 1997b). In Retinex theory, the image is made up of the incident element which represents the brightness information around the object and the reflection element which reflects the reflection ability of itself, then the single scale Retinex algorithm (SSR) is proposed. And then, multiscale Retinex with color restoration (MSRCR) and the multiscale Retinex (MSR) algorithm have both been developed on the foundation of SSR (Jobson et al., 1997a). Defogging algorithm based on image restoration is more commonly used at present. Such algorithms need to consider the physical processes of fog formation, and reasonably estimate the transmission and atmospheric light. In the end, the atmospheric scattering model's calculations provide the restored image. Please note that the word "transmission" mentioned here is not the same as the word "transmission" in the electricity transmission line mentioned above. The "transmission" mentioned here is a parameter in the atmospheric scattering model that reflects the distance between the object in the image and the observation point (such as the camera). Without special circumstances, the t appears later to refer to transmission in atmospheric scattering models.

Multiple image defogging is mainly based on polarization method. Schechner proposed a method of defogging by using two polarized images taken vertically and horizontally (Schechner et al., 2001). Miyazaki et al. (2013) suggested a fog removal method based on the polarization data of two known photographs taken at various distances to predict the characteristics of fog. Shwartz and Schechner (2006) suggested a polarization defogging technique for images without sky areas that choose two comparable characteristics in the scene to estimate atmospheric scattering model parameters. However, the polarization-based image defogging algorithm needs to take multiple polarized images in the same weather condition, which is hard to fulfill the practical needs.

Due to the large limitations of multiple image defogging, it has not been widely used. The more commonly used defogging method is the restoration-based single image defogging method. To estimate necessary parameters based on atmospheric scattering model, Fattal (2008) created the concept of surface shading and the assumption that the transmission and surface shadow are unrelated. Based on the supposition that fog-covered images have less contrast than those taken in clear skies, Tan (2008) proposed an defogging algorithm for images based on the Markov random field optimization atmospheric scattering model to maximize local contrast. Meng et al. (2013) offered a technique to calculate the transmission of unknown scenes by combining the boundary constraint of single image defogging with context regularization based on weighted L1 norm. He et al. (2010) proposed a defogging algorithm called dark channel prior. For single image defogging, the dark channel prior algorithm has developed as one of the most popular methods.

In order to reduce halo and block artifacts generated by coarse transmission estimation, He uses "soft matting" to smooth up the coarse transmission. However, the soft matting technique has the disadvantage of consuming too much time, so it is hard to apply in actual situations. To resolve this issue, He et al. (2012) proposed a guide filter and a fast guide filter (He and Sun, 2015). The neighborhood pixels relationship of hazy images may be transferred by the guided filter to improve air light and transmission smoothness. However, dark channel prior algorithm has some limitations. Dark channel prior algorithm is ineffective for sky region or bright ground region, and the result of defogging in this region is often oversaturated. And dark channel prior algorithm is poor in the processing of depth discontinuous region, and in the area where the foreground and background of the image meet, "halo" phenomena are simple to create. Tarel and Hautiere (2009) proposed a median filter and its variants to replace soft matting, which can improve the calculation speed. Ehsan et al. (2021) proposed a fog removal method that uses local patches of different sizes to calculate the two transmission maps and refine the transmission map with gradient-domain guided image filtering. With the help of training the sum of squared residual error, Raikwar and Tapaswi (2020) suggested a method to determine the lower limit of transmission based on the peak signal-to-noise ratio. Berman and Avidan (2016) assumed that an image can be approximated by hundreds of different colors, which form close clusters in RGB space, and thus proposed a non-local prior method of defogging.

More and more fog removal algorithms based on machine learning have been presented as a result of the advancement of computer neural networks and deep learning. Li et al. (2017) reconstructed the atmospheric scattering model. Then, to estimate the pertinent parameters of fog, an All-in-One Dehazing Network was created utilizing residual learning and convolutional neural network. GridDehazeNet is Liu's proposed end-to-end trainable convolutional neural network for removing fog from a single image. It has pre-processing, backbone, and post-processing, it is a multi-scale network image defogging algorithm based on attention (Liu et al., 2019). Cai et al. (2016) proposed a deep CNN structure for fog removal, named Dehaze Net, to achieve end-to-end fog removal. Zhang and Patel (2018) proposed an edge-preserving densely connected encoder-decoder structure fusion end-to-end densely connected pyramid defogging network, named DCPDN. Pang et al. (2020) suggested a binocular image dehazing Network, which requires the simultaneous use of multiple images for defogging. Ren et al. (2018) suggested a Gated Fusion Network for image defogging, which fuses the three inputs preprocessed for foggy images to avoid halo artifacts. Qin et al. (2020) proposed an attention-based feature fusion single image dehazing network, named FFA-Net.

In order to promote the construction of smart cities, we propose a defogging algorithm for electricity transmission line monitoring. The following are the paper's contributions:

- In order to solve the problem of inaccurate calculation of global atmospheric light in the original dark channel prior algorithm, according to the assumption that the sky area of the electricity transmission line image is usually in the upper half of the image, an improved quadtree segmentation is proposed to calculate the global atmospheric light value. The algorithm can avoid the interference caused by the bright objects on the ground to the solution of the global atmospheric light;
- The concept of dark pixel is introduced for the problem that the dark channel prior is prone to the "halo" effect. Dark pixels are located using super pixel segmentation and a fidelity function is proposed to calculate the transmission;
- Due to the size of the electricity transmission line in the image is tiny and difficult to observe, a detail sharpening post-processing based on visibility and air light is introduced to improve the image details of the electricity transmission line.

The remainder of this paper is organized as shown below. (Section "2 Related works) reviews atmospheric scattering models and dark channel priors, and points out the limitations of dark channel priors. (Section "3 Proposed method) presents a defogging method for electricity transmission line images based on improved quadtree segmentation and dark pixels, and enhances image details. (Section "4 Experimental results and discussion) evaluates the efficacy of the proposed method using both qualitative and quantitative analyses. And the entire study is summarized in (Section "5 Conclusion).

2. Related works

2.1. Physical model

The physical model of atmospheric scattering based on Mie scattering theory was initially put out by McCartney (1976). Narasimhan and Nayar (2001) believes that the wavelength of visible light in a uniform atmosphere has nothing to do with the scattering coefficient, and proposed a simplified version of the atmospheric scattering model:

$$I\left(x\right) = I_{\infty}\rho\left(x\right)e^{-\beta d(x)} + I_{\infty}\left(1 - e^{-\beta d(x)}\right) \tag{1}$$

In formula (1), I is the brightness of the sky, $\rho(x)$ denotes the normalized radiance of a scene point x, β is the scattering coefficient of the atmosphere, and d is the scene depth. However, this model is too complicated, so a simplified atmospheric scattering model is proposed. The simplified atmospheric

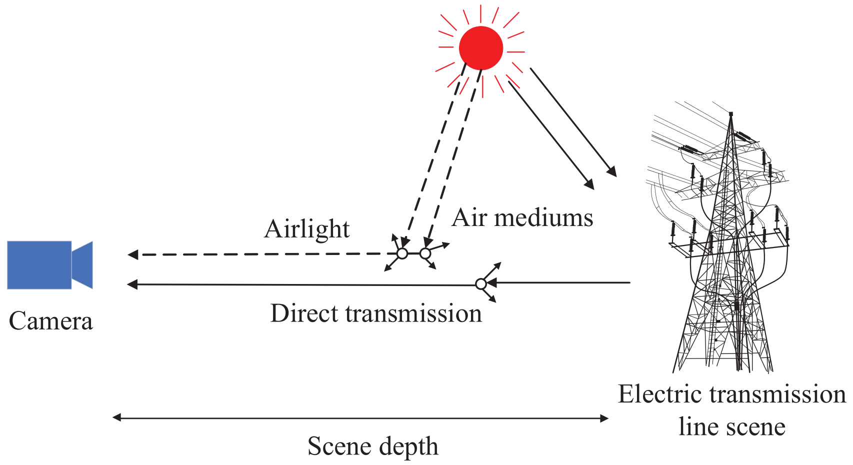


FIGURE 1
Atmospheric scattering model.

scattering model developed by he is now the most used atmospheric scattering model for expressing the principle of fog (He et al., 2010). It is shown in the following formula:

$$I(x) = J(x)\,t(x) + A(1 - t(x)) \tag{2}$$

where A is the global atmospheric light, which represents the background lighting in the atmosphere, and $I(x)$ and $J(x)$ are the fogging and defogging images, respectively. And $x = (m, n)$ is the coordinate of the image. $t(x)$ is transmission. It represent the transmission of a medium that is not scattered and successfully entries into vision systems such as monitoring systems and cameras. As per the atmospheric scattering theory, the scattering of air light during the process of reaching the vision system and the attenuation process of the reflected light from the surface of the object reaching the vision system are the two main divisions of the scattering of atmospheric particles. For equation (2), $J(x)t(x)$ is direct transmission, and $A[1\text{-}\ t(x)]$ is airlight, denoted as $a(x)$. Direct transmission means the attenuation of the foggy image directly passing through the air medium, and the airlight is generated by the scattered light. The schematic diagram for the atmospheric scattering model is shown in **Figure 1**. Note that the solid line represents direct transmission and the dashed line represents airlight.

For transmission $t(x)$, we have:

$$t(x) = e^{-\beta d(x)} \tag{3}$$

In the above formula, $d(x)$ represents the scene depth and, at the same time, β is the atmospheric scattering coefficient. The formula shows that the transmission decreases gradually as the depth of the scene increases.

Trying to imply the transmission $t(x)$ and the global atmospheric light value A into the atmospheric scattering physical model yields the defogging image J, which is the essential step in image defogging based on the atmospheric scattering model. The following formula can be obtained by deriving formula (2):

$$J(x) = \frac{I(x) - A}{t(x)} + A(x) \tag{4}$$

It can be seen from formula (4) that the key to calculating the defogging image is to reasonably estimate the transmission $t(x)$ of the foggy image and the global atmospheric light value A. At present, the most commonly used method of defogging is the dark channel prior theory proposed by He et al. (2010).

2.2. Dark channel prior theory

He gained a statistical rule by observing a significant number of images without fog: for a large number of non-sky local patches, there is always at least one color channel with pixel intensity so low that it is close to 0. So the dark channel $J^{dark}(x)$ is defined by the following formula:

$$J^{dark}(x) = \min_{y \in \Omega(x)} \left\{ \min_{c \in \{r,g,b\}} \left[J^c(y) \right] \right\} \tag{5}$$

where $\Omega(x)$ is the local area centered at x, y is the pixel in the local area $\Omega(x)$, J^C is the color channel of the fog-free image J, C is the three channels of the RGB image. And r, g, b represent the red, green and blue channels of the RGB image, respectively.

He draws the following conclusion through observation: for an outdoor fog-free image J, due to the shadows caused by buildings in the city or leaves in the natural landscape, the surfaces of colored objects with low reflectivity, and the surfaces of dark objects, dark channel intensity of J for non-sky regions is exceedingly low, almost nothing. So there is the following formula:

$$J^{dark} \to 0 \tag{6}$$

The transmission calculation formula may be constructed using the dark channel prior theory and the atmospheric scattering model above as follows:

$$t(x) = 1 - \omega \min_{y \in \Omega(x)} \left\{ \min_{c} \left[\frac{I^c(y)}{A^c} \right] \right\} \tag{7}$$

The role of ω is to retain some fog to make the image appear more natural. The value range of ω is (0, 1), and the value is generally set to 0.95.

The transmission obtained by this method is not accurate, and is prone to "halo" effect. The "halo" phenomenon is an effect that tends to occur in images after the fog has been removed. Since the foreground is close to the observation point and the background is far from the observation point in the image, the depth of field of different positions in the image has a large gap, especially for the junction of the foreground and background. Therefore, the "halo" phenomenon is usually generated at the junction of the foreground and background of the image, resulting in abnormal color distortion at the edge of the observed object in the image after fog removal, and the "halo" phenomenon gradually weakens when the image is far away from the edge. Therefore, He optimized the transmission using "soft matting" to get rid of the "halo" effect. However, the "soft matting" consumes a lot of time, so it is not suitable or practical applications. Therefore, He proposed the guided filter, through which the transmission optimization time can be greatly shortened, and the resulting image edges are sharper.

In order to prevent the image from being enhanced too much due to too small transmission, it is required to define the bottom bound of transmission t_0, which is usually set to 0.1. Then the final result can be obtained from the following equation:

$$J(x) = \frac{I(x) - A}{\max(t(x), t_0)} + A \tag{8}$$

It can be seen from Formula (8) that for a given fogged image $I[x]$, to obtain the image after defogging [that is, $J(x)$], only two unknown quantities need to be solved: global atmospheric light value A and transmission t. Therefore, when using the atmospheric scattering model for image defogging, the most important two steps are the calculation of global atmospheric light A and the calculation of transmission t.

2.3. Disadvantages of dark channel priors

In the dark channel prior algorithm, the global atmospheric light is chosen in the brightest color channel in the image. He picks the pixels with the highest intensity as the global atmospheric light after first detecting the brightest top 0.1 percent of the dark channel pixels. However, this process suffers from large areas of white objects or objects that are too bright in the image. At this point the global atmospheric light is misestimated, resulting in a color shift in the recovered image. Second, it is common to create a "halo" phenomenon in the region separating the image's foreground and background when employing the dark channel prior algorithm for regions with discontinuous depths. Finally, the atmospheric scattering in the real situation is multiple scattering. The single scattering model is the most often used atmospheric scattering model since it is challenging to compute multiple atmospheric scattering. As a result, the defogging images obtained by the dark channel prior algorithm are often too smooth and lack of image details. Therefore, this paper will optimize the dark channel prior algorithm for these three aspects.

3. Proposed method

The defogging algorithm flowchart from this work is shown in **Figure 2**. Electricity transmission lines and power towers are often located outdoors, and their images often have large areas of the sky. According to the statistical law that the sky area often exists in the upper part of the image, in order to address the issue of erroneous estimation of the global atmospheric light due to the influence of a large area of white objects, a global atmospheric light solution based on the optimized quadtree algorithm is proposed. This ensures correct estimation of global atmospheric light. Then we define dark pixels, perform superpixel segmentation on the input foggy image, and locate dark pixels in the segmented superpixel block. The transmission is calculated through a fidelity function, and the solved transmission is optimized for color correction. The next step is to invert the atmospheric scattering model to produce a preliminary defogging image. Due to the thin size of electricity transmission line, which is not suitable for observation, and the defogging image lacks details, a detail sharpening post-processing algorithm based on airlight constraints and visibility constraints are used for the preliminary defogging image to improve the texture details of the image. Finally, the final defogging image $J(x)$ is obtained.

3.1. Global atmospheric light estimation

The presence of fog in the image will lead to a brighter area in the image, so the global atmospheric light is usually selected in the bright area, usually in the sky area. However, white objects on the ground or high-brightness objects can easily induce an incorrect selection of the global atmospheric light, resulting

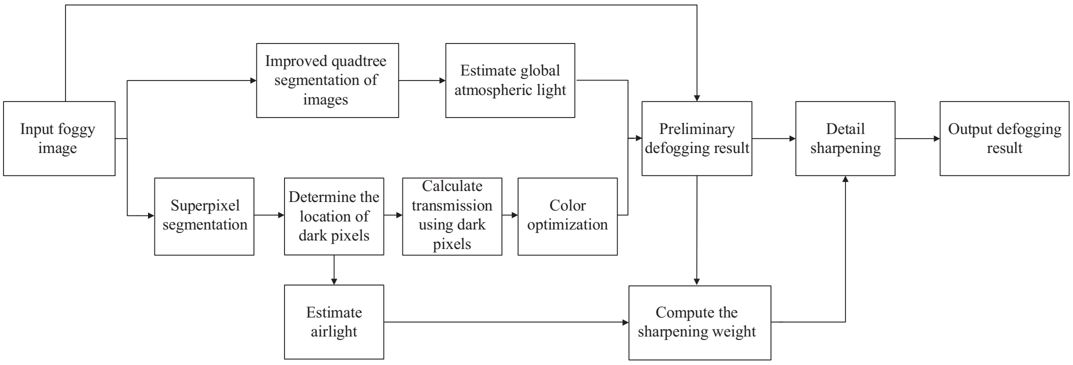

Input foggy image
Improved quadtree segmentation of images
Estimate global atmospheric light
Superpixel segmentation
Determine the location of dark pixels
Calculate transmission using dark pixels
Color optimization
Preliminary defogging result
Detail sharpening
Output defogging result
Estimate airlight
Compute the sharpening weight

FIGURE 2
The flow chart of the proposed method.

in chromatic aberration in the fog-free image. Therefore, it is necessary to improve the selection of global atmospheric light.

An optimized quadtree segmentation algorithm is used to estimate global atmospheric light in this study. Based on the principle that the pixel value variance of the image is often low in the foggy area, Kim et al. (2013) suggested an algorithm based on quadtree segmentation to select the global atmospheric light. The input image is first divided into four sections. Then we name the upper left area as area A, the upper right area as area B, the lower left area as area C, and the lower right area as area D. Then, for each region, we determine the standard deviation and mean of the pixel values for the three R, G, and B channels, and subtract the standard deviation from the mean to obtain a region score. The region with the highest score is first determined. It is then divided into four smaller parts, and the region with the highest score is selected from those four. Repeat the aforementioned procedure up until the size of the selected area is below the predetermined threshold. In the final selected area, we look for the value of the pixel closest to the white area as the global atmospheric light. For an RGB image, the white part is the area where the three channels of R, G, and B are all 255. Hence the estimate of global atmospheric light can be transformed into finding the minimum of the following formula:

$$\left|\left|\left(I_r\left(x\right), I_g\left(x\right), I_b\left(x\right)\right) - \left(255, 255, 255\right)\right|\right| \tag{9}$$

In Formula (9), I represents the input image, r, g, b represent the RGB color channel of the input image, x represents the pixel value of each pixel of the input image, and 255 represents the white in the RGB space. Through the quadtree segmentation method, the global atmospheric light can be selected in a brighter area as much as possible. However, when there are large areas of white objects or high-contrast objects in the non-sky area, the quadtree segmentation algorithm will still select the non-sky area as the global atmospheric light, as shown in Figure 3. Images are from the OTS dataset in the Realistic Single Image Dehazing dataset. We usually call it RESIDE. The green line in the image represents the quadtree segmentation process, and the red fill represents the final selected area. It can be seen from Figure 3 that the estimation of the global atmospheric light may be disturbed by the white objects on the ground, and the final result is chosen on the ground and the lake surface instead of the sky.

We optimize the quadtree segmentation for estimating the global atmospheric light to address this issue. For the four regions A, B, C, and D, we calculate their regional scores and record them as $score_A$, $score_B$, $score_C$, and $score_D$, and then compare the scores. Because the sky is mostly concentrated in the upper portion of the image, if the area with the highest score in the first step is located in the upper half of the image, that is, the $score_A$ or the $score_B$ has the highest score, the subsequent segmentation operation will be continued. If the region with the highest score in the first step is located in the lower half of the image, that is, the $score_C$ or $score_D$ has the highest score, then we assign the calculation weights ξ_A and ξ_B to the $score_A$ and $score_B$, respectively, and the scores are recorded as $\xi_A{\cdot}score_A$ and $\xi_B{\cdot}score_B$. Then calculation process returns to the first segmentation process, and re-compare the scores of $\xi_A{\cdot}score_A$, $\xi_B{\cdot}score_B$, $score_C$, and $score_D$, so that the global atmospheric light can be located at the top half of the image in the first step. To ensure that the recalculated $\xi_A{\cdot}score_A$ and $\xi_B{\cdot}score_B$ can be larger than $score_C$ and $score_D$, the calculation weight ξ is set to 1.5. Finally, the average value of the selected area is used as the global atmospheric light. Figure 4 depicts the process mentioned previously.

Figure 5 shows the global atmospheric light selection result for the three foggy images given in Figure 3 using the optimized quadtree segmentation algorithm. The selected area of the global atmospheric light is changed from bottom half of images to the sky area. This shows that for foggy images with a sky, this method can locate the global atmospheric light in the sky area in the upper half of the image, and avoid locating it on the

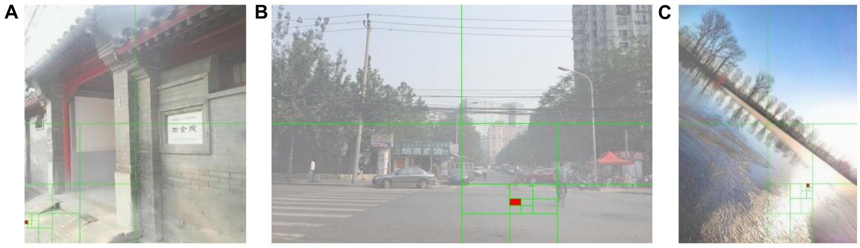


FIGURE 3
Estimation of global atmospheric light using quadtree segmentation algorithm.

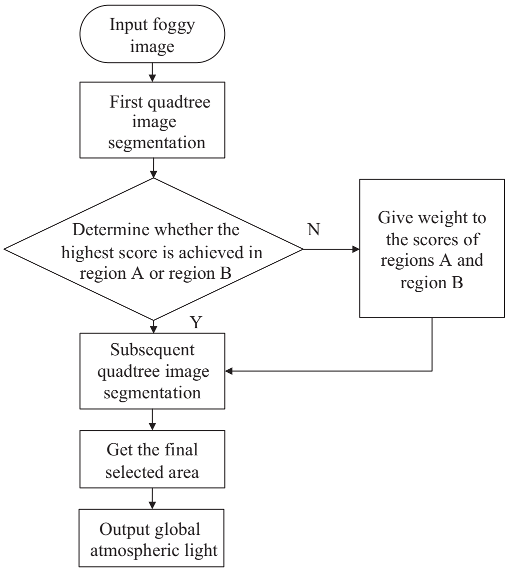

Input foggy image
First quadtree image segmentation
Determine whether the highest score is achieved in region A or region B
N
Give weight to the scores of regions A and region B
Y
Subsequent quadtree image segmentation
Get the final selected area
Output global atmospheric light

FIGURE 4
Flowchart of the optimized quadtree segmentation algorithm.

ground or large areas of white objects and other interfering objects.

3.2. Transmission optimization

Since the dark channel prior uses the minimum filter to calculate the transmission, it is common for a depth discontinuity to emerge at the image's boundaries. Hence it is easy to produce a "halo" phenomenon at the boundary between the foreground and background. When the foggy image is converted into a dark channel map to extract the clear part, the size of the local patch Ω is difficult to determine. When deriving the transmission calculation formula, the original dark channel prior method assumes that the transmission in the local patch Ω is a constant, which is not consistent with the real situation. To optimize transmission, a method combining super-pixel segmentation and the dark pixel is utilized in this research.

Zhu pointed out in Zhu et al. (2019) that dark pixels are widespread. First, this paper defines dark pixels as follows:

$$\min_{c} J^{c}(z) \rightarrow 0 \tag{10}$$

Simple linear iterative clustering (Achanta et al., 2012) is used for super-pixel segmentation of images. The technology can be called SLIC for short. Superpixel segmentation uses adjacent pixels with the same brightness and texture characteristics to form irregular pixel blocks, and aggregates pixels with similar characteristics to achieve the purpose of using a small number of superpixel blocks to replace a large number of pixels in original images. When the superpixel block is too large, block artifacts and "halo" phenomena may also occur, which are caused by the discontinuity of depth caused by the excessively large superpixel, so the size of the superpixel block needs to be selected reasonably. We partition the foggy image into 1,000 superpixels in this study. Next, we need to locate dark pixels in the generated superpixel block. We locate dark pixels using the local constant assumption (Zhu et al., 2019). Note that the local constant assumption is only used to locate dark pixels, not to estimate transmission. For each superpixel local patch Ω, there is at least one dark pixel in it. From the assumption that the amount of transmission in the local patch Ω of each superpixel is constant, it can be known that dark pixels are found in each superpixel local area Ω by finding a local minimum in $\min_{c} I_{w}^{c}$, where $\min_{c} I_{w}^{c} = \min_{c}\left[I^{c}(x)\, A^{c}\right]$.

For each dark pixel, there is the following formula:

$$\min_{c} \frac{I^{c}(z)}{A^{c}} = \left[1 + t(z)\right] - t(z) \min_{c} \frac{J^{c}(z)}{A^{c}} \tag{11}$$

A certain amount of fog should be preserved in order to give the image a more realistic appearance, and take $\mathrm{J^{c}}(z)/\mathrm{A^{c}} = 0.05$ (Zhu et al., 2019). Bringing formula (10) into formula (11), there is the following formula:

$$0.95t(z) \approx 1 - \min_{c} \frac{I^{c}(z)}{A^{c}} \tag{12}$$

For any pixel x, the smaller the value of $\min_{c}\left[I^{c}(x)/A^{c}\right] = \min_{\Omega,c}\left[I^{c}(x)/A^{c}\right]$ is, the closer

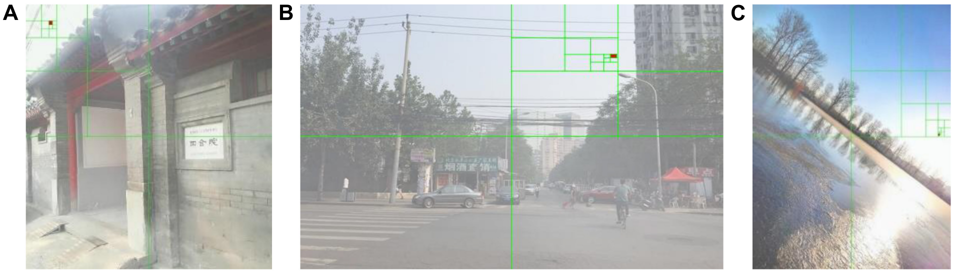


FIGURE 5
Estimation of global atmospheric light using an optimized quadtree segmentation algorithm.

$\min_c [I^c(x)/A^c]$ is to the minimum value of the local patch Ω, and the more likely the pixel is to be a dark pixel (Zhu et al., 2019). To ensure that x is a dark pixel, $\min_c [I^c(x)/A^c]$ and $\min_{\Omega,c} [I^c(x)/A^c]$ should be close enough. Therefore we define the fidelity function $F(x)$ for the dark pixel x as follows:

$$F(x) = \log_{0.001}\left\{\max\left[\min_c \frac{I^c(x)}{A^c} - \min_{\Omega,c}\frac{I^c(x)}{A^c}, 0.001\right]\right\} \tag{13}$$

As can be seen from the above, the closer $\min_c [I^c(x)/A^c]$ and $\min_{\Omega,c} [I^c(x)/A^c]$ are, the more likely pixel x is to be a dark pixel. Formula (13) is a fidelity function. According to this formula, when the difference between $\min_c [I^c(x)/A^c]$ and $\min_{\Omega,c} [I^c(x)/A^c]$ is less than 0.001, it can be seen from the property of logarithmic function that the value of $F(x)$ is 1, thus $\min_c [I^c(x)/A^c] = \min_{\Omega,c} [I^c(x)/A^c]$. Therefore, it can be approximately considered that the pixel x is the expected dark pixel. Therefore, there is the following formula:

$$\tilde{t}(x) \approx \frac{\left[1 - \min_c \frac{I^c(x)}{A^c}\right]}{0.95} = \frac{1 - \min_\Omega \left[\min_c \frac{I^c(x)}{A^c}\right]}{0.95} \tag{14}$$

The final transmission is obtained by optimizing the following energy function:

$$E(t) = \sum_x F(x)\left[t(x) - \tilde{t}(x)\right]^2 + \lambda\left[a_{x,N(\tilde{t})}\left(\frac{\partial t}{\partial x}\right)^2 + a_{y,N(\tilde{t})}\left(\frac{\partial t}{\partial y}\right)^2\right] \tag{15}$$

where $a_{x,N(\tilde{t})}$ and $a_{y,N(\tilde{t})}$ are weight coefficients, defined as:

$$a_{x,N(\tilde{t})} = \left[\left|\frac{\partial\left(I^c/A^c\right)}{\partial x}\right|^2 + \varepsilon\right]^{-1} \tag{16}$$

$$a_{y,N(\tilde{t})} = \left[\left|\frac{\partial\left(I^c/A^c\right)}{\partial y}\right|^2 + \varepsilon\right]^{-1} \tag{17}$$

The final transmission can be obtained from the above formula, as shown in the following formula:

$$\overrightarrow{t} = \left(\overrightarrow{F} + \lambda\overrightarrow{L}\right)^{-1}\overrightarrow{F}\overrightarrow{\tilde{t}} \tag{18}$$

where $\overrightarrow{t}$ is the vector form of t, $\overrightarrow{\tilde{t}}$ is the vector form of $\tilde{t}$. And $\overrightarrow{F}$ is a sparse diagonal matrix composed of elements in F, $\overrightarrow{L}$ is the Laplace matrix. The value of λ is 0.02.

The restored fog-free image may have color offset problems such as too dark color in non-sky area and overexposure color in bright sky area. It is necessary to perform color correction on the obtained transmission. Color correction for transmission t is performed by the following formula:

$$t = \frac{\max\left(t, 1 - \min_c \frac{I^c(x)}{A^c}\right) + \sigma}{1 + \sigma} \tag{19}$$

where 0.2 is used as the value for σ.

Figure 6 shows the comparison of the dark channel prior method and proposed method on transmission and recovered images. As shown in Figure 6C, when the transmission is estimated using the dark channel prior algorithm, the dark channel map contains some depth-independent details. And for thin overhead lines of electricity transmission lines, the range of the transmission map will be overestimated, resulting in halo artifacts near the overhead lines in the defogging results. As shown in Figure 6E, the proposed method can avoid overestimating the transmission of the overhead line, avoid halo artifacts near overhead line, and provide better transition at the junction of overhead line and sky. Comparing the method to the ground truth in Figure 6B, color saturation may also be avoided.

3.3. Detail sharpening

The transmission line is thin in size and difficult to observe, so it is necessary to enhance the details of the defogging image in order to better observe the electricity transmission line. The actual atmospheric scattering is multiple scattering, while the commonly used atmospheric scattering model is only one scattering, which will lead to the loss of details and blurred images in the defogging image. Therefore, it is necessary to sharpen the details of the defogging image. Image blur caused by multiple scattering is mainly related to two factors: visibility level and airlight level. The visibility level is related to detail,

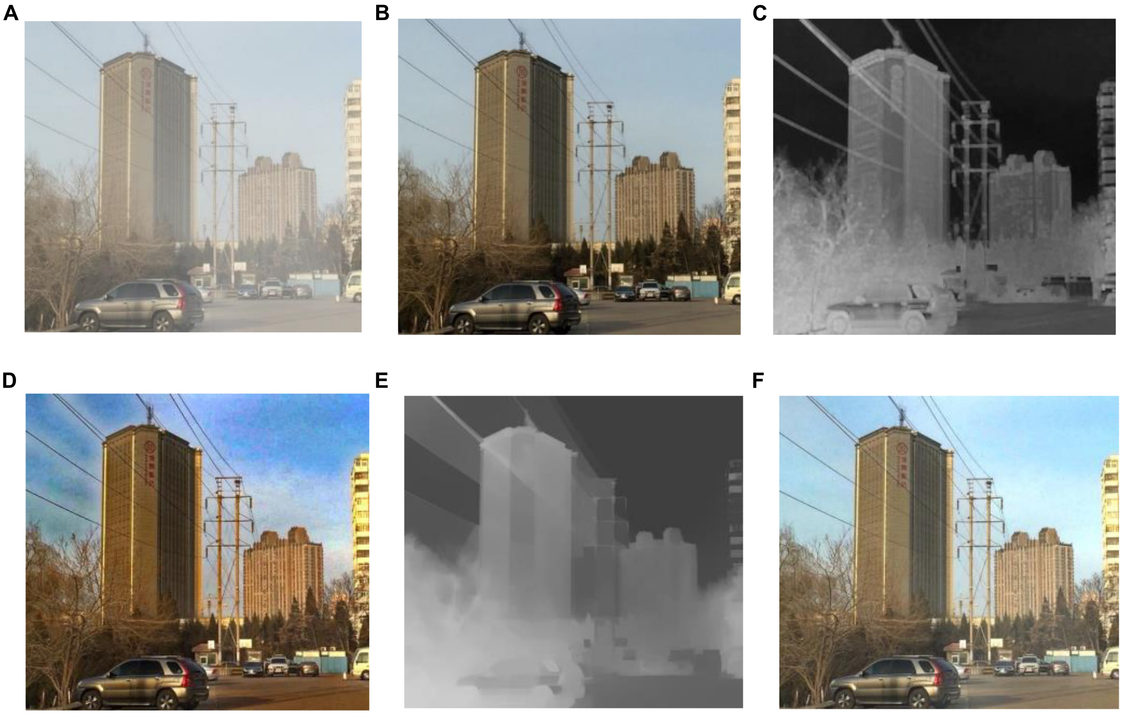


FIGURE 6
The effect of this method on transmission line transmittance optimization. **(A)** Hazy image; **(B)** ground truth **(C)** transmission of dark channel prior; **(D)** result of dark channel prior; **(E)** transmission of our method; **(F)** preliminary result of our method.

and the airlight level is related to depth (Gao et al., 2018). If the airlight level in a certain area of the image is high, the image details in that area will be smoother, so the degree of image sharpening is proportional to the airlight level. And the smoothness of the image details is poor when there is high visibility in a certain area, hence the degree of image sharpening is inversely related to the visibility level. The following is a definition of the sharpening coefficient:

$$S\left(x,y\right)=S_1\left(x,y\right)\circ S_2\left(x,y\right) \tag{20}$$

In formula (20), $S(x, y)$ represents the sharpening coefficient matrix. The function determined by the airlight level is represented by $S_1(x, y)$, while the function determined by the visibility level is represented by $S_2(x, y)$. $S(x, y)$ means the multiplication of the corresponding elements of the $S_1(x, y)$ and $S_2(x, y)$ matrices.

Sigmoid function can satisfy the requirement that the airlight level is proportional to the sharpening coefficient, and the visibility level is inversely proportional to the sharpening coefficient. We use the cumulative distribution function as the constraint functions for the airlight level and visibility level. This function is a sigmoid function, expressed as follows:

$$\phi\left(x\right)=\frac{1}{2}\left[1+erf\left(\frac{x}{\sqrt{2}}\right)\right] \tag{21}$$

The following formula is the error function $erf(x)$:

$$erf\left(x\right)=\frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt \tag{22}$$

The cumulative distribution function $\Phi(x)$ is an sigmoid function that increases monotonically with x. The cumulative distribution function can meet the requirement that the airlight level is proportional to the sharpening coefficient and the visibility level is inversely proportional to the sharpening coefficient. For the cumulative distribution function, it approaches 0 as x approaches $-\infty$ and 1 as x approaches ∞, and the cumulative distribution function is a monotonically increasing function. If we add a minus sign to the cumulative distribution function, we get a monotonically decreasing function. Therefore, the cumulative distribution function can be used as the constraint function of airlight level and visibility level. As a result, the following definitions apply to the airlight level and visibility level constraints:

$$S_1\left(x,y\right)=\frac{1}{2}\left\{1+erf\left[\frac{a\left(x,y\right)-a_{ave}}{\sqrt{2}k_1}\right]\right\} \tag{23}$$

$$S_2\left(x,y\right)=1-\frac{1}{2}\left\{1+erf\left[\frac{C\left(x,y\right)-C_{ave}}{\sqrt{2}k_2}\right]\right\} \tag{24}$$

where $a(x, y)$ denotes the airlight level, and $C(x, y)$ represents the visibility level. a_{ave} represents the average value of the airlight

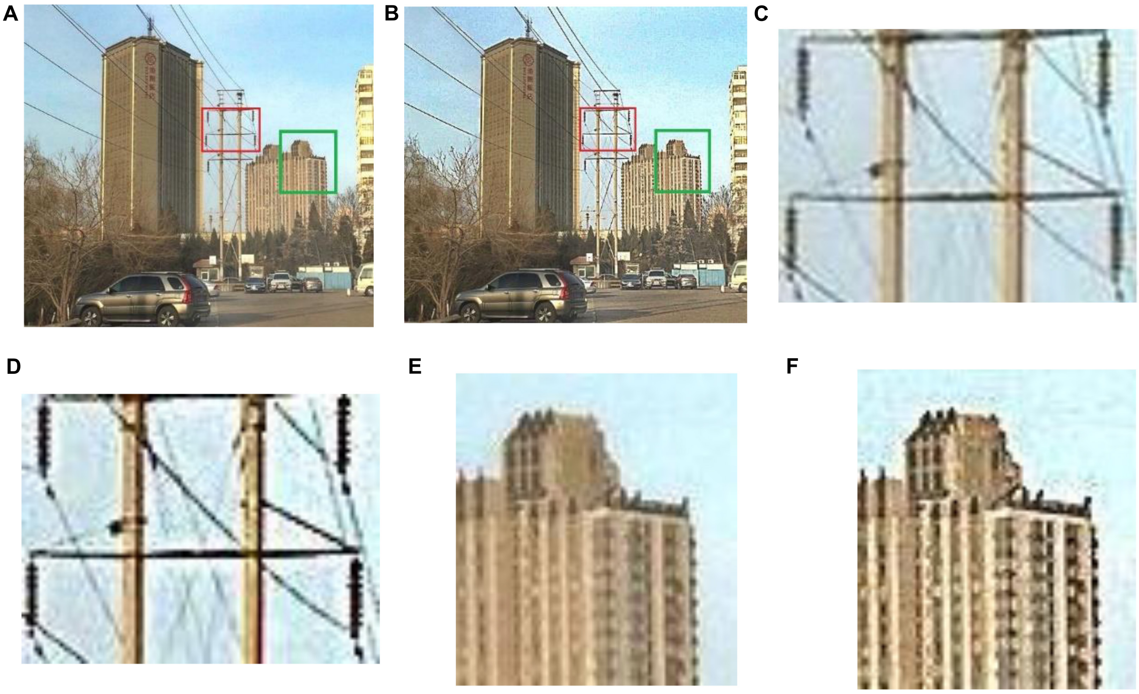


FIGURE 7

Comparison of local details between the two methods. **(A)** Preliminary defogging result without sharpening; **(B)** Final defogging result after sharpening; **(C)** Local detail of intermediate defogging result (insulators); **(D)** Local detail of final defogging result (insulators); **(E)** Local detail of intermediate defogging result (buildings); **(F)** Local detail of final defogging result (buildings).

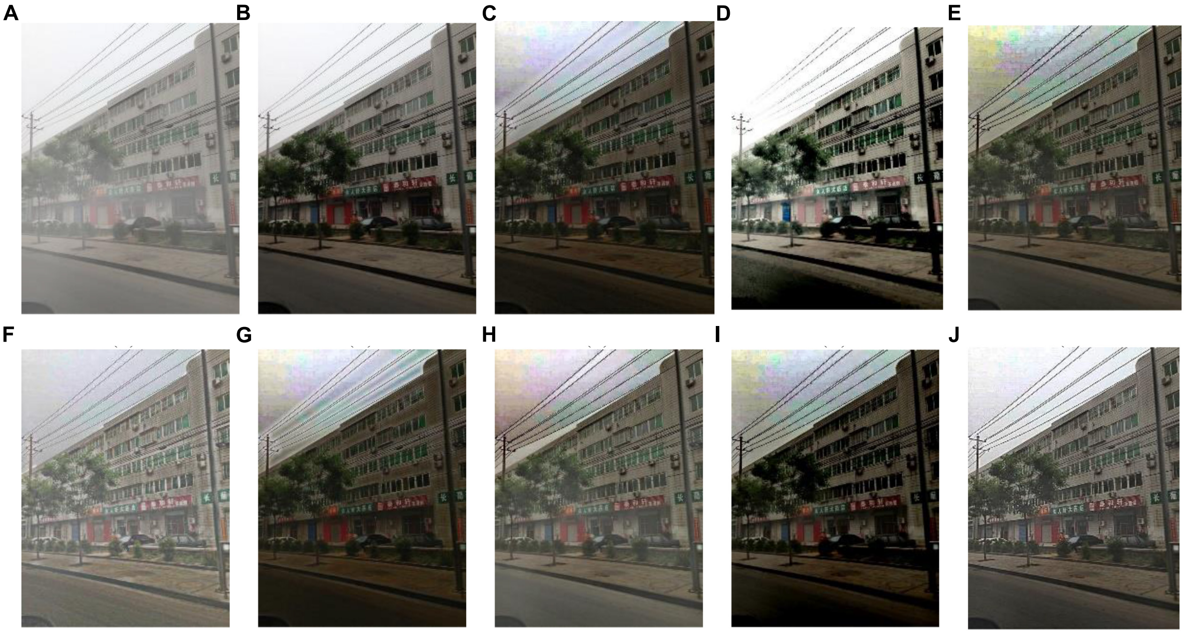


FIGURE 8

Experimental results of different methods for Figure 1: **(A)** input image; **(B)** ground truth; **(C)** dark channel prior; **(D)** Fattal et al.; **(E)** Meng et al.; **(F)** Tarel et al.; **(G)** Ehsan et al.; **(H)** Berman et al.; **(I)** Raikwar et al.; **(J)** proposed.

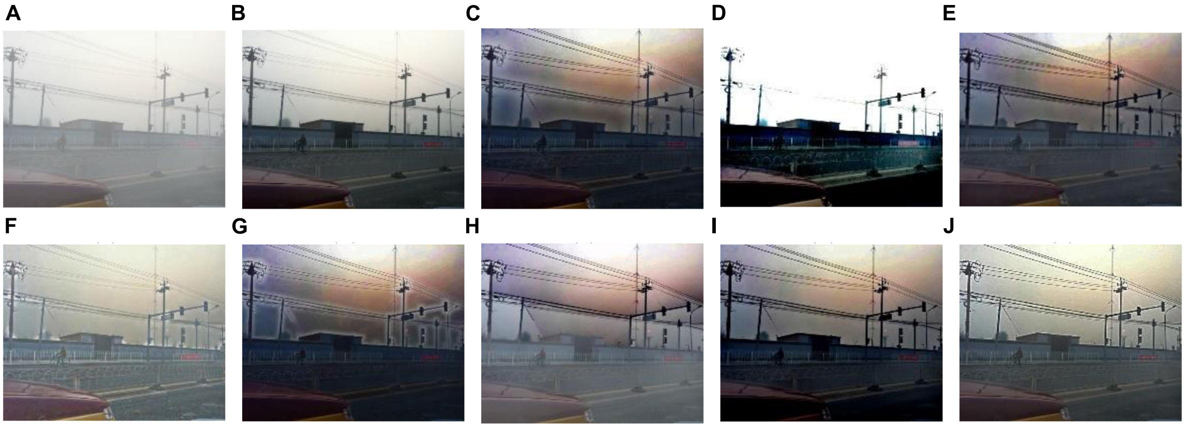


FIGURE 9
Experimental results of different methods for Figure 2: **(A)** input image; **(B)** ground truth; **(C)** dark channel prior; **(D)** Fattal et al.; **(E)** Meng et al.; **(F)** Tarel et al.; **(G)** Ehsan et al.; **(H)** Berman et al.; **(I)** Raikwar et al.; **(J)** proposed.

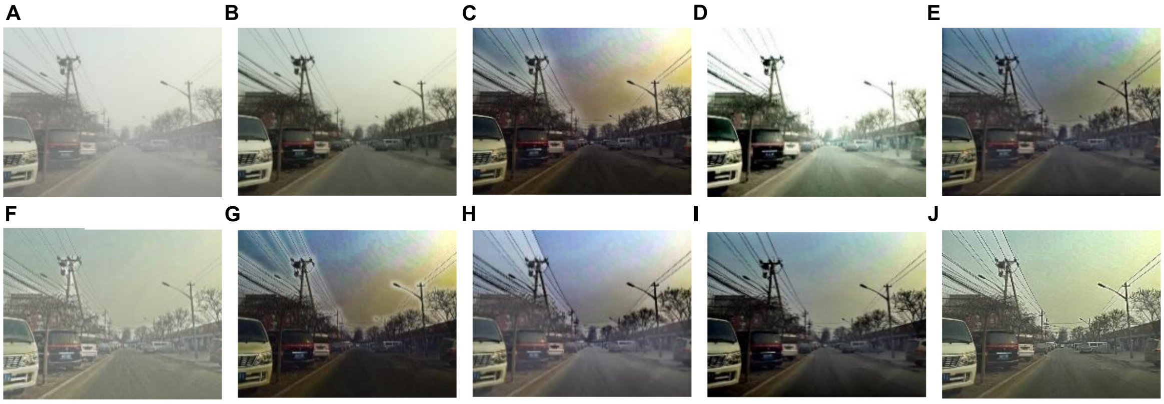


FIGURE 10
Experimental results of different methods for Figure 3: **(A)** input image; **(B)** ground truth; **(C)** dark channel prior; **(D)** Fattal et al.; **(E)** Meng et al.; **(F)** Tarel et al.; **(G)** Ehsan et al.; **(H)** Berman et al.; **(I)** Raikwar et al.; **(J)** proposed.

level, C_{ave} represents the average value of the visibility level, and k_1 and k_2 are the slope control coefficients. The visibility level C has a relationship with the Weber brightness (Hautiere et al., 2008). The expression for visibility level C is as follows:

$$C(x) = \frac{\Delta L(x,y)}{L_b(x,y)} = \frac{L_t(x,y) - L_b(x,y)}{L_b(x,y)} \tag{25}$$

In the above formula, ΔL is the brightness difference between the preliminary defogging result and the background image, L_t is the brightness of the preliminary defogging result, and L_b is the brightness of the image background. RGB space is the most commonly used color space, including three basic colors: red (R), green (G), and blue (B), while YCbCr is another color space, including luminance component (Y), blue chrominance component (Cb), and red chrominance component (Cr). To calculate the brightness difference, the image needs to be transferred from RGB space to YcbCr space. The preliminary defogging result is converted from RGB to YCbCr space, and the brightness component L_t is extracted, then the preliminary defogging result is low-pass filtered to produce L_b.

The final enhancement result is as follows:

$$J_{final}(x,y) = J(x,y) + \theta \cdot S(x,y) \circ T(x,y) \tag{26}$$

For formula (26), $J_{final}(x,y)$ is the final defogging result, $J(x,y)$ is the preliminary defogging result, and the upper bound on the enhancement is constrained using θ. $T(x,y)$ is the high-frequency value of the preliminary result $J(x,y)$, which is obtained by Gaussian filtering on $J(x,y)$ to prevent excessive enhancement of flat areas such as the sky. The preliminary defogging result $J(x, y)$ in Formula (26) refers to the defogging result obtained by formula (8) after calculating the global atmospheric light A and transmission t of the input image $I(x)$

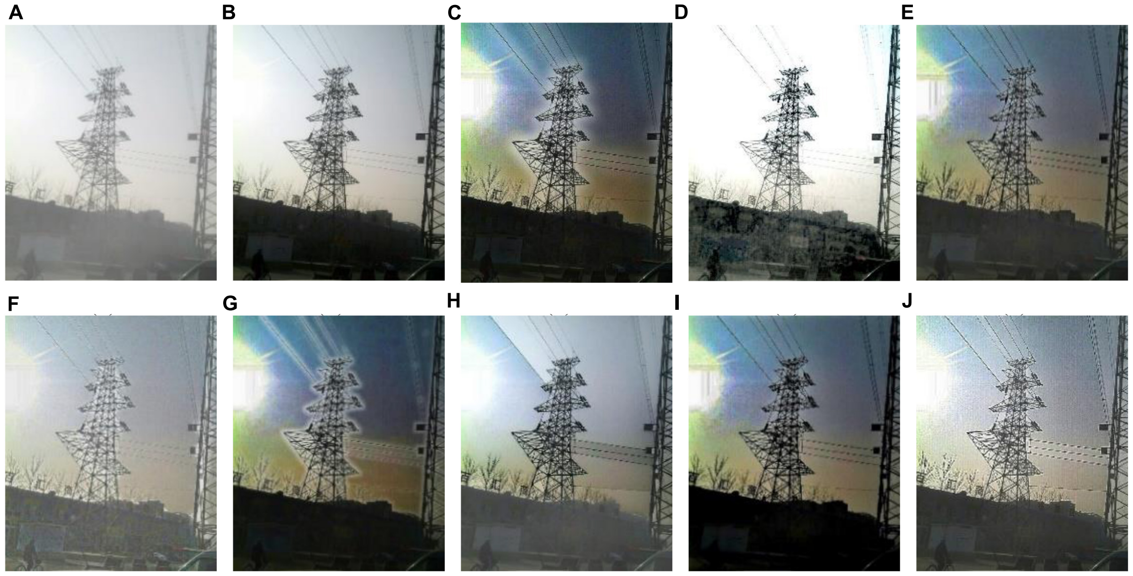


FIGURE 11
Experimental results of different methods for Figure 4: **(A)** input image; **(B)** ground truth; **(C)** dark channel prior; **(D)** Fattal et al.; **(E)** Meng et al.; **(F)** Tarel et al.; **(G)** Ehsan et al.; **(H)** Berman et al.; **(I)** Raikwar et al.; **(J)** proposed.

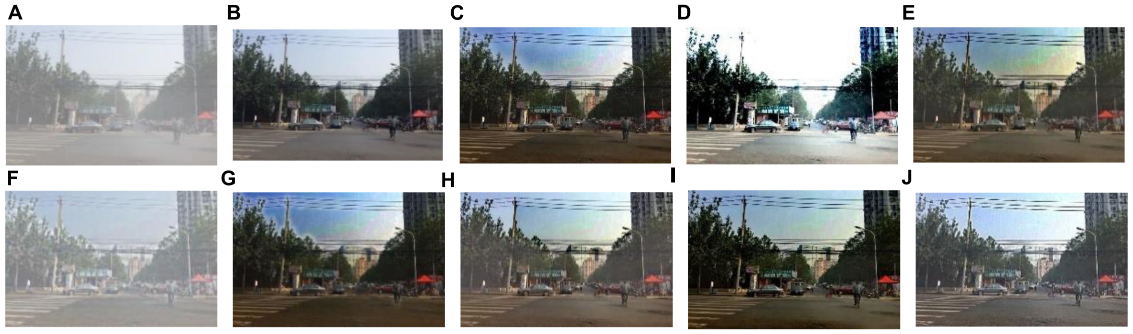


FIGURE 12
Experimental results of different methods for Figure 5: **(A)** input image; **(B)** ground truth; **(C)** dark channel prior; **(D)** Fattal et al.; **(E)** Meng et al.; **(F)** Tarel et al.; **(G)** Ehsan et al.; **(H)** Berman et al.; **(I)** Raikwar et al.; **(J)** proposed.

through the methods of (Section "3.1 Global atmospheric light estimation") and (Section "3.2 Transmission optimization"). That is, the fog-free image without detail sharpening.

The final defogging result after detail sharpening has more accurate local details. Figure 7 shows the comparison between the preliminary defogging results without sharpening and the final defogging results after sharpening. The details of the electricity transmission lines and insulators after sharpening are richer. According to Figure 7D, the electricity transmission line and insulator have more prominent image details after sharpening. In addition, the details of distant buildings have also become clearer. As shown in Figures 7E, F, the details of buildings in the image after detail sharpening are more prominent.

4. Experimental results and discussion

The efficiency of proposed defogging algorithm is examined through qualitative and quantitative comparisons with widely utilized defogging methods. We select foggy images with electricity transmission lines in the RESIDE dataset and compare with the methods of He et al. (2010), Fattal (2008), Meng et al. (2013), Tarel and Hautiere (2009), Ehsan et al. (2021), Berman and Avidan (2016), and Raikwar and Tapaswi (2020). The following parameters are selected for this study: $\xi = 1.5$, $\lambda = 0.02$, $\varepsilon = 0.00001$, $\sigma = 0.2$, $k_1 = 25$, $k_2 = 0.01$, $\theta = 3$. The experimental platform is a 64-bit Windows 10 operating system laptop.

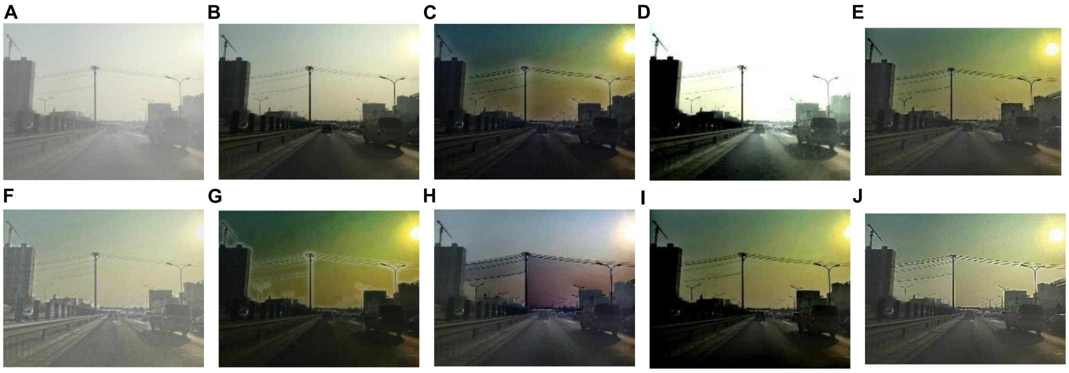


FIGURE 13
Experimental results of different methods for Figure 6: **(A)** input image; **(B)** ground truth; **(C)** dark channel prior; **(D)** Fattal et al.; **(E)** Meng et al.; **(F)** Tarel et al.; **(G)** Ehsan et al.; **(H)** Berman et al.; **(I)** Raikwar et al.; **(J)** proposed.

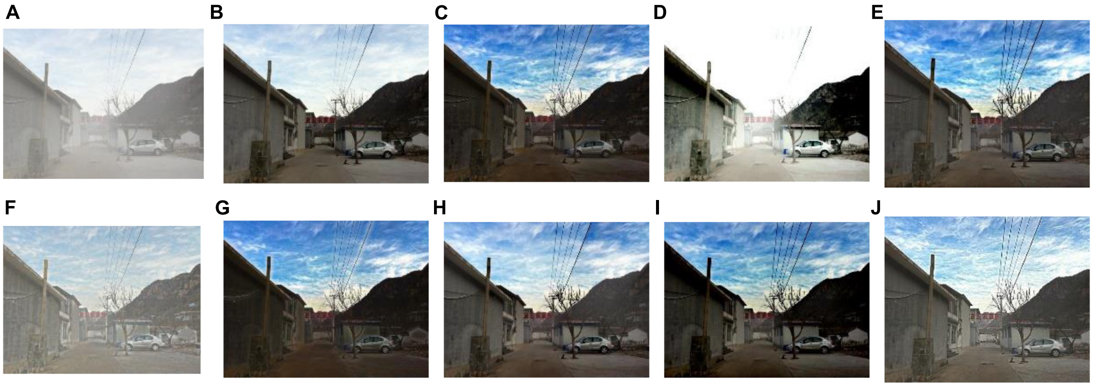


FIGURE 14
Experimental results of different methods for Figure 7: **(A)** input image; **(B)** ground truth; **(C)** dark channel prior; **(D)** Fattal et al.; **(E)** Meng et al.; **(F)** Tarel et al.; **(G)** Ehsan et al.; **(H)** Berman et al.; **(I)** Raikwar et al.; **(J)** proposed.

The CPU is Inter(R) Core i7-11800 H and clocked at 2.30 GHz. The GPU is NVIDIA RTX3060. The computer memory is 40 GB. The software platform is MATLAB 2021b.

4.1. Qualitative comparison

We select 8 foggy images with electricity transmission lines or power towers from the OTS dataset in the RESIDE dataset with ground truth as research objects. We name images as Figures 1–8. The defogging results are shown in Figures 8–15. Note that the post-processing of the transmission of He's method employs guided filter, rather than soft matting.

It can be seen from Figures 8–15 that dark channel prior and Meng's method will over enhance the sky area, resulting in color deviation or over saturation of the sky area, while it is too dark for the non-sky area. Therefore, the fog removal image is very different from the ground truth. The reason for this is that the transmission is often underestimated when using these methods. At the same time, it is noted that when He's method is used to defog the electricity transmission line, because the power tower and electricity transmission line are used as the foreground and the sky area is used as the background, there is a depth discontinuity between the foreground and the background, so there will be an obvious "halo" effect at the edge of the electricity transmission line, which is not conducive to observation. The "halo" phenomenon will lead to the transition area of color deviation between the image to be observed and the background, and produce abnormal colors of white or other colors around the image, affecting the observation of the image. This phenomenon is particularly obvious when dealing with Figures 2, 3, 4, 8.

For Fattal's method, the sky area in the defogging image will be overexposed, resulting in chromatic aberration in the sky area. In addition, due to the tine size of the transmission line, the image of the electricity transmission line occupies a small proportion in the entire image, so the image information of the

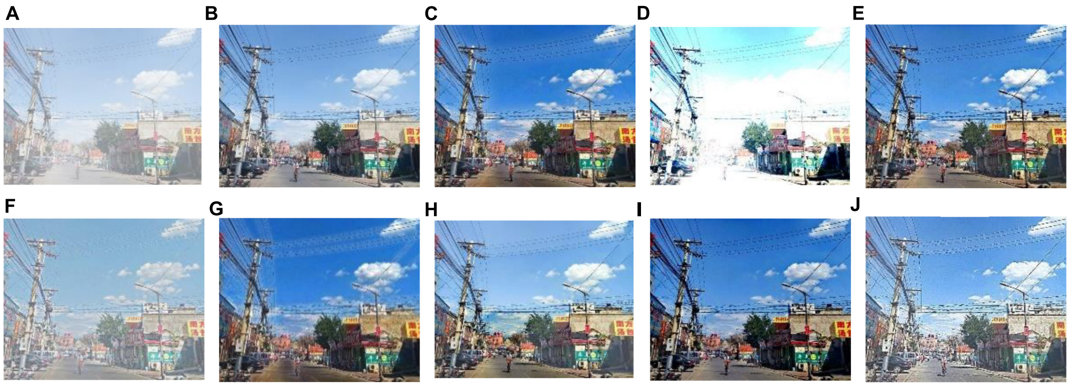


FIGURE 15
Experimental results of different methods for Figure 8: **(A)** input image; **(B)** ground truth; **(C)** dark channel prior; **(D)** Fattal et al.; **(E)** Meng et al.; **(E)** Tarel et al.; **(G)** Ehsan et al.; **(H)** Berman et al.; **(I)** Raikwar et al.; **(J)** proposed.

TABLE 1 Comparison of peak signal-to-noise ratio (PSNR) of the defogging images.

Figure	He	Fatal	Meng	Tarel	Ehsan	Berman	Raikwar	Our
I	17.4379	16.3903	15.4995	12.5983	16.2771	14.9417	13.8952	**19.82**
II	12.4694	14.655	12.1856	15.3734	12.0424	13.0669	12.7189	**18.5631**
III	14.8967	13.5464	13.5908	14.4461	13.599	16.1208	14.8689	**20.7617**
IV	11.8169	14.5616	11.985	14.6685	11.2372	15.6598	11.4271	**18.2309**
V	16.8005	12.7247	16.6877	10.9627	15.965	18.8939	16.0547	**19.2684**
VI	11.6662	12.5876	13.9475	13.1485	11.4975	15.1161	13.0413	**17.9578**
VII	14.7942	12.2326	15.6319	12.8542	14.4614	19.006	13.8794	**19.6325**
VIII	16.0436	11.5157	16.8198	16.5496	15.0584	20.0116	14.0607	**20.1302**
Average	14.4907	13.5267	14.5435	13.8252	13.7673	16.6021	13.7433	**19.2956**

TABLE 2 Comparison of structural similarity index measurement (SSIM) of the defogging images.

Figure	He	Fatal	Meng	Tarel	Ehsan	Berman	Raikwar	Our
I	**0.7908**	0.6541	0.7542	0.6952	0.7373	0.7186	0.5402	0.7857
II	0.6939	0.4905	0.6952	0.7566	0.6577	0.652	0.5856	**0.7621**
III	0.6798	0.6411	0.6309	**0.8271**	0.6425	0.5822	0.6662	0.8238
IV	0.706	0.6852	0.6622	**0.7349**	0.7057	0.7057	0.7057	0.6986
V	0.7357	0.487	0.6886	0.7378	0.7095	0.7411	0.6692	**0.785**
VI	0.6548	0.5038	0.795	0.7476	0.6958	0.5672	0.5852	**0.8185**
VII	0.7135	0.4965	0.7285	0.7521	0.6884	**0.8242**	0.6088	0.7868
VIII	0.8501	0.5455	0.8467	0.8562	0.8287	**0.9023**	0.7999	0.862
Average	0.7281	0.563	0.7252	0.7634	0.7082	0.7117	0.6451	**0.7903**

electricity transmission line in the defogging image is partially or completely lost, which is not conducive to the observation of the electricity transmission line. This phenomenon has a particularly obvious impact on the results of the Fattal's method for defogging in Figures 5–8. For Tarel's method, this method cannot completely remove fog, the image still retains a part of fog after defogging, and the overall image looks hazy with low saturation. For Ehsan's method, there is a large chromatic aberration in the sky area, and there is a "halo" effect around the electricity transmission line, which is particularly evident in the defogging images of Figures 1, 3, 4, 6, 8. For Berman's method, serious color distortion and color shift occur in the sky area of some defogging images, as shown in Figures 2, 3, 5, which are quite different from the ground truth. This is because this

TABLE 3 Comparison of information entropy of the defogging images.

Figure	He	Fatal	Meng	Tarel	Ehsan	Berman	Raikwar	Our
I	7.1639	6.586	7.3179	7.0902	7.0475	7.2592	6.9901	**7.3549**
II	7.4548	4.1126	7.4348	7.3702	7.3647	7.3263	7.3466	**7.5985**
III	7.5324	5.6041	7.4242	6.8307	7.3969	**7.5966**	7.5963	7.4889
IV	7.5499	5.5866	7.4739	7.2516	7.438	**7.7022**	6.9906	7.5764
V	7.3755	6.5764	7.3821	6.5202	7.2797	7.3142	**7.5267**	7.4763
VI	7.1002	5.2216	7.246	6.9646	7.1061	7.3705	7.1291	**7.4727**
VII	7.4989	5.6138	7.5771	6.9565	7.3666	7.5943	7.5273	**7.6219**
VIII	7.5316	5.6522	7.5435	6.678	7.4648	7.4519	7.5322	**7.6213**
Average	7.4009	5.6192	7.4249	6.9578	7.308	7.4519	7.3299	**7.5264**

TABLE 4 Comparison of mean squared error (MSE) of the defogging images.

Figure	He	Fatal	Meng	Tarel	Ehsan	Berman	Raikwar	Our
I	1172.9902	1492.9522	1832.8653	3574.8292	1532.4007	2084.0716	2651.9413	**677.7674**
II	3682.4637	2226.287	3931.1625	1886.8427	4062.9096	3209.1541	3476.8593	**905.2525**
III	2105.7584	2873.7249	2844.4478	2335.9851	2839.0908	1588.5627	2119.313	**545.6503**
IV	4279.4657	2274.6594	4117.0035	2219.371	4890.5339	2256.9768	4681.292	**977.2104**
V	1358.4117	3472.2773	1394.1472	5209.7195	1646.554	1646.554	1612.9123	**769.5488**
VI	4430.6094	3583.5926	2620.1912	3149.4204	4606.1166	2002.0382	3228.1386	**1040.6388**
VII	2156.0511	3888.8094	1777.8414	3370.2747	2327.7683	817.48	2661.5958	**707.6773**
VIII	1617.0443	4586.8511	1352.3898	1439.1915	2028.7847	648.512	2552.7462	**631.0411**
Average	2600.3493	3049.8942	2483.7561	2898.2043	2991.7698	1781.6687	2873.0998	**781.8483**

TABLE 5 Comparison of universal quality index (UQI) of the defogging images.

Figure	He	Fatal	Meng	Tarel	Ehsan	Berman	Raikwar	Our
I	0.8314	0.7429	0.8439	0.6734	0.7999	0.7679	0.5632	**0.8665**
II	0.726	0.6864	0.8223	0.7922	0.7022	0.7529	0.6101	**0.9369**
III	0.8077	0.8525	0.8725	0.8331	0.7643	0.902	0.7829	**0.9638**
IV	0.7683	0.8261	0.7391	0.7753	0.7278	0.7912	0.5586	**0.8571**
V	0.8791	0.7446	0.9149	0.6979	0.8565	0.898	0.7825	**0.9234**
VI	0.6906	0.7417	0.8542	0.7017	0.7494	0.8552	0.6078	**0.883**
VII	0.8367	0.7472	0.8857	0.7334	0.8224	**0.9041**	0.6716	0.8886
VIII	0.9238	0.8622	0.9433	0.9232	0.9023	0.9645	0.8122	**0.9781**
Average	0.808	0.7755	0.8595	0.7663	0.7906	0.8545	0.6736	**0.9122**

method needs to preset a gamma value for each image, and the most suitable gamma value for different images is an unknown quantity. If the best gamma value for each image is unknown, Berma recommends trying to set the default gamma value to 1. Therefore, Berman's algorithm cannot satisfy all situations, and has limitations for practical use. Raikwar's method can eliminate the "halo" phenomenon effectively, but it will still cause color saturation in the sky area, resulting in color shift. At the same time, the method also has low contrast in the non-sky area, which leads to darkness in the non-sky area of the defogging image and affects the observation of power towers on the ground.

For our algorithm, the color saturation of the image after defogging is moderate, the sky area is not over-saturated, and the ground area is not too low in brightness, the "halo" effect can be reduced at the same time. And the visual effect is most similar to the ground truth. In addition, because the proposed method sharpens and enhances the details of the defogging image, the power towers and electricity transmission lines in the defogging image have a clearer visual effect, retaining clearer

TABLE 6 Comparison of average gradient (AG) of the defogging images.

Figure	He	Fatal	Meng	Tarel	Ehsan	Berman	Raikwar	Our
I	6.2546	9.3965	7.9195	7.2134	6.1852	7.6195	7.6144	**10.391**
II	4.8603	4.7456	6.3885	5.1114	4.6492	5.3087	5.7916	**8.4631**
III	7.138	7.2709	8.193	5.6889	7.006	8.3407	8.1677	**10.0117**
IV	8.6812	8.9836	10.0807	8.1328	8.8091	9.1037	10.1014	**12.2277**
V	8.215	12.1108	9.1919	6.1002	7.9448	8.7118	9.3035	**12.5756**
VI	2.7866	3.4014	2.8994	2.9219	2.5547	3.1448	2.8588	**4.902**
VII	5.4091	4.6841	6.4485	4.4309	5.0904	5.7151	6.1237	**7.6858**
VIII	11.7685	10.2167	14.1335	7.9979	11.8835	12.1639	13.4144	**17.8921**
Average	6.8892	7.6012	8.1569	5.9497	6.7654	7.5135	7.9219	**10.5186**

outlines and more detailed details. It is convenient to observe electricity transmission lines and power towers. The qualitative study above shows that our method has better visual effects, and the fog removal effect is better and more realistic.

4.2. Quantitative comparison

The proposed defogging algorithm will be compared and analyzed with previous defogging algorithms utilizing various test metrics in the section on quantitative comparison. The following are the evaluation methods used in this study: peak signal-to-noise ratio (PSNR), information entropy, structural similarity index measurement (SSIM), mean squared error (MSE), universal quality index (UQI), average gradient (AG). According to whether there is a reference image, image evaluation methods can be divided into full reference image quality assessment and no reference image quality assessment. Full reference image quality assessment refers to comparing the difference between the image to be evaluated and the reference image when there is an ideal image as the reference image. No reference image quality assessment refers to directly calculating the visual quality of an image when there is no reference image. PSNR, SSIM, MSE, and UQI belong to full reference image quality assessment, while information entropy and AG belong to no reference image quality assessment. PSNR is the most commonly used image quality evaluation metric, which is an objective standard to measure the level of image distortion. The similarity between the fog removal image and the ground truth is directly proportional to the value of PSNR. A larger PSNR value means that the smaller the distortion of the defogging image and the better the defogging effect. SSIM is a measurement metric that objectively compares the brightness, contrast, and structure of two images to determine how similar they are to one another. The value of SSIM is a number in the range of 0 to 1, and the closer the value is to 1, the more closely the defogging images resemble the ground truth image. For an image, the average amount of information can be determined *via* the information entropy. The more details and richer colors of the image after defogging, the greater the information entropy. The UQI can reflect the structural similarity between two images. The larger the value of UQI, the closer the two images are. The AG is related to the changing characteristics of the image detail texture and reflects the sharpness. The clearer the image, the higher the value of AG. Note that SSIM and UQI belong to the full reference image quality assessment and need to be compared with the reference image when calculating. Therefore, we selected the ground truth of OTS data set as the reference image, compared the defogging images obtained by different methods with the ground truth, and obtained the evaluation results. Similarly, PSNR and MSE also chose ground truth as the reference image.

Tables 1–6 show the results of evaluation metrics obtained when different defogging algorithms are adopted in Figures 1–8. For each row in the table, the bold value represents that the evaluation metric can obtain the optimal result when the defogging algorithm corresponding to the value is adopted for the image.

As shown in Tables 1–6, for PSNR, MSE and AG, compared with other comparison algorithms, the algorithm proposed in this paper achieves the best effect for each image, and obviously the average evaluation results also achieve the best effect. For SSIM, information entropy and UQI, the proposed method achieves the highest or relatively high performance on single image metrics, respectively, and the best performance on average score. For SSIM, the average value obtained by the method proposed in this paper is 0.7903, which is 3.4% higher than that of Tarel, the second highest ranking method, and 28.76% higher than that of Fattal, the lowest ranking method. For information entropy, the method proposed in this paper achieves an average value of 7.5264, which is 0.99% higher than Berman's method with the second highest ranking and 25.34% higher than Fattal's method with the lowest ranking. For UQI, the method proposed in this paper achieves an average value of 0.9122, which is 5.78% higher than that of Meng, the second highest ranking method, and 26.16% higher than that of Raikwar, the lowest ranking method. Quantitative analysis show s that the image obtained by using the proposed method has better structure similarity, rich information content, better color restoration and clarity. As a result, the proposed method has good visual effect.

For the evaluation results of a single defogging image, Berman's method and Raikwar's method will cause color shift and over-saturation in the sky area, making the sky area more yellow or blue. And the information entropy is an indicator that reflects the richness of the color, so the information entropy is sometimes higher when using Berman's method and Raikwar's method. At the same time, Berman's method requires a gamma value to be set in advance, so the application scenarios are limited. Although Tarel's method is used for some images to obtain the best SSIM, Tarel's method cannot completely remove the fog, and the details of electricity transmission lines in the defogging image are not obvious, which is not conducive to observation.

In general, this method can enhance image details, avoid image distortion and color offset, and has a good defogging effect.

5. Conclusion

In this study, we propose an image defogging algorithm for power towers and electricity transmission lines in video monitoring system. First of all, in view of the statistical law that most of the outdoor electricity transmission line images have a sky area in the upper part of the image, the proposed algorithm uses an improved quadtree segmentation algorithm to find the global atmospheric light, then locates the global atmospheric light in the sky area containing the electricity transmission line, preventing the white or bright objects on the ground interfere with the calculation of global atmospheric light. Second, to solve the "halo" effect when the transmission is computed by the dark channel prior, the algorithm in this paper introduces the concept of dark pixels, and uses superpixel segmentation to locate the dark pixels and use a fidelity function to compute the transmission. Finally, in view of the problem that the size of outdoor electricity transmission lines is tiny and unsuitable for observation, this paper introduces a detail enhancement post-processing based on visibility level and air light level to enhance the details of defogging images. We assess the efficacy of the proposed method by quantitatively and qualitatively assessing defogging images of power towers and transmission lines that were acquired using various methods. The results of the experiment proved that the defogging images restored by suggested algorithm have better detail level, structural similarity and color reproduction, and can effectively remove fog, which is superior to existing algorithms. In addition, the algorithm proposed in this paper can not only be used in power system online monitoring, but also can be extended to community monitoring, UAV monitoring, automatic driving, industrial production, Internet of Things and other fields, with broad application space. In the further work, we are going to do research on image defogging combined with dark image enhancement to expand the application range of the algorithm.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/h4nwei/MEF-SSIMd.

Author contributions

MZ and MG: conceptualization. ZS: methodology. JY: software, formal analysis, and writing—original draft preparation. JY and MG: validation and data curation. CY and ZJ: investigation. ZJ: resources. MG: writing—review and editing. YH: visualization and supervision. MZ and WC: project administration. MZ: funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding

The authors declare that this study received funding from the Science and Technology Project of State Grid Hubei Power Supply Limited Company Ezhou Power Supply Company (Grant Number: B315F0222836). The funder was not involved in the study design, collection, analysis, interpretation of data, and the writing of this article or the decision to submit it for publication.

Acknowledgments

The authors would like to express sincere thanks to colleague from transmission line operation and maintenance, State Grid Hubei Power Supply Limited Company Ezhou Power Supply Company. Thanks to Prof. Gwanggil Jeon (employed at Shandong University of Technology) for his guidance in this manuscript.

Conflict of interest

MZ, ZS, CY, ZJ, and WC were employed by the State Grid Hubei Power Supply Limited Company Ezhou Power Supply Company.

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

Publisher's note

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

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2012). SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* 34, 2274–2282. doi: 10.1109/TPAMI.2012.120

Berman, D., and Avidan, S. (2016). "Non-local image dehazing," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NEV. doi: 10.1109/CVPR.2016.185

Cai, B., Xu, X., Jia, K., Qing, C., and Tao, D. (2016). Dehazenet: An end-to-end system for single image haze removal. *IEEE Trans. Image Process.* 25, 5187–5198. doi: 10.1109/TIP.2016.2598681

Ehsan, S. M., Imran, M., Ullah, A., and Elbasi, E. A. (2021). Single image dehazing technique using the dual transmission maps strategy and gradient-domain guided image filtering. *IEEE Access* 9, 89055–89063. doi: 10.1109/ACCESS.2021.3090078

Fattal, R. (2008). Single image dehazing. *ACM Trans. Graph.* 27, 1–9. doi: 10.1145/1360612.1360671

Gao, Y., Hu, H. M., Li, B., Guo, Q., and Pu, S. (2018). Detail preserved single image dehazing algorithm based on airlight refinement. *IEEE Trans. Multimedia* 21, 351–362. doi: 10.1109/TMM.2018.2856095

Hautiere, N., Tarel, J. P., Aubert, D., and Dumont, E. (2008). Blind contrast enhancement assessment by gradient ratioing at visible edges. *Image Anal. Stereol.* 27, 87–95. doi: 10.5566/ias.v27.p87-95

He, K., and Sun, J. (2015). Fast guided filter. *arXiv* [Preprint]. arXiv:1505.00996.

He, K., Sun, J., and Tang, X. (2010). Single image haze removal using dark channel prior. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 2341–2353. doi: 10.1109/TPAMI.2010.168

He, K., Sun, J., and Tang, X. (2012). Guided image filtering. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 1397–1409. doi: 10.1109/TPAMI.2012.213

Jobson, D. J., Rahman, Z., and Woodell, G. A. (1997b). Properties and performance of a center/surround retinex. *IEEE Trans. Image Process.* 6, 451–462. doi: 10.1109/83.557356

Jobson, D. J., Rahman, Z., and Woodell, G. A. (1997a). A multiscale retinex for bridging the gap between color images and the human observation of scenes. *IEEE Trans. Image Process.* 6, 965–976. doi: 10.1109/83.597272

Jun, W. L., and Rong, Z. (2013). Image defogging algorithm of single color image based on wavelet transform and histogram equalization. *Appl. Math. Sci.* 7, 3913–3921. doi: 10.12988/ams.2013.34206

Kim, J. H., Jang, W. D., Sim, J. Y., and Kim, C. S. (2013). Optimized contrast enhancement for real-time image and video dehazing. *J. Vis. Commun. Image Represent.* 24, 410–425. doi: 10.1016/j.jvcir.2013.02.004

Li, B., Peng, X., Wang, Z., Xu, J., and Feng, D. (2017). "Aod-net: All-in-one dehazing network," in *Proceedings of the IEEE international conference on computer vision*, Venice, 22–29. doi: 10.1109/ICCV.2017.511

Liu, X., Ma, Y., Shi, Z., and Chen, J. (2019). "Griddehazenet: Attention-based multi-scale network for image dehazing," in *Proceedings of the IEEE/CVF international conference on computer vision*, Seoul. doi: 10.1109/ICCV.2019.00741

McCartney, E. J. (1976). *Optics of the atmosphere: Scattering by molecules and particles*. New York, NY, 421.

Meng, G., Wang, Y., Duan, J., Xiang, S., and Pan, C. (2013). "Efficient image dehazing with boundary constraint and contextual regularization," in *Proceedings of the IEEE international conference on computer vision*, Sydney, NSW. doi: 10.1109/ICCV.2013.82

Miyazaki, D., Akiyama, D., Baba, M., Furukawa, R., Hiura, S., and Asada, N. (2013). "Polarization-based dehazing using two reference objects," in *Proceedings of the IEEE international conference on computer vision workshops*, Sydney, NSW. doi: 10.1109/ICCVW.2013.117

Narasimhan, S. G., and Nayar, S. K. (2001). "Removing weather effects from monochrome images," in *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition*, Kauai.

Pang, Y., Nie, J., Xie, J., Han, J., and Li, X. (2020). "BidNet: Binocular image dehazing without explicit disparity estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Seattle, DC. doi: 10.1109/CVPR42600.2020.00597

Qin, X., Wang, Z., Bai, Y., Xie, X., and Jia, H. (2020). "FFA-Net: Feature fusion attention network for single image dehazing," in *Proceedings of the AAAI conference on artificial intelligence*, New York, NY. doi: 10.1609/aaai.v34i07.6865

Raikwar, S. C., and Tapaswi, S. (2020). Lower bound on transmission using non-linear bounding function in single image dehazing. *IEEE Trans. Image Process.* 29, 4832–4847. doi: 10.1109/TIP.2020.2975909

Ren, W., Ma, L., Zhang, J., Pan, J., Cao, X., Liu, W., et al. (2018). "Gated fusion network for single image dehazing," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Salt Lake City. doi: 10.1109/CVPR.2018.00343

Sangeetha, N., and Anusudha, K. (2017). "Image defogging using enhancement techniques," in *Proceedings of 2017 international conference on computer, communication and signal processing (ICCCSP)*, Chennai. doi: 10.1109/ICCCSP.2017.7944087

Schechner, Y. Y., Narasimhan, S. G., and Nayar, S. K. (2001). "Instant dehazing of images using polarization," in *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition*, Kauai.

Seow, M. J., and Asari, V. K. (2006). Ratio rule and homomorphic filter for enhancement of digital colour image. *Neurocomputing* 69, 954–958. doi: 10.1016/j.neucom.2005.07.003

Sharma, N., Kumar, V., and Singla, S. K. (2021). Single image defogging using deep learning techniques: Past, present and future. *Arch. Comput. Methods Eng.* 28, 4449–4469. doi: 10.1007/s11831-021-09541-6

Shwartz, S., and Schechner, Y. Y. (2006). "Blind haze separation," in *Proceedings of 2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, New York, NY.

Su, C., Wang, W., Zhang, X., and Jin, L. (2020). Dehazing with offset correction and a weighted residual map. *Electronics* 9:1419. doi: 10.3390/electronics9091419

Tan, R. T. (2008). "Visibility in bad weather from a single image," in *Proceedings of 2008 IEEE conference on computer vision and pattern recognition*, Anchorage, AK. doi: 10.1109/CVPR.2008.4587643

Tarel, J. P., and Hautiere, N. (2009). "Fast visibility restoration from a single color or gray level image," in *Proceedings of 2009 IEEE 12th international conference on computer vision*, Kyoto. doi: 10.1109/ICCV.2009.5459251

Xu, Y., Wen, J., Fei, L., and Zhang, Z. (2015). Review of video and image defogging algorithms and related studies on image restoration and enhancement. *IEEE Access* 4, 165–188. doi: 10.1109/ACCESS.2015.2511558

Zhang, H., and Patel, V. M. (2018). "Densely connected pyramid dehazing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Salt Lake City, GA. doi: 10.1109/CVPR.2018.00337

Zhou, L., Bi, D. Y., and He, L. Y. (2016). Variational histogram equalization for single color image defogging. *Math. Probl. Eng.* 2016, 1–17. doi: 10.1155/2016/9897064

Zhu, M., He, B., Liu, J., and Zhang, L. (2019). Dark channel: The devil is in the details. *IEEE Signal Process. Lett.* 26, 981–985. doi: 10.1109/LSP.2019.2914559

frontiers | Frontiers in Neurorobotics

TYPE Original Research
PUBLISHED 10 January 2023
DOI 10.3389/fnbot.2022.1105385

Check for updates

OPEN ACCESS

EDITED BY
Xin Jin,
Yunnan University, China

REVIEWED BY
Fang Chunhua,
China Three Gorges University, China
Kangjian He,
Yunnan University, China

*CORRESPONDENCE
Yunzhu An
✉ anuyunzhu2006@163.com

RECEIVED 22 November 2022
ACCEPTED 13 December 2022
PUBLISHED 10 January 2023

CITATION
Chen S, Li Z, Shen D, An Y, Yang J, Lv B and Zhou G (2023) Multi-exposure electric power monitoring image fusion method without ghosting based on exposure fusion framework and color dissimilarity feature.
Front. Neurorobot. 16:1105385.
doi: 10.3389/fnbot.2022.1105385

Multi-exposure electric power monitoring image fusion method without ghosting based on exposure fusion framework and color dissimilarity feature

Sichao Chen[1], Zhenfei Li[2], Dilong Shen[1], Yunzhu An[2]*, Jian Yang[1], Bin Lv[1] and Guohua Zhou[1]

[1]Hangzhou Xinmei Complete Electric Appliance Manufacturing Co., Ltd., Hangzhou, China, [2]School of Electrical and Electronic Engineering, Shandong University of Technology, Zibo, China

To solve the ghosting artifacts problem in dynamic scene multi-scale exposure fusion, an improved multi-exposure fusion method has been proposed without ghosting based on the exposure fusion framework and the color dissimilarity feature of this study. This fusion method can be further applied to power system monitoring and unmanned aerial vehicle monitoring. In this study, first, an improved exposure fusion framework based on the camera response model was applied to preprocess the input image sequence. Second, the initial weight map was estimated by multiplying four weight items. In removing the ghosting weight term, an improved color dissimilarity feature was used to detect the object motion features in dynamic scenes. Finally, the improved pyramid model as adopted to retain detailed information about the poor exposure areas. Experimental results indicated that the proposed method improves the performance of images in terms of sharpness, detail processing, and ghosting artifacts removal and is superior to the five existing multi-exposure image fusion (MEF) methods in quality evaluation.

KEYWORDS

ghosting artifacts, electric power monitoring, camera response model, color dissimilarity feature, pyramid, multi-exposure image fusion

1. Introduction

Since the objects are constantly in motion, compared with most natural scenes, the dynamic range of the existing ordinary cameras is very narrow (Akçay et al., 2017). Therefore, the captured image cannot have all the details in the high dynamic range (HDR) scene at disposable. Dynamic range refers to the ratio between the brightness in the brightest and darkest areas of the images. To address the issue of low dynamic range (LDR) images, we used HDR imaging technology to merge LDR images of different scenes captured into HDR images (Debevec and Malik, 2008).

At present, there are two methods for HDR imaging, namely, MEF and tone mapping. The tone mapping method requires the camera response function (CRF) for correction in the HDR imaging process and also uses the tone mapping operator to convert HDR images to LDR images that can be shown on traditional LDR devices. The MEF method directly fuses images taken at different exposure levels in the same scene to generate HDR images with rich information. It makes up for the shortcomings of the tone mapping method. Exposure evaluation, CRF correction, and tone mapping operation are not required during HDR imaging. Therefore, it saves computation costs and is widely used in high-dynamic-range imaging.

In recent years, many MEF methods have been successfully developed. According to whether the objects in the input image sequence are moving or not, they were divided into two methods, namely, the static scene MEF method (Mertens et al., 2007; Heo et al., 2010; Gu et al., 2012; Zhang and Cham, 2012; Shen et al., 2014; Ma and Wang, 2015; Nejati et al., 2017; Huang et al., 2018; Lee et al., 2018; Ma et al., 2018; Wang et al., 2019; Ulucan et al., 2021; Wu et al., 2021; Hu et al., 2022) and the dynamic scene MEF method (Li and Kang, 2012; Qin et al., 2014; Liu and Wang, 2015; Vanmali et al., 2015; Fu et al., 2016; Ma et al., 2017; Zhang et al., 2017; Hayat and Imran, 2019; Li et al., 2020; Qi et al., 2020; Jiang et al., 2022; Luo et al., 2022; Yin et al., 2022). Mertens et al. (2007) proposed a technique for fusing exposure sequences into high-quality images using multi-scale resolution. It can generate natural-color images, but the edge texture details of the fusion image are largely lost. Zhang and Cham (2012) proposed a method to process static and dynamic exposure compositions using image gradient information. This method can reduce the tedious tone mapping steps but cannot deal with the ghosts caused by the movement of objects and cameras. Gu et al. (2012) proposed a MEF method using the Euclidean metric to measure intensity distance in gradient domain feature space. It can produce fused images with rich information. Shen et al. (2014) proposed an advanced exposure fusion method. The method integrates local, global, and saliency weights into the weight processing problem. Ma and Wang (2015) proposed a patch decomposition MEF method to save running time. It improves the color appearance of the fusion image based on the decomposition of the image patches into three components, namely, average intensity, signal structure, and signal strength. Later, Ma et al. combined structural similarity with patch structure. Ma et al. (2018) proposed a MEF method to increase the perceptual quality by optimizing the color structure similarity index (MEF-SSIMc). Nejati et al. (2017) first disaggregated the source input image into basic and detail levels. Second, the exposure function is adopted to handle the weight problem. Although this method improves computational efficiency, it cannot remove the ghosts of dynamic scenes. Lee et al. (2018) designed an advanced weight function. Its function is to increase the weights of the bright regions in underexposure images and the dark regions in overexposure images while suppressing the oversaturation of these regions. Huang et al. (2018) proposed the color multi-exposure image fusion method to enhance the detailed information of fusion images. The method is based on decomposing the images into three weights, including intensity adjustment, structure preservation, and contrast extraction, and fusing them separately, preserving a great deal of detailed information for the input images. Wang et al. (2019) proposed a multi-exposure image fusion method in YUV color space. Simple detail components are used to strengthen the fused image details, which can retain the brightest and darkest area details in the HDR scene. A few pieces of literature (Ulucan et al., 2021; Wu et al., 2021; Hu et al., 2022) describe the recent results of the MEF method. Ulucan et al. (2021) designed a MEF technology to obtain accurate weights of fused images. The weight map is constructed by watershed masking and linear embedding weights. Then, the weight map and the input image are fused. This method can produce fusion images with lots of details and a good color appearance. Wu et al. (2021) presented a MEF method based on the improved exposure evaluation and the dual-pyramid model. The method can be applied in the computer vision field and the medical, remote sensing, and electrical fields. Hu et al. (2022) proposed a MEF method for detail enhancement based on homomorphic filtering. In terms of weight map calculation, threshold segmentation and Gaussian curves are utilized for processing. In terms of detail enhancement, the pyramid model of homomorphic filtering is used for processing weight maps and input image sequences.

In the dynamic scene MEF process, there is an object motion phenomenon in the input image sequence. Therefore, we should consider removing ghosting caused by object motion. Heo et al. (2010) proposed a high-dynamic-range imaging (HDRI) algorithm using a global intensity transfer function to remove ghosting artifacts. Li and Kang (2012) proposed a MEF method to remove ghosting utilizing histogram equalization and color dissimilarity feature using median filtering. Qin et al. (2014) used a random walk algorithm to maintain the content of the moving objects and provide more details. Therefore, this method can process dynamic scenes and reduce the ghosting artifacts of fused images. To increase the color brightness of the fused image, Vanmali et al. (2015) proposed a weight-forced MEF method without ghosting. Mertens et al. (2007) presented an algorithm to obtain the weighted map and used the weight-forced technology to force the weight of newly detected objects to zero. Therefore, it can produce ghost-free images with good color and texture details. Li and Kang (2012) presented a multi-exposure image fusion method based on DSIFT deghosting. It was adopted to extract the local contrast of the source image and remove the ghosting artifacts in the dynamic scene using the dense SIFT descriptor. To enhance the quality of ghost-free fusion images, Ma et al. (2017) proposed a MEF method (SPD-MEF) based on structural patch decomposition. It uses the direction of signal structure in the patch vector space to detect motion consistency, which removes ghosts. Zhang et al. (2017) introduced the inter-consistency of pixel intensity similarity in input image sequences and the intra-consistency of the interrelationships between adjacent pixels. To reduce the cost of motion estimation and accelerate MEF efficiency, Hayat and Imran (2019) presented a MEF method (MEF-DSIFT) based on dense SIFT descriptors and guided filtering. The method calculates the color dissimilarity feature using histogram equalization and median filtering, which removes the ghosting phenomenon in the MEF of dynamic scenes. Recently, Qi et al. (2020) proposed a MEF method based on feature patches. This method removes ghosts in dynamic scenes by prior exposure quality and structural consistency checking, which improves the performance of ghost removal. Li et al. (2020) proposed a fast multi-scale SPD-MEF method. It can decrease halos in static scenes and ghosting in dynamic scenes.

The available MEF methods are mainly suitable for static scene fusion, but they lack robustness to dynamic scenes, which causes a poor ghost removal effect. Therefore, this study adopts the multi-exposure image fusion method of weighted term deghosting. Based on the Ying method, an improved exposure fusion framework based on the camera response model is proposed to process input image sequences. Based on the Hayat method, an improved color dissimilarity feature is proposed for dynamic scenes, which is used to remove ghosting artifacts caused by object motion. In this study, the proposed method can generate images without ghosting fusion with pleasing naturalness and sharp texture details. Overall, the main advantages of the proposed method are summarized as follows:

(1) This study proposes an improved exposure fusion framework based on the camera response model. For the first time, the input image sequences processed by the fusion framework are used as multi-exposure input source image sequences. Through the fusion framework processing, the brightness and contrast of the source image are enhanced, and vast details are retained.

(2) The initial weight map is designed. It is obtained by calculating four weight terms, namely, local contrast, exposure feature, brightness feature, and improved color dissimilarity feature, of the input image and multiplying the four weight terms together. For dynamic scenes, an improved color dissimilarity feature is proposed based on a hybrid median filter and histogram equalization, which strengthens the sharpness of the image and has a better deghosting effect.

(3) Weighted guided image filtering (WGIF) is utilized to refine the initial weight map. The improved multi-scale pyramid decomposition model is used to add the Laplacian pyramid information to the highest level of the weighted mapping pyramid to weaken halo artifacts and retain details.

The rest of the study is organized as follows: Section 2 describes in detail the proposed multi-scale fusion deghosting method. In section 3, the effectiveness of the proposed method is obtained by analyzing the experiment results. Finally, section 4 concludes this study and makes prospects for the future.

2. Multi-scale image fusion ghosting removal

2.1. Improved exposure fusion framework based on the camera response model

There are overexposure/underexposure areas in the input image sequence. The input image sequence used for direct multi-scale image fusion may affect the contrast and sharpness of the fused images. Therefore, we transform the brightness of all images in the exposure sequence and carry out a weighted fusion of images before and after brightness transform to enhance image contrast, as in Equation (1).

$$\begin{cases} I_i(x,y) = \mathrm{M}(x,y) \circ P_i^c(x,y)+(1\text{-}\mathrm{M}(x,y)) \circ P_i^{c'}(x,y) \\ P_i^{c'}(x,y)=g(P^c,k_i) = \beta P^{\gamma} = \mathrm{e}^{b(1-k^a)}P^{k^a}(x,y) \end{cases} \tag{1}$$

where g is the brightness transfer function, which uses the β-γ correction model. $P_i(x,y)$, $I = 1, 2, 3 ...$; N is the input image; $P_i^{'}(x,y)$ is the image of $P_i(x,y)$ brightness change in the exposure sequence; and k_i is the exposure rate of the i-th image. M is the weight map of the input image of $P_i(x,y)$; "$\circ$" indicates the dot product operator; c is the index of three-color channels; $a = -0.3293$ and $b = 1.1258$ are the parameters of the CRF; and $I_i(x,y)$ is the enhancement result.

For low-light images, image brightness $L_i(x,y)$ is obtained using the maximal value in the three color channels in Equation (2).

$$L_i(x,y) = \max_{c \in \{R,G,B\}} P_i{}^c(x,y) \tag{2}$$

The illumination map T estimation algorithm has been extensively studied. This study adopts the morphological closure operation to calculate the initial illumination map T_i by Fu et al. (2016), as shown in Equation (3).

$$\mathrm{T}_i(x,y) = \frac{L_i(x,y) \cdot Q_i(x,y)}{255} \tag{3}$$

where $Q_i(x,y)$ denotes a structural element, and "$\cdot$" denotes an end operation. The range is mapped to [0,1] downstream operations by dividing by 255. Then, weighted guided image filtering (WGIF) (Li et al., 2014) is used to optimize the initial illumination map $\mathrm{T}_i(x,y)$, which can better remove the halo phenomenon than the existing guided image filter (GIF). The V level in the HSV color space for the input images is regarded as the guiding image in WGIF.

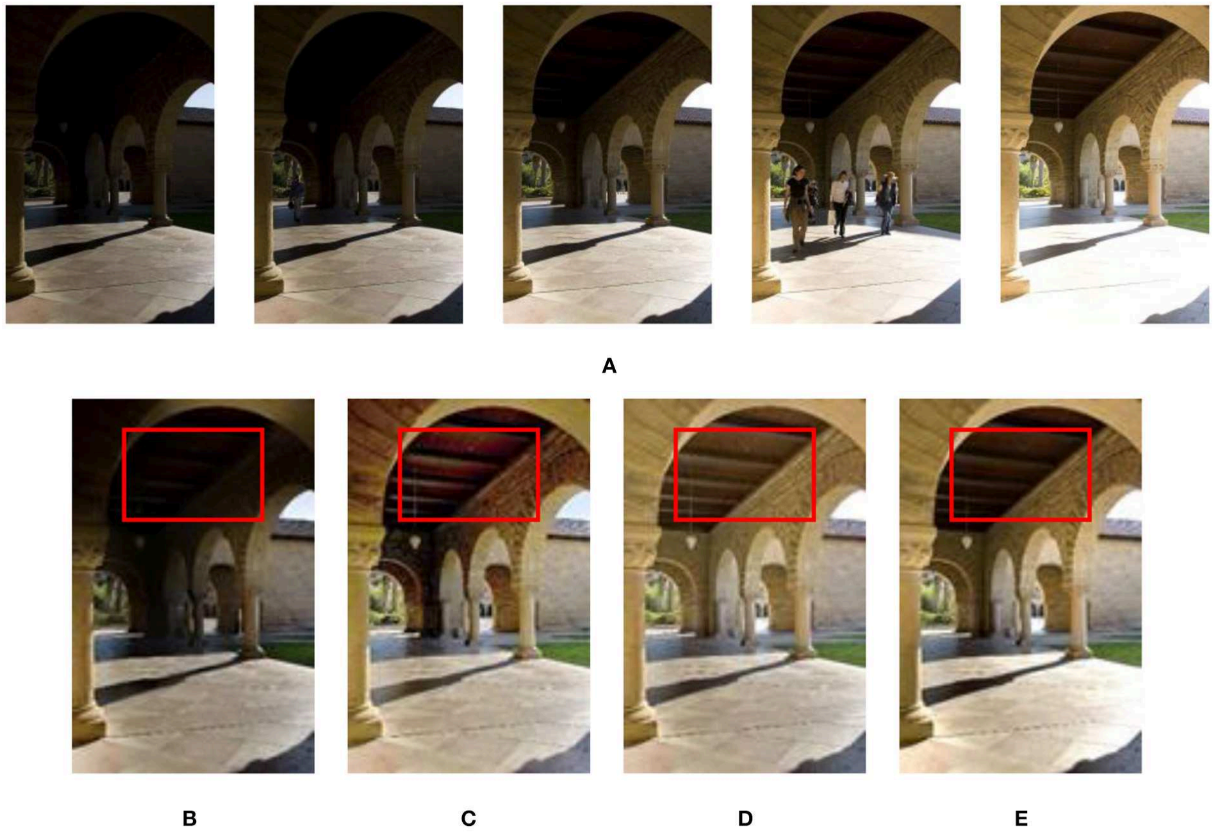


FIGURE 1
Results of dynamic scene "Arch" image sequence processed with/without CRF exposure fusion framework. **(A)** "Arch" image sequence; **(B)** Without CRF exposure fusion framework processing method (Hayat and Imran, 2019); **(C)** ICCV image processing method (Ying et al., 2017b); **(D)** CAIP image processing method (Ying et al., 2017a); **(E)** The image processing method proposed in this study.

It should be noted that the key point of image fusion enhancement is the design of the weight map M(*x*,*y*). The weight map M(*x*,*y*) is calculated using the method proposed by Ying et al. (2017a) in Equation (4).

$$M(x,y)=(T_i^{\mathrm{op}}(x,y))^{\theta} \tag{4}$$

where $\theta = 0.5$ is a parameter to control the enhanced intensity and $T_i^{op}(x,y)$ represents the optimized illumination map. Besides, we used the Ying et al. (2017a) exposure rate determination method to obtain the best exposure rate *k*. To obtain images with good sharpness, the non-linear unsharp masking algorithm (Ngo et al., 2020) proposed by Ngo et al. is used to increase the naturalness and sharpness of fused images.

Figure 1 shows the effect with/without CRF exposure fusion framework on experiment results. Figure 1B shows the results of the without CRF exposure fusion framework. Figures 1C–E are the result of the CRF exposure fusion framework. In Figure 1D, although the contrast of the image is improved, the image suffers from oversaturation distortion. The proposed fusion framework (see Figure 1E) significantly improves the brightness and sharpness of over/underexposure regions in the source input image sequences. Therefore, we used the proposed exposure fusion framework for related experiments in the following algorithm.

2.2. Multi-exposure image fusion without ghosting based on improved color dissimilarity feature and improved pyramid model

This section proposes an improved multi-exposure image fusion method without ghosting. The proposed method is mainly for motion scenes in multi-exposure images. Figure 2 shows the flow schematic drawing of the proposed method.

2.2.1. Improved color dissimilarity feature

An improved color dissimilarity feature based on fast multi-exposure image fusion with a median filter and recursive filter is

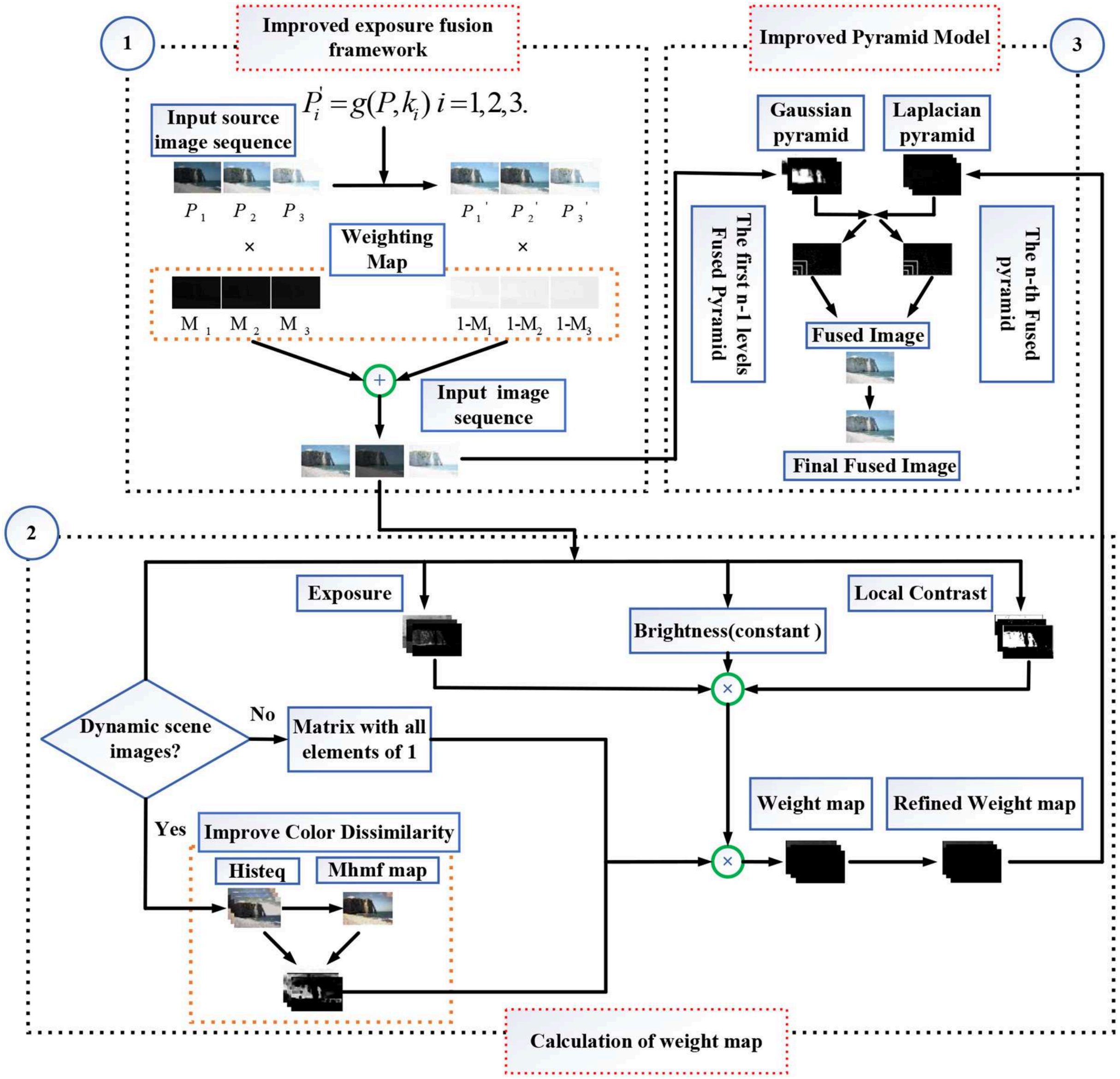


FIGURE 2
Schematic diagram of the proposed method.

proposed by Li and Kang (2012). Unlike the method proposed by Li and Kang (2012), static background images I^S of the scene are processed by a hybrid median filter (mHMF) (Kim et al., 2018) as in Equation (5).

$$I^S = mhmf(I^{HE}_{\min}(x,y)) \tag{5}$$

where I^S represents the static background of the scene and *mhmf* $(\cdot)$ denotes an operator. The hybrid median filter (mHMF) (Kim et al., 2018) was applied to the worst image $I^{HE}_{\min}(x,y)$ in the histogram equalized exposure sequence $I^{HE}_i(x,y)$, which is more beneficial to preserving the image edges in regions such as mutation than the median filter. Besides, the color dissimilarity feature $D_i(x,y)$ of moving objects is calculated between the static background image I^S and histogram equalized image $I^{HE}_i(x,y)$ in Li and Kang (2012) and Hayat and Imran (2019).

Comparisons of the color dissimilarity feature by Li and Kang (2012) and the proposed method have been conducted, as shown in Figure 3. The fused image in Figure 3B generated by the method of Li and Kang (2012) has ghosting artifacts at the ellipsoid. The proposed algorithm is validated by adopting underexposure, exposure normal, and overexposure source images, as shown in Figures 3C–E. According to Figure 3E, it can be seen that the results generated by underexposure images have a good effect on deghosting and are better

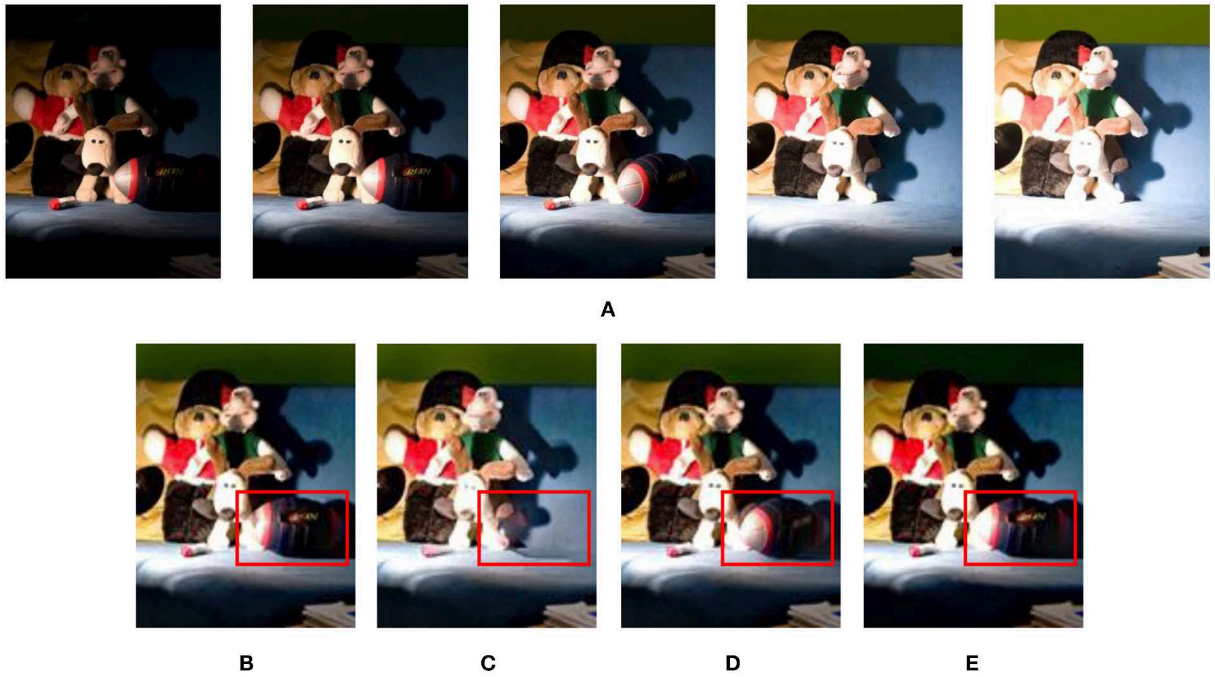


FIGURE 3
The results of processing the dynamic scene of the "Puppets" image sequence using the original/improved color dissimilarity features. **(A)** "Puppets" image sequence; **(B)** using the original color dissimilarity feature (Li and Kang, 2012); **(C)** hybrid median filter processing the brightest exposure image; **(D)** hybrid median filter processing the good exposure image; **(E)** hybrid median filter processing the darkest exposure image.

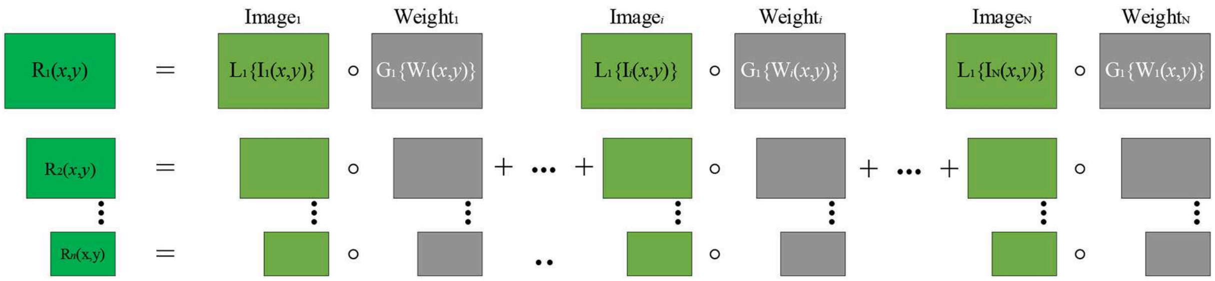


FIGURE 4
General flow of multi-scale exposure fusion. $I_i(x,y)$ is an LDR image. $W_i(x,y)$ is a weighted mapping. The Laplacian pyramid is obtained by LDR image decomposition, and the weighted mapping decomposition obtains the Gaussian pyramid. $R_1(x,y)$–$R_n(x,y)$ is the resulting level of the Laplacian pyramid.

than in Figure 3B. Therefore, in the following algorithm, we utilized the mHMF to handle underexposure images for related experiments.

2.2.2. Exposure feature and brightness feature

Because of the correlation between the three channels in *RGB* color space, which affects the final multi-scale pyramid decomposition and fusion, the input source image is converted from *RGB* to *YUV* color space. The exposure feature weight item $E_i(x,y)$ of the input image is measured in the *Y* channel as in Equation (6).

$$E_i(x,y) = e^{-\frac{[Y_i(x,y)-(1-\overline{Y_i})]^2}{2\sigma^2}} \tag{6}$$

where $Y_i(x,y)$ is the standardized value of the Y channel, Y_i is the mean value of $Y_i(x,y)$, and σ is a Gaussian kernel parameter taken as $\sigma = 0.2$. Besides, to increase the SNR of the input image sequence and retain the detailed information of the brightest/darkest regions, this method uses the brightness quality metric $B_i = Y_i^{2}$ in Kou et al. (2018).

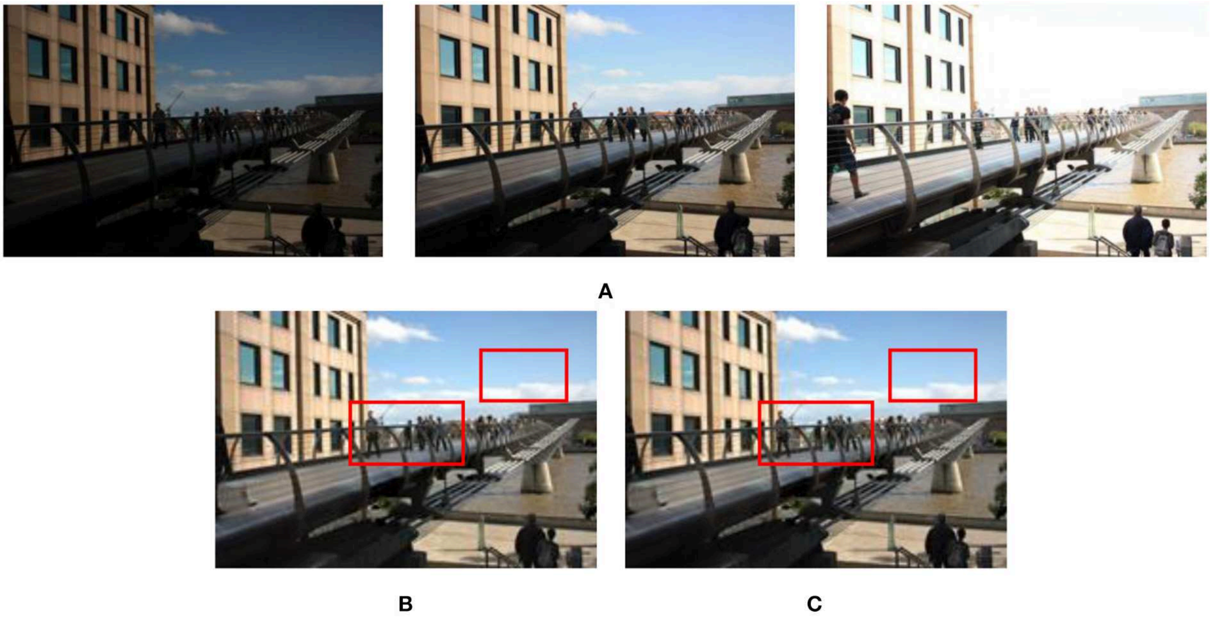


FIGURE 5
Experimental results of processing a dynamic scene "Tate" image sequence using the original/improved pyramid model. **(A)** "Tate" image sequence; **(B)** using the original pyramid model (Mertens et al., 2007); **(C)** using the improved pyramid model.

2.2.3. Local contrast using dense SIFT descriptor

The local contrast is measured using Equation (7), which is extracted by non-standardized dense filtering in dense SIFT descriptor (Liu et al., 2010).

$$C_i(x, y) = \left\| DSIFT(I_i^{\text{gray}}(x, y)) \right\|_1 \tag{7}$$

where $DSIFT(.)$ represents the operator that computes the non-normalized dense SIFT source image mapping, $C_i(x,y)$ represents a simple indicator vector for local contrast measurement, and $I_i^{gary}(x, y)$ denotes the grayscale image corresponding to the input image sequence $I_i(x,y)$. At each pixel, the $I_i^{gary}(x, y)$ mapping is regarded as the l_1 norm of $C_i(x,y)$. Besides, this study selects a winner-take-all weight allocation strategy (Liu and Wang, 2015; Hayat and Imran, 2019) to obtain the final local contrast weight term $C_i^{final}(x, y)$.

2.2.4. Estimation and refinement of the weight map

First, the following four weight items of the input image sequence are calculated: color dissimilarity feature, exposure feature, brightness feature, and local contrast. Second, weight items are multiplied to generate a weighted mapping, as in Equation (8).

$$\begin{cases} W_i(x, y) = C_i^{final}(x, y) \times B_i \times E_i(x, y), \text{ for static scene} \\ W_i(x, y) = C_i^{final}(x, y) \times B_i \times E_i(x, y) \times D_i(x, y), \\ \qquad\qquad \text{for dynamic scene} \end{cases} \tag{8}$$

Using WGIF (Li et al., 2014) directly refines and filters the weight map obtained by Equation (8), which is different from the refinement of the weight map in Liu and Wang (2015) and Hayat and Imran (2019). In the process of filter refinement, both the source image and the guide image are used $W_i(x, y)$. Then, normalizing refined weight maps makes weight maps sum to 1 at every pixel. The final weight map is shown in Equation (9).

$$\overline{W}_i(x, y) = \left[\sum_{i=1}^{N} \hat{W}_i^{\text{WF}}(x, y) + \varepsilon \right]^{-1} (\hat{W}_i^{\text{WF}}(x, y) + \varepsilon) \tag{9}$$

where $\hat{W}_i^{WF}(x, y)$ denotes the weight map after WGIF refinement, $W_i(x, y)$ denotes the final normalized weight map, and $\varepsilon = 10^{-5}$ is a small positive value, avoiding a zero denominator in the calculation process.

2.2.5. Improved pyramid decomposition fusion model

Utilizing the original multi-scale pyramid model (Mertens et al., 2007) may produce fusion images with a loss of details

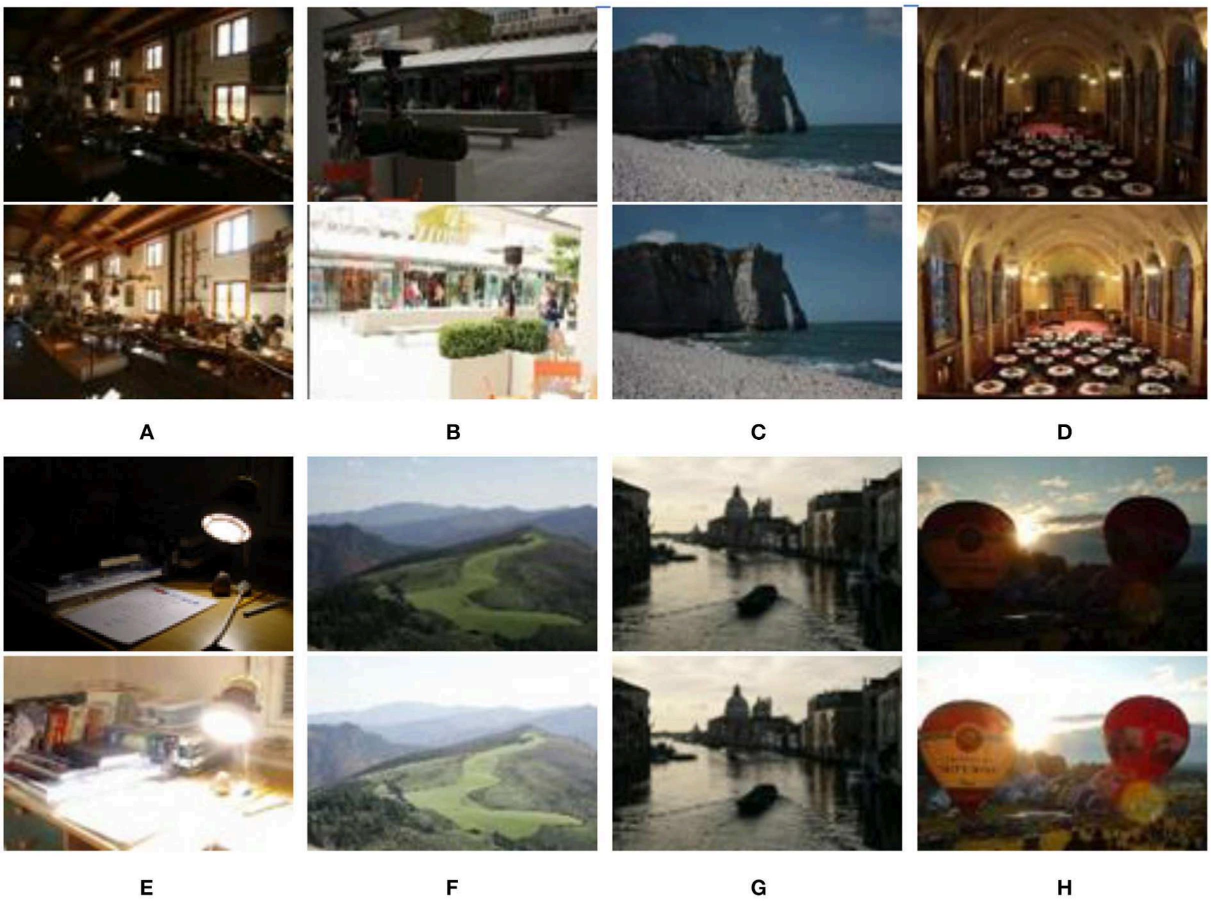


FIGURE 6
Source image sequences used in experiments. **(A)** Farmhouse; **(B)** Brunswick; **(C)** Cliff; **(D)** Llandudno; **(E)** Cadik; **(F)** Landscape; **(G)** Venice; **(H)** Balloons.

and the halo phenomenon. Therefore, an improved pyramid fusion model is used. In this pyramid model, the Laplacian and Gaussian pyramids are disaggregated into n levels, as shown in Figure 4. The total number of levels n is defined by Equation (10).

$$n = [log_2(min(r_o, c_o))] - 2 \qquad (10)$$

where r_o and c_o are the number of rows and columns of input image pixels, respectively.

It is considered that, at the highest level of the Gaussian pyramid, improper smoothing of edges is the main reason for producing halos. On the lower levels of the Gaussian pyramid, the improper smoothing of the edges is not evident for the generation of halos. Therefore, on the n-th level of the RGB color space pyramid, using the single-scale fusion algorithm in Ancuti et al. (2016) adds the Laplacian pyramid information of the source image to the Gaussian pyramid weighted mapping as in Equation (11).

$$R_n^i = [G_{\overline{n}}\{\overline{W}_i(x,y)\} + \lambda \left|L_1\{I_i(x,y)\}\right|]I_i(x,y) \qquad (11)$$

where I_i is the input image of LDR, R_n^i is the result of fusing the i-th image and the i-th image weight on the n-th level, and $G_n\left\{W_i(x,y)\right\}$ is the n-th Gaussian pyramid of $W_i(x,y)$. In Ancuti et al. (2016), n is the maximum number of levels of the Gaussian pyramid, $L_1\left\{I_i(x,y)\right\}$ is the first level of the input image $I_i(x,y)$ Laplacian pyramid, and λ is the coefficient of $L_1\left\{I_i(x,y)\right\}$, which controls the amplitude of the high-frequency signal $L_1\left\{I_i(x,y)\right\}$.

To retain detailed information on overexposed/underexposed areas, on the n-th level, the improved multi-scale exposure fusion algorithm proposed by Wang et al. (2019) is used as in Equation (12).

$$R_n^i(x,y) = [G_{\overline{n}}\{G_n\{\overline{W}_i(x,y)\}\} + \lambda \left|L_1\{L_n\{I_i(x,y)\}\}\right|]L_n\{I_i(x,y)\} \qquad (12)$$

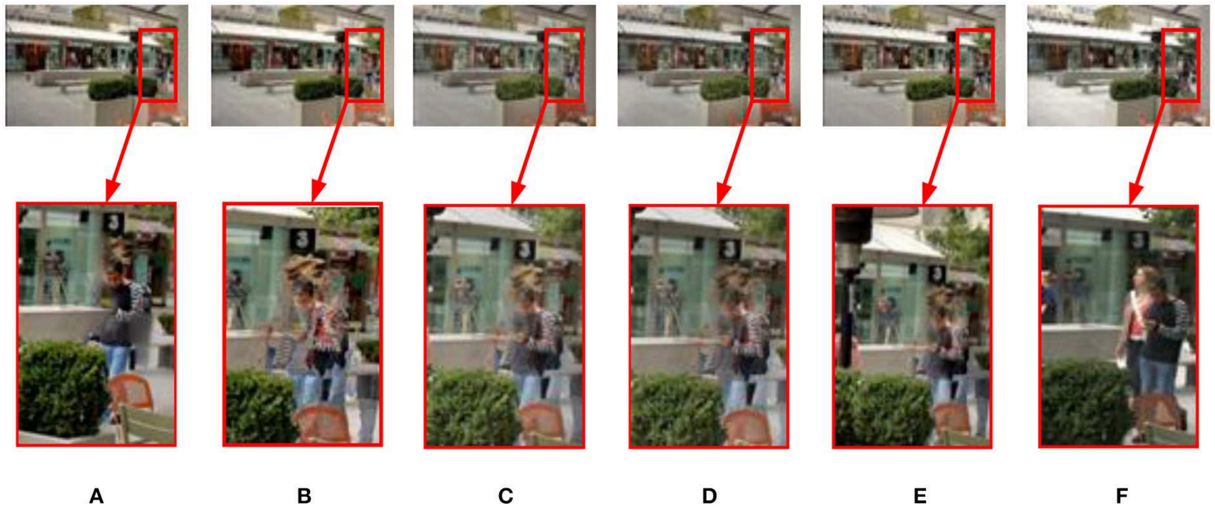


FIGURE 7
Comparison results of different methods on the dynamic "Brunswick" image sequence. **(A)** Hayat and Imran (2019); **(B)** Mertens et al. (2007); **(C)** Li and Kang (2012); **(D)** Liu and Wang (2015); **(E)** Lee et al. (2018); **(F)** the proposed method in this study.

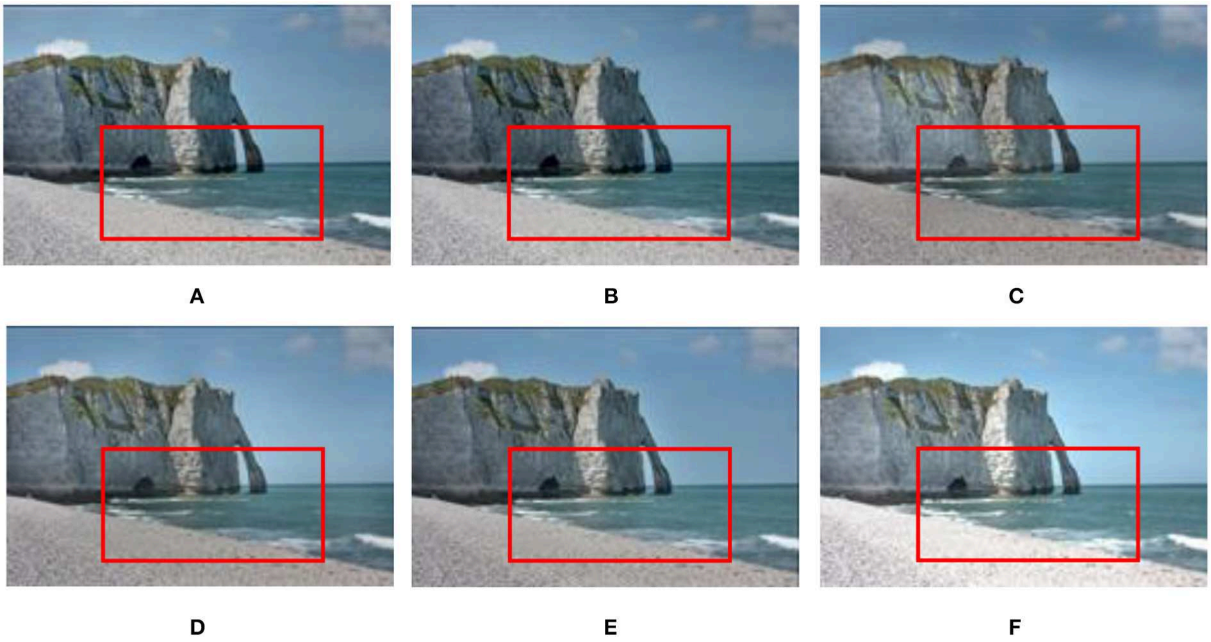


FIGURE 8
On dynamic "Cliff" image sequence, the available MEF methods compare with the proposed method. **(A)** Hayat and Imran (2019); **(B)** Mertens et al. (2007); **(C)** Li and Kang (2012); **(D)** Liu and Wang (2015); **(E)** Lee et al. (2018); **(F)** the proposed method in this study.

For underexposure source images, $\left|L_1\left\{L_n\left\{I_i(x,y)\right\}\right\}\right|$ in Equation (12) is introduced at the n-th level to correct the incorrect weights introduced by the weighted mapping smoothed by the Gaussian smoothing filter. It also reasonably enhances the weight of the well-exposure areas in the underexposure image, which retains the details of the underexposure areas. For overexposure images, the weight map of the n-th level adopts the primary Gaussian smoothing filter to smooth, which retains the details of the overexposure area.

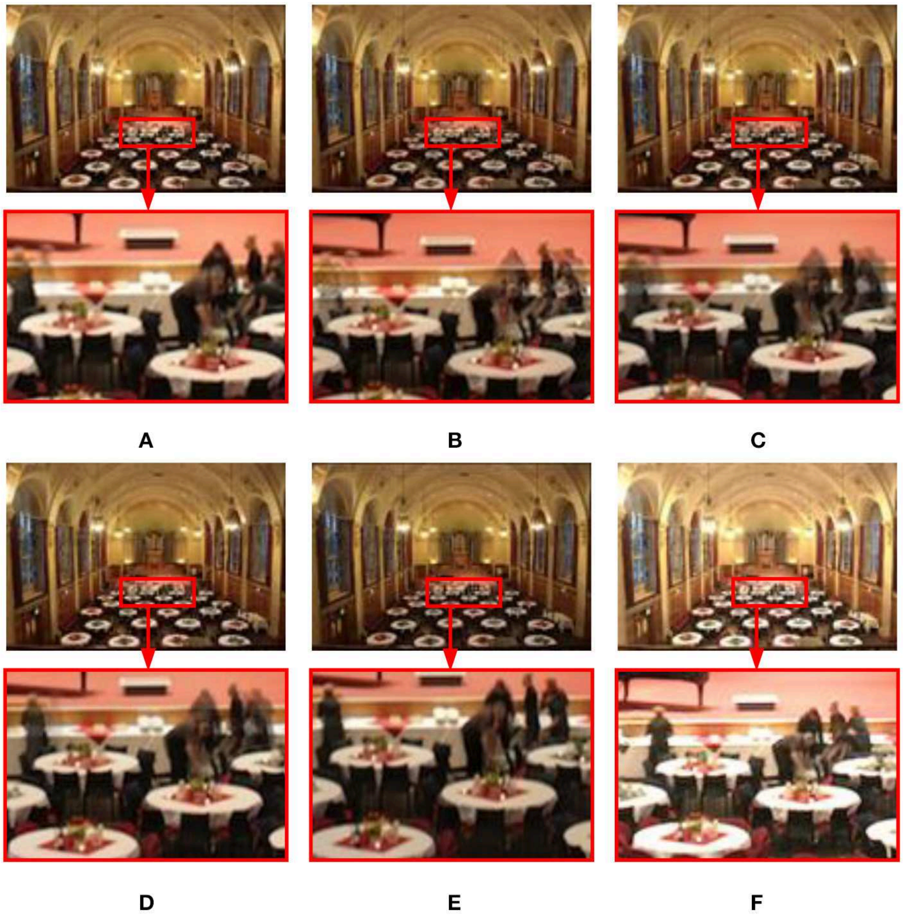


FIGURE 9
Fusion results of different methods on the dynamic "Llandudno" image sequence. **(A)** Hayat and Imran (2019); **(B)** Mertens et al. (2007); **(C)** Li and Kang (2012); **(D)** Liu and Wang (2015); **(E)** Lee et al. (2018); **(F)** the proposed method in this study.

For other scales, the improved pyramid fusion is the same as the original pyramid fusion (Mertens et al., 2007). Finally, reconstructing the Laplacian pyramid composed of $R_l(x,y)$ in Equation (13) generates the fused image R.

$$R_l(x,y) = \sum_{i=1}^{N} R_l^i(x,y), l = 1, 2, \ldots, n \tag{13}$$

where l represents the level number of the pyramid. The image details and brightness enhancement method proposed by Li et al. (2017) is adopted to enhance fusion image detail information, which obtains the final multi-scale exposure fusion image.

Comparisons of the original and improved pyramid models have been conducted, as shown in Figure 5. Compared with the original pyramid model (see Figure 5B), the generated image in Figure 5C by the improved pyramid model performs well in contrast and detail processing aspects, especially in pedestrian and white cloud areas. It is considered that multi-scale pyramid decomposition and fusion, loss of details, and the halo phenomenon are complex problems in pyramid decomposition and fusion. Therefore, this study selects the improved pyramid model to decompose and fuse the input image.

3. Experimental analysis

3.1. Experimental setup

In our experiments, five and six image groups were selected from seventeen static scene (Kede, 2018) and twenty dynamic scene (DeghostingIQADatabase, 2019) image groups, respectively. As shown in Figure 6, two images with different brightnesses are extracted from the above input image sequences. We utilized eleven image groups to test five existing MEF methods and the proposed method. The five MEF methods were presented by Mertens et al. (2007), Li and Kang (2012), Liu and Wang (2015), Lee et al. (2018), and Hayat

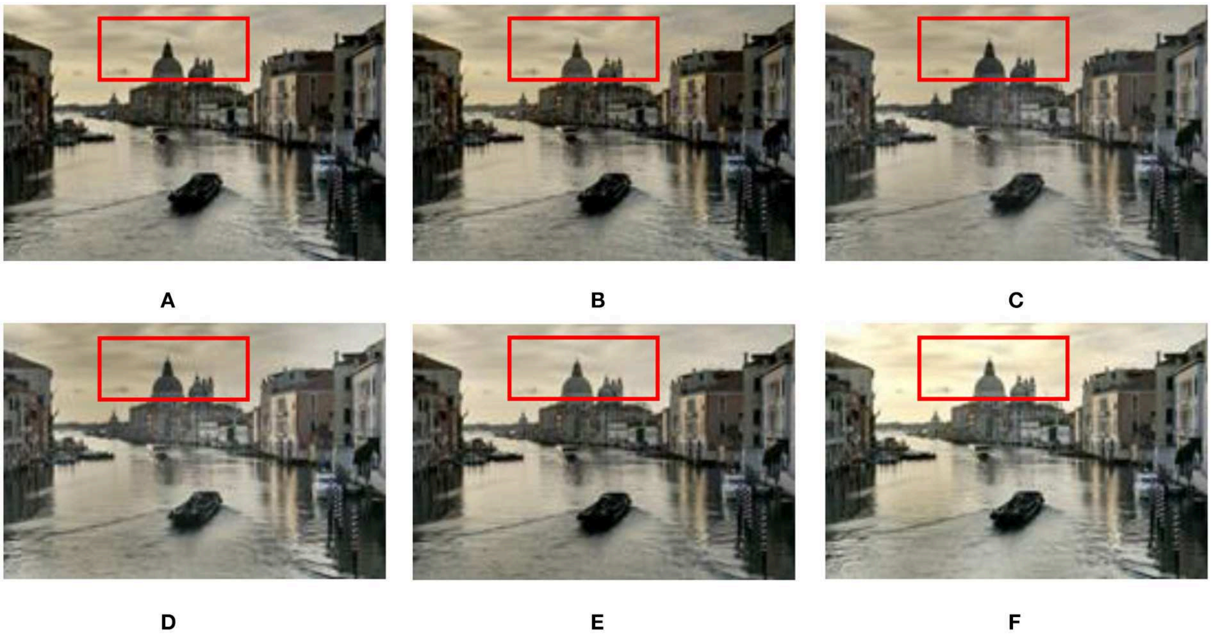


FIGURE 10
Comparison of the proposed method with Mertens et al. (2007), Li and Kang (2012), Hayat and Imran (2019), Liu and Wang (2015), and Lee et al. (2018) in the static "Venice" image sequence. **(A)** Hayat and Imran (2019); **(B)** Mertens et al. (2007); **(C)** Li and Kang (2012); **(D)** Liu and Wang (2015); **(E)** Lee et al. (2018); **(F)** the proposed method in this study.

and Imran (2019), respectively. All experiments are run on MATLAB 2019a [Intel Xeon X5675 3.07 GHz desktop with 32.00 GB RAM].

3.2. Subjective evaluation

In this section, to thoroughly discuss the content of the experimental results, we performed a local amplification close-up shot of the results of most sequence images.

3.2.1. Dynamic scenes

Figure 7 shows the experimental results of different methods in the dynamic Brunswick sequence. In terms of ghost removal, the methods (see Figures 7B–E) presented by Mertens et al. (2007), Li and Kang (2012), Liu and Wang (2015), and Lee et al. (2018) have poor effects and cannot effectively remove ghosts in pedestrian areas. The pixel oversaturation distortion in Figure 7A significantly reduces the visual quality. The proposed method can produce a good result (see Figure 7F). No ghosting artifact phenomenon exists in the image, and human visual perception is natural.

Figure 8 shows the fusion results of different methods in the dynamic Cliff sequence. The images in Figures 8A, B generated by the methods of Mertens et al. (2007) and Hayat and Imran (2019) are dark in color, the local contrast is not apparent, and the ghosting phenomenon exists in the water waves, which reduces the visual observation effect to a certain extent. Although the methods (see Figures 8C–E) of Li and Kang (2012), Liu and Wang (2015), and Lee et al. (2018) increase the contrast of the image, there are still darker colors and ghost phenomena. Figure 8F is the method proposed in this study. In contrast, the ghost removal performance significantly improved. On the waves and beaches, detailed information, local contrast, and naturalness are maintained, consistent with human visual observation.

Figure 9 shows the performance comparison of different methods in the dynamic Llandudno sequence. The results (see Figures 9B–D) acquired by Mertens et al. (2007), Li and Kang (2012), and Liu and Wang (2015) show that there are apparent ghosting artifacts in the area of characters and that there is a loss of detail information and color distortion. In Figure 9A, the overall image deghosting effect is good, but the color above the house is dark. The image in Figure 9E is unclear, and there is a color distortion phenomenon. The proposed method can produce a good result (see Figure 9F). The characters in the image have no noticeable ghosting artifacts, details are well

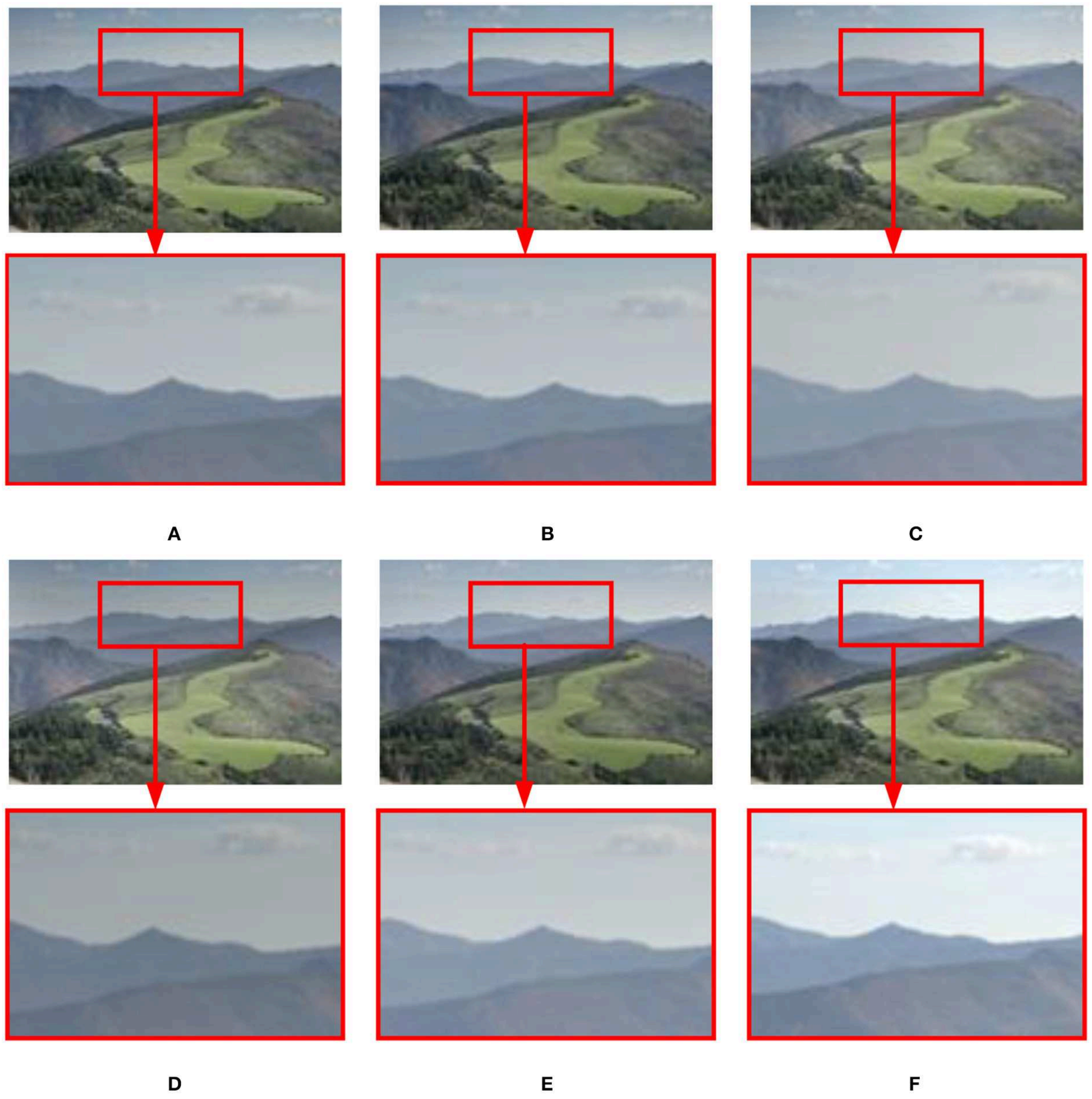


FIGURE 11
Comparison results of different methods on the static "Landscape" image sequence. **(A)** Hayat and Imran (2019); **(B)** Mertens et al. (2007); **(C)** Li and Kang (2012); **(D)** Liu and Wang (2015); **(E)** Lee et al. (2018); **(F)** the proposed method in this study.

preserved, and the exposure level is consistent with human visual observation.

3.2.2. Static scenes

Experimental results on the static Venice sequence using different methods are shown in Figure 10. In terms of image sharpness and detail processing, the proposed method (see Figure 10F) is superior to the methods (see Figures 10A–E) proposed by Mertens et al. (2007), Li and Kang (2012), Liu and Wang (2015), Lee et al. (2018), and Hayat and Imran (2019). Especially in Figures 10B–D, in the sky and church areas of the image, exposure and sharpness are poor, local contrast is not apparent, and fused image details are lost. In the results of the

TABLE 1 MEF-SSIMd of six MEF methods.

Dataset	Hayat	Mertens	Li	Liu	Lee	Proposed
Arch	**0.9503**	0.8423	0.9464	0.9417	0.8711	0.9267
Brunswick	0.8592	0.8834	0.8586	0.8261	0.8378	**0.9270**
Cliff	0.8873	0.9401	0.9243	0.9006	0.9035	**0.9687**
Llandudno	0.9072	0.8483	0.8926	0.8746	**0.9771**	0.9260
Puppets	0.8357	0.7791	0.8085	0.8035	0.8481	**0.8900**
Tate	0.8306	0.8076	0.8044	0.8298	0.8258	**0.9123**
Cadik	0.9247	0.9268	0.9032	**0.9474**	0.9290	0.9004
Landscape	0.9936	**0.9941**	0.9924	0.9883	0.9935	**0.9941**
Venice	0.8141	0.8612	0.8654	0.8250	0.8631	**0.9170**
Balloons	**0.9756**	0.9597	0.9429	0.9205	0.9539	0.9651
Farmhouse	0.9824	0.9824	0.9824	**0.9872**	0.9791	0.9588
Average	0.9055	0.8932	0.9019	0.8950	0.9075	**0.9351**
Rank	3	6	4	5	2	**1**
Total	9.9607	9.825	9.9211	9.8447	9.982	**10.2861**

The bold value indicates the maximum value, and the larger the value, the better the image fusion.

TABLE 2 NIQE comparison results of the MEF method.

Dataset	Hayat	Mertens	Li	Liu	Lee	Proposed
Arch	2.4484	2.6802	2.5456	2.6763	2.4354	2.2921
Brunswick	2.7688	**2.4740**	3.0447	3.0847	2.9462	2.8631
Cliff	3.4004	3.5940	3.4480	3.5294	3.5791	**2.9777**
Llandudno	3.1018	3.9349	3.3978	3.4781	3.9544	**2.8940**
Puppets	3.0152	**2.9780**	3.2068	3.2526	3.0909	3.2955
Tate	3.0066	**2.6109**	2.9594	2.9954	2.7553	2.8802
Cadik	3.5912	3.5379	3.7545	3.7202	3.5309	**3.4117**
Landscape	2.7917	2.8495	2.8220	2.7845	2.8128	**2.7497**
Venice	3.4862	3.8663	3.3296	3.3251	3.4182	**3.2924**
Balloons	3.3137	3.2691	3.5863	3.4309	3.4333	**3.0047**
Farmhouse	2.9762	2.9537	3.017	2.9744	2.9261	**2.7657**
Average	3.0818	3.159	3.1920	3.2047	3.1744	**2.9479**
Rank	2	3	5	6	4	**1**
Total	33.9002	34.7485	35.1117	35.2516	34.8826	32.4268

The bold value indicates the minimum value, the smaller the NIQE value is, the better the image quality is, and the image more accords with the requirements of the visible human system to observe the scene.

method proposed by Lee et al. (2018) and Hayat and Imran (2019), the sharpness of the fused image has improved, but there is still local contrast that is not obvious, and details are lost (see Figures 10A, E).

The fusion results of six MEF methods on static scene landscape sequences are shown in Figure 11. In Figures 11B–E, in the sky area (white cloud parts), the sharpness is not good enough. In the method (see Figure 11A) proposed by Hayat and Imran (2019), although the sharpness and naturalness of the image are enhanced in the sky area, the fused image details are seriously lost. Compared with the methods (see Figures 11A–E) presented by Mertens et al. (2007), Li and Kang (2012), Liu and Wang (2015), Lee et al. (2018), and Hayat and Imran (2019), the proposed method in this study (see Figure 11F) has good saturation and contrast in the sky area, and the detailed information is retained better.

TABLE 3 Test results of LPC-SI.

Dataset	Hayat	Mertens	Li	Liu	Lee	Proposed
Arch	0.9767	0.9710	0.9758	**0.9774**	0.9728	0.9770
Brunswick	0.9691	0.9620	0.9699	0.9703	0.9678	**0.9785**
Cliff	0.9671	0.9619	0.9637	0.9653	0.9643	**0.9777**
Llandudno	0.9737	0.9724	0.9737	0.9736	0.9734	**0.9767**
Puppets	0.9785	0.9731	0.9782	0.9763	0.9759	**0.9821**
Tate	0.9739	0.9686	0.9736	0.9723	0.9702	**0.9795**
Cadik	0.9691	0.9626	0.9655	0.9650	0.9687	**0.9700**
Landscape	0.9516	0.9484	0.9516	**0.9522**	0.9477	0.9512
Venice	0.9692	0.9633	0.9675	0.9659	0.9537	**0.9709**
Balloons	**0.9701**	0.9689	0.9696	0.9681	0.9690	0.9700
Farmhouse	0.9729	0.9728	0.9752	0.9760	0.9754	**0.9780**
Average	0.9711	0.9659	0.9695	0.9693	0.9672	**0.9738**
Rank	2	6	3	4	5	**1**
Total	10.6819	10.625	10.6643	10.6624	10.6389	**10.7116**

The bold value indicates the maximum value, a more considerable LPC-SI value of the fused image represents a clearer image, which conforms to the evaluation of human visual observation.

3.3. Objective evaluation

3.3.1. Evaluation using dynamic scene structural similarity index (MEF-SSIMd)

The structural similarity index (MEF-SSIMd) (Fang et al., 2019) is applied to measure structural similarity between input image sequences and fused images in dynamic ranges. The overall MEF-SSIMd is defined in Equation (14).

$$q_{overall} = \frac{q_s + q_d}{2} \tag{14}$$

where q_d represents MEF-SSIMd of dynamic scenes and q_s represents MEF-SSIMd of static scenes.

The data range of MEF-SSIMd is [0,1]. The greater the value, the better the deghosting efficiency, and the stronger the robustness of the dynamic scene. The smaller the value is, the opposite is true. As shown in Table 1, using MEF-SSIMd objectively evaluates six MEF methods for the quality of generating fused images. Overall, the proposed method is superior to the other five existing MEF methods in the performance evaluation of MEF-SSIMd.

3.3.2. Evaluation using natural image quality evaluator (NIQE)

In multi-exposure image fusion, the fused image should meet the requirements of the human visual system to observe the scene. Since the general purpose does not reference the IQA (image quality assessment), the algorithm requires much training to meet the IQA. Thus, a non-reference quality metric, NIQE (Mittal et al., 2012) was proposed. The smaller the NIQE value is, the better the image quality is, and the image more closely accords with the requirements of the visible human system to observe the scene. On the contrary, the greater the NIQE value is, the fewer images conform requirements of the human visual system observation scene. As shown in Table 2, NIQE is used to evaluate the quality of fusion images produced by different MEF methods. Overall, the proposed method can acquire images with better naturalness.

3.3.3. Evaluation of image sharpness using local phase coherence (LPC)

In multi-exposure image fusion, sharpness is a critical factor in the visual evaluation of image quality. The sharpness of the image to achieve the human visual system can effortlessly detect blur and observe visual images. Therefore, Hassen et al. (2013) used sharpness in the complex wavelet transform domain to evaluate the local solid phase coherence (LPC) of the image features. Then, the overall sharpness index of LPC (LPC-SI) is proposed. A more considerable LPC-SI value of the fused image represents a clearer image, which conforms to the evaluation of human visual observation. A smaller LPC-SI value of the fused image represents a blurred image. The value range of LPC-SI is [0,100]. Table 3 shows the comparison results of LPC-SI values between the other five MEF methods and the presented method. A comprehensive comparison shows that the proposed method in this study outperforms the other five existing MEF methods.

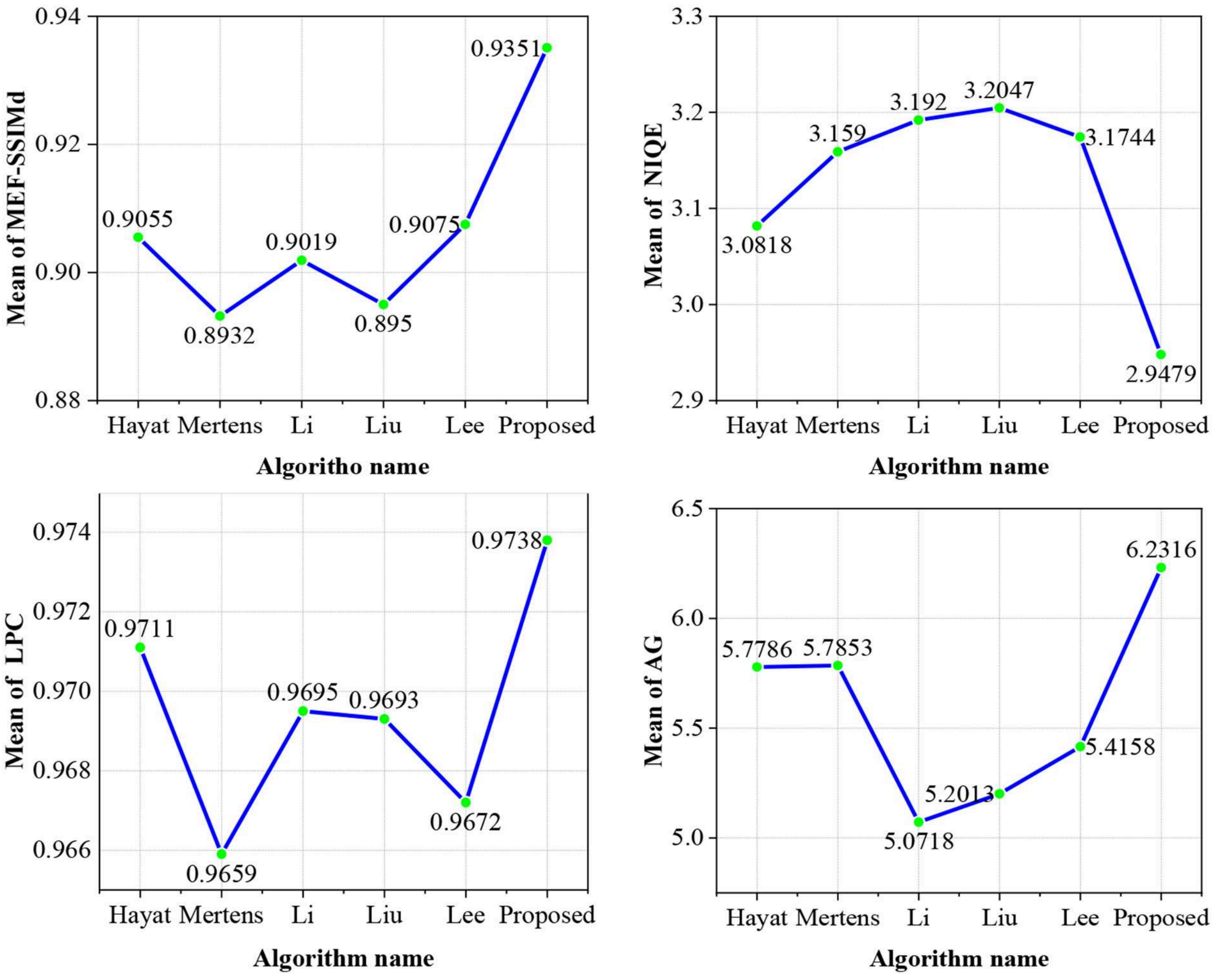


FIGURE 12
The mean values of MEF-SSIMd, NIQE, LPC-SI, and AG are obtained by different methods.

3.3.4. Mean value analysis of objective evaluation indexes

As shown in Figure 12, the proposed method in this study ranks first in the line graph of the mean values of the entire reference objective evaluation index MEF-SSIMd and non-reference objective evaluation index NIQE, LPC, and average gradient (AG). The proposed MEF method without ghosting based on the exposure fusion framework and color dissimilarity feature can effectively remove ghosting in dynamic scene MEF. It also improves the sharpness and naturalness of the fused image and retains many details.

4. Conclusion

An improved MEF method has been proposed in this study without ghosting based on the exposure fusion framework and color dissimilarity feature. It generates ghost-free, high-quality images with good sharpness and rich details. The proposed algorithm in this study can be further applied to power system monitoring and unmanned aerial vehicle monitoring fields. An improved exposure fusion framework based on the camera response model has been utilized to improve the contrast and sharpness of over/underexposure regions in the input image sequence. The WGIF refined weight map with an improved color dissimilarity feature was adopted to remove ghosting artifacts and to retain more image details utilizing an improved pyramid model. In the experimental tests of qualitative and quantitative evaluation for eleven image groups, including five static scene image groups and six dynamic scene image groups, this method ranks first compared with the five available MEF methods. However, when objects move frequently or move more widely, the fusion results may produce ghosting artifacts. Therefore, we hope that the researchers further study to overcome the above problems.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/h4nwei/MEF-SSIMd.

Author contributions

SC and ZL: conceptualization, methodology, software, and validation. DS: data curation. ZL: writing and original draft preparation. YA: writing, review, and editing. JY, BL, and SC: visualization. GZ: funding acquisition. All authors agreed to be accountable for the content of the study. All authors contributed to the article and approved the submitted version.

Funding

This research was funded by the National Natural Science Foundation of China (Grant Number: 51807113).

Acknowledgments

The authors thank the editors and the reviewers for their careful work and valuable suggestions for this study.

Conflict of interest

SC, DS, JY, BL, and GZ were employed by the company Hangzhou Xinmei Complete Electric Appliance Manufacturing 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

Akçay, Ö., Erenoglu, R. C., and Avşar, E. Ö. (2017). The effect of Jpeg compression in close range photogrammetry. *Int. J. Eng. Geosci.* 2, 35–40. doi: 10.26833/ijeg.287308

Ancuti, C. O., Ancuti, C., De Vleeschouwer, C., and Bovik, A. C. (2016). Single-scale fusion: an effective approach to merging images. *IEEE Trans. Image Process.* 26, 65–78. doi: 10.1109/TIP.2016.2621674

Debevec, P. E., and Malik, J. (2008). "Rendering high dynamic range radiance maps from photographs," in *Proceedings of the SIGGRAPH 1997: 24th Annual Conference on Computer Graphics and Interactive Techniques* (Los Angeles, CA: SIGGRAPH 1997), 369–378.

DeghostingIQADatabase (2019). Available online at: https://github.com/h4nwei/MEF-SSIMd (accessed March 5, 2022).

Fang, Y., Zhu, H., Ma, K., Wang, Z., and Li, S. (2019). Perceptual evaluation for multi-exposure image fusion of dynamic scenes. *IEEE Trans. Image Process.* 29, 1127–1138. doi: 10.1109/TIP.2019.2940678

Fu, X., Zeng, D., Huang, Y., Liao, Y., Ding, X., and Paisley, J. (2016). A fusion-based enhancing method for weakly illuminated images. *Signal Process.* 129, 82–96. doi: 10.1016/j.sigpro.2016.05.031

Gu, B., Li, W., Wong, J., Zhu, M., and Wang, M. (2012). Gradient field multi-exposure images fusion for high dynamic range image visualization. *J. Vis. Commun. Image Represent.* 23, 604–610. doi: 10.1016/j.jvcir.2012.02.009

Hassen, R., Wang, Z., and Salama, M. M. (2013). Image sharpness assessment based on local phase coherence. *IEEE Trans. Image Process.* 22, 2798–2810. doi: 10.1109/TIP.2013.2251643

Hayat, N., and Imran, M. (2019). Ghost-free multi exposure image fusion technique using dense SIFT descriptor and guided filter. *J. Vis. Commun. Image Represent.* 62, 295–308. doi: 10.1016/j.jvcir.2019.06.002

Heo, Y. S., Lee, K. M., Lee, S. U., Moon, Y., and Cha, J. (2010). "Ghost-free high dynamic range imaging," in *Proceedings of the 10th Asian Conference on Computer Vision* (Queenstown: Springer, Berlin, Heidelberg), 486–500.

Hu, Y., Xu, C., Li, Z., Lei, F., Feng, B., Chu, L., et al. (2022). Detail enhancement multi-exposure image fusion based on homomorphic filtering. *Electronics* 11, 1211. doi: 10.3390/electronics11081211

Huang, F., Zhou, D., Nie, R., and Yu, C. (2018). A color multi-exposure image fusion approach using structural patch decomposition. *IEEE Access* 6, 42877–42885. doi: 10.1109/ACCESS.2018.2859355

Jiang, Q., Lee, S., Zeng, X., Jin, X., Hou, J., Zhou, W., et al. (2022). A multi-focus image fusion scheme based on similarity measure of transformed isosceles triangles between intuitionistic fuzzy sets. *IEEE Trans. Instrument. Meas.* 71, 1–15. doi: 10.1109/TIM.2022.3169571

Kede, M. (2018). Available online at: https://ece.uwaterloo.ca/$\sim$k29ma/ (accessed March 15, 2022).

Kim, G. J., Lee, S., and Kang, B. (2018). Single image haze removal using hazy particle maps. *IEICE Trans. Fund. Electr.* 101, 1999–2002. doi: 10.1587/transfun.E101.A.1999

Kou, F., Li, Z., Wen, C., and Chen, W. (2018). Edge-preserving smoothing pyramid based multi-scale exposure fusion. *J. Vis. Commun. Image Represent.* 53, 235–244. doi: 10.1016/j.jvcir.2018.03.020

Lee, S. H., Park, J. S., and Cho, N. I. (2018). "A multi-exposure image fusion based on the adaptive weights reflecting the relative pixel intensity and global gradient," in *Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP)* (Athens: IEEE), 1737–1741.

Li, H., Ma, K., Yong, H., and Zhang, L. (2020). Fast multi-scale structural patch decomposition for multi-exposure image fusion. *IEEE Trans. Image Process.* 29, 5805–5816. doi: 10.1109/TIP.2020.2987133

Li, S., and Kang, X. (2012). Fast multi-exposure image fusion with median filter and recursive filter. *IEEE Trans. Consum. Electron.* 58, 626–632. doi: 10.1109/TCE.2012.6227469

Li, Z., Wei, Z., Wen, C., and Zheng, J. (2017). Detail-enhanced multi-scale exposure fusion. *IEEE Trans. Image Process*. 26, 1243–1252. doi: 10.1109/TIP.2017.2651366

Li, Z., Zheng, J., Zhu, Z., Yao, W., and Wu, S. (2014). Weighted guided image filtering. *IEEE Trans. Image Process*. 24, 120–129. doi: 10.1109/TIP.2014.2371234

Liu, Y., and Wang, Z. (2015). Dense SIFT for ghost-free multi-exposure fusion. *J. Vis. Commun. Image Represent*. 31, 208–224. doi: 10.1016/j.jvcir.2015.06.021

Liu, C., Yuen, J., and Torralba, A. (2010). Sift flow: dense correspondence across scenes and its applications. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 978–994. doi: 10.1109/TPAMI.2010.147

Luo, Y., He, K., Xu, D., Yin, W., and Liu, W. (2022). Infrared and visible image fusion based on visibility enhancement and hybrid multiscale decomposition. *Optik* 258, 168914. doi: 10.1016/j.ijleo.2022.168914

Ma, K., Duanmu, Z., Yeganeh, H., and Wang, Z. (2018). Multi-exposure image fusion by optimizing a structural similarity index. *IEEE Trans. Comput. Imaging* 4, 60–72. doi: 10.1109/TCI.2017.2786138

Ma, K., Li, H., Yong, H., Wang, Z., Meng, D., and Zhang, L. (2017). Robust multi-exposure image fusion: a structural patch decomposition approach. *IEEE Trans. Image Process*. 26, 2519–2532. doi: 10.1109/TIP.2017.671921

Ma, K., and Wang, Z. (2015). "Multi-exposure image fusion: A patch-wise approach," in *Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP)* (Quebec, QC: IEEE), 1717–1721.

Mertens, T., Kautz, J., and Van Reeth, F. (2007). "Exposure fusion," in *Proceedings of the 15th Pacific Conference on Computer Graphics and Applications (PG'07)* (Maui, HI: IEEE), 382–390.

Mittal, A., Soundararajan, R., and Bovik, A. C. (2012). Making a "completely blind" image quality analyzer. *IEEE Signal Process. Lett*. 20, 209–212. doi: 10.1109/LSP.2012.2227726

Nejati, M., Karimi, M., Soroushmehr, S. R., Karimi, N., Samavi, S., and Najarian, K. (2017). "Fast exposure fusion using exposedness function," in *Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP)* (Beijing: IEEE), 2234–2238.

Ngo, D., Lee, S., and Kang, B. (2020). "Nonlinear unsharp masking algorithm," in *Proceedings of the 2020 International Conference on Electronics, Information, and Communication (ICEIC)* (Barcelona: IEEE), 1–6.

Qi, G., Chang, L., Luo, Y., Chen, Y., Zhu, Z., and Wang, S. (2020). A precise multi-exposure image fusion method based on low-level features. *Sensors* 20, 1597. doi: 10.3390/s20061597

Qin, X., Shen, J., Mao, X., Li, X., and Jia, Y. (2014). Robust match fusion using optimization. *IEEE Trans. Cybern*. 45, 1549–1560. doi: 10.1109/TCYB.2014.2355140

Shen. J., Zhao, Y., Yan, S. and Li, X. (2014). Exposure fusion using boosting Laplacian pyramid. *IEEE Trans. Cybern.* 44, 1579–1590. doi: 10.1109/TCYB.2013.2290435

Ulucan, O., Karakaya, D., and Turkan, M. (2021). Multi-exposure image fusion based on linear embeddings and watershed masking. *Signal Process*. 178, 107791. doi: 10.1016/j.sigpro.2020.107791

Vanmali, A. V., Kelkar, S. G., and Gadre, V. M. (2015). "Multi-exposure image fusion for dynamic scenes without ghost effect," in *Proceedings of the 2015 Twenty First National Conference on Communications (NCC)* (Mumbai: IEEE), 1–6.

Wang, Q., Chen, W., Wu, X., and Li, Z. (2019). Detail-enhanced multi-scale exposure fusion in YUV color space. *IEEE Trans. Circ. Syst. Video Technol*. 30, 2418–2429. doi: 10.1109/TCSVT.2019.2919310

Wu, L., Hu, J., Yuan, C., and Shao, Z. (2021). Details-preserving multi-exposure image fusion based on dual-pyramid using improved exposure evaluation. *Results Opt*. 2, 100046. doi: 10.1016/j.rio.2020.100046

Yin, W., He, K., Xu, D., Luo, Y., and Gong, J. (2022). Significant target analysis and detail preserving based infrared and visible image fusion. *Infrared Phys. Technol.* 121, 104041. doi: 10.1016/j.infrared.2022.104041

Ying, Z., Li, G., Ren, Y., Wang, R., and Wang, W. (2017a). "A new image contrast enhancement algorithm using exposure fusion framework," in *Proceedings of the 17th International Conference on Computer Analysis of Images and Patterns (CAIP 2017)* (Ystad: Springer, Cham), 36–46.

Ying, Z., Li, G., Ren, Y., Wang, R., and Wang, W. (2017b). "A new low-light image enhancement algorithm using camera response model," in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)* (Venice: IEEE), 3015–3022.

Zhang, W., and Cham, W. K. (2012). Gradient-directed multiexposure composition. *IEEE Trans. Image Process.* 21, 2318–2323. doi: 10.1109/TIP.2011.2170079

Zhang, W., Hu, S., Liu, K., and Yao, J. (2017). Motion-free exposure fusion based on inter-consistency and intra-consistency. *Inf. Sci*. 376, 190–201. doi: 10.1016/j.ins.2016.10.020

Frontiers in Neurorobotics

TYPE Original Research
PUBLISHED 23 January 2023
DOI 10.3389/fnbot.2022.1042429

Check for updates

OPEN ACCESS

EDITED BY
Shin-Jye Lee,
National Chiao Tung University, Taiwan

REVIEWED BY
Hu Pei,
Shandong University of Science and Technology, China
Wenfeng Zheng,
University of Electronic Science and Technology of China, China

*CORRESPONDENCE
Fangxiu Jia
✉ jiafangxiu@126.com

RECEIVED 12 September 2022
ACCEPTED 26 December 2022
PUBLISHED 23 January 2023

CITATION
Tang W, Jia F and Wang X (2023) An improved adaptive triangular mesh-based image warping method. *Front. Neurorobot.* 16:1042429.
doi: 10.3389/fnbot.2022.1042429

An improved adaptive triangular mesh-based image warping method

Wei Tang, Fangxiu Jia* and Xiaoming Wang

College of Mechanical Engineering, Nanjing University of Science and Technology, Nanjing, China

It is of vital importance to stitch the two images into a panorama in many computer vision applications of motion detection and tracking and virtual reality, panoramic photography, and virtual tours. To preserve more local details and with few artifacts in panoramas, this article presents an improved mesh-based joint optimization image stitching model. Since the uniform vertices are usually used in mesh-based warps, we consider the matched feature points and uniform points as grid vertices to strengthen constraints on deformed vertices. Simultaneously, we define an improved energy function and add a color similarity term to perform the alignment. In addition to good alignment and minimal local distortion, a regularization parameter strategy of combining our method with an as-projective-as-possible (APAP) warp is introduced. Then, controlling the proportion of each part by calculating the distance between the vertex and the nearest matched feature point to the vertex. This ensures a more natural stitching effect in non-overlapping areas. A comprehensive evaluation shows that the proposed method achieves more accurate image stitching, with significantly reduced ghosting effects in the overlapping regions and more natural results in the other areas. The comparative experiments demonstrate that the proposed method outperforms the state-of-the-art image stitching warps and achieves higher precision panorama stitching and less distortion in the overlapping. The proposed algorithm illustrates great application potential in image stitching, which can achieve higher precision panoramic image stitching.

KEYWORDS

image stitching, mesh deformation, image alignment, color consistency, combining strategy

Introduction

Image stitching algorithm to mosaic two or more images into a panorama image to create a larger image with a wider field of view is the oldest and most widely used in computer vision (Szeliski, 2007; Nie et al., 2022; Ren et al., 2022). Earlier, the methods estimate a 2D transformation between two images focus on the global warps that include similarity, affine, and projective ones (Brown and Lowe, 2007; Chen and Chuang, 2016). Thus, the global warps are usually not flexible enough for all types of scenes like low-alignment quality images and parallax images. Furthermore, the holy grail of image stitching is to seamlessly blend overlapping images, even in scenes of distortion and parallax, to provide a panorama image that looks as natural as possible (Zaragoza et al., 2013).

While image stitching based on global warps (Zhu et al., 2001; Brown and Lowe, 2007; Kopf et al., 2007) can achieve good results, it still suffers from local distortion and is unnatural. The global warps estimate the global transformation, and they are robust but often not flexible enough. To address the model problem of global warps, many local warp models have been proposed, such as the dual-homography warping (DHW) (Gao et al., 2011), smoothly varying affine (SVA) (Lin et al., 2011) stitching, as-projective-as-possible (APAP), single-perspective warps (SPW), and so on. Unlike global warps, the above methods adopt multiple local parametric warps as the primary (Zaragoza et al., 2013; Liao and Li, 2019; Li et al., 2019; Guo et al., 2021), which is more flexible than the global warps. The DHW divides the image into two parts: a distant back plane and a ground plane, and it can seamlessly stitch most scenes. To achieve flexibility, Lin et al. (2011) proposed a smoothly varying affine stitching field that is defined over the entire coordinate frame, which is better for local deformation and alignment. Therefore, it is more tolerant of parallax than traditional global homography stitching. Instead of adopting an optimal global transformation, APAP estimates local space transformations to align every local image patch accurately.

Local parametric methods use spatially varying models to represent the motion of different image regions (Gao et al., 2011; Zaragoza et al., 2013; Chen et al., 2018). Compared to global methods, the higher degrees of freedom make them more flexible in handling motion in complex scenes but also make the model estimation more difficult (Chen et al., 2018; Liao and Li, 2019) proposed two single-perspectives warps for image stitching. The first parametric warp combines dual-feature-based APAP with quasi-homography. The second mesh-based warp is to achieve image stitching by optimizing a sparse and quadratic total energy function. Inspired by the Liu et al. (2009), many mesh-based warps (Li et al., 2015; Lin et al., 2016) have been proposed, which divide the source image into a uniform grid mesh.

In Liao and Li (2019), the stitching panorama looks as natural as possible when the source image has lots of lines; on the contrary, the stitching results represent noticeable ghosting in the curved areas and irregular object regions, such as the curve on the ground and the orange bag in the blue and red box in Figure 6. Meanwhile, Figure 6 illustrates the results of APAP which looks much better than global alignment, but visible ghosting still appears in some areas, such as the orange bag in the blue box picture.

To address the above problem with distortion and ghosting in the stitched images, we improved our method's meshing and combined our warps with APAP. In this study, we propose an improved mesh-based image stitching method. To optimize the quadrilateral grid cells, we introduce an innovative triangular mesh strategy. The mesh vertices include two parts: APAP and matched feature vertices. The APAP vertices belong to uniform vertices, which can preserve the flexibility of the APAP algorithm. Thus, the matched feature vertices, which are non-uniform, can make a few artifacts in overlapping regions. We then design a color constraint term in the energy function, and the global alignment term includes two transformations for the mesh vertices. The matched feature vertices can reduce ghosting in overlapping areas in the function term. Finally, to reduce distortion in non-overlapping areas, we combine our method with APAP warp and give the weight value by calculating the distance between the vertex and the nearest matched feature point to the vertex. The comparative experiments prove that the alignment accuracy of our method is higher than the APAP warp. In summary, our three contributions are as follows:

(1) We introduce an improved mesh deformation model, including two-part vertices: non-uniform and uniform vertices. Then, the cell in our method is changed from quads to triangles, which is a novel mesh different from the conventional ways. Thus, results show that our model makes few artifacts in overlapping regions.

(2) We also design a new deformation function, which includes the data term, global alignment term, and color smoothness term. Unlike other warps, the color smoothness term can constrain the overlapping regions' smoothness.

(3) We give a new strategy of combining our method with APAP warp to obtain its flexibility.

We compare our method with the state-of-the-art image stitching methods, and the comparison experiments illustrate that our method outperforms all other methods in preserving local details and with few artifacts in overlapping regions. This syudy is organized as follows. Section is the introduction. Section shows the related work of image stitching. Section introduces the proposed method for image stitching in detail. In Section , the results and comparison experiments with other algorithms were presented. Finally, Section shows the conclusion of this article.

Related work

Image stitching has been widely used in computer vision and many applications. This section will give a brief finding on image stitching.

Multi-homography method for image stitching

A single global homography matrix can be used to express the relationship between images when the scenes are approximately in the same plane. The actual scenes are often complex with multiple planes; thus, employing the global homography to align images in the overlapping region is usually not flexible enough to provide high-precision alignment. Gao et al. (2011) proposed a dual-homography warping, which divides the image into two parts: a distant back plane and a ground plane, and it can seamlessly stitch most scenes. The method can improve alignment accuracy, but for complex scenes with multiple planes, this method incorrectly divides the different planes into one structure, which will lead to alignment errors. Hence, Yan et al. (2017) proposed a robust multi-homography image composition method. By calculating different homographies from different types of features, multiple homographies are then blended with Gaussian weights to construct a panorama. When the scene is complex, and there are multiple planes, the method based on the simple multiple homographies is ineffective for alignment. Many methods (Chen and Chuang, 2016; Medeiros et al., 2016; Zheng et al., 2019) based on planar segmentation were provided to align images. Zheng et al. (2019) proposed a novel projective-consistent plane-based image stitching method. According to the normal vector direction of the local area and the reprojection error of the aligned image, the overlapping area of the input image is divided into several projection-uniform planes.

Image stitching based on mesh deformation

The main idea of image stitching based on mesh deformation (Liu et al., 2009; Zaragoza et al., 2013; Chen and Chuang, 2016; Chen et al., 2018; Liao and Li, 2019) is to mesh the image, transform the deformation of the image into the redrawing of the mesh, and then correspond the deformation of the mesh to the deformation of the image. This method enables the vast majority of matched feature point pairs to be completely aligned. Such methods realize image stitching by constructing an energy function for mesh vertices, and different results can be achieved by adding different constraints to the energy function. Liu et al. (2009) proposed a content-preserving warp (CPW) for video stabilization. This method divides the aligned image into multiple grid units and then constructs an energy function for the grid vertices consisting of data items, similar transformation items, and global alignment items and obtains the redrawn vertex coordinates by minimizing the energy function. The vertex coordinates of the grid where the feature points are located are optimized by the energy function, which can protect the shape of the important area of the image from being changed during the transformation. Zaragoza et al. (2013) proposed a moving direct linear transformation (Moving DLT) method to obtain the local homography matrix for each grid cell. The method added a weight value for each grid when calculating the local homography matrix. Liao and Li (2019) and Jia et al. (2021) proposed an image stitching method combining point features and line features and introduced global collinear structures into an energy function to specify and balance the desired characters for image stitching.

Seam-driven image stitching

When the image parallax is large, the image stitching method based on spatial transformation can no longer obtain accurate results. For such image stitching problems with large parallax, the more effective method is the image stitching approach based on stitching seam (Gao et al., 2013; Zhang and Liu, 2014; Lin et al., 2016; Chen et al., 2022). Gao et al. (2013) proposed an image stitching method based on seam driven, which obtains the final homography matrix based on the quality of the stitching seam. Zhang and Liu (2014) proposed a method for local alignment using CPW near stitching seam to achieve large parallax image stitching and combined homography transformation with content-preserving warp. The experiment results illustrated that their method could stitch images with large parallax well. A superpixel-based feature grouping method (Lin et al., 2016) was proposed to optimize the generation of initial alignment hypotheses. To avoid generating only potentially biased local homography hypotheses, the hypothesis set was enriched by combining different sets of superpixels to generate additional alignment hypotheses. Then, the method evaluated the alignment quality of the stitching seam to achieve the final panorama stitching. Chen et al. (2022) proposed a novel warping model based on multi-homography and structure preserving. The homographies at different depth regions were estimated by dividing matched feature pairs into multiple layers. Collinear structures were added to the objective function to preserve salient line structures. Thus, an optimal stitching seam search method based on stitching seam quality assessment was proposed.

Our approach

This section will give a detailed presentation of our image stitching approach. We first describe the traditional global homography model to pre-align the reference and the target image; a roughly global homography is obtained to help refine image stitching in the later sections. Then, we introduce the triangular mesh deformation and give the total energy function to get the coordinates of triangular mesh vertices after deformation. Finally, a regularization parameter is introduced to balance the global and local vertices after deformation; hence, the final result can be automatically adjusted by the input images. Major steps of our proposed scheme, as shown in Figure 1.

The similarity projective transformations

Given a pair of matching points $p = \left[x\, y\right]^T$ and $p' = \left[x'\, y'\right]^T$ across overlapping images I and I'. The homography model can be represented as follows

$$\widetilde{p}' = \mathbf{H}\tilde{p}, \tag{1}$$

Where $\widetilde{p}$ is p in homogeneous coordinates, $\widetilde{p} = \left[x\, y\, 1\right]^T$, and $\widetilde{p}' = \left[x'\, y'\, 1\right]^T$. $\mathbf{H} \in \mathbb{R}^{3\times3}$ denotes the homography matrix and $\mathbf{H} = \left[h_1\, h_2\, h_3\right]^T$. In inhomogeneous coordinates,

$$x' = \frac{h_1^T\left[\begin{array}{ccc} x & y & 1\end{array}\right]^T}{h_3^T\left[\begin{array}{ccc} x & y & 1\end{array}\right]^T} \text{ and } y' = \frac{h_2^T\left[\begin{array}{ccc} x & y & 1\end{array}\right]^T}{h_3^T\left[\begin{array}{ccc} x & y & 1\end{array}\right]^T}. \tag{2}$$

Taking a cross product on both sides of Equation (1), we can obtain the following:

$$0_{1\times3} = \begin{bmatrix} 0_{1\times3} & -\tilde{p}^T & y'\tilde{p}^T \\ \tilde{p}^T & 0_{1\times3} & -x'\tilde{p}^T \\ -y'\tilde{p}^T & x'\tilde{p}^T & 0_{1\times3} \end{bmatrix} \begin{bmatrix} \mathrm{h}_1 \\ \mathrm{h}_2 \\ \mathrm{h}_3 \end{bmatrix}. \tag{3}$$

There only two rows of the 3×9 matrix in Equat9ion (3) are linearly independent, and we let $a_i \in \mathbb{R}^{2\times9}$ be the first-two rows of Equation (3) computed for the i-th datum for a set of N matched points $\left\{p_i\right\}_{i=1}^N$ and $\left\{p_i'\right\}_{i=1}^N$, we can obtain h by the following

$$\hat{\mathrm{h}} = \arg\min_{\mathrm{h}} \sum \left\|a_i \mathrm{h}\right\|^2 = \arg\min_{\mathrm{h}} \|\mathbf{A}\mathrm{h}\|^2. \tag{4}$$

With the constraint $\left\|\mathrm{h}\right\| = 1$, where matrix $\mathbf{A} = [a_1\, a_2 \ldots\, a_i]^T$. Given the estimated $\mathbf{H}$ (reshaped from $\hat{\mathrm{h}}$), to align the images, the arbitrary pixel in the source image I is warped to the target image I' by Equation (1). Thus, the details can be found in Lin et al. (2015).

Triangular mesh deformation

The image stitching based on mesh deformation usually uses the quadrilateral grid, but the warp could still have less distortion at the position of the matched feature points. Therefore, we propose a triangular mesh cell, including APAP and matched feature vertices.

Mathematical setup

Inspire by the work of Li et al. (2019), they introduced the planar and spherical triangulation strategies and approximated the scene as a combination of adjacent triangular facets. This inspired us, so we partitioned the source image into a triangular mesh of a series of cells and took the matching points and APAP's vertices as our

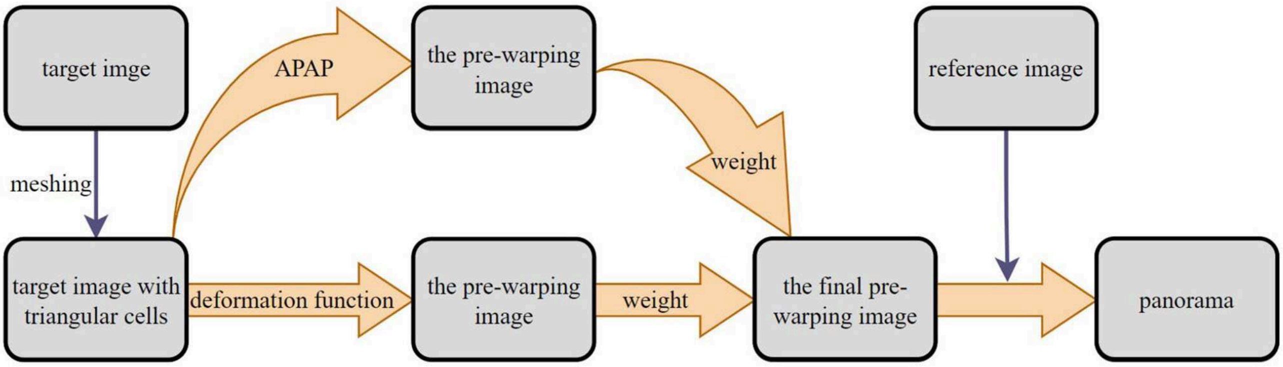


FIGURE 1
The schematic diagram of the proposed image stitching method.

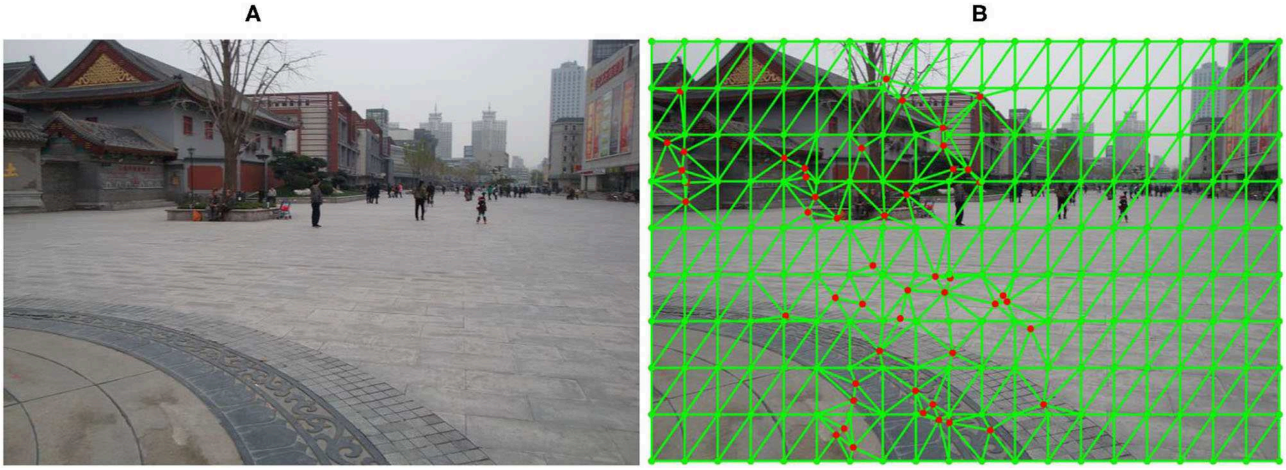


FIGURE 2
View triangulation results on the target image. **(A)** The template image and **(B)** the triangular mesh image. The green dots are APAP vertices, and the red dots denote matched feature vertices.

triangular mesh vertices. Then, a triangulation-based local alignment algorithm for image stitching is proposed, which could compensate for the weaknesses of the quadrilateral grid deformation.

For ease of explanation, we take the two image stitching pair as an example and let I', I, and $\hat{I}$ to denote the reference image, the target image, and the final warping image. We keep the reference image I' fixed and warp the target image I. Thus, the vertices in the image I, I', and $\hat{I}$ are denoted as V, V', and $\hat{V}$.

Unlike traditional quadrilateral grid deformation warps, we partition the source target image I into a series of triangular cells by Delaunay triangulation (Edelsbrunner et al., 1990). For each cell, three vertices are more stable than the four vertices in the quadrilateral cell. To make the image stitching warp more stable, we choose a series of APAP's vertices as the triangular cell vertices and add n-matched feature points as vertices into the original vertices. Therefore, the target image is partitioned into many cells, including two parts: APAP and matched feature vertices. Figure 2 illustrates a warp learned with 250 vertices cells for an image pair.

In addition, after building mesh grids for the target image I, where $V_{i,j}$ is the grid vertex at position (i,j). The target image is composed of many cells which have three vertices, and we index the grid vertices from 1 up to n; we reshape all vertices into a 2n-dimension vector $V = \left[x_1\, y_1\ \ldots\ x_n\, y_n\right]^T$; then, the mesh deformation vertices which correspond to the target image vertices are formed into $\hat{V} = \left[\hat{x}_1\, \hat{y}_1\ \cdots\ \hat{x}_n\, \hat{y}_n\right]^T$. Each cell has four vertices in Liao and Li (2019), so different from Liao and Li (2019), the mesh deformation cell has three vertices in our approach.

In Liao and Li (2019), each feature point p can be characterized as a bilinear interpolation of its four enclosing grid vertices. Thus, similar to Liao and Li (2019), for any feature point p in the triangular cell, which can be expressed as a linear interpolation of the triangular vertices v1, v2, and v3. Different from the bilinear interpolation, barycentric coordinate system (Koecher and Krieg, 2007) can denote any point which is inside the triangle cell well. So, the feature point p can be characterized as follows:

$$\varphi(p) = w_1 v_1 + w_2 v_2 + w_3 v_3, \tag{5}$$

Where w_1, w_2, and w_3 denote the weight of each vertex, respectively, the higher the weight, the closer the point is to the vertex, and $w_1 + w_2 + w_3 = 1$. If we get a known point inside the triangle, the weights will be obtained by solving a binary system of linear equations.

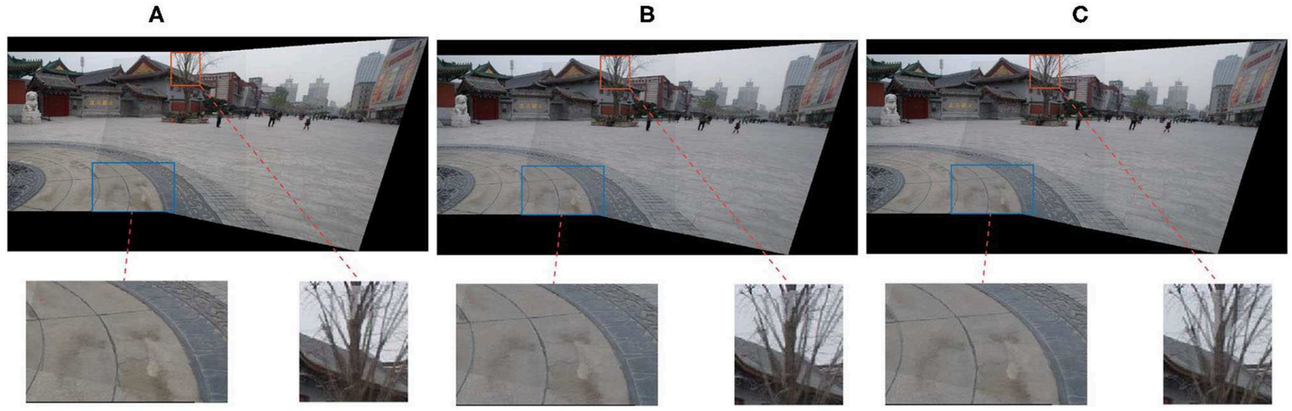


FIGURE 3
Comparison of stitching results with different ω_G. **(A)** $\omega_G = 0$, **(B)** $\omega_G = 10$, and **(C)** $\omega_G = 5000$.

Assuming that the weights are fixed, thus the corresponding point p' that is after mesh deformation can also be characterized as $\varphi(\hat{p}) = w_1\hat{v}_1 + w_2\hat{v}_2 + w_3\hat{v}_3$. Subsequently, any constraint on the point correspondences, which are inside the triangle can be expressed as a constraint on the three vertex correspondences.

Energy function definition

Inspired by the study of the content-preserving warps Liu et al. (2009), we construct the total energy function E that includes the following three parts: data term, global alignment term, and color smoothness term.

$$E\left(\hat{V}\right) = E_D\left(\hat{V}\right) + \omega_G E_G\left(\hat{V}\right) + E_C\left(\hat{V}\right), \tag{6}$$

Where E_D denotes the data term that addresses the alignment issue by enhancing the feature point correspondences, E_G is the global alignment term, and E_C addresses a color smoothness issue by protecting the vertices' intensity and its neighboring region. The deformed vertex $\hat{V}$ can be calculated by the above formula, then mapping the deformation of the mesh to the deformation of the image to obtain the final panorama. The above minimization problem is easily solved using a standard spares linear solver. We use texture mapping to extract the final image when we get the deformed vertices. The weight $\omega_G = 10$ in our implementation. Figure 3 shows the stitching results of different ω_G. Theoretically, the larger ω_G is, the better the alignment at the matched feature vertex positions of the stitching results; the blue box in Figure 3 verifies this point. Thus, ω_G is too large, which means the weight of the global alignment term is too large. As shown in the red box in Figure 3, too much weight of data items will affect the stitching effect of other regions.

A. Data term

The data term E_D is defined the same way as Liu et al. (2009). Thus, the feature point p which is in the mesh cell can be denoted by the triangular vertices of its enclosing grid cell. To align p to its matched location p' after deformation, we define the data term as follows:

$$E_D = \sum_i \left\| \sum_{i=1}^{3} w_{i,k}\hat{V}_{i,k} - p'_i \right\|^2 \tag{7}$$

Where $\hat{V}$ is the unknown coordinate of mesh vertices to be estimated, $\omega_{i,k}$ is the interpolation coefficient, which is obtained by the mesh cell, that contains p_i in the target image (Equation 5), and p'_i is the corresponding feature point in the reference image.

B. Global alignment term

To align the grid vertices and avoid unnecessary moving of the vertices from their pre-warped positions, we construct an improved global term to provide a good estimation. We redefine the global term E_G as the summation in the L2 norm of the difference between the origin vertex and its deformation.

$$E_G = \sum_j \left\| V^*\hat{V}_j - \left(V^*\right)^2 \right\|^2 \tag{8}$$

$$V_j^* = \begin{cases} p'_j, & \text{if } V_j \text{ is feature point vertex} \\ H_{APAP}V_j, & \text{other vertices} \end{cases}, \tag{9}$$

Where p'_i denotes the matching feature point in the reference image I', H_{apap} is the local homography in Zaragoza et al. (2013) and j is the cell vertices index. V and $\hat{V}$ are the corresponding vertex in the target image triangular cell and its deformation.

C. Color similarity term

To constrain the smoothness of color models with a connected neighboring region and let these selected intensities remain close after the mesh deformation, we design this color similarity term. Assuming that the overlapping image region with any points has the

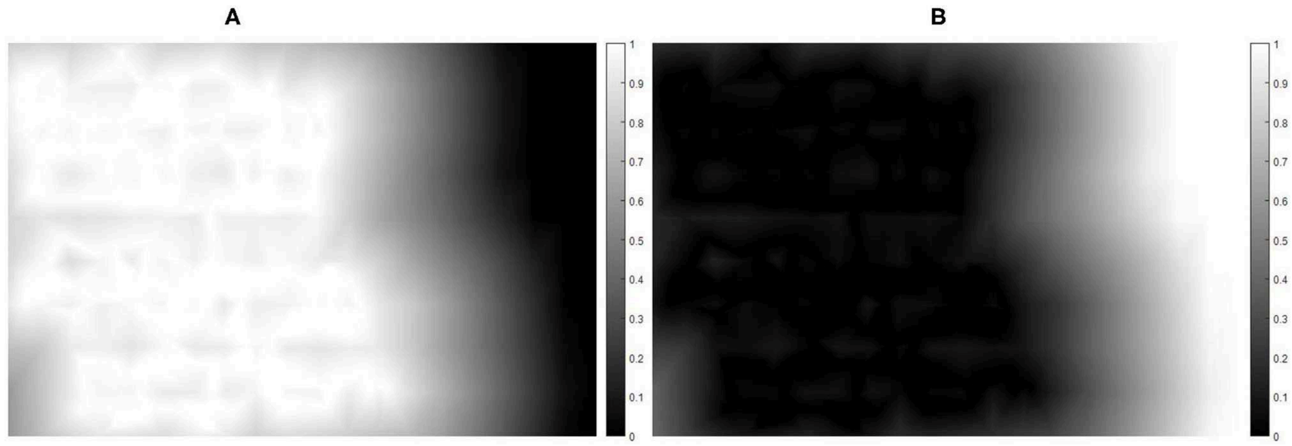


FIGURE 4
Weight map of the target image. **(A)** Weight map of content-preserving warps and **(B)** weight map of APAP warps. The color denotes the weight value, which is between 0 and 1.

same intensities. Thus, we can obtain the intensity difference value between the two overlapping image parts.

$$E_c = \sum_{\Omega} \sum_{(x,y)=Q} \left\| \hat{I}_{\Omega}(\hat{x},\hat{y}) - I'_{\Omega}\left(x',y'\right) \right\|^2 \tag{10}$$

Where Q denotes the feature point set, which is in the overlapping image region. Here, Ω denotes the point connected neighboring area at position $(\hat{x},\hat{y})$ and its corresponding (x',y'). Ω is set to 9×9 in our experiment.

Joint optimization

After we obtain a warped version of this triangular mesh vertices by the above energy function. The overlapping image area in the target image and reference image can stitch well, and the mosaic image has a good performance. The feature points have a good match pair only on the overlapping region, and if we only get the warped version by the energy function, the stitching result may have an unnatural visual effect on the non-overlapping area. Hence, we update the final warped vertices by controlling the relative amount of the vertices obtained with APAP warps injected into the vertices obtained by the energy function way in a soft manner, which can be auto-adjusted further by the origin image pair. The final vertices can be denoted as follows:

$$\widetilde{V}_i = c_i^1 \hat{V}_i + c_i^2 \bar{V}_i, \tag{11}$$

Where, $\widetilde{V}_i$ is the final triangular cell vertex after deformation, V_i is the cell vertex in the target image I, $\bar{V}_i = H_{apap} V_i$, and $\hat{V}_i$ denotes the vertices after deformation by the energy function. H_{apap} can find the details in Zaragoza et al. (2013), APAP computes a local homography for each image patch for high-precision local alignment, so we use each homography in this study. c^1 and c^2 are weighting coefficients. We also make $c^1 + c^2 = 1$, and c^1 and c^2 are between 0 and 1. They are identified by the following equations:

$$c_i^2 = \frac{\min\left(\frac{D_i}{\max(D_i)}, \gamma\right)}{\gamma}, c_i^1 = 1 - c_i^2 \tag{12}$$

$$D_i = \min\left(d_i(k)\right), k = 1, 2, 3\ldots \tag{13}$$

$$d_i(k) = Dist(V_i, P(k)), \tag{14}$$

Where Dist($\cdot$) represents the function to calculate the distance between two points, P is the feature point sequence of $p_1, p_2, \ldots, \gamma$ is an adjustable parameter, in fact, as $\gamma \rightarrow 1$ the shortest distance when the weight is equal to 1 between vertex and the matched feature points is the largest. Thus, V_i is the location of the i-th location in the image cell vertices. As shown in Figure 4, when the vertex is near the over from the matched feature points regions (the overlapping regions), the content-preserving warps have a high weight to ensure accurate alignment. On the contrary, the APAP warps have a high weight for fewer distortions for vertices far from the overlapping regions. Therefore, the final warp has good performance by using the weight combination. Figure 5 shows the comparison results with APAP and global homography.

Experiments

To verify the effectiveness of the proposed image stitching method, we test the method by subjective and objective assessments on pairwise datasets. In this section, we illustrate several representative image pair stitching results for comparing our warp for image stitching with several state-of-the-art stitching methods. First, we show a quantitative evaluation of the alignment accuracy for comparing our method against the state-of-the-art image stitching methods, namely, APAP, global homography, APAP, AutoStich, and SPW. Second, we give a quantitative evaluation of pairwise alignment by our image stitching way and several state-of-the-art methods. The mesh-based warps have a good performance; therefore, we ran a series of tests. Thus, the experimental parameters of the comparative paper are also consistent with the original paper.

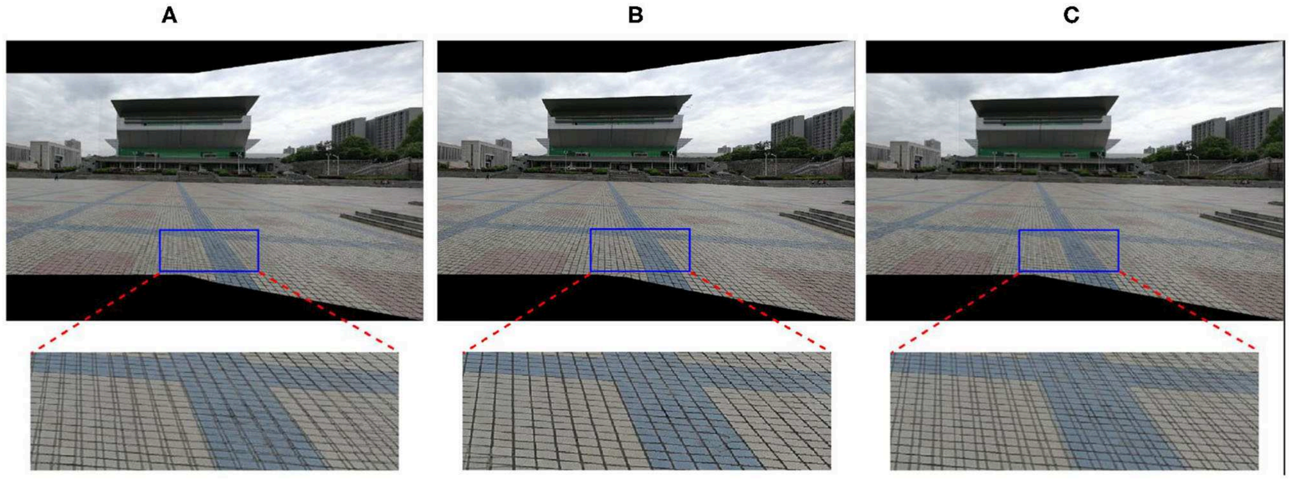


FIGURE 5
Comparisons with APAP and global homography. **(A)** APAP, **(B)** our method, and **(C)** global homography.

In our experiment, we use VLFeat (Vedaldi and Fulkerson, 2010) library to extract and match SIFT (Lowe, 2004) feature key points and run RANSAC to remove mismatches and match feature points by Jia et al. (2016). Codes are implemented in MATLAB (some codes are in C++ for efficiency) and run on a desktop PC with Intel i3-10100 3.6 GHz CPU and 16GB RAM. Then, all the image pairs in our test are contributed by the authors of Li et al. (2017). For parameter settings, $\gamma = 0.8$, the number of the APAP vertex is set to 5×6, and the matched feature vertex is set to 0.7x the total number of the matched feature points. As shown in Figure 3, if ω_G is too small then the vertices distortion becomes serious, and if ω_G is too large, then the region outside the vertex is severely distorted. Thus, ω_G is set to 10 in the experiment. The experimental parameters of the comparison algorithm are consistent with its original paper.

Qualitative evaluation of pairwise stitching

Figure 6 depicts the result of image stitching on the Temp image pair. Each row illustrates a panorama result of different methods, and the green and blue box regions are enlarged for a wide view of the local details. As we can see, all the results have a good performance. Nevertheless, our method has a better performance on the details. The global homography and AutoStitch could not align two images well using a global 2D transformation, in addition to the stitching results suffering from ghosting, such as the curves on the ground in the green rectangle and the orange bag being duplicated in the blue zoomed-in rectangle. Considering the limitations of global transformation, the APAP method shows a fine stitching result as shown in Figure 6C; however, the details in the APAP results are not good as our method, comparing the white arched logo in Figures 6A, C, it can be seen that our result has few artifacts. As shown in Figures 6A, B, D, the orange bag in the blue zoomed-in rectangle has few artifacts in our results. The SPW method has a weakness in the image with few lines, the detail is illustrated in Figure 6E, and there is obvious misalignment. Contrast the above methods with our method, which has less "ghostly" with few artifacts. Especially, the curves on the ground, the white arched logo on the wall, and the orange bag in the blue zoomed-in rectangle have few artifacts, as shown in the first row of Figure 6A, so our method has the best stitching quality. The better performance is due to our approach adding a tight constraint into the mesh warps and combining our method with the APAP warp.

To comprehensively demonstrate the effectiveness of our image stitching method, we compare the final stitching results on a different scene. As shown in Figure 7, from left to right, the stitching results are the tower, riverbank, and theater, respectively. In the results of the riverbank, the round pillar misalignments are shown in the AutoStitch method. The other stitching method has a good performance on the riverbank. However, our method shows the *roads*, *wires*, and *buildings* on the riverbank more clearly. As shown in *tower*, the global homography method shows an obvious "ghostly," and the gaps in the paving exhibit non-uniform distortions over the image. In the SPW result, the top of the tower is duplicated. Thus, all of the results introduce obvious distortion or ghosting, as indicated in Figure 7. As for scene *theater*, the gaps in the paving show less ghosting than the other methods because the authors of SPW combine point and line features in the mesh-based warp. Then, the building on the overlapping region exhibited more ghosting than our method. Generally speaking, our method shows less distortion and ghosting results.

Quantitative evaluation of alignment

To quantify the alignment accuracy of our proposed method, we calculate the structural similarity index (SSIM) (Wang et al., 2004) along the overlapping region points as an evaluation standard. The SSIM is usually used to describe the alignment accuracy on the different images. The quantitative results are shown in Table 1, which includes five methods tested data from seven scenes. As shown in Table 1, our method yields the highest similarity value in five scenes, and our method is next to the highest value in the other two scenes. Our average similarity value is 0.9426, 1.5% higher than SPW,

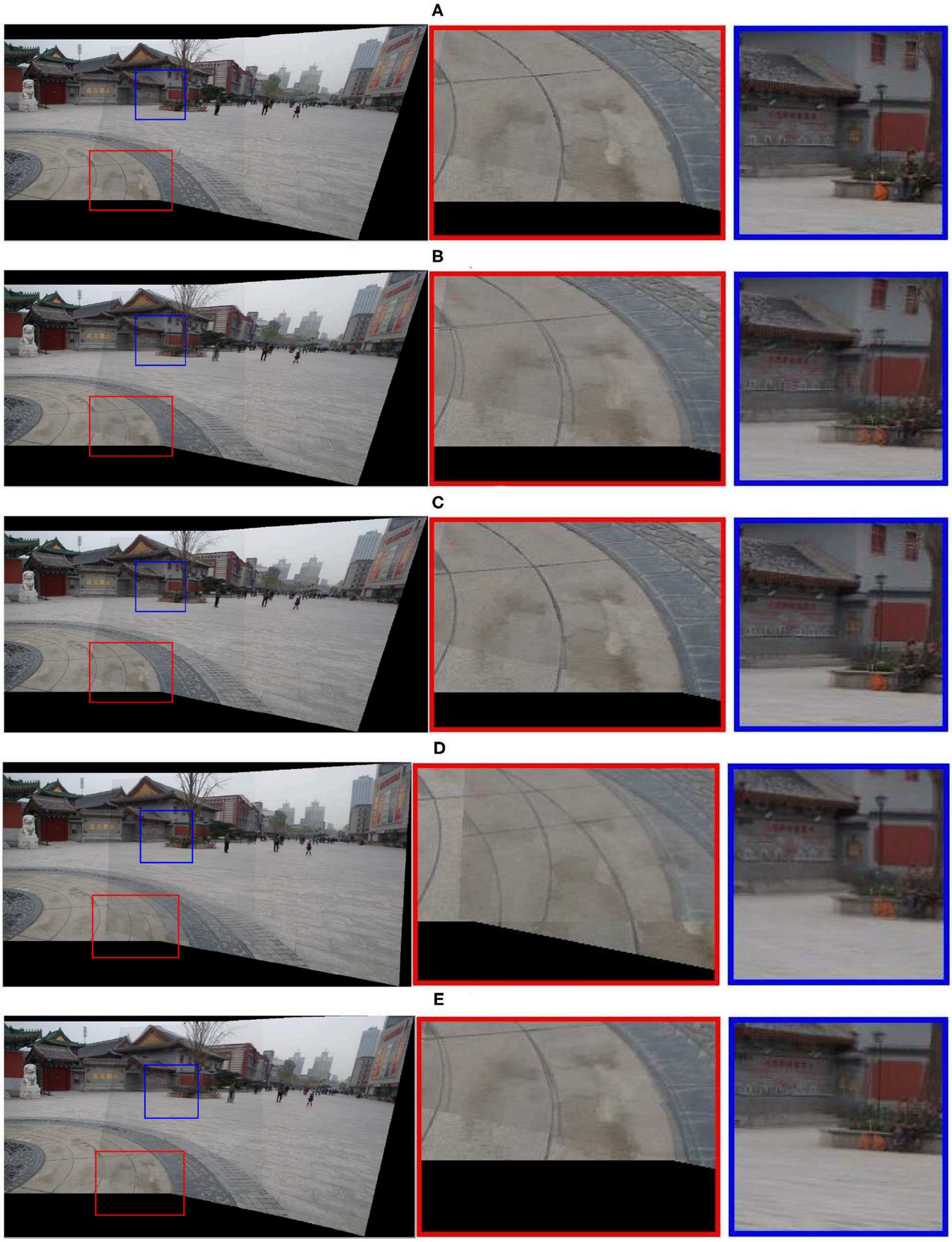


FIGURE 6
Comparisons with state-of-the-art image stitching techniques on the Temp image dataset. From top to bottom, each row is **(A)** our method, **(B)** global homography, **(C)** APAP, **(D)** AutoStich, and **(E)** SPW. The red boxes and blue boxes show the stitching details clearly stated.

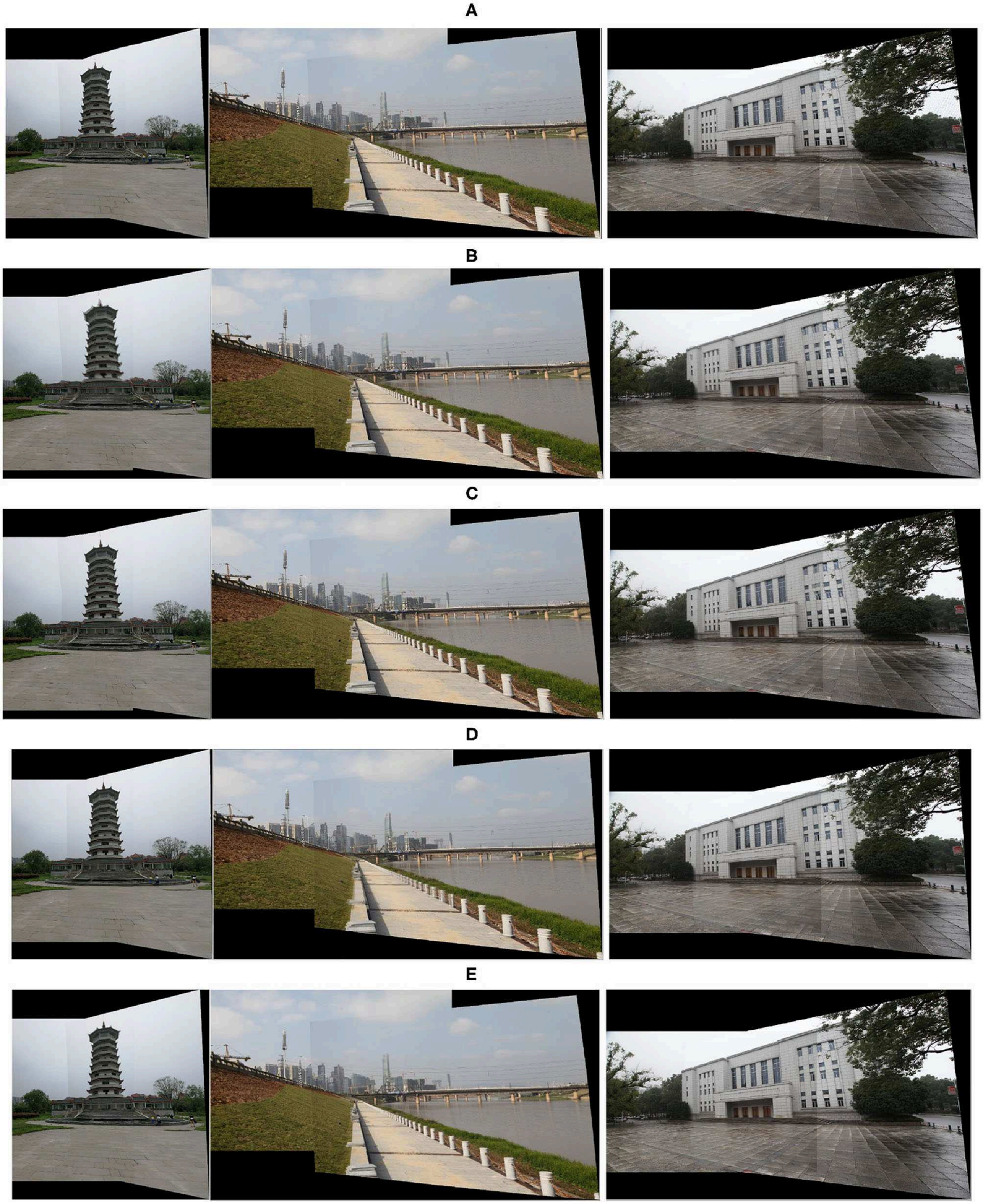


FIGURE 7
Comparison results for different scenes. From top to bottom, the image stitching results are **(A)** our method, **(B)** global homography, **(C)** APAP, **(D)** AutoStitch, and **(E)** SPW, respectively. Here, from left to right, the scenes are the *tower*, *riverbank*, and *theater*.

TABLE 1 Comparison of the SSIM of different scenes (the global homography is abbreviated as GH).

	Railtracks	Temp	Tower	Theater	Riverbank	Racetracks	Worktable	Average
Our	**0.936**	**0.945**	**0.963**	**0.947**	0.959	**0.898**	0.949	**0.943**
GH	0.884	0.905	0.945	0.896	0.949	0.867	0.936	0.913
APAP	0.909	0.939	0.912	0.918	**0.965**	0.887	**0.953**	0.926
AutoStitch	0.898	0.913	0.946	0.921	0.959	0.864	0.751	0.893
SPW	0.922	0.911	0.946	0.933	0.960	0.880	0.947	0.928

The best value is shown in bold.

5.5% higher than AutoStitch, 3.3% higher than global homography, and 1.8% higher than APAP. A comprehensive visual comparison is demonstrated in Figures 6, 7. Our method performs better than all the other methods in preserving local details and being artifact-free in overlapping regions.

Conclusion

We have proposed an improved adaptive triangular mesh-based image stitching method. First, without sacrificing the accuracy of alignment, a non-uniform triangular mesh is set over the image to improve alignment accuracy. The non-uniform grid includes uniform and non-uniform vertices, and the non-uniform vertices are from the matched feature points, which provide good constraints on overlapping areas and is a novel method. Second, an improved deformation function is constructed to obtain deformed vertices. To constrain the smoothness of the color model, we introduced a color similarity term in the deformation function. Finally, we give a novel strategy for combining our method with APAP warp to obtain its flexibility. The combining strategy not only absorbs the advantages of the good alignment of APAP but also can adaptively adjust its weight value. The proposed algorithm is proved on different images and compared with other methods. The experimental results illustrate that the image stitching method in this study can achieve more accurate panoramic stitching and less overlapping distortion and improve the accuracy of panoramic image stitching. The proposed method has an improvement in accuracy compared to the other methods. The mean SSIM of the proposed method is 0.9426, which is 1.5% higher than SPW, 5.5% higher than AutoStitch, 3.3% higher than global homography, and 1.8% higher than APAP. For further work, we expect to apply this method to large parallax image stitching and image stitching with moving targets.

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

WT created the improved model and the provided initial idea, conducted the experiments, and wrote the article. FJ and XW put forward some effective suggestions for improving the structure of the article. 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

Brown, M., and Lowe, D. G. (2007). Automatic panoramic image stitching using invariant features. *Int. J. Comput. Vis.* 74, 59–73. doi: 10.1007/s11263-006-0002-3

Chen, K., Tu, J., Yao, J., and Li, J. (2018). Generalized content-preserving warp: direct photometric alignment beyond color consistency. *IEEE Access* 6, 69835–69849. doi: 10.1109/ACCESS.2018.2877794

Chen, X., Yu, M., and Song, Y. (2022). Optimized seam-driven image stitching method based on scene depth information. *Electronics* 11, 1876. doi: 10.3390/electronics11121876

Chen, Y.-S., and Chuang, Y.-Y. (2016). "Natural image stitching with the global similarity prior," in *European Conference on Computer Vision* (Amsterdam: Springer), 186–201.

Edelsbrunner, H., Tan, T. S., and Waupotitsch, R. (1990). "An o (n 2log n) time algorithm for the minmax angle triangulation," in *Proceedings of the Sixth Annual Symposium on Computational Geometry* (Berkley, CA), 44–52.

Gao, J., Kim, S. J., and Brown, M. S. (2011). "Constructing image panoramas using dual-homography warping," in *CVPR 2011* (Colorado Springs, CO: IEEE), 49–56.

Gao, J., Li, Y., Chin, T.-J., and Brown, M. S. (2013). "Seam-driven image stitching," in *Eurographics (Short Papers)* (Girona), 45–48.

Guo, D., Chen, J., Luo, L., Gong, W., and Wei, L. (2021). Uav image stitching using shape-preserving warp combined with global alignment. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2021.3094977

Jia, Q., Gao, X., Fan, X., Luo, Z., Li, H., and Chen, Z. (2016). "Novel coplanar line-points invariants for robust line matching across views," in *European Conference on Computer Vision* (Amsterdam: Springer), 599–611.

Jia, Q., Li, Z., Fan, X., Zhao, H., Teng, S., Ye, X., et al. (2021). "Leveraging line-point consistence to preserve structures for wide parallax image stitching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Nashville, TN: IEEE), 12186–12195.

Koecher, M., and Krieg, A. (2007). *Ebene Geometrie*. Berlin: Springer-Verlag.

Kopf, J., Uyttendaele, M., Deussen, O., and Cohen, M. F. (2007). Capturing and viewing gigapixel images. *aCm Trans. Graph*. 26, 93-es. doi: 10.1145/1276377.1276494

Li, D., He, K., Sun, J., and Zhou, K. (2015). "A geodesic-preserving method for image warping," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Boston, MA: IEEE), 213–221.

Li, J., Deng, B., Tang, R., Wang, Z., and Yan, Y. (2019). Local-adaptive image alignment based on triangular facet approximation. *IEEE Trans. Image Process*. 29, 2356–2369. doi: 10.1109/TIP.2019.2949424

Li, J., Wang, Z., Lai, S., Zhai, Y., and Zhang, M. (2017). Parallax-tolerant image stitching based on robust elastic warping. *IEEE Trans. Multimedia* 20, 1672–1687. doi: 10.1109/TMM.2017.2777461

Liao, T., and Li, N. (2019). Single-perspective warps in natural image stitching. *IEEE Trans. Image Process*. 29, 724–735. doi: 10.1109/TIP.2019.2934344

Lin, C.-C., Pankanti, S. U., Natesan Ramamurthy, K., and Aravkin, A. Y. (2015). "Adaptive as-natural-as-possible image stitching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Boston, MA: IEEE), 1155–1163.

Lin, K., Jiang, N., Cheong, L.-F., Do, M., and Lu, J. (2016). "Seagull: seam-guided local alignment for parallax-tolerant image stitching," in *European Conference on Computer Vision* (Amsterdam: Springer), 370–385.

Lin, W.-Y., Liu, S., Matsushita, Y., Ng, T.-T., and Cheong, L.-F. (2011). "Smoothly varying affine stitching," in *CVPR 2011* (Colorado Springs, CO: IEEE), 345–352.

Liu, F., Gleicher, M., Jin, H., and Agarwala, A. (2009). Content-preserving warps for 3D video stabilization. *ACM Trans. Graph*. 28, 1–9. doi: 10.1145/1576246.1531350

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis*. 60, 91–110. doi: 10.1023/B:VISI.0000029664.99615.94

Medeiros, R., Scharcanski, J., and Wong, A. (2016). Image segmentation via multi-scale stochastic regional texture appearance models. *Comput. Vis. Image Understand*. 142, 23–36. doi: 10.1016/j.cviu.2015.06.001

Nie, L., Lin, C., Liao, K., Liu, S., and Zhao, Y. (2022). "Deep rectangling for image stitching: a learning baseline," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (New Orleans, LA: IEEE), 5740–5748.

Ren, M., Li, J., Song, L., Li, H., and Xu, T. (2022). Mlp-based efficient stitching method for uav images. *IEEE Geosci. Remote Sens. Lett*. 19, 1–5. doi: 10.1109/LGRS.2022.3141890

Szeliski, R. (2007). Image alignment and stitching: a tutorial. *Foundat. Trends®Comput. Graph. Vis*. 2, 1–104. doi: 10.1561/0600000009

Vedaldi, A., and Fulkerson, B. (2010). "Vlfeat: an open and portable library of computer vision algorithms," in *Proceedings of the 18th ACM international conference on Multimedia* (Firenze), 1469–1472.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process*. 13, 600–612. doi: 10.1109/TIP.2003.819861

Yan, W., Liu, C., and Peng, F. (2017). Robust multi-homography method for image stitching under large viewpoint changes. *Int. J. Hybrid Inf. Technol*. 10, 1–18. doi: 10.14257/ijhit.2017.10.9.01

Zaragoza, J., Chin, T.-J., Brown, M. S., and Suter, D. (2013). "As-projective-as-possible image stitching with moving dlt," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Portland, OR: IEEE), 2339–2346.

Zhang, F., and Liu, F. (2014). "Parallax-tolerant image stitching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Columbus, OH: IEEE), 3262–3269.

Zheng, J., Wang, Y., Wang, H., Li, B., and Hu, H.-M. (2019). A novel projective-consistent plane based image stitching method. *IEEE Trans. Multimedia* 21, 2561–2575. doi: 10.1109/TMM.2019.2905692

Zhu, Z., Riseman, E. M., and Hanson, A. R. (2001). "Parallel-perspective stereo mosaics," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, Vol. 1* (Vancouver, BC: IEEE), 345–352.

frontiers | Frontiers in Neurorobotics

TYPE Original Research
PUBLISHED 09 February 2023
DOI 10.3389/fnbot.2023.1119231

Check for updates

OPEN ACCESS

EDITED BY
Zhou Wei,
Yunnan University, China

REVIEWED BY
Rohit Shukla,
University of Wisconsin–Madison, United States
Lidong Yang,
Inner Mongolia University of Science
and Technology, China
Feihu Zhang,
Northwestern Polytechnical University, China

*CORRESPONDENCE
Chunyu Zhang
✉ 1910654@stu.neu.edu.cn

RECEIVED 08 December 2022
ACCEPTED 26 January 2023
PUBLISHED 09 February 2023

CITATION
Zhang C, Xu F, Wu C and Xu C (2023) Rethinking 1D convolution for lightweight semantic segmentation.
Front. Neurorobot. 17:1119231.
doi: 10.3389/fnbot.2023.1119231

Rethinking 1D convolution for lightweight semantic segmentation

Chunyu Zhang[1]*, Fang Xu[2], Chengdong Wu[1] and Chenglong Xu[3]

[1]Faculty of Robot Science and Engineering, Northeastern University, Shenyang, China, [2]Shenyang Siasun Robot & Automation Company Ltd., Shenyang, China, [3]College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin, China

Lightweight semantic segmentation promotes the application of semantic segmentation in tiny devices. The existing lightweight semantic segmentation network (LSNet) has the problems of low precision and a large number of parameters. In response to the above problems, we designed a full 1D convolutional LSNet. The tremendous success of this network is attributed to the following three modules: 1D multi-layer space module (1D-MS), 1D multi-layer channel module (1D-MC), and flow alignment module (FA). The 1D-MS and the 1D-MC add global feature extraction operations based on the multi-layer perceptron (MLP) idea. This module uses 1D convolutional coding, which is more flexible than MLP. It increases the global information operation, improving features' coding ability. The FA module fuses high-level and low-level semantic information, which solves the problem of precision loss caused by the misalignment of features. We designed a 1D-mixer encoder based on the transformer structure. It performed fusion encoding of the feature space information extracted by the 1D-MS module and the channel information extracted by the 1D-MC module. 1D-mixer obtains high-quality encoded features with very few parameters, which is the key to the network's success. The attention pyramid with FA (AP-FA) uses an AP to decode features and adds a FA module to solve the problem of feature misalignment. Our network requires no pre-training and only needs a 1080Ti GPU for training. It achieved 72.6 mIoU and 95.6 FPS on the Cityscapes dataset and 70.5 mIoU and 122 FPS on the CamVid dataset. We ported the network trained on the ADE2K dataset to mobile devices, and the latency of 224 ms proves the application value of the network on mobile devices. The results on the three datasets prove that the network generalization ability we designed is powerful. Compared to state-of-the-art lightweight semantic segmentation algorithms, our designed network achieves the best balance between segmentation accuracy and parameters. The parameters of LSNet are only 0.62 M, which is currently the network with the highest segmentation accuracy within 1 M parameters.

KEYWORDS

semantic segmentation, lightweight network, 1D convolution, encoder-decoder, feature alignment

1. Introduction

Semantic segmentation is one of the essential tasks in computer vision, which requires the classification of each pixel of an image. There are many problems in practical applications: application equipment has a small storage capacity and cannot store large-scale networks; equipment needs to complete the calculation of semantic segmentation; reasoning speed needs to be faster to meet actual needs. Based on the above problems, the researchers adjusted the research direction accordingly and proposed lightweight semantic segmentation. The lightweight network has the advantages of fewer parameters, fast operation speed, and segmentation accuracy that meets engineering needs. The earliest lightweight semantic segmentation networks (LSNets) are SegNet (Badrinarayanan et al., 2017), ENet (Paszke et al., 2016), SQNet (Treml et al., 2016), ERFNet (Romera et al., 2017), LinkNet (Chaurasia and Culurciello, 2017), and BiSeNet (Yu et al., 2018). Their segmentation accuracy is around 65 mIoU, and their inference speed is 50 FPS. The segmentation accuracy and inference speed of LSNets that have emerged in recent years have significantly improved. Typical networks include HyperSeg-S (Nirkin et al., 2021), STDC1 (Fan et al., 2021), STDC2, SFNet (Li et al., 2020), and PIDNet (Xu et al., 2022). By reading a lot of semantic segmentation papers, we summarized several directions for lightweight semantic segmentation design: (1) downsampling: reduce the resolution of the input image and reduce the amount of calculation; (2) design efficient convolution: expand the receptive field of convolution, reduce model parameters, and calculations; (3) residual connection: reuse features, improve gradient propagation; (4) design backbone encoding module: standard backbones include ResNet (He et al., 2016), SqueezeNet (Iandola et al., 2016), ShuffleNetV2 (Ma et al., 2018), MobileNet (Howard et al., 2019), and EfficientNet (Tan and Le, 2019).

In this paper, we rethink the application of 1D convolution in lightweight semantic segmentation and design a 1D multi-layer spatial module (1D-MS) and 1D multi-layer channel module (1D-MC). 1D-MS and 1D-MC adopt the idea of the multi-layer perceptron (MLP), simultaneously adds global information. They obtain the best balance in terms of encoding performance and parameters. We also propose a feature alignment module (FA), which solves the problem of feature misalignment on the network, improving segmentation accuracy. Based on the above modules, we designed a 1D-mixer module and an attention pyramid with FA (AP-FA). 1D-mixer adopts the coding structure of the transformer. The first residual connection contains 1D-MSs, and the channel separation operation aims to extract spatial information and reduce the amount of calculation. The second residual connection contains 1D-MCs to facilitate information fusion between channels. The AP-FA module contains an AP and a FA to decode and upsample features. The purpose of our design of the AP-FA module is to fuse multi-scale information, reduce the loss of details, solve the problem of misalignment, and improve the segmentation accuracy. Based on the 1D-mixer and AP-FA modules, we propose an efficient, LSNet consisting entirely of 1D convolutions. The 1D-LSNet network we designed is trained and predicted on only one 1080Ti GPU, and there are no other pre-training operations. On the Cityscapes dataset, a segmentation accuracy of 72.6 mIoU has been achieved, and the number of parameters is 0.62 M. It is currently the lightweight network with the highest segmentation accuracy within 1 M parameters. On the CamVid dataset, our accuracy is 70.5 mIoU, and the inference speed reaches 122 FPS, the model with the highest accuracy among all lightweight networks. On the ADE2K dataset, our network achieves an accuracy of 36.4 mIoU. We transplanted the trained network to the Qualcomm Snapdragon 865 mobile processing device, and the delay time was 224 ms, which met the requirements for mobile devices. Compared with advanced semantic segmentation algorithms, LSNet outperforms the latest lightweight networks regarding segmentation accuracy and parameter balance.

Our contributions can be summarized in the following points:

1. A 1D-MS and a 1D-MC are proposed, which inherit the design idea of MLP and integrate global feature operations. Since this module uses 1D convolution, it is not limited by the input size. This module has the advantages of fewer parameters and strong coding ability.
2. We designed the 1D-mixer module, which adopts the structure of the visual transformer, and combines the 1D-MS module, the 1D-MC module, and the channel separation technology. This module encodes and fuses the feature map along the space and channel direction, which has the advantages of strong encoding ability and few parameters.
3. An AP-FA is proposed. The purpose of the AP is to expand the network receptive field, reduce the loss of details, and improve the segmentation accuracy. At the same time, to solve the loss of accuracy caused by feature misalignment, a FA is proposed for upsampling.
4. Based on the above modules, we designed a LSNet. The network performed well on the Cityscapes and CamVid datasets compared with the advanced LSNet, and it obtained the best balance between accuracy and parameters. The network trained in the ADE2K data set is transplanted to the mobile device, and the delay time is 224 ms, which meets the requirements of the mobile device. The number of parameters of the network we designed is 0.62 M, and the accuracy is the highest among the networks within 1 M parameters.

2. Related work

2.1. Semantic segmentation

Semantic segmentation (Brempong et al., 2022; Mo et al., 2022; Sheng et al., 2022; Ulku and Akagündüz, 2022) is the vision task of classifying image pixels. FCN (Noh et al., 2015) replaces the FC of the classification network with convolution, enabling the development of end-to-end convolutional networks. Recently, MLP-based networks have shown great potential in object detection and surpassed transformer-based semantic segmentation methods. LEDNet (Wang et al., 2019) is a typical lightweight network. The encoder uses a combination of residual modules and decomposed convolutions, and the decoder uses a simple pyramid structure. The algorithm's structure conforms to the design principle of lightweight semantic segmentation structure and has the advantages of high segmentation accuracy and few parameters. We summarized the main design ideas of lightweight semantic segmentation through many research papers, mainly multi-scale receptive field fusion, multi-scale semantics, expanding receptive field, strengthening edge features, and obtaining global information.

2.2. Attention mechanism

The purpose of the attention mechanism (Guo et al., 2022a,b) is to select features and make reasonable use of computing resources. There are two types of attention mechanisms in semantic segmentation networks, channel attention and spatial attention, which play different roles in the network. Spatial attention focuses on the central region from the perspective of feature space. Channel attention focuses on selecting feature channels and using some channels as the primary encoding object. CBAM (Woo et al., 2018) uses a mixture of typical channels and spatial attention. The most significant advantage of this module is that it has a small number of parameters. It can be seamlessly integrated into any CNN architecture, ignoring additional overhead.

2.3. Transformer

The transformer (Han et al., 2022; Khan et al., 2022) was first used in the field of NLP to encode the input sequence. ViT (Dosovitskiy et al., 2020) demonstrates that transformers can also be applied to image classification. ViT treats an image as a sequence and sends it to a transformer layer for classification. ViT-based variants include CPVT (Chu et al., 2021), TNT (Han et al., 2021), and LocalViT (Li et al., 2021), improving image classification accuracy. For semantic segmentation, the core architecture of SETR (Zheng et al., 2021) is still the encoder-decoder structure. However, compared to the traditional CNN-led encoder structure, SETR uses transformer to replace it, but this method could be more efficient. Recently, SegFormer (Xie et al., 2021) designed a novel hierarchical transformer encoder that outputs multi-scale features. It does not require positional encoding, thus avoiding interpolation of positional encodings. SegFormer also has disadvantages: the output resolution is fixed, and the resolution is too low, which affects the detail segmentation.

3. Method

3.1. 1D-MS and 1D-MC

Lightweight semantic segmentation research aims to design a neural network with small parameters and high segmentation accuracy. The current lightweight segmentation network can be divided into two categories: (1) the number of parameters is more than 5 M, and the segmentation accuracy is between 72 and 80 mIoU. The utilization rate of such network parameters is low, and it may be necessary to increase the parameters by about 10 M for every 1 mIoU increase in accuracy. Although the accuracy can meet the application requirements, it deviates from the original intention of lightweight. (2) The number of parameters is below 5 M, and the segmentation accuracy is less than 72 mIoU. The parameter utilization rate of this type of network is high, but the segmentation accuracy could be better. The parameters and segmentation accuracy are challenging to balance. MLP has recently become a new research direction, and its advantages are high segmentation accuracy and a small number of parameters, as shown in **Figure 1A**. MLP has a fatal shortcoming. It has strict requirements on the input feature size and requires additional feature cropping to be applied to the semantic segmentation network.

Based on the above analysis, we designed a 1D-MS and a 1D-MC. The purpose of our design of these two modules is to inherit the excellent performance of MLP and solve the shortcomings of MLP. The design process is as follows: 1D-MS is divided into a local feature extraction branch and a global information extraction branch, as shown in **Figure 1C**. The local feature extraction branch adopts the structure of MLP and replaces the fully connected layer with 1D depth separation convolution (convolution kernel size is 3 × 1 and 1 × 3). This not only fits the coding performance of MLP but also solves the problem of input size. Since 1D convolution is used for spatial encoding, there will be decoupling problems in extracting features. To solve this problem, we design the global information extraction branch. This branch uses max-pooling and avg-pooling to obtain global feature information and generates global features through 1 × 1 convolution. The addition of the output features of the two branches not only solves the decoupling problem but also integrates the local and global features to improve the coding performance. The design concept of 1D-MC is similar to that of 1D-MS. As shown in **Figure 1B**, its channel fusion branch replaces the MLP fully connected layer with 1 × 1 convolution, and the channel selection branch uses the global max-pooling operation. It is worth noting that the number of intermediate feature output channels of our designed channel fusion branch is half the number of input channels. The output of the two branches is multiplied, and 1D-MC not only performs information fusion between channels but also selects feature channels.

The 1D-MS and 1D-MC we designed to have the following advantages: they inherit MLP's advantages of solid coding ability and fewer parameters; there is no requirement for the input feature size, which is more flexible than MLP; it adds a global feature branch and channel selection branch to improve the overall coding performance of the module.

3.2. 1D-mixer module

The design of the encoder is key to the success of the network. Visual transformer is the coding structure that has recently received the most attention and is widely used in object detection and semantic segmentation. The 1D-mixer module we designed uses the transformer architecture. The 1D-mixer module comprises 1D convolution, which extracts and fuses the feature's spatial and channel information. The 1D-mixer spatial feature encoding part includes the 1D-MS module, channel separation, and residual connection. The role of channel separation is to reduce the number of feature channels and the parameters required for later encoding. 1D-MS is used for encoding in the direction of feature space. This encoding module integrates local and global information and has strong encoding ability. Using residual connections increases the utilization of features and speeds up network training. The 1D-mixer channel information fusion part is composed of 1D-MC and residual connection. This part helps feature information flow between different channels and feature selection along the direction of the channel. The overall structure of the 1D-mixer is shown in **Figure 1D**, and the specific calculation process is as follows:

$$SF = Concat\left(MS\left(Split\left(X\right)\right)\right) + X \tag{1}$$

$$OUT = MC\left(SA\right) + SF \tag{2}$$

Where X represents the feature input.SF and OUT denote spatially encoded features and 1D-mixer encoded output. $Split$ means distinct channel separation, MS means 1D-MS module, and MC is the

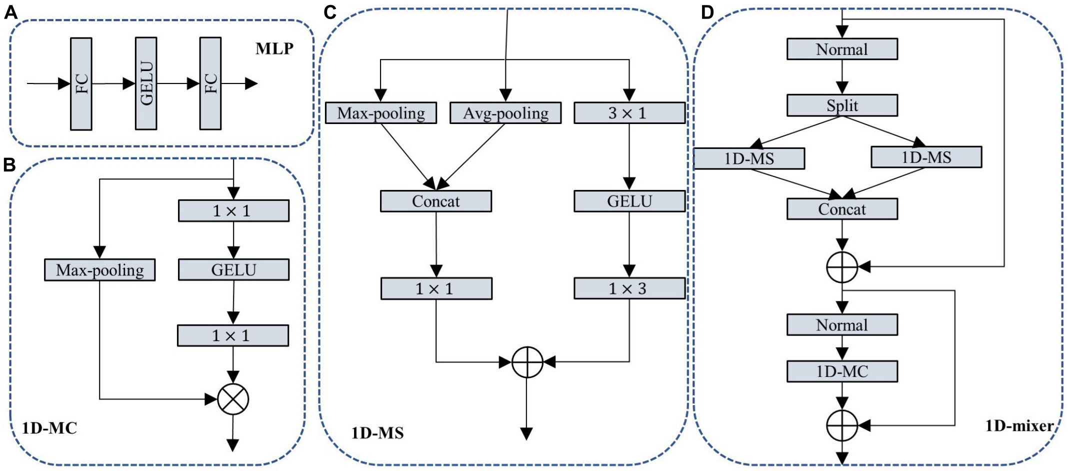


FIGURE 1
(A) Multi-layer perceptron (MLP); **(B)** 1D multi-layer channel module (1D-MC); **(C)** 1D multi-layer space module (1D-MS), and **(D)** 1D-mixer. ⊗ Means pixel multiplication; ⊕ means pixel addition; split means channel separation; concat means channel splicing.

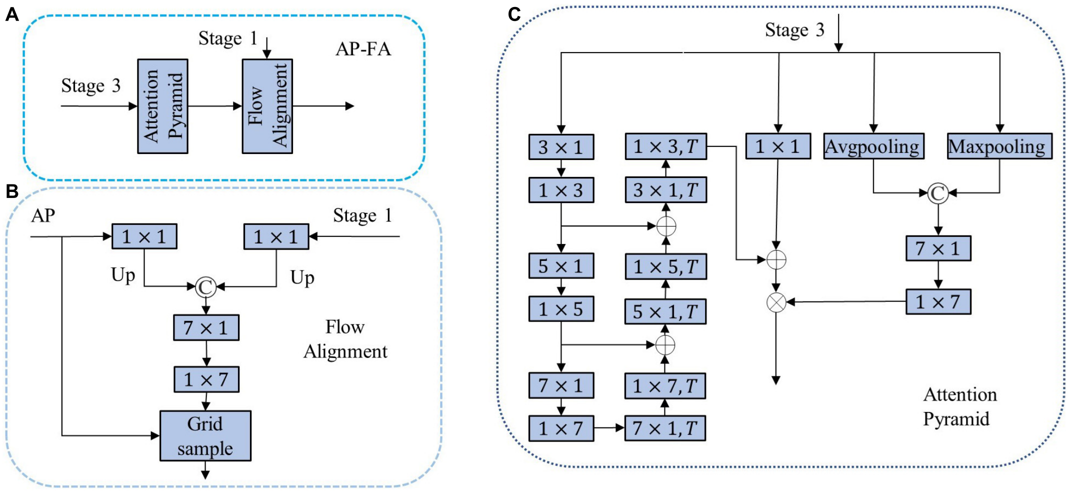


FIGURE 2
(A) Attention pyramid with flow alignment module (AP-FA); **(B)** FA; **(C)** AP. ⊗ Means pixel multiplication; ⊕ means pixel addition; © means channel splicing; *T* means deconvolution.

1D-MC module. + Means residual connection, and *Concat* means channel splicing.

Our 1D-mixer has the following advantages: (1) it adopts transformer structure to fuse spatial feature information and channel information to improve segmentation accuracy; (2) 1D-MS fuses local and global information of feature space direction with very few parameters; (3) 1D-MC module promotes the flow of feature information in the channel direction and selects effective feature channels; (4) it adopts channel separation operation to reduce model parameters and calculation further.

3.3. AP-FA module

In order to further extract high-level semantic information and adapt to different tasks, the network usually connects a decoder after the encoder, for which we designed a novel AP-FA, as shown in Figure 2A. The decoder consists of two main parts, one is the attention feature pyramid, and the other is the FA.

3.3.1. Attention pyramid

The AP consists of three branches: 1D pyramid structure, which can further encode features to obtain global information and detailed information; 1 × 1 convolution, which fuses channel information on the output of the encoder; the spatial attention branch acquires features. The spatial position relationship reduces the loss of details. The specific operation process is shown in Equation (3).

$$OUT = [C_{1\times1}(X) + P(X)] \times SA(X) \quad (3)$$

Where X and OUT represent the output feature of the Stage 3 and output of AP, P is the pyramid structure, $C_{1\times1}$is 1 × 1 convolution, SA is spatial attention, + represents the addition of corresponding elements, and represents the multiplication of corresponding elements. In the pyramid structure, the convolution

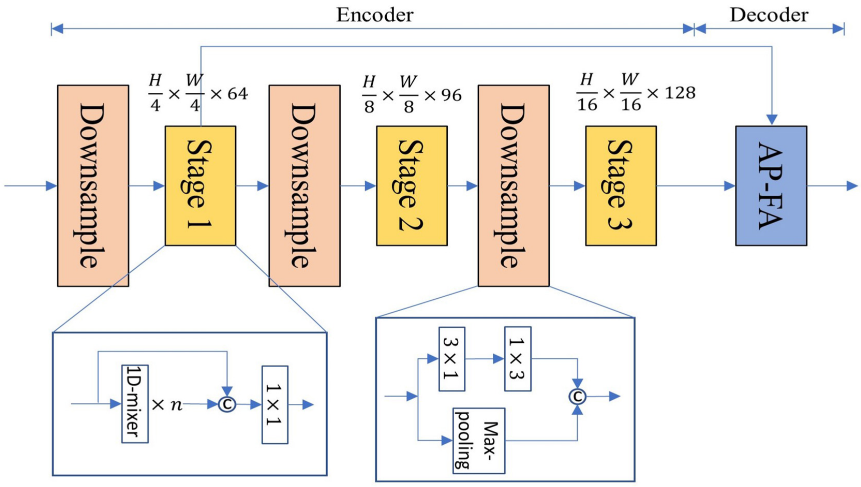

Encoder
Decoder
Downsample
$\frac{H}{4} \times \frac{W}{4} \times 64$
Stage 1
Downsample
$\frac{H}{8} \times \frac{W}{8} \times 96$
Stage 2
Downsample
$\frac{H}{16} \times \frac{W}{16} \times 128$
Stage 3
AP-FA
1D-mixer
× n
©
1 × 1
3 × 1
1 × 3
Max-pooling
©

FIGURE 3
The overall network architecture of lightweight semantic segmentation network (LSNet).

and deconvolution of the depth-wise convolution kernel sizes we use are (3 × 1, 5 × 1, and 7 × 1). There are two main reasons for using decomposed convolution here. One is that banded convolution meets the needs of lightweight networks, and the second is that most detected targets are banded. Therefore, using banded convolution is helpful for feature decoding. In the spatial attention branch, two kinds of pooling are used to obtain global information from multiple aspects and are encoded by 1 × 7 and 7 × 1 convolutions. 1 × 7 and 7 × 1 large convolutions can extract spatial features very well. AP related details are shown in Figure 2C.

3.3.2. Flow alignment

Ordinary upsampling will cause the problem of feature misalignment, resulting in decreased segmentation accuracy. We design a FA to restore the resolution and solve the misalignment problem by predicting the flow field inside the network. The specific process is shown in Figure 2B. The input of FA is the output feature (F_1) of Stage 1 and the output feature map (D) of AP. The feature map is obtained through a 1 × 1 convolutional layer to obtain a feature map with a channel number of 1. The resulting feature map is upsampled to ensure that the resolution of the two features is equal to the resolution of the input image. We concatenate them together and feed the concatenated feature maps into 7 × 1 and 1 × 7 concatenated convolutional networks. The above steps can be written as follows:

$$offset = Conv\left(U\left(C_{1\times 1}\left(F_1, D\right)\right)\right) \tag{4}$$

Among them, U represents the connection and upsampling operation, $C_{1\times 1}$is a 1 × 1 convolutional layer, $Conv$ is a series network of 7 × 1 and 1 × 7. $offset$ is the offset required for bilinear interpolation. We normalize $offset$ and sum it with the grid to generate an upsampling index. The features output by the AP is upsampled through the grid sample operation. The FA we designed combines high-level semantic features and low-level structural features to solve the problem of feature misalignment perfectly.

The AP-FA structure we designed has the following advantages: first, the pyramid structure is used to extract features, and the purpose is to expand the network receptive field and obtain more decoding features; second, the spatial attention structure suppresses unnecessary information, highlights important information, and

TABLE 1 The detailed architecture of lightweight semantic segmentation network (LSNet).

Stage	Type	Channel	Output size
Encoder	Downsampling	64	512 ×256
	1D-mixer ×3	64	512 ×256
	Downsampling	96	256 ×128
	1D-mixer ×3	96	256 ×128
	Downsampling	128	128 ×64
	1D-mixer ×21	128	128 ×64
Decoder	AP-FA	C	1,024 ×512

"Channel" denotes the number of output feature maps and "C" is the number of classes. "Output size" denotes the output size with an input size of 1,024 × 512.

TABLE 2 Ablation study results of 1D-mixer module.

Type	Model	mIoU (%)	Params (M)
Baseline	LSNet	72.6	0.62
Ablation for typical module	SS-nbt	69.8	2.52
	DAB	71.2	2.15
	CG	64.4	0.48
Ablation for depth	3, 9	65.6	0.40
	3, 12	67.2	0.46
	6, 12	67.4	0.49
	3, 15	68.8	0.51
	6, 15	67.5	0.54
	3, 18	70.2	0.57
	3, 24	72.3	0.67
Ablation for 1D-MS	3 × 3	70.9	2.31
	3 × 3 depth-wise	69.8	0.64
Ablation for 1D-MC	1 × 1	71.4	0.62

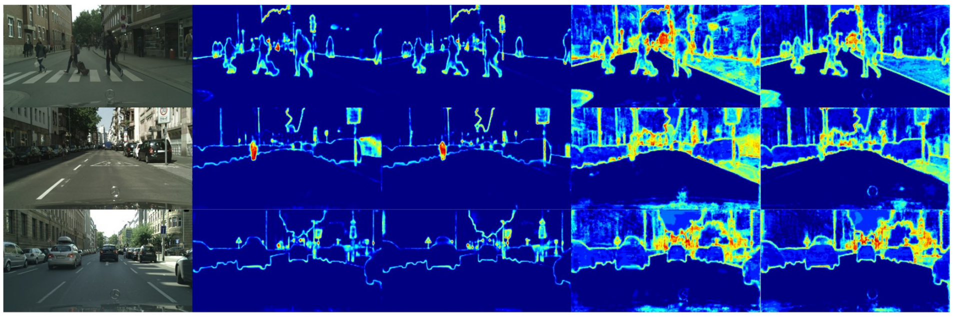


FIGURE 4
The lightweight semantic segmentation network (LSNet) feature visualization. The picture from left to right is: the original image, the encoder feature map using DAB, the encoder feature map using 1D-mixer, the network output feature map using DAB, and the network output feature map using 1D-mixer.

improves segmentation precision. Third, the FA method solves the misalignment problem when bilinear interpolation is used for upsampling and improving segmentation accuracy.

3.4. Network architecture

Figure 3 is a structural diagram of LSNet, which uses an asymmetric encoder-decoder structure. The details of the specific design are shown in Table 1. The encoding part uses three stages to encode different resolution features, and the number of 1D-mixer in each stage is 3, 3, 21. The input resolutions of each stage are $(H_{\frac{1}{4}} \times W_{\frac{1}{4}}, H_{\frac{1}{8}} \times W_{\frac{1}{8}}, \mathit{and}\ H_{1/16} \times W_{1/16})$, where H and W are the height and width of the input image, respectively. The downsampling is 3 × 1 and 1 × 3 convolution concatenation, the step size is 2, and the max-pooling output is spliced simultaneously.

The input of the AP-FA decoder comes from the feature maps of Stage 1 and Stage 3, and the final scene parsing is performed through the attention feature pyramid and the FA. Much lightweight semantic segmentation ignores the decoder in order to reduce network parameters. A dense decoder can help improve segmentation accuracy without generating too many parameters. Many lightweight networks use three-stage encoders to cause the network's receptive field to be too small, and bilinear interpolation has problems with upsampling misalignment. Aiming at the problem of the decrease in segmentation accuracy caused by the above, we designed the AP module to expand the network receptive field and increase the global information. We design a FA to restore feature resolution and improve segmentation accuracy.

TABLE 3 Ablation study results of attention pyramid with flow alignment module (AP-FA) module.

Type	Model	mIoU (%)	Params (M)
Baseline	LSNet	72.6	0.62
Ablation for AP	1 ×1	70.5	0.59
Ablation for attention	–	72.2	0.62
Ablation for feature pyramid	–	70.9	0.59
	333	71.9	0.61
	235	72.0	0.61
	135	71.5	0.61
	3,579	72.5	0.62
Ablation for FA	Bilinear interpolation	70.8	0.62

4. Experiments

4.1. Datasets and implementation details

4.1.1. Cityscapes

Cityscapes (Cordts et al., 2016) is an urban scene parsing dataset commonly used for semantic segmentation training. It contains street scenes in multiple cities and 5,000 car-driving images collected from the driver's perspective. This network splits the dataset into 2,975, 500, and 1,525 for training, validation, and testing. We select 19 of these semantic categories for training. We convert the resolution of the original image from $2,048 \times 1,024$ to $1,024 \times 512$ to improve the running speed. We do not introduce additional pre-training during training.

4.1.2. CamVid

CamVid (Brostow et al., 2008) contains 701 street view images, of which 367 are used for training, 101 for validation, and 233 for testing. The data set semantically annotates 32 objects in the picture, and we only train 11 semantic objects. We reduce the resolution of the original image from 960 × 720 to 480 × 360 to improve the inference speed.

4.1.3. ADE2K

ADE2K contains 25,000 pictures, and the resolution of each picture is not uniform. We unified the size of the pictures to 512 × 512 to facilitate model training. The training set contains 20,000 images, the validation set contains 2,000 images, and the test set contains 3,000 images.

4.1.4. Implementation details

All our experiments are run on a 1080Ti GPU. PyTorch 1.7, CUDA 9.0, cuDNN 8.0, and Anaconda environment are specific configurations. For fairness, we adopted the training configuration widely used by everyone. The details are as follows: the stochastic

TABLE 4 Evaluation results of our lightweight semantic segmentation network (LSNet) and other state-of-the-art real-time semantic segmentation models on the Cityscapes test set.

Model	Input size	Pre-train	GPU	mIoU (%)	FPS	Params (M)
SegNet (Badrinarayanan et al., 2017)	640 x 360	ImageNet	TitanX	57	16.7	29.5
ENet (Paszke et al., 2016)	640 x 360	No	TitanX	58.3	135.4	0.4
ICNet (Zhao et al., 2018)	1,024 x 2,048	ImageNet	TitanX	69.5	30.3	26.5
ERFNet (Romera et al., 2017)	512 x 1,024	No	TitanX	68	41.7	2.1
ESPNet (Mehta et al., 2018)	512 x 1,024	No	TitanX	60.3	112	2.1
BiSeNet (Yu et al., 2018)	768 x 1,536	ImageNet	TitanX	68.4	72.3	5.8
Fast-SCNN (Poudel et al., 2019)	1,024 x 2,408	ImageNet	TitanX	68	123.5	1.11
ESPNetV2 (Mehta et al., 2019)	512 x 1,024	No	TitanX	66.2	67	1.25
DFANet (Li H. et al., 2019)	512 x 1,024	ImageNet	TitanX	70.3	160	7.8
LEDNet (Wang et al., 2019)	512 x 1,024	No	1080Ti	69.2	71	0.94
ESNet (Lyu et al., 2019)	512 x 1,024	No	1080Ti	69.1	63	1.66
DABNet (Li G. et al., 2019)	512 x 1,024	No	1080Ti	70.1	104	0.76
FDDWNet (Liu et al., 2020)	512 x 1,024	No	2080Ti	71.5	60	0.8
DDPNet (Yang et al., 2020)	768 x 1,536	No	1080Ti	74.0	85.4	2.52
LEANet (Zhang et al., 2022)	512 x 1,024	No	1080Ti	71.9	77.3	0.74
SFNet (Li et al., 2020)	1,024 x 2,048	No	1080Ti	78.9	26	12.87
PIDNet-S (Xu et al., 2022)	1,024 x 2,048	No	3,090	78.8	93.2	7.6
LSNet (Our)	512 x 1,024	No	1080Ti	72.6	95.6	0.62

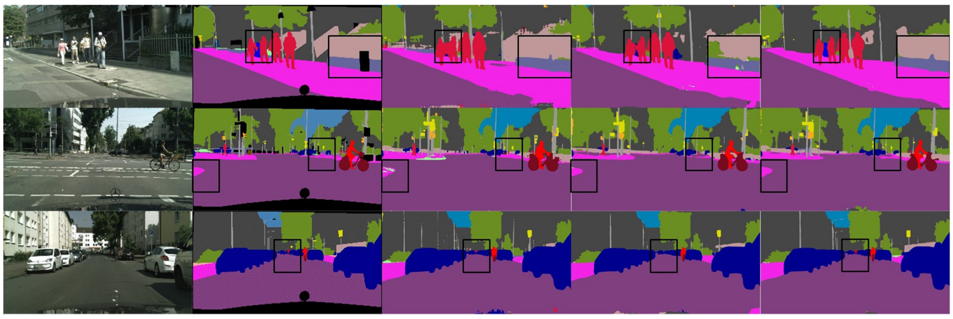


FIGURE 5
Some visual comparisons on the Cityscapes validation set. From left to right are input images, ground truth, predicted results from LEDNet, DABNet, and our lightweight semantic segmentation network (LSNet).

TABLE 5 Evaluation results of our lightweight semantic segmentation network (LSNet) and other state-of-the-art real-time semantic segmentation models on the CamVid test set.

Model	Input size	Pre-train	GPU	mIoU (%)	FPS	Params (M)
SegNet (Badrinarayanan et al., 2017)	360 x 480	ImageNet	TitanX	55.6	–	29.5
ENet (Paszke et al., 2016)	360 x 480	No	TitanX	51.3	–	0.4
ICNet (Zhao et al., 2018)	720 x 960	ImageNet	TitanX	67.1	27.8	26.5
CGNet (Wu et al., 2020)	360 x 480	No	2 x V100	65.6	–	0.5
BiSeNet (Yu et al., 2018)	720 x 960	ImageNet	TitanX	65.6	175	5.8
BiSeNetV2 (Yu et al., 2021)	720 x 960	ImageNet	TitanX	68.7	124.5	49.0
DFANet (Li H. et al., 2019)	720 x 960	ImageNet	TitanX	64.7	120	7.8
DABNet (Li G. et al., 2019)	360 x 480	No	1080Ti	66.2	124.4	0.76
LRNNet (Jiang et al., 2020)	360 x 480	No	1080Ti	67.6	83	0.67
DDPNet (Yang et al., 2020)	360 x 480	No	1080Ti	67.3	–	1.1
LEANet (Zhang et al., 2022)	360 x 480	No	1080Ti	67.5	98.6	0.74
LSNet (Our)	360 x 480	No	1080Ti	70.5	122	0.62

gradient descent method (SGD) is used, the loss function is the cross-entropy, and the learning rate update strategy uses "poly." The input image is randomly cropped, inverted, and scaled, and the scaling range is $0.75 - 2$. The initial learning rate of training Cityscapes is $1e - 2$, the weight decay is $5e - 4$, the cropping size is 512×512, and the number of input images is eight. The initial learning rate of training. Initial learning rate of CamVid is $1e - 3$, the weight decay is $5e - 4$, the cropping size is 480×360, and the number of input images is 16. Initial learning rate of ADE20K is $1.2e - 4$, the weight decay is $1e - 2$, the cropping size is 512×512, and the number of input images is eight.

4.2. Ablation study

4.2.1. Ablation study for 1D-mixer module

4.2.1.1. Ablation for typical module

We compare LEDNet's (Wang et al., 2019) encoding structure SS-nbt, DABNet's (Li G. et al., 2019) encoding structure DAB, and CGNet's (Wu et al., 2020) CG encoder with our designed 1D-mixer. We trained on the Cityscapes dataset, replacing the classic module 1D-mixer in the LSNet network. As shown in Table 2, the LSNet network with the CG module has minor parameters, but the accuracy is 8.2 mIoU lower than the network with 1D-mixer. The parameters of the remaining two modules are more than three times that of the 1D-mixer, and the accuracy is also lower than the modules we designed. Figure 4 is a feature visualization diagram of the LSNet network using the 1D-mixer module and the DAB module. Through the above comparative analysis, the 1D-mixer we designed outperforms the classic lightweight encoding modules in feature extraction and parameters.

4.2.1.2. Ablation for depth

The LSNet network contains three encoding stages, and the number of layers set in the first stage is three, which is consistent with the design of most classic lightweight networks. We experimented with the number of modules in the second and third stages of the network, hoping to find a suitable number of layers to achieve a certain balance between the segmentation accuracy and parameters of the network. As shown in Table 2, the segmentation accuracy and model parameters increase as the number of network layers increases. When the network exceeds a certain number of layers, the segmentation accuracy does not increase. We denote the number of encoders in the second stage by N, and M is the number of encoders in the third stage. When $M = 12$, the network accuracy of $N = 3$ is 0.2 mIoU higher than that of $N = 6$. The network accuracy is the highest when $N = 3$ and $M = 21$. After the above analysis, we set to $N = 3$ and $M = 21$ in Stage 2 and 3.

TABLE 6 Results of typical networks on the ADE20K validation set.

Model	Params (M)	FLOPs (G)	mIoU (%)	Latency (ms)
FCN-8s (Noh et al., 2015)	9.8	39.6	19.7	1,015
PSPNet (Zhao et al., 2017)	13.7	52.2	29.6	1,065
R-ASPP (Sandler et al., 2018)	2.2	2.8	32.0	177
Lite-ASPP (Chen et al., 2018)	2.9	4.4	36.6	235
LR-ASPP (Howard et al., 2019)	3.2	2.0	33.1	126
SegFormer (Xie et al., 2021)	3.8	8.4	37.4	770
Semantic FPN (Kirillov et al., 2019)	12.8	33.8	35.8	777
LSNet (Our)	0.65	3.8	36.4	224

All networks are trained on the server and ported to mobile devices through TNN. Latency and GFLOPs calculations take 512×512 resolution images as input. Latency measured based on a single Qualcomm Snapdragon 865 processor. All results are evaluated using a single thread.

4.2.1.3. Ablation for 1D-MS

According to the idea of MLP and global information fusion technology, we designed the 1D-MS module. The 1D-MS module plays the role of spatial feature extraction in the encoder. To explore the superiority of our designed 1D-MS block encoding, we replace 1D-MS with 3 × 3 convolution and 3 × 3 depth-wise convolution. As shown in Table 2, 3 × 3 depth-wise convolution has the same parameters as our designed 1D-MS module, but the accuracy drops by 2.8 mIoU. The 3 × 3 convolution is not as powerful as the 1D-MS module in terms of accuracy and parameters. The above experimental results prove that the encoding effect of our designed 1D-MS exceeds that of ordinary convolution.

4.2.1.4. Ablation for 1D-MC

Information fusion between channels can improve network accuracy. We design the 1D-MC module, adopting the ideas of MLP and channel selection. Ordinary channel information fusion uses 1 × 1 convolution, and here we compare 1D-MC with it. As shown in Table 2, 1D-mixer with 1 × 1 convolution has the same parameters as 1D-MC, but the accuracy is reduced by 1.2 mIoU. It can be seen from the experiments that efficient channel information fusion can improve segmentation accuracy, and our designed 1D-MC is more suitable for channel information fusion than 1 × 1 convolution.

4.2.2 Ablation study for AP-FA module

4.2.2.1 Ablation study for AP

Attention pyramid can fuse multi-scale information and perform feature screening simultaneously to improve network accuracy. We conduct ablation experiments on the AP structure, replacing the AP module with 1 × 1 convolution. As can be seen from Table 3, the accuracy of the network without the AP module drops by 2.1 mIoU. From the experiments, it can be seen that adequately designing the decoder can improve network accuracy.

4.2.2.2 Ablation study for attention

We introduced spatial attention in AP-FA; the purpose is to extract the overall structural features of the feature map and filter the features to improve the segmentation accuracy. To demonstrate the role of spatial attention in the decoder, we compare LSNet with LSNet without attention. Table 3 shows that the accuracy of the network without spatial attention drops by 0.4 mIoU. This test shows that our spatial attention branch can improve network segmentation accuracy.

4.2.2.3 Ablation study for feature pyramid

We use 3 × 1, 5 × 1, and 7 × 1 convolution and deconvolution to form a feature pyramid, the purpose of which is to increase the depth of the network and integrate contextual scale information. We designed five sets of 1D convolution, and the convolution kernel sizes are ((3, 3, 3), (1, 3, 5), (2, 3, 5), (3, 5, 7), (3, 5, 7, 9)). In order to further prove the value of the pyramid, we designed LSNet to remove the pyramid structure. It can be seen from Table 3 that introducing the pyramid structure can increase 1.7 mIoU. Comparing the experimental results of the LSNet network using these five sets of convolution kernels, the segmentation accuracy of the convolution kernel (3, 5, 7) is the highest, and it is proved that further increasing the depth of the pyramid has little effect on the segmentation accuracy.

4.2.2.4 Ablation study for FA

Since the output resolution of the encoder is smaller than the resolution of the original image, bilinear interpolation is usually used to restore the feature resolution at the end of the network. There is a problem of feature misalignment in bilinear upsampling, which affects the segmentation accuracy. We design a FA in the decoder to solve this problem. We compared bilinear interpolation with FA, and the specific results are shown in Table 3. The FA we designed is 1.8 mIoU higher than the bilinear interpolation algorithm, which shows that the design of the alignment module is effective.

4.3 Evaluation results on Cityscapes

We designed an LSNet with a parameter of 0.62 M, an inference speed of 95.6 FPS, and a segmentation accuracy of 72.6 mIoU on a 1080Ti. It can be seen from Table 4 that the network we designed has the highest accuracy among the networks with less than 1 M parameters. Under the same experimental conditions of 1080Ti, the network we designed is 69.6 FPS faster than SFNet, and the parameters are also reduced by 12.25 M. From the balance of network parameters and segmentation accuracy, the parameter expression ability of the LSNet we designed is better than that of SFNet. For PIDNet, the segmentation accuracy is 6.2 mIoU higher than LSNet, but 6.98 M increases the number of parameters. From the perspective of accuracy and parameter balance, the parameters of PIDNet are 11 times that of LSNet, but the accuracy increases very little. The network we designed has a better balance. It is worth noting that the resolution of our network input is 1, 024 × 512, and the resolution of PIDNet and SFNet input is 2, 048 × 1, 024, which is an important reason why their accuracy is higher than our network. We compare the visualization results of DABNet, LEDNet, and our designed LSNet, as shown in Figure 5.

4.4 Evaluation results on CamVid

Table 5 compares the performance of LSNet on the CamVid dataset with other models. The network we designed has the highest accuracy in the current LSNet, which is 3 mIoU higher than LEANet (Zhang et al., 2022). Without any pre-training, the LSNet network has an accuracy of 70.5 mIoU and a speed of 122 FPS. Our training is only done on a 1080Ti GPU, and the input resolution uses low-resolution images. Unlike most real-time semantic segmentation models, LSNet has apparent advantages: fewer parameters and high segmentation accuracy. Whether it is the Cityscapes or CamVid dataset, our LSNet has excellent performance and strong robustness.

4.5 Evaluation results on ADE20K

We train all networks on the server and use TNN to port the trained networks to mobile devices. The LSNet we designed and the advanced algorithm are compared on the validation dataset on ADE20K, and the latency (ms) is tested on a mobile device with a single Qualcomm Snapdragon 865 processor. The experimental results are shown in Table 6. FCN-8s, PSPNet (Zhao et al., 2017), R-ASPP (Sandler et al., 2018), and Lite-ASPP (Chen et al., 2018), use MobileV2 as the encoder. LR-ASPP (Howard et al., 2019) uses MoblieV3 as the encoder. We also compare with the advanced lightweight transformer algorithm, where SegFormer uses MiT-B0 as the encoder, and Semantic FPN (Kirillov et al., 2019) uses ConvMLP-S as the encoder. As can be seen from Table 6, LSNet and Lite-ASPP are comparable in latency and segmentation accuracy. However, LSNet has more advantages in calculation amount (GFLOPs) and parameter amount. This experiment proves that the network we designed can be used on mobile devices, and the calculation amount, parameter amount, and segmentation accuracy achieve the best balance.

5. Conclusion

In this paper, we designed a LSNet. The network's success is attributed to the combination design of 1D convolution. Our network transforms the MLP idea into a 1D convolution multi-layer combination, which solves problems where MLP is challenging to apply in semantic segmentation. At the same time, the design of the decoder increases the network's depth, solves the misalignment of upsampling, and further improves the accuracy of network segmentation. Experimental results show that our designed network achieves the best balance of accuracy and parameters, surpassing the current state-of-the-art lightweight language segmentation network. This paper shows that the proper use of multi-layer 1D convolution is more suitable for semantic segmentation than MLP. Clever decoder design is also an essential part of improving segmentation accuracy. We hope this paper encourages researchers to investigate the potential of 1D convolutions further.

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.

Author contributions

CZ, FX, CW, and CX performed the material preparation, data collection, and analysis. CZ wrote the first draft of the manuscript. All authors have study conception and design, commented on previous versions of the manuscript, read, and approved the final manuscript.

Acknowledgments

We would like to thank the reviewers for their constructive and valuable suggestions on the earlier drafts of this manuscript.

Conflict of interest

FX was employed by Shenyang Siasun Robot & Automation Company 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

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 2481–2495. doi: 10.1109/TPAMI.2016.2644615

Brempong, E. A., Kornblith, S., Chen, T., Parmar, N., Minderer, M., and Norouzi, M. (2022). "Denoising pretraining for semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, New Orleans, LA, 4175–4186. doi: 10.1109/CVPRW56347.2022.00462

Brostow, G. J., Shotton, J., Fauqueur, J., and Cipolla, R. (2008). "Segmentation and recognition using structure from motion point clouds," in *European conference on computer vision* (Berlin: Springer), 44–57. doi: 10.1007/978-3-540-88682-2_5

Chaurasia, A., and Culurciello, E. (2017). "LinkNet: Exploiting encoder representations for efficient semantic segmentation," in *Proceedings of the 2017 IEEE visual communications and image processing (VCIP)* (St. Petersburg, FL: IEEE), 1–4. doi: 10.1109/VCIP.2017.8305148

Chen, L. C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, eds V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss (Cham: Springer), 801–818. doi: 10.1007/978-3-030-01234-2_49

Chu, X., Tian, Z., Zhang, B., Wang, X., Wei, X., Xia, H., et al. (2021). Conditional positional encodings for vision transformers. *arXiv* [Preprint]. arXiv:2102.10882

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., et al. (2016). "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3213–3223. doi: 10.1109/CVPR.2016.350

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., and Houlsby, N. (2020). *An image is worth 16x16 words: Transformers for image recognition at scale.*

Fan, M., Lai, S., Huang, J., Wei, X., Chai, Z., Luo, J., et al. (2021). "Rethinking BiSeNet for real-time semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Nashville, TN, 9716–9725. doi: 10.1109/CVPR46437.2021.00959

Guo, M. H., Lu, C. Z., Liu, Z. N., Cheng, M. M., and Hu, S. M. (2022a). Visual attention network. *arXiv* [Preprint]. arXiv:2202.09741

Guo, M. H., Xu, T. X., Liu, J. J., Liu, Z. N., Jiang, P. T., Mu, T. J., et al. (2022b). Attention mechanisms in computer vision: A survey. *Comput. Vis. Media* 8, 331–368. doi: 10.1007/s41095-022-0271-y

Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., et al. (2022). "A survey on vision transformer," in *Proceedings of the IEEE transactions on pattern analysis and machine intelligence* (Piscataway, NJ: IEEE).

Han, K., Xiao, A., Wu, E., Guo, J., Xu, C., and Wang, Y. (2021). Transformer in transformer. *Adv. Neural Inf. Proc. Syst.* 34, 15908–15919.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, 770–778. doi: 10.1109/CVPR.2016.90

Howard, A., Sandler, M., Chu, G., Chen, L. C., Chen, B., Tan, M., et al. (2019). "Searching for MobileNetV3," in *Proceedings of the IEEE/CVF international conference on computer vision* (Seoul: IEEE), 1314–1324. doi: 10.1109/ICCV.2019.00140

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. *arXiv* [Preprint]. arXiv:1602.07360

Jiang, W., Xie, Z., Li, Y., Liu, C., and Lu, H. (2020). "Lrnnet: A light-weighted network with efficient reduced non-local operation for real-time semantic segmentation," in *Proceedings of the 2020 IEEE international conference on multimedia & expo workshops (ICMEW)* (Piscataway, NJ: IEEE), 1–6. doi: 10.1109/ICMEW46912.2020.9106038

Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S., and Shah, M. (2022). Transformers in vision: A survey. *ACM Comput. Surv.* 54, 1–41. doi: 10.1145/3505244

Kirillov, A., Girshick, R., He, K., and Dollár, P. (2019). "Panoptic feature pyramid networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Long Beach, CA, 6399–6408. doi: 10.1109/CVPR.2019.00656

Li, G., Yun, I., Kim, J., and Kim, J. (2019). DabNet: Depth-wise asymmetric bottleneck for real-time semantic segmentation. *arXiv* [Preprint]. arXiv:1907.11357

Li, H., Xiong, P., Fan, H., and Sun, J. (2019). "DFANet: Deep feature aggregation for real-time semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Long Beach, CA, 9522–9531. doi: 10.1109/CVPR.2019.00975

Li, X., You, A., Zhu, Z., Zhao, H., Yang, M., Yang, K., et al. (2020). "Semantic flow for fast and accurate scene parsing," in *Proceedings of the European conference on computer vision* (Cham: Springer), 775–793. doi: 10.1007/978-3-030-58452-8_45

Li, Y., Zhang, K., Cao, J., Timofte, R., and Van Gool, L. (2021). Localvit: Bringing locality to vision transformers. *arXiv* [Preprint]. arXiv:2104.05707

Liu, J., Zhou, Q., Qiang, Y., Kang, B., Wu, X., and Zheng, B. (2020). "FDDWNet: A lightweight convolutional neural network for real-time semantic segmentation," in *Proceedings of the ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (Barcelona: IEEE), 2373–2377. doi: 10.1109/ICASSP40776.2020.9053838

Lyu, H., Fu, H., Hu, X., and Liu, L. (2019). "Esnet: Edge-based segmentation network for real-time semantic segmentation in traffic scenes," in *Proceedings of the 2019 IEEE international conference on image processing (ICIP)* (Taipei: IEEE), 1855–1859. doi: 10.1109/ICIP.2019.8803132

Ma, N., Zhang, X., Zheng, H. T., and Sun, J. (2018). "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proceedings of the European conference on computer vision (ECCV)*, eds V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss (Cham: Springer), 116–131. doi: 10.1007/978-3-030-01264-9_8

Mehta, S., Rastegari, M., Caspi, A., Shapiro, L., and Hajishirzi, H. (2018). "ESPNet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, eds V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss (Cham: Springer), 552–568. doi: 10.1007/978-3-030-01249-6_34

Mehta, S., Rastegari, M., Shapiro, L., and Hajishirzi, H. (2019). "ESPNetv2: A light-weight, power efficient, and general purpose convolutional neural network," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9190–9200. doi: 10.1109/CVPR.2019.00941

Mo, Y., Wu, Y., Yang, X., Liu, F., and Liao, Y. (2022). Review the state-of-the-art technologies of semantic segmentation based on deep learning. *Neurocomputing* 493, 626–646. doi: 10.1016/j.neucom.2022.01.005

Nirkin, Y., Wolf, L., and Hassner, T. (2021). "Hyperseg: Patch-wise hypernetwork for real-time semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Nashville, TN, 4061–4070. doi: 10.1109/CVPR46437.2021.00405

Noh, H., Hong, S., and Han, B. (2015). "Learning deconvolution network for semantic segmentation," in *Proceedings of the IEEE international conference on computer vision*, Santiago, 1520–1528. doi: 10.1109/ICCV.2015.178

Paszke, A., Chaurasia, A., Kim, S., and Culurciello, E. (2016). ENet: A deep neural network architecture for real-time semantic segmentation. *arXiv* [Preprint]. arXiv:1606.02147

Poudel, R. P., Liwicki, S., and Cipolla, R. (2019). Fast-SCNN: Fast semantic segmentation network. *arXiv* [Preprint]. arXiv:1902.04502

Romera, E., Alvarez, J. M., Bergasa, L. M., and Arroyo, R. (2017). ERFNet: Efficient residual factorized convNet for real-time semantic segmentation. *IEEE Trans. Intell. Transp. Syst.* 19, 263–272. doi: 10.1109/TITS.2017.2750080

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L. C. (2018). "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Salt Lake City, UT, 4510–4520. doi: 10.1109/CVPR.2018.00474

Sheng, H., Cong, R., Yang, D., Chen, R., Wang, S., and Cui, Z. (2022). UrbanLF: A comprehensive light field dataset for semantic segmentation of urban scenes. *IEEE Trans. Circuits Syst. Video Technol.* 32, 7880–7893. doi: 10.1109/TCSVT.2022.3187664

Tan, M., and Le, Q. (2019). "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th international conference on machine learning, ICML 2019* (Long Beach, CA: PMLR), 6105–6114.

Treml, M., Arjona-Medina, J., Unterthiner, T., Durgesh, R., Friedmann, F., Schuberth, P., et al. (2016). *Speeding up semantic segmentation for autonomous driving*.

Ulku, I., and Akagündüz, E. (2022). A survey on deep learning-based architectures for semantic segmentation on 2D images. *Appl. Artif. Intell.* 1–45. doi: 10.1080/08839514.2022.2032924

Wang, Y., Zhou, Q., Liu, J., Xiong, J., Gao, G., Wu, X., et al. (2019). "LEDNet: A lightweight encoder-decoder network for real-time semantic segmentation," in *Proceedings of the 2019 IEEE international conference on image processing (ICIP)* (Piscataway, NJ: IEEE), 1860–1864. doi: 10.1109/ICIP.2019.8803154

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). "CBAM: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, eds V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss (Cham: Springer), 3–19. doi: 10.1007/978-3-030-01234-2_1

Wu, T., Tang, S., Zhang, R., Cao, J., and Zhang, Y. (2020). CGNet: A light-weight context guided network for semantic segmentation. *IEEE Trans. Image Process.* 30, 1169–1179. doi: 10.1109/TIP.2020.3042065

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., and Luo, P. (2021). SegFormer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inf. Process. Syst.* 34, 12077–12090.

Xu, J., Xiong, Z., and Bhattacharyya, S. P. (2022). PIDNet: A real-time semantic segmentation network inspired from PID controller. *arXiv* [Preprint]. arXiv:2206.02066

Yang, X., Wu, Y., Zhao, J., and Liu, F. (2020). "Dense dual-path network for real-time semantic segmentation," in *Proceedings of the Asian conference on computer vision*, Kyoto.

Yu, C., Gao, C., Wang, J., Yu, G., Shen, C., and Sang, N. (2021). BiSeNet V2: Bilateral network with guided aggregation for real-time semantic segmentation. *Int. J. Comput. Vis.* 129, 3051–3068. doi: 10.1007/s11263-021-01515-2

Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., and Sang, N. (2018). "BiSeNet: Bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, eds V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss (Cham: Springer), 325–341. doi: 10.1007/978-3-030-01261-8_20

Zhang, X. L., Du, B. C., Luo, Z. C., and Ma, K. (2022). Lightweight and efficient asymmetric network design for real-time semantic segmentation. *Appl. Intell.* 52, 564–579. doi: 10.1007/s10489-021-02437-9

Zhao, H., Qi, X., Shen, X., Shi, J., and Jia, J. (2018). "ICNet for real-time semantic segmentation on high-resolution images," in *Proceedings of the European conference on computer vision (ECCV)*, Munich, 405–420. doi: 10.1007/978-3-030-01219-9_25

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2881–2890. doi: 10.1109/CVPR.2017.660

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., et al. (2021). "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, Nashville, TN, 6881–6890. doi: 10.1109/CVPR46437.2021.00681

TYPE Original Research
PUBLISHED 09 February 2023
DOI 10.3389/fnbot.2023.1129720

Check for updates

OPEN ACCESS

EDITED BY
Di Wu,
Chongqing Institute of Green and Intelligent Technology (CAS), China

REVIEWED BY
Ye Yuan,
Southwest University, China
Linyuan Wang,
National Digital Switching System Engineering and Technological Research Centre, China
Krishnaraj Nagappan,
SRM Institute of Science and Technology, India

*CORRESPONDENCE
Wei Zhou
✉ zwei@ynu.edu.cn

RECEIVED 22 December 2022
ACCEPTED 24 January 2023
PUBLISHED 09 February 2023

CITATION
Liu R, Jin X, Hu D, Zhang J, Wang Y, Zhang J and Zhou W (2023) DualFlow: Generating imperceptible adversarial examples by flow field and normalize flow-based model. *Front. Neurorobot.* 17:1129720. doi: 10.3389/fnbot.2023.1129720

DualFlow: Generating imperceptible adversarial examples by flow field and normalize flow-based model

Renyang Liu[1,2], Xin Jin[2,3], Dongting Hu[4], Jinhong Zhang[2,3], Yuanyu Wang[5], Jin Zhang[5] and Wei Zhou[2,3]*

[1]School of Information Science and Engineering, Yunnan University, Kunming, China, [2]Engineering Research Center of Cyberspace, Yunnan University, Kunming, China, [3]National Pilot School of Software, Yunnan University, Kunming, China, [4]School of Mathematics and Statistics, University of Melbourne, Melbourne, VIC, Australia, [5]Kunming Institute of Physics, Yunnan University, Kunming, China

Recent adversarial attack research reveals the vulnerability of learning-based deep learning models (DNN) against well-designed perturbations. However, most existing attack methods have inherent limitations in image quality as they rely on a relatively loose noise budget, i.e., limit the perturbations by L_p-norm. Resulting that the perturbations generated by these methods can be easily detected by defense mechanisms and are easily perceptible to the human visual system (HVS). To circumvent the former problem, we propose a novel framework, called **DualFlow**, to craft adversarial examples by disturbing the image's latent representations with spatial transform techniques. In this way, we are able to fool classifiers with human imperceptible adversarial examples and step forward in exploring the existing DNN's fragility. For imperceptibility, we introduce the flow-based model and spatial transform strategy to ensure the calculated adversarial examples are perceptually distinguishable from the original clean images. Extensive experiments on three computer vision benchmark datasets (CIFAR-10, CIFAR-100 and ImageNet) indicate that our method can yield superior attack performance in most situations. Additionally, the visualization results and quantitative performance (in terms of six different metrics) show that the proposed method can generate more imperceptible adversarial examples than the existing imperceptible attack methods.

KEYWORDS

deep learning, adversarial attack, adversarial example, normalize flow, spatial transform

1. Introduction

Deep neural networks (DNNs) have achieved remarkable achievements in theories and applications. However, the DNNs have been proven to be easily fooled by adversarial examples (AEs), which are generated by adding well-designed unwanted perturbations to the original clean data (Zhou et al., 2019). In these years, many studies dabbled in crafting adversarial examples and revealed that many DNN applications are vulnerable to them. Such as Computer Vision (CV) (Kurakin et al., 2017; Eykholt et al., 2018; Duan et al., 2020), Neural Language Processing (NLP) (Xu H. et al., 2020; Shao et al., 2022; Yi et al., 2022), and Autonomous Driving (Liu A. et al., 2019; Zhao et al., 2019; Yan et al., 2022). Generally, in CV, the AE needs to meet the following two properties, one is that it can attack the target model successfully, resulting in the target model outputting wrong predictions; another one is its perturbations should be invisible to human eyes (Goodfellow et al., 2015; Carlini and Wagner, 2017).

Unfortunately, most existing works (Kurakin et al., 2017; Dong et al., 2018, 2019) are focused on promoting the generated adversarial examples' attack ability but ignored the visual aspects of the crafted evil examples. Typically, the calculated adversarial noise is limited by a small L_p-norm ball, which tries to keep the built adversarial examples looking like the original image as possible. However, the L_p-norm limited adversarial perturbations blur the images to a large extent and are so conspicuous to human eyes and not harmonious with the whole image. Furthermore, these L_p-norm-based methods, which modify the entire image at the pixel level, seriously affect the quality of the generated adversarial images. Resulting in the vivid details of the original image can not be preserved. Besides, the adversarial examples crafted in these settings can be easily detected by the defense mechanism or immediately discarded by the target model and further encounter the "denied to service." All the mentioned above can lead the attack to be failed. Furthermore, most existing methods adopt L_p-norm, i.e., L_2 and L_{inf}-norm, distance as the metrics to constraint the image's distortion. Indeed, the L_p-norm can ensure the similarity between the clean and adversarial images. However, it does not perform well in evaluating an adversarial example.

Recently, some studies have attempted to generate adversarial examples beyond the L_p-norm ball limited way. For instance, patch-based adversarial attacks, which usually extend into the physical world, do not limit the intensity of perturbation but the range scope. Such as adversarial-Yolo (Thys et al., 2019), DPatch (Liu X. et al., 2019), AdvCam (Duan et al., 2020), Sparse-RS (Croce et al., 2022). To obtain more human harmonious adversarial examples with acceptable attack success rate in the digital world, Xiao et al. (2018) proposed the stAdv to generate adversarial examples by spatial transform to modify each pixel's position in the whole image. The overall visual effect of the adversarial example generated by stAdv is good. However, the adversarial examples generated by stAdv usually have serration modifications and are visible to the naked eye. Later, the Chroma-Shift (Aydin et al., 2021) made a forward step by applying the spatial transform to the image's YUV space rather than RGB space. Unfortunately, these attacks have destroyed the semantic information and data distribution of the image, resulting that the generated adversarial noise that can be easily detected by the defense mechanism (Arvinte et al., 2020; Xu Z. et al., 2020; Besnier et al., 2021) and leading the attack failed.

To gap this bridge, we formulate the issue of synthesizing invisible adversarial examples beyond noise-adding at pixel level and propose a novel attack method called **DualFlow**. More specifically, DualFlow uses spatial transform techniques to disturb the latent representation of the image rather than directly adding well-designed noise to the benign image, which can significantly improve the adversarial noise's concealment and preserve the adversarial examples' vivid details at the same time. The spatial transform can learn a smooth flow field vector $\boldsymbol{f}$ for each value's new location in the latent space to optimize an eligible adversarial example. Furthermore, the adversarial examples are not limited to L_p-norm rules, which can guarantee the image quality and details of the generated examples. Empirically, the proposed DualFlow can remarkably preserve the images' vivid details while achieving an admirable attack success rate.

We conduct extensive experiments on three different computer vision benchmark datasets. Results illustrate that the adversarial perturbations generated by the proposed method take into account the data structure and only appear around the target object. We draw the adversarial examples and their corresponding noise from the noise-adding method MI-FGSM and the DualFlow in Figure 1. As shown in Figure 1, our proposed method slightly alters this area around the target object, thus ensuring the invisibility of the adversarial perturbations. Furthermore, the statistical results demonstrate that the DualFlow can guarantee the generated adversarial examples' image quality compared to the existing imperceptible attack methods on the target models while outperforming them both on the ordinary and defense models concerning attack success rate. The main contributions could be summarized as follows:

- We propose a novel attack method, named DualFlow, which generates adversarial examples by directly disturbing the latent representation of the clean examples rather than performing an attack on the pixel level.
- We craft the adversarial examples by applying the spatial transform techniques to the latent value to preserve the details of original images and guarantee the adversarial images' quality.
- Comparing with the existing attack methods, experimental results show our method's superiority in synthesizing adversarial examples with the highest attack ability, best invisibility, and remarkable image quality.

The rest of this paper is organized as follows. First, we briefly review the methods relating to adversarial attacks and imperceptible adversarial attacks in Section 2. Then, Sections 3 and 4, introduce the preliminary knowledge and the details of the proposed DualFlow framework. Finally, the experimental results are presented in Section 5, with the conclusion drawn in Section 6.

2. Related work

In this section, we briefly review the most pertinent attack methods to the proposed work: the adversarial attacks and the techniques used for crafting inconspicuous adversarial perturbations.

2.1. Adversarial attack

Previous researchers contend that deep neural networks (DNN) are sensitive to adversarial examples (Goodfellow et al., 2015), which are crafted by disturbing the clean data slightly but can fool the well-trained DNN models. The classical adversarial attack methods can be classified into two categories, white-box attacks (Kurakin et al., 2017; Madry et al., 2018) and black-box attacks (Narodytska and Kasiviswanathan, 2017; Bai et al., 2023). In white-box settings, the attackers can generate adversarial examples with a nearly 100% attack success rate because they can access the complete information of the target DNN model, while for the physical world, the black-box attack is more threatening to the DNN applications because they don't need too much information about the DNN models' details (Ilyas et al., 2018, 2019; Guo et al., 2019).

2.2. Imperceptible adversarial attacks

Recently, some studies have attempted to generate adversarial examples beyond the L_p-norm ball limit for obtaining humanly imperceptible adversarial examples. LowProFool (Ballet et al., 2019) propose an imperceptibility attack to craft invisible adversarial

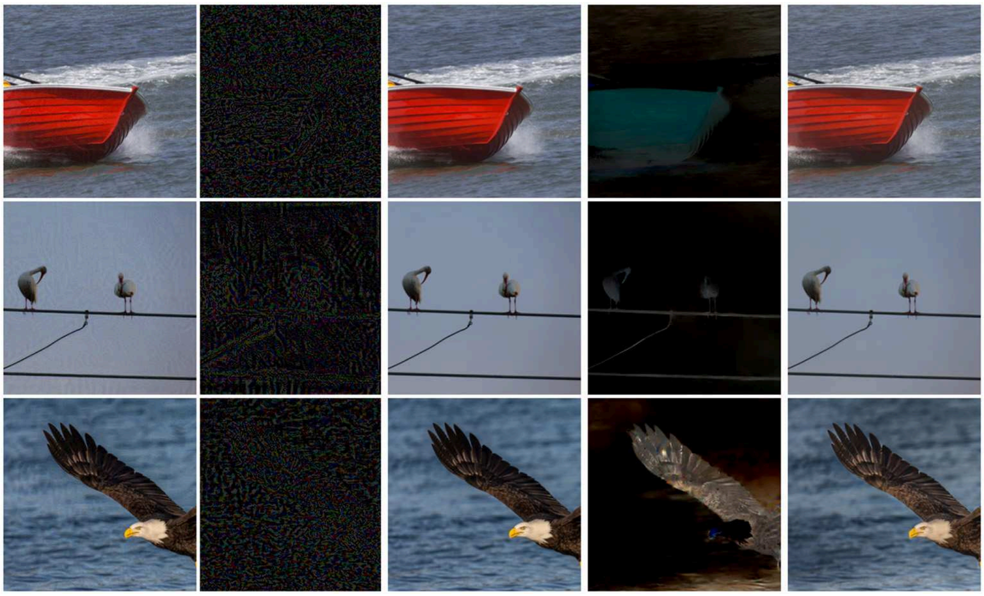


FIGURE 1
The adversarial examples generated by the MI-FGSM (Aydin et al., 2021) and the proposed DualFlow for the ResNet-152 (He et al., 2016) model. Specifically, the first column and the second column are the adversarial examples and their corresponding adversarial perturbations generated by MI-FGSM, respectively. The middle column is the clean images. The last two columns are the adversarial perturbations and their corresponding adversarial examples, respectively.

examples in the tabular domain. Its empirical results show that LowProFool can generate imperceptible adversarial examples while keeping a high fooling rate. For computer vision tasks the attackers will also consider the human perception of the generated adversarial examples. In Luo et al. (2018), the authors propose a new approach to craft adversarial examples, which design a new distance metric that considers the human perceptual system and maximizes the noise tolerance of the generated adversarial examples. This metric evaluates the sensitivity of image pixels to the human eye and can ensure that the crafted adversarial examples are highly imperceptible and robust to the physical world. stAdv (Xiao et al., 2018) focuses on generating different adversarial perturbations through spatial transform and claims that such adversarial examples are perceptually realistic and more challenging to defend against with existing defense systems. Later, the Chroma-Shift (Aydin et al., 2021) made a forward step by applying the spatial transform to the image's YUV space rather than RGB space. AdvCam (Duan et al., 2020) crafts and disguises adversarial examples of the physical world into natural styles to make them appear legitimate to a human observer. It transfers large adversarial perturbations into a custom style and then “hides" them in a background other than the target object. Moreover, its experimental results that AEs produced by AdvCam are well camouflaged and highly concealed in both digital and physical world scenarios while still being effective in deceiving state-of-the-art DNN image detectors. SSAH (Luo et al., 2022) crafts adversarial examples and disguises adversarial noise in a low-frequency constraints manner. This method limits the adversarial perturbations to the high-frequency components of the specific image to ensure low human perceptual similarity. The SSAH also jumps out of the original L_p-norm constraint-based attack way and provides a new idea for calculating adversarial noise.

Therefore, crafting adversarial examples, especially for the imperceptible ones, poses the request for a method that can efficiently and effectively build adversarial examples with high invisibility and image quality efficiently and effectively. On the other hand, with the development of defense mechanisms, higher requirements are placed on the defense resistance of adversarial examples. To achieve these goals, we learn from the previous studies that adversarial examples can be gained beyond noise-adding ways. Hence, we are well motivated to develop a novel method to disturb the original image latent representation obtained by a well-trained normalizing flow-based model, and then apply a well-calculated flow field to it to generate adversarial examples. Our method can build adversarial examples with high invisibility and image quality without losing attack performance.

3. Preliminary

Before introducing the details of the proposed framework, in this section, we first present the preliminary knowledge about adversarial attacks and normalizing flows.

3.1. Adversarial attack

Given a well-trained DNN classifier $\mathcal{C}$ and a correctly classified input $(x, y) \sim D$, we have $\mathcal{C}(x) = y$, where D denotes the accessible dataset. The adversarial example x_{adv} is a neighbor of x and satisfies that $\mathcal{C}(x_{adv}) \neq y$ and $\|x_{adv} - x\|_p \leq \epsilon$, where the ℓ_p norm is used as the metric function and ϵ is usually a small value such as 8 and 16 with the image intensity [0, 255]. With this definition, the problem of calculating an adversarial example becomes a constrained optimization problem:

$$x_{adv} = \underset{\|x_{adv}-x\|_p \leq \epsilon}{arg\ max}\, \ell\,(\mathcal{C}(x_{adv}) \neq y), \tag{1}$$

Where $\boldsymbol{\ell}$ stands for a loss function that measures the confidence of the model outputs.

In the optimization-based methods, the above problem is solved by computing the gradients of the loss function in Equation (1) to generate the adversarial example. Furthermore, most traditional attack methods craft adversarial examples by optimizing a noise δ and adding it to the clean image, i.e., $x_{adv} = x + \delta$. By contrast, in this work, we formulate the x_{adv} by disturbing the image's latent representation with spatial transform techniques.

3.2. Normalizing flow

The normalizing flows (Dinh et al., 2015; Kingma and Dhariwal, 2018; Xu H. et al., 2020) are a class of probabilistic generative models, which are constructed based on a series of entirely reversible components. The reversible property allows to transform from the original distribution to a new one and vice versa. By optimizing the model, a simple distribution (such as the Gaussian distribution) can be transformed into a complex distribution of real data. The training process of normalizing flows is indeed an explicit likelihood maximization. Considering that the model is expressed by a fully invertible and differentiable function that transfers a random vector z from the Gaussian distribution to another vector x, we can employ such a model to generate high dimensional and complex data.

⊮ Specifically, given a reversible function $\boldsymbol{F}: \mathbb{R}^d \rightarrow \mathbb{R}^d$ and two random variables $z \sim p(z)$ and $z' \sim p(z')$ where $z' = f(z)$, the change of variable rule tells that

$$p(z') = p(z) \left| det \frac{\partial \boldsymbol{F}^{-1}}{\partial z'} \right|, \tag{2}$$

$$p(z) = p(z') \left| det \frac{\partial \boldsymbol{F}}{\partial z} \right|, \tag{3}$$

Where *det* denotes the determinant operation. The above equation follows a chaining rule, in which a series of invertible mappings can be chained to approximate a sufficiently complex distribution, i.e.,

$$z_K = \boldsymbol{F}_K \odot \ldots \odot \boldsymbol{F}_2 \odot \boldsymbol{F}_1(z_0), \tag{4}$$

Where each $\boldsymbol{F}$ is a reversible function called a flow step. Equation (4) is the shorthand of $\boldsymbol{F}_K(\boldsymbol{F}_{k-1}(\ldots \boldsymbol{F}_1(x)))$. Assuming that x is the observed example and z is the hidden representation, we write the generative process as

$$x = \boldsymbol{F}_\theta(z), \tag{5}$$

Where $\boldsymbol{F}_\theta$ is the accumulate sum of all $\boldsymbol{F}$ in Equation (4). Based on the change-of-variables theorem, we write the log-density function of $x = z_K$ as follows:

$$-\log p_K(z_K) = -\log p_0(z_0) - \sum_{k=1}^{K} \log \left| det \frac{\partial z_{k-1}}{\partial z_k} \right|, \tag{6}$$

Where we use $z_k = \boldsymbol{F}_k(z_{k-1})$ implicitly. The training process of normalizing flow is minimizing the above function, which exactly maximizes the likelihood of the observed training data. Hence, the optimization is stable and easy to implement.

TABLE 1 The notations used in this paper.

x	clean example	$\mathcal{C}$	the classifier	z_{adv}	the disturbed latent value
x_{adv}	adversarial example	$\mathcal{L}$	loss function	δ	the noise
y	clean label	$\boldsymbol{F}$	Pretrained Flow Model	$\boldsymbol{f}$	the flow field
t	the target label	z	the latent value	$\mathcal{N}(\cdot)$	the four neighborhood

3.3. Spatial transform

The concept of spatial transform is firstly mentioned in Fawzi and Frossard (2015), which indicates that the conventional neural networks are not robust to rotation, translation and dilation. Next, Xiao et al. (2018) utilized the spatial transform techniques and proposed the stAdv to craft adversarial examples with a high fooling rate and perceptually realistic beyond noise-adding way. StAdv changes each pixel position in the clean image by applying a well-optimized flow field matrix to the original image. Later, Zhang et al. (2020) proposed a new method to produce the universal adversarial examples by combining the spatial transform and pixel distortion, and it successfully increased the attack success rate against universal perturbation to more than 90%. In the literature (Aydin et al., 2021), the authors applied spatial transform to the YUV space to generate adversarial examples with higher superiority in image quality.

We summarized the adopted symbols in Table 1 to increase the readability.

4. Methodology

In this section, we propose our attack method. First, we take an overview of our method. Next, we go over the detail of each part step by step. Finally, we discuss our objective function and summarize the whole process as Algorithm 1.

4.1. The DualFlow framework

The proposed DualFlow attack framework can be divided into three parts, the first one is to map clean image x to its latent space z by the well-trained normalizing flow model. The second part is to optimize the flow field f, and apply it to the images' latent representation z and inverse the transformed z to generate its corresponding RGB space counterpart x_t. Note that step 2 needs to be worked in an iterative manner to update the flow field f guided by the adv_loss until the adversarial candidate x_t can fool the target model. Finally, apply the optimized flow field $\boldsymbol{f}$ to the image's latent counterpart z and do the inverse operation of normalizing flow to obtain the adversarial image. The whole process is shown in Figure 2.

4.2. Normalizing flow model training

As introduced in Section 3.2., the training of the normalizing flow is to maximize the likelihood function on the training data

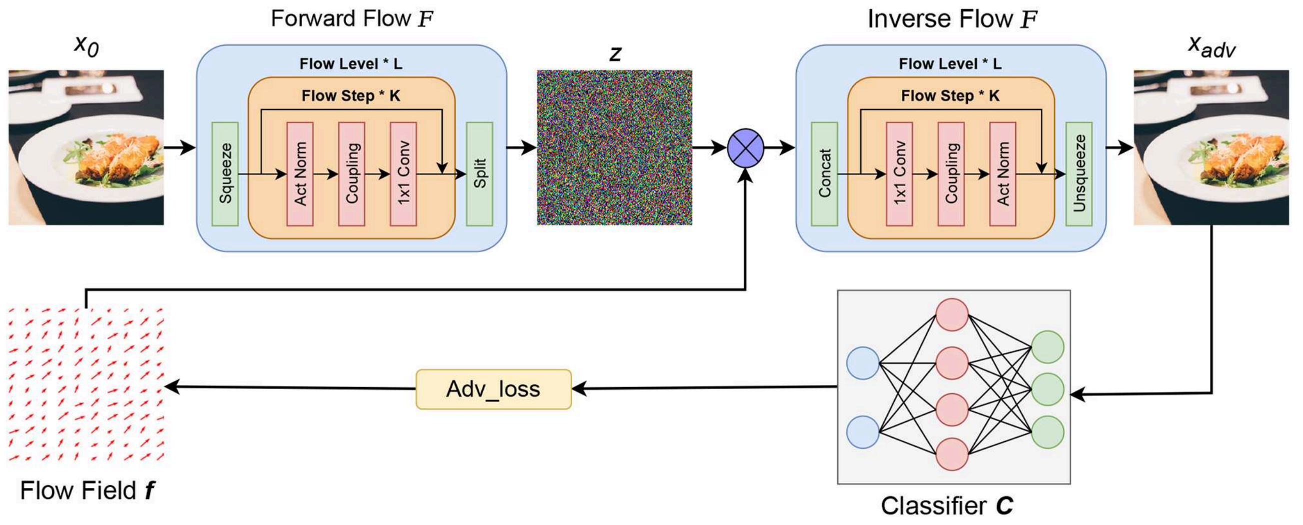


FIGURE 2
The framework of proposed DualFlow. x represent the image, among them, x_0 is the benign image, x_{adv} is the corresponding adversarial counterpart; z is the hidden representation of the image; $\boldsymbol{F}$ is the well-trained Normalize Flow model and $\mathcal{C}$ is the pre-trained classifier; f is the flow field need to be optimized and ⊗ represents the spatial transform operation.

with respect to the model parameters. Formally, assume that the collected dataset is denoted by $x \sim X$. The hidden representation follows the Gaussian distribution, i.e., $z \sim \mathcal{N}(0, 1)$. The flow model is denoted by $\boldsymbol{F}$, parameterized θ, which have $x = \boldsymbol{F}_\theta(z)$ and $z = \boldsymbol{F}^{-1}(x)$. Then, the loss function to be minimized is expressed as:

$$L(\theta; z, x) = -\log p(x|z, \theta) = -\log p_z(\boldsymbol{F}_\theta^{-1}(x)) - \log \left| det \frac{\partial \boldsymbol{F}_\theta^{-1}(x)}{\partial x} \right|, \tag{7}$$

By optimizing the above objective, the learned distribution $p(x|z, \theta)$ characterizes the data distribution as expected.

In the training process, we use the Adam algorithm to optimize the model parameters; while the learning rate is set as 10^{-4}, the momentum is set to 0.999, and the maximal iteration number is 100,000.

4.3. Generating adversarial examples with DualFlow

For a clean image x, to obtain its corresponding adversarial example x_{adv}, we first calculate its corresponding latent space vector z by performing a forward flow process *via* $z = \boldsymbol{F}_\theta(x)$. Once the z is calculated, we can disturb it with the spatial transform techniques, the core is to optimize the flow filed vector $\boldsymbol{f}$, which will be applied to z to get the transformed latent representation z_{st} according to x. In this paper, the flow filed vector f is directly optimized with the Adam optimizer iteratively. We will repeat the above process to optimize flow field $\boldsymbol{f}$ until z_{st} becomes an eligible adversarial latent value, that is, make the z_{st} becomes z_{adv}. Finally, when the optimal flow filed $\boldsymbol{f}$ is calculated, we restore the transformed latent representation z_{adv} to the image space through the inverse operation of the normalizing flow model, that is, $x_{adv} = \boldsymbol{F}_\theta(z_{adv})$, to get its perturbed example x_{adv} in pixel level.

Moore specifically, the spatial transform techniques using a flow field matrix $\boldsymbol{f} = [2, h, w]$ to transform the original image x to x_{st} (Xiao et al., 2018). In this paper, we adopt the spatial transform from the pixel level to the latent space. Specifically, assume the latent representation of input x is z and its transformed counterpart z_{st}, for the i-th value in z_{st} at the value location (u_{st}^i, v_{st}^i), we need to calculate the flow field matrix $\boldsymbol{f}_i = (\Delta u^i, \Delta v^i)$. So, the i-th value z^i's location in the transformed image can be indicated as:

$$(u^i, v^i) = (u_{st}^i + \Delta u^i, v_{st}^i + \Delta v^i). \tag{8}$$

To ensure the flow field $\boldsymbol{f}$ is differentiable, the bi-linear interpolation (Jaderberg et al., 2015) is used to obtain the four neighboring values surrounding the location $(u_{st}^i + \Delta u^i, v_{st}^i + \Delta v^i)$ for the transformed latent value z_{st} as:

$$z_{st}^i = \sum_{q \in \mathcal{N}(u^i, v^i)} z^q (1 - |u^i - u^q|)(1 - |v^i - v^q|), \tag{9}$$

Where $\mathcal{N}(u^i, v^i)$ is the neighborhood, that is, the four positions (top-left, top-right, bottom-left, bottom-right) tightly surrounding the target value (u^i, v^i). In our adversarial attack settings, the calculated z_{st} is the final adversarial latent representation z_{adv}. Once the $\boldsymbol{f}$ has been computed, we can obtain the z_{adv} by applying the calculated flow field $\boldsymbol{f}$ to the original z, which is given by:

$$z_{adv} = \sum_{q \in \mathcal{N}(u^i, v^i)} z^q (1 - |u^i - u^q|)(1 - |v^i - v^q|)), \tag{10}$$

and the adversarial examples x_{adv} can be obtained by:

$$x_{adv} = clip(\boldsymbol{F}^{-1}(z_{adv}), 0, 1), \tag{11}$$

Where $clip(\cdot)$ is the clip operation to keep the generated value belonging to $[0, 1]$.

4.4. Objective functions

Taking the attack success rate and visual invisibility of the generated adversarial examples into account, we divide the objective function into two parts, where one is the adversarial loss and the other is a constraint for the flow field. Unlike other flow field-based attack methods, which constrain the flow field by the flow loss proposed in Xiao et al. (2018), in our method, we use a dynamically updated flow field budget ξ (a small number, like $1 * 10^{-3}$) to regularize the flow field $\boldsymbol{f}$. For adversarial attacks, the goal is making $\mathcal{C}(x_{adv}) \neq y$. We give the objective function as follows:

for un-targeted attacks:

$$\mathcal{L}_{adv}(X, y, \boldsymbol{f}) = max[\mathcal{C}(X_{adv})_y - \underset{k\neq y}{max}\mathcal{C}(X_{adv})_k, k], \qquad s.t.\|\boldsymbol{f}\| \leq \xi. \tag{12}$$

for target attacks:

$$\mathcal{L}_{adv}(X, y, t, \boldsymbol{f}) = min[\underset{k=t}{max}\mathcal{C}(X_{adv})_k - \mathcal{C}(X_{adv})_y, k], \qquad s.t.\|\boldsymbol{f}\| \leq \xi. \tag{13}$$

The whole algorithm of LFFA is listed in Algorithm 1 for easy reproducing of our results, where lines 11-18 depict the core optimization process.

```
Input: $X_{tr}$: a batch of clean examples used for training;
    $\alpha$: the learning rate; $T$: the maximal training
    iterations; $Q$: the maximal steps for attack; $\xi$:
    the flow budget; $X_{te}$: a clean example used for test;
    $\mathcal{C}$: the target model to be attacked.
Output: The adversarial example $x_{adv}$ is used for attack.
Parameter: The flow model $\boldsymbol{F}_\theta$.
 1: Initialize the parameters of the flow model $\boldsymbol{F}_\theta$;
 2: for $i = 1$ to $T$ do
 3:    Optimize $\boldsymbol{F}_\theta$ according to Equation (6);
 4:    if Convergence reached then
 5:       break;
 6:    end if
 7: end for
 8: Obtain optimized $\boldsymbol{F}_\theta$;
 9: Compute the hidden representation of examples in $X_{te}$
    via $z = \boldsymbol{F}^{-1}(x_{te})$;
10: $z_0^{'} = z$
11: Initialize the flow filed $f$ with zeros;
12: for $i = 1$ to $Q$ do
13:    Optimize $f$ via Equations (12) or 13;
14:    Compute the adversarial example candidate $x_i^{'}$ via
    Equation (11);
15:    if Successfully attack $\mathcal{C}$ by $x_i^{'}$ then
16:       $x_{adv} = x_i^{'}$
17:       break.
18:    end if
19: end for
```

Algorithm 1. DualFlow attack.

5. Experiments

In this section, we evaluate the proposed DualFlow on three benchmark image classification datasets. We first compare our proposed method with several baseline techniques concerned with Attack Success Rate (ASR) on clean models and robust models on three CV baseline datasets (CIFAR-10, CIFAR-100 and ImageNet). Then, we first provide a comparative experiment to the existing attack methods in image quality aspects with regard to LPIPS, DISTS, SCC, SSIM, VIPF and et al. Through these experimental results, we show the superiority of our method in attack ability, human inception and image quality.

5.1. Settings

Dataset

We verify the performance of our method on three benchmark datasets for computer vision task, named CIFAR-10[1] (Krizhevsky and Hinton, 2009), CIFAR-100[1] (Krizhevsky and Hinton, 2009) and ImageNet-1k[2] (Deng et al., 2009). In detail, CIFAR-10 contains 50,000 training images and 10,000 testing images with the size of 3x32x32 from 10 classes; CIFAR-100 has 100 classes, including the same number of training and testing images as the CIFAR-10; ImageNet-1K has 1,000 categories, containing about 1.3M samples for training and 50,000 samples for validation. In particular, in this paper, we extend our attack on the whole images in testing datasets of CIFAR-10 and CIFAR-100, in terms of ImageNet-1k, we are using its subset datasets from ImageNet Adversarial Learning Challenge, which is commonly used in work related to adversarial attacks.

All the experiments are conducted on a GPU server with 4 * Tesla A100 40GB GPU, 2 * Xeon Glod 6112 CPU, and RAM 512GB.

Models

For CIFAR-10 and CIFAR-100, the pre-trained VGG-19 (Simonyan and Zisserman, 2015), ResNet-56 (He et al., 2016), MobileNet-V2 (Sandler et al., 2018) and ShuffleNet-V2 (Ma N. et al., 2018) are adopted, with top-1 classification accuracy 93.91, 94.37, 93.91, and 93.98% on CIFAR-10 and 73.87, 72.60, 71.13, and 75.49% on CIFAR-100, respectively, all the models' parameters are provided in the GitHub Repository[3]. For ImageNet, we use the PyTorch pre-trained clean model VGG-16, VGG-19 (Simonyan and Zisserman, 2015), ResNet-152 (He et al., 2016), MobileNet-V2 (Sandler et al., 2018) and DenseNet-121 (Huang et al., 2017), achieving 87.40, 89.00, 94.40, 87.80, and 91.60% classification accuracy rate on ImageNet, respectively. And in terms of robust models, they include Hendrycks2019Using (Hendrycks et al., 2019), Wu2020Adversarial (Wu et al., 2020), Chen2020Efficient (Chen et al., 2022) and Rice2020Overfitting (Rice et al., 2020) for CIFAR-10 and CIFAR-100, And Engstrom2019Robustness (Croce et al., 2021), Salman2020Do_R18 (Salman et al., 2020), Salman2020Do_R50 (Salman et al., 2020), and Wong2020Fast (Wong et al., 2020) for

[1] http://www.cs.toronto.edu/~kriz/cifar.html

[2] https://image-net.org/

[3] https://github.com/chenyaofo/pytorch-cifar-models

ImageNet. All the models we use are implemented in the robustbench toolbox[4] (Croce et al., 2021) and the models' parameters are also provided in Croce et al. (2021). For all these models, we chose their L_{inf} version parameters due to most baselines being extended L_{inf} attacks in this paper.

Baselines

The baseline methods are FGSM (Goodfellow et al., 2015), MI-FGSM (Dong et al., 2018), TI-FGSM (Dong et al., 2019), Jitter (Schwinn et al., 2021), stAdv (Xiao et al., 2018), Chroma-Shift (Aydin et al., 2021), and GUAP (Zhang et al., 2020). The experimental results of those methods are reproduced by the Torchattacks toolkit[5] and the code provided by the authors with default settings.

Metrics

Unlike the pixel-based attack methods, which only use L_p norm to evaluate the adversarial examples' perceptual similarity to its corresponding benign image. The adversarial examples generated by spatial transform always use other metrics referring to image quality. To be exact, in this paper, we follow the work in Aydin et al. (2021) using the following perceptual metrics to evaluate the adversarial examples generated by our method, including Learned Perceptual Image Patch Similarity (LPIPS) metric (Zhang et al., 2018) and Deep Image Structure and Texture Similarity (DISTS) index (Ding et al., 2022). LPIPS is a technique that measures the Euclidean distance of deep representations (i.e., VGG network Simonyan and Zisserman, 2015) calibrated by human perception. LPIPS has already been used on spatially transformed adversarial examples generating studies (Jordan et al., 2019; Laidlaw and Feizi, 2019; Aydin et al., 2021). DISTS is a method that combines texture similarity with structure similarity (i.e., feature maps) using deep networks with the optimization of human perception. We used the implementation of Ding et al. for both perceptual metrics (Ding et al., 2021). Moreover, we use other metrics like Spatial Correlation Coefficient (SCC) (Li, 2000), Structure Similarity Index Measure (SSIM) and Pixel Based Visual Information Fidelity (VIFP) (Sheikh and Bovik, 2004) to assess the generated images' qualities. SCC reflects the indirect correlation based on the spatial contiguity between any two geographical entities. SSIM is used to assess the generated images' qualities concerning luminance, contrast and structure. VIFP is used to assess the adversarial examples' image quality. The primary toolkits we used in the experiments of this part are IQA_pytorch[6] and sewar[7].

5.2. Quantitative comparison with the existing attacks

In this subsection, we will evaluate the proposed DualFlow and the baselines FGSM, MI-FGSM, TI-FGSM (Dong et al., 2019), Jitter, stAdv, Chroma-shift and GUAP in attack success rate on CIFAR-10, CIFAR-100 and the whole ImageNet dataset. We set the noise budget as $\epsilon = 0.031$ for all L_{inf}-based attacks baseline methods. The other attack methods, such as stAdv and Chroma-shift, follow their default settings in the code provided by the authors.

Tables 2–4 show the ASR of DualFlow and the baselines on CIFAR-10, CIFAR-100 and ImageNet, respectively. As the results illustrated, DualFlow can perform better in most situations on the three benchmark datasets. Take the attack results on ImageNet as an example, refer to Table 3. The BIM, MI-FGSM, TI-FGSM, Jitter, stAdv, Chroma-shift and GUAP can achieve 91.954, 98.556, 93.94, 95.172, 97.356, 98.678, and 94.606% average attack success rate on ImageNet dataset, respectively, vice versa, our DualFlow can achieve 99.364% average attack success rate. On the other two benchmark datasets, CIFAR-10 and CIFAR-100, the DualFlow also can get a better average attack performance. To further explore the attack performance of the proposed DualFlow, we also extend the targeted attack on ImageNet, and the results are presented in Table 4. The empirical results show that DualFlow can generate more powerful adversarial examples and obtain a superior attack success rate in most cases. It can get an ASR range from 94.12 to 99.52% on five benchmark DL models, but the most competitive baseline MI-FGSM can achieve an ASR of 83.90 to 99.34%. It is indicated that the proposed method is more threatening to DNNs and meaningful for exploring the existing DNNs' vulnerability and guiding the new DNNs' design.

5.3. Attack on defense models

Next, we investigate the performance of the proposed method in attacking robust image classifiers. Thus we select some of the most recent defense techniques that are from the robustbench toolbox as follows, for CIFAR-10 and CIFAR-100 are Hendrycks2019Using (Hendrycks et al., 2019), Wu2020Adversarial (Wu et al., 2020), Chen2020Efficient (Chen et al., 2022) and Rice2020Overfitting (Rice et al., 2020); for ImageNet are Engstrom2019Robustness (Croce et al., 2021), Salman2020Do_R18 (Salman et al., 2020), Salman2020Do_R50 (Salman et al., 2020) and Wong2020Fast (Wong et al., 2020). We compare our proposed method with the baseline methods.

Following the results shown in Table 5, we derive that DualFlow exhibits the best performance of all the baseline methods in terms of the attack success rate in most cases. The attack success rate of the baseline method stAdv and Chroma-Shift range from 95.41 to 99.12% and 17.22% from 74.80 in ImageNet, respectively. However, the DualFlow can obtain a higher performance range from 97.50 to 100%. It demonstrates the superiority of our method when attacking robust models.

5.4. Evaluation of human perceptual and image quality

Unlike the noise-adding attack methods, which usually use L_p norm to evaluate the victim examples' perceptual similarity to its corresponding benign image. The adversarial examples generated by noise-beyond ways always use other metrics referring to image quality. To be exact, we follow the work in Aydin et al. (2021)

[4] https://github.com/RobustBench/robustbench

[5] https://github.com/Harry24k/adversarial-attacks-pytorch

[6] https://www.cnpython.com/pypi/iqa-pytorch

[7] https://github.com/andrewekhalel/sewar

TABLE 2 Experimental results on attack success rate (ASR) of un-targeted attack of CIFAR-10 and CIFAR-100.

	CIFAR-10				CIFAR-100			
	VGG19	ResNet56	MobileNetV2	ShuffleNetV2	VGG19	ResNet56	MobileNetV2	ShuffleNetV2
FGSM	55.28	65.58	71.46	54.85	75.42	91.23	90.40	85.72
MI-FGSM	76.43	93.11	94.12	78.47	87.69	99.78	99.47	93.68
TI-FGSM	59.63	71.03	80.01	76.10	83.43	97.46	93.92	92.77
Jitter	83.70	94.87	**96.92**	86.25	98.31	**100.00**	**99.76**	94.63
stAdv	86.04	63.77	69.43	66.11	97.66	93.26	93.55	95.61
Chroma-shift	84.87	68.36	73.57	64.58	98.84	98.37	96.39	96.86
GUAP	82.55	89.34	87.61	87.02	92.26	94.59	96.89	92.20
DualFlow	**97.07**	**95.31**	93.65	**96.19**	**99.32**	99.02	98.83	**97.36**

The victim models are VGG19, ResNet56, MobileNetV2 and ShuffleNetV2, respectively, pre-trained by a GitHub Repository, named pytorch-cifar-models. Note that for the FGSM-based baselines, we synthesize their adversarial examples under L_{inf}-norm=0.031 limitation; the others are not subject to the L_{inf}-norm restrictions. Bold values indicates the best result.

TABLE 3 Experimental results on attack success rate (ASR) of un-targeted attack of ImageNet.

	GSM	MI-FGSM	TI-FGSM	Jitter	stAdv	Chroma-shift	GUAP	DualFlow
VGG16	93.56	98.64	97.16	95.27	97.62	98.62	97.73	**99.37**
VGG19	95.31	99.42	96.34	91.76	98.74	98.98	96.10	**99.43**
ResNet152	84	96.82	85.17	94.28	97.46	97.79	88.90	**98.63**
MobileNetV2	91.92	98.29	91.47	94.99	96.13	99.35	97.60	**99.61**
DenseNet121	94.98	99.61	99.56	99.56	96.83	98.65	92.70	**99.78**

The victim models are VGG19, ResNet152, MobileNetV2 and DenseNet121, respectively, which are pre-trained by PyTorch. Note that for the FGSM-based baselines, we synthesize their adversarial examples under L_{inf}-norm=0.031 limitation; the others are not subject to the L_{inf}-norm restrictions. Bold values indicates the best result.

TABLE 4 Experimental results on the attack success rate of targeted attack on dataset ImageNet.

Methods	FGSM	MI-FGSM	TI-FGSM	Jitter	stAdv	Chroma-Shift	DualFlow
VGG16	80.78	73.11	96.34	67.51	54.74	65.10	**96.67**
VGG19	60.59	49.36	83.90	46.50	53.23	55.39	**98.85**
ResNet152	80.22	73.93	**94.72**	70.45	65.87	69.60	94.12
MobileNetV2	72.70	63.94	92.38	60.86	70.63	76.00	**99.52**
DenseNet121	78.06	74.56	**99.34**	63.86	75.94	80.79	99.06

The baselines are FGSM, MI-FGSM, TI-FGSM, Jitter, stAdv, Chroma-shift and DualFlow. Note that for the FGSM-based baselines, we synthesize their adversarial examples under L_{inf}-norm=0.031 limitation; the others are not subject to the restrictions. Bold values indicates the best result.

using the following perceptual metrics to evaluate the adversarial examples generated by baseline methods and the proposed method, including Learned Perceptual Image Patch Similarity (LPIPS) metric (Zhang et al., 2018) and Deep Image Structure and Texture Similarity (DISTS) index (Ding et al., 2022). In addition, L_{inf}-norm, Spatial Correlation Coefficient (SCC) (Li, 2000), Structure Similarity Index Measure (SSIM) (Wang et al., 2004), and Pixel Based Visual Information Fidelity (VIFP) (Sheikh and Bovik, 2004) are also involved in evaluating the difference between the generated adversarial examples and their benign counterparts and the quality of the generated adversarial examples.

The generated images' quality results can be seen in Table 6, which indicated that the proposed method has the lowest LPIPS, DISTS perceptual loss and L_{inf} (the lower is better) are 0.0188, 0.0324 and 0.1642, respectively, on VGG-19 model; and has the highest SCC, SSIM and VIFP (the higher is better), achieving 0.9452, 0.7876 and 0.8192, respectively, on VGG-19 model. All the empirical data are obtained on the ImageNet dataset. The results show that the proposed method is superior to the existing attack methods.

To visualize the difference between the adversarial examples generated by our method and the baselines, we also draw the adversarial perturbation generated on NIPS2107 by FGSM, MI-FGSM, TI-FGSM, Jitter stAdv, Chroma-shift, GUAP and the proposed method in Figure 3, the target model is pre-trained VGG-19. The first two columns is the adversarial examples and the following are the adversarial noises of FGSM, MI-FGSM, TI-FGSM, Jitter stAdv, Chroma-shift, GUAP and our method, respectively. Noted that, for better observation, we magnified the noise by a factor of 10. From Figure 3, we can clearly observe that stAdv and Chroma-Shift distort the whole image. In contrast, the adversarial examples generated by our method are focused on the salient region and its noise is milder, and they are similar to the original clean counterparts and are more imperceptible to human eyes. These simulations of the proposed method take place under diverse aspects and the

TABLE 5 Experimental results on the attack success rate of un-targeted attack on CIFAR-10, CIFAR-100 and ImageNet dataset to robust models.

		FGSM	MIFGSM	TIFGSM	Jitter	stAdv	Chroma-shift	DualFlow
CIFAR-10	Hendrycks2019Using	27.06	16.90	18.54	32.67	99.12	20.70	**100**
	Wu2020Adversarial	25.63	16.28	19.10	31.02	99.12	18.36	**100**
	Chen2020Efficient	28.59	18.93	20.94	35.59	99.02	24.90	**100**
	Rice2020Overfitting	27.38	16.87	16.92	33.02	98.93	25.98	**100**
CIFAR-100	Hendrycks2019Using	37.67	25.57	28.88	48.89	95.41	35.16	**100**
	Wu2020Adversarial	40.13	27.06	30.71	50.13	97.66	30.86	**100**
	Chen2020Efficient	42.24	30.51	34.24	54.66	97.75	34.57	**100**
	Rice2020Overfitting	52.55	38.92	46.63	62.66	97.75	34.67	**100**
ImageNet	Engstrom2019Robustness	62.92	51.03	65.50	83.85	95.41	22.61	**97.50**
	Salman2020Do_R18	65.61	51.82	62.44	82.09	97.66	42.16	**100**
	Salman2020Do_R50	57.58	44.99	55.66	76.48	97.75	17.22	**99.19**
	Wong2020Fast	61.24	50.08	70.02	82.30	**97.75**	74.80	97.5

Bold values indicates the best result.

TABLE 6 Perceptual distances were calculated on fooled examples by FGSM, MI-FGSM, TI-FGSM, Jitter, stAdv, Chroma-shift, GUAP, and the proposed DualFlow on ImageNet.

	VGG19						ResNet152					
	LPIPS	DISTS	L_{inf}	SCC	SSIM	VIFP	LPIPS	DISTS	L_{inf}	SCC	SSIM	VIFP
FGSM	0.3036	0.1916	–	0.5572	0.8273	0.4705	0.2688	0.1679	–	0.5796	0.8348	0.4753
MI-FGSM	0.1962	0.1444	–	0.7135	0.9474	0.6575	0.1589	0.1078	–	0.7180	0.9466	0.6597
TI-FGSM	0.2179	0.1849	–	0.8153	0.9199	0.5576	0.1684	0.1451	–	0.8216	0.9330	0.5943
Jitter	0.2461	0.1617	–	0.6342	0.9076	0.5864	0.2001	0.1305	–	0.6480	0.9107	0.5792
stAdv	0.0581	0.0757	0.2420	0.8954	0.9873	0.7290	0.0490	0.0690	0.2420	0.8954	0.9873	0.7290
Chroma-shift	0.0231	0.5943	0.0275	0.9142	0.9834	0.8079	0.0.0203	0.0246	0.0.2250	0.9126	0.0.9848	0.0.8027
GUAP	0.4349	0.2838	0.4984	0.2768	0.7630	0.2955	0.4205	0.2501	0.6443	0.2289	0.7274	0.2674
DualFlow	**0.0188**	**0.0324**	**0.1642**	**0.9451**	**0.9876**	**0.8192**	**0.0169**	**0.0312**	**0.1550**	**0.9451**	**0.9876**	**0.8192**

Note that for the FGSM-based baselines, we synthesize their adversarial examples under L_{inf} -norm=0.031 limitation; the others are not subject to the restrictions. Bold values indicates the best result.

outcome verified the betterment of the presented method over the compared baselines.

5.5. Detectability

Adversarial examples can be viewed as data outside the clean data distribution, so the defender can easily check whether each input is an adversarial example. Therefore, generating adversarial examples with high concealment means that they have the same or similar distribution as the original data (Ma X. et al., 2018; Dolatabadi et al., 2020). To verify whether the carefully crafted examples satisfy this rule, we follow (Dolatabadi et al., 2020) and select LID (Ma X. et al., 2018), Mahalanobis (Lee et al., 2018), and Res-Flow (Zisselman and Tamar, 2020) adversarial attack detectors to evaluate the performance of the adversarial examples crafted by DualFlow. For comparison, we choose FGSM (Goodfellow et al., 2015), MI-FGSM (Dong et al., 2018), stAdv (Xiao et al., 2018), and Chroma-Shift (Aydin et al., 2021) as baseline methods. The test results are shown in the Table 7, including the area under the receiver operating characteristic curve (AUROC) and detection accuracy. Table 7, we can find that these adversarial detectors struggle to detect malicious examples constructed with DualFlow, compared to the baseline in all cases. Empirical results precisely demonstrate the superiority of our method, which generates adversarial examples closer to the distribution of original clean images than other methods, and the optimized adversarial perturbations have better hiding ability. The classifier is ResNet-34, and the code used in this experiment is modified from deep_Mahalanobis_detector[8] and Residual-Flow[9], respectively.

6. Conclusions

In this paper, we propose a novel framework named DualFlow for generating imperceptible adversarial examples with strong attack ability. It aims to perturb images by disturbing their latent representation space rather than adding noise to the clean

[8] https://github.com/pokaxpoka/deep_Mahalanobis_detector

[9] https://github.com/EvZissel/Residual-Flow

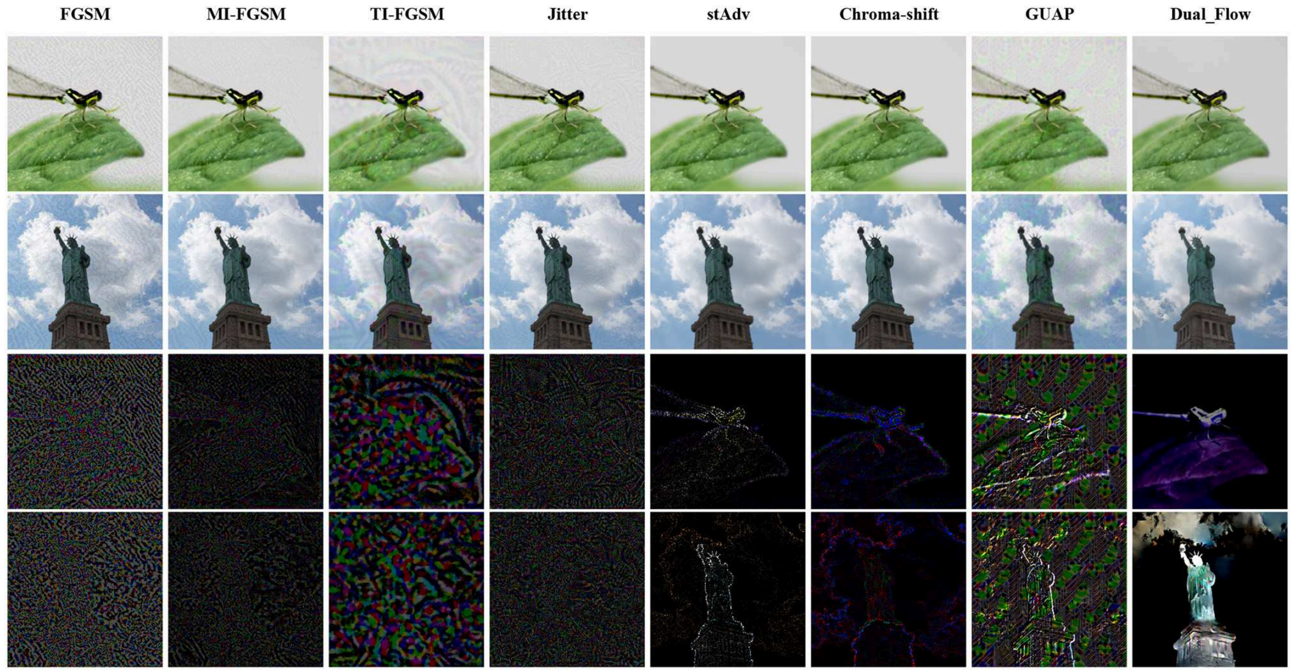


FIGURE 3
Adversarial examples and their corresponding perturbations. The first two columns are the adversarial examples, and the followings are the adversarial noise of FGSM, MI-FGSM, TI-FGSM, Jitter, stAdv, Chroma-shift, GUAP and our method, respectively.

TABLE 7 The detect results of DualFlow and the baselines on CIFAR-10 and CIFAR-100, Where the Chroma represent the Chroma-Shift.

Datasets	Methods	AUROC (%) ↑					Detection Acc. (%) ↑				
		FGSM	MI-FGSM	stAdv	Chroma	DualFlow	FGSM	MI-FGSM	stAdv	Chroma	DualFlow
CIFAR-10	LID	99.67	95.36	82.13	70.61	**52.23**	99.73	90.42	78.95	65.42	**58.42**
	Mahalanobis	96.54	98.54	85.64	75.61	**58.49**	90.42	97.26	79.67	76.13	**64.23**
	Res-Flow	94.47	97.59	78.96	72.37	**64.95**	88.56	91.54	76.38	73.64	**59.78**
CIFAR-100	LID	97.86	91.67	75.85	73.84	**62.37**	93.34	82.6	76.71	69.57	**57.78**
	Mahalanobis	99.61	97.64	76.17	72.32	**65.48**	98.62	92.49	80.65	71.48	**63.15**
	Res-Flow	99.07	99.76	78.53	78.56	**65.74**	95.92	96.99	83.43	69.72	**62.94**

↑ means that the larger the value, the better the detection method. Bold values indicates the best result.

image at the pixel level. Combining the normalizing flow and the spatial transform techniques, DualFlow can attack images' latent representations by changing the position of each value in the latent vector to craft adversarial examples. Besides, the empirical results of defense models show that DualFlow has stronger attack capability than noise-adding-based methods, which is meaningful for exploring the DNN's vulnerability sufficiently. Therefore, developing a more effective method to generate invisible, both for human eyes and the machine, is fascinating. Extensive experiments show that the adversarial examples obtained by DualFlow have superiority in imperceptibility and attack ability compared with the existing methods.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found at: CIFAR-10 and CIFAR-100, http://www.cs.toronto.edu/~kriz/cifar.html; ImageNet, https://image-net.org/.

Author contributions

YW and JinZ performed computer simulations. DH analyzed the data. RL and JinhZ wrote the original draft. RL and XJ revised and edited the manuscript. WZ polished the manuscript. All authors confirmed the submitted version.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 62162067 and 62101480, in part by the Yunnan Province Science Foundation under Grant Nos. 202005AC160007, 202001BB050076, and

Research and Application of Object detection based on Artificial Intelligence.

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

Arvinte, M., Tewfik, A. H., and Vishwanath, S. (2020). Detecting patch adversarial attacks with image residuals. *CoRR*, abs/2002.12504. doi: 10.48550/arXiv.2002.12504

Aydin, A., Sen, D., Karli, B. T., Hanoglu, O., and Temizel, A. (2021). "Imperceptible adversarial examples by spatial chroma-shift," in *ADVM '21: Proceedings of the 1st International Workshop on Adversarial Learning for Multimedia*, eds D. Song, D. Tao, A. L. Yuille, A. Anandkumar, A. Liu, X. Chen, Y. Li, C. Xiao, X. Yang, and X. Liu (Beijing: ACM) 8–14.

Bai, Y., Wang, Y., Zeng, Y., Jiang, Y., and Xia, S. (2023). Query efficient black-box adversarial attack on deep neural networks. *Pattern Recognit.* 133, 109037. doi: 10.1016/j.patcog.2022.109037

Ballet, V., Renard, X., Aigrain, J., Laugel, T., Frossard, P., and Detyniecki, M. (2019). Imperceptible adversarial attacks on tabular data. *CoRR*, abs/1911.03274. doi: 10.48550/arXiv.1911.03274

Besnier, V., Bursuc, A., Picard, D., and Briot, A. (2021). "Triggering failures: out-of-distribution detection by learning from local adversarial attacks in semantic segmentation," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* (Montreal, QC: IEEE), 15681–15690.

Carlini, N., and Wagner, D. A. (2017). "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy* (San Jose, CA: IEEE), 39–57.

Chen, J., Cheng, Y., Gan, Z., Gu, Q., and Liu, J. (2022). "Efficient robust training via backward smoothing," in *Thirty-Sixth AAAI Conference on Artificial Intelligence, (AAAI) 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence* (AAAI Press), 6222–6230.

Croce, F., Andriushchenko, M., Sehwag, V., Debenedetti, E., Flammarion, N., Chiang, M., et al. (2021). "Robustbench: a standardized adversarial robustness benchmark," in *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1*, eds J. Vanschoren and S.-K. Yeung.

Croce, F., Andriushchenko, M., Singh, N. D., Flammarion, N., and Hein, M. (2022). "Sparse-rs: a versatile framework for query-efficient sparse black-box adversarial attacks," in *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence* (AAAI Press), 6437–6445.

Deng, J., Dong, W., Socher, R., Li, L., Li, K., and Fei-Fei, L. (2009). "Imagenet: a large-scale hierarchical image database," in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (Miami, FL: IEEE), 248–255.

Ding, K., Ma, K., Wang, S., and Simoncelli, E. P. (2021). Comparison of full-reference image quality models for optimization of image processing systems. *Int. J. Comput. Vis.* 129, 1258–1281. doi: 10.1007/s11263-020-01419-7

Ding, K., Ma, K., Wang, S., and Simoncelli, E. P. (2022). Image quality assessment: unifying structure and texture similarity. *IEEE Trans. Pattern Anal. Mach. Intell.* 44, 2567–2581. doi: 10.1109/TPAMI.2020.3045810

Dinh, L., Krueger, D., and Bengio, Y. (2015). "NICE: non-linear independent components estimation," in *3rd International Conference on Learning Representations* (San Diego, CA: ICLR).

Dolatabadi, H. M., Erfani, S. M., and Leckie, C. (2020). "AdvFlow: Inconspicuous black-box adversarial attacks using normalizing flows," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020*, eds H. Larochelle, M. A. Rantzato, R. Hadsell, M.-F. Balcan, and H.-T. Lin.

Dong, Y., Liao, F., Pang, T., Su, H., Zhu, J., Hu, X., et al. (2018). "Boosting adversarial attacks with momentum," in *2018 IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT: IEEE), 9185–9193.

Dong, Y., Pang, T., Su, H., and Zhu, J. (2019). "Evading defenses to transferable adversarial examples by translation-invariant attacks," in *IEEE Conference on Computer Vision and Pattern Recognition* (Long Beach, CA: Computer Vision Foundation; IEEE), 4312–4321.

Duan, R., Ma, X., Wang, Y., Bailey, J., Qin, A. K., and Yang, Y. (2020). "Adversarial camouflage: Hiding physical-world attacks with natural styles," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Seattle, WA: Computer Vision Foundation; IEEE), 97–1005.

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., et al. (2018). "Robust physical-world attacks on deep learning visual classification," in *2018 IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT: Computer Vision Foundation; IEEE), 1625–1634.

Fawzi, A., and Frossard, P. (2015). "Manitest: are classifiers really invariant?," in *Proceedings of the British Machine Vision Conference 2015* (Swansea), 106.1–106.13.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2015). "Explaining and harnessing adversarial examples," in *3rd International Conference on Learning Representations*, eds Y. Bengio and Y. Lecum (San Diego, CA).

Guo, C., Gardener, J. R., You, Y., Wilson, A. G., and Weinberger, K. Q. (2019). "Simple black-box adversarial attacks," in *Proceedings of the 36th International Conference on Machine Learning*, eds K. Chaudhari and R. Salakhutdinov (Long Beach, CA: ICML).

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV: IEEE Computer Society), 770–778.

Hendrycks, D., Lee, K., and Mazeika, M. (2019). "Using pre-training can improve model robustness and uncertainty," in *in Proceedings of the 36th International Conference on Machine Learning*, eds K. Chaudhuri and R. Salakhutdinov (Long Beach, CA: PMLR), 2712–2721.

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* (Honolulu, HI: IEEE Computer Society), 2261–2269.

Ilyas, A., Engstrom, L., Athalye, A., and Lin, J. (2018). "Black-box adversarial attacks with limited queries and information," in *Proceedings of the 35th International Conference on Machine Learning*, eds J. G. Dy and A. Krause (Stockholm: PMLR) 2142–2151.

Ilyas, A., Engstrom, L., and Madry, A. (2019). "Prior convictions: black-box adversarial attacks with bandits and priors," in *7th International Conference on Learning Representations* (New Orleans, LA: OpenReview.net).

Jaderberg, M., Simonyan, K., Zisserman, A., and Kavukcuoglu, K. (2015). "Spatial transformer networks," in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, eds C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (Montreal, QC), 2017–2025.

Jordan, M., Manoj, N., Goel, S., and Dimakis, A. G. (2019). Quantifying perceptual distortion of adversarial examples. *CoRR*, abs/1902.08265. doi: 10.48550/arXiv.1902.08265

Kingma, D. P., and Dhariwal, P. (2018). "Glow: generative flow with invertible 1x1 convolutions," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018* (Montreal, QC), 10236–10245.

Krizhevsky, A., and Hinton, G. (2009). *Learning multiple layers of features from tiny images*. Computer Science Department, University of Toronto, Techchnical Report 1.

Kurakin, A., Goodfellow, I. J., and Bengio, S. (2017). "Adversarial examples in the physical world," in *5th International Conference on Learning Representations* (Toulon).

Laidlaw, C., and Feizi, S. (2019). "Functional adversarial attacks," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019*, eds H. M. Wallach, H. Larochelle, A. Beydelzimer, F. d'Alche-Bec, E. B. Fox, and R. Garnett (Vancouver, BC), 10408–10418.

Lee, K., Lee, K., Lee, H., and Shin, J. (2018). "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018*, eds S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Montreal, QC), 7167–7177.

Li, J. (2000). Spatial quality evaluation of fusion of different resolution images. *Int. Arch. Photogramm. Remot. Sens.* 33, 339–346.

Liu, A., Liu, X., Fan, J., Ma, Y., Zhang, A., Xie, H., et al. (2019). "Perceptual-sensitive gan for generating adversarial patches," in *The Thirty-Third AAAI Conference on Artificial Intelligence 2019, The Thirty-First Innovative Applications of Artificial Intelligence* (Honolulu, HI: AAAI Press), 1028–1035.

Liu, X., Yang, H., Liu, Z., Song, L., Chen, Y., and Li, H. (2019). "DPATCH: an adversarial patch attack on object detectors," in *Workshop on Artificial Intelligence Safety 2019 co-located with the Thirty-Third AAAI Conference on Artificial Intelligence 2019* (Honolulu, HI).

Luo, B., Liu, Y., Wei, L., and Xu, Q. (2018). "Towards imperceptible and robust adversarial example attacks against neural networks," in *Proceedings of the Thirty-Second Conference on Artificial Intelligence, (AAAI-18), the 30th Innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18)* (New Orleans, LA: AAAI Press), 1652–1659.

Luo, C., Lin, Q., Xie, W., Wu, B., Xie, J., and Shen, L. (2022). "Frequency-driven imperceptible adversarial attack on semantic similarity," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (New Orleans, LA: IEEE), 15294–15303.

Ma, N., Zhang, X., Zheng, H., and Sun, J. (2018). "Shufflenet V2: practical guidelines for efficient CNN architecture design," in *ECCV, Vol. 11218*, 122–138.

Ma, X., Li, B., Wang, Y., Erfani, S. M., Wijewickrema, S. N. R., Schoenebeck, G., et al. (2018). "Characterizing adversarial subspaces using local intrinsic dimensionality," in *6th International Conference on Learning Representations* (Vancouver, BC: OpenReview.net).

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018). "Towards deep learning models resistant to adversarial attacks," in *6th International Conference on Learning Representations* (Vancouver, BC: OpenReview.net).

Narodytska, N., and Kasiviswanathan, S. P. (2017). "Simple black-box adversarial attacks on deep neural networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops* (IEEE Computer Society), 1310–1318.

Rice, L., Wong, E., and Kolter, J. Z. (2020). "Overfitting in adversarially robust deep learning," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops* (PMLR), 8093–8104.

Salman, H., Ilyas, A., Engstrom, L., Kapoor, A., and Madry, A. (2020). "Do adversarially robust imagenet models transfer better?," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020*, eds H. Larochelle, M. Ranzato, R. Hadsell, M.-F. Balcan, and H.-T. Lin.

Sandler, M., Howard, A. G., Zhu, M., Zhmoginov, A., and Chen, L. (2018). Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381. doi: 10.1109/CVPR.2018.00474

Schwinn, L., Raab, R., Nguyen, A., Zanca, D., and Eskofier, B. M. (2021). Exploring misclassifications of robust neural networks to enhance adversarial attacks. *CoRR*, abs/2105.10304. doi: 10.48550/arXiv.2105.10304

Shao, Z., Wu, Z., and Huang, M. (2022). Advexpander: Generating natural language adversarial examples by expanding text. *IEEE ACM Trans. Audio Speech Lang. Process*. 30, 1184–1196. doi: 10.1109/TASLP.2021.3129339

Sheikh, H. R., and Bovik, A. C. (2004). "Image information and visual quality," in *ICASSP*, 709–712.

Simonyan, K., and Zisserman, A. (2015). "Very deep convolutional networks for large-scale image recognition," in *ICLR*.

Thys, S., Ranst, W. V., and Goedemé, T. (2019). "Fooling automated surveillance cameras: adversarial patches to attack person detection," in *CVPR* (IEEE; Computer Vision Foundation)49–55.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Trans. on Image Process*. 13, 600–612. doi: 10.1109/TIP.2003.819861

Wong, E., Rice, L., and Kolter, J. Z. (2020). "Fast is better than free: revisiting adversarial training," in *ICLR*.

Wu, D., Xia, S., and Wang, Y. (2020). "Adversarial weight perturbation helps robust generalization," in *NeurIPS*.

Xiao, C., Zhu, J., Li, B., He, W., Liu, M., and Song, D. (2018). "Spatially transformed adversarial examples," in *ICLR*.

Xu, H., Ma, Y., Liu, H., Deb, D., Liu, H., Tang, J., et al. (2020). Adversarial attacks and defenses in images, graphs and text: a review. *Inte. J. Autom. Comput*. 17, 151–178. doi: 10.1007/s11633-019-1211-x

Xu, Z., Yu, F., and Chen, X. (2020). "Lance: a comprehensive and lightweight CNN defense methodology against physical adversarial attacks on embedded multimedia applications," in *ASP-DAC* (IEEE), 470–475.

Yan, C., Xu, Z., Yin, Z., Ji, X., and Xu, W. (2022). "Rolling colors: adversarial laser exploits against traffic light recognition," in *USENIX Security*, 1957–1974.

Yi, Z., Yu, J., Tan, Y., and Wu, Q. (2022). Fine-tuning more stable neural text classifiers for defending word level adversarial attacks. *Appl. Intell*. 52, 11948–11965. doi: 10.1007/s10489-021-02800-w

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018). "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 586–595.

Zhang, Y., Ruan, W., Wang, F., and Huang, X. (2020). "Generalizing universal adversarial attacks beyond additive perturbations," in *ICDM*, 1412–1417.

Zhao, Y., Zhu, H., Liang, R., Shen, Q., Zhang, S., and Chen, K. (2019). "Seeing isn't believing: Towards more robust adversarial attack against real world object detectors," in *CCS*, eds L. Cavallaro, J. Kinder, X. Wang, and J. Katz, 1989–2004.

Zhou, Y., Han, M., Liu, L., He, J., and Gao, X. (2019). "The adversarial attacks threats on computer vision: a survey," in *MASS*, 25–30.

Zisselman, E., and Tamar, A. (2020). "Deep residual flow for out of distribution detection," in *CVPR*, 13991–14000.

TYPE Original Research
PUBLISHED 09 February 2023
DOI 10.3389/fnbot.2023.1119896

OPEN ACCESS

EDITED BY
Xin Jin,
Yunnan University, China

REVIEWED BY
Il Song Han,
Independent Researcher, London,
United Kingdom
Qinjun Zhao,
University of Jinan, China

*CORRESPONDENCE
Zengzhen Mi
✉ mizengzhen@163.com
Ren Chen
✉ 1309146643@qq.com

RECEIVED 09 December 2022
ACCEPTED 23 January 2023
PUBLISHED 09 February 2023

CITATION
Mi Z, Chen R and Zhao S (2023) Research on steel rail surface defects detection based on improved YOLOv4 network.
Front. Neurorobot. 17:1119896.
doi: 10.3389/fnbot.2023.1119896

Research on steel rail surface defects detection based on improved YOLOv4 network

Zengzhen Mi*, Ren Chen* and Shanshan Zhao

College of Mechanical Engineering, Chongqing University of Technology, Chongqing, China

Introduction: The surface images of steel rails are extremely difficult to detect and recognize due to the presence of interference such as light changes and texture background clutter during the acquisition process.

Methods: To improve the accuracy of railway defects detection, a deep learning algorithm is proposed to detect the rail defects. Aiming at the problems of inconspicuous rail defects edges, small size and background texture interference, the rail region extraction, improved Retinex image enhancement, background modeling difference, and threshold segmentation are performed sequentially to obtain the segmentation map of defects. For the classification of defects, Res2Net and CBAM attention mechanism are introduced to improve the receptive field and small target position weights. The bottom-up path enhancement structure is removed from the PANet structure to reduce the parameter redundancy and enhance the feature extraction of small targets.

Results: The results show the average accuracy of rail defects detection reaches 92.68%, the recall rate reaches 92.33%, and the average detection time reaches an average of 0.068 s per image, which can meet the real-time of rail defects detection.

Discussion: Comparing the improved method with the mainstream target detection algorithms such as Faster RCNN, SSD, YOLOv3 and other algorithms, the improved YOLOv4 has excellent comprehensive performance for rail defects detection, the improved YOLOv4 model obviously better than several others in P_r, R_c, and F1 value, and can be well-applied to rail defect detection projects.

KEYWORDS

rail defects, machine vision, defects detection, image enhancement, convolutional neural network (CNN)

1. Introduction

With the development of rail network layout and the rapid development of high speed rail technology, the importance of rail quality to train safety is becoming more and more obvious. According to the relevant safety statistics, the train safety accidents caused by rail surface defects account for about 30% of all accidents (Popović et al., 2022). Therefore, to ensure the security of traffic, accurate and dynamic detection of rail surface defects has become an urgent problem for railway development, and has important practical application value and research significance.

Due to the influence of rail manufacturing process, or by the wheel rail extrusion, impact, wear and other contact stress and natural weathering, its health status and quality deteriorate continuously, thus forming cracks, scars, wear, peeling, and other defects on the surface, with the passage of time, these defects will further deteriorate the rail surface quality, which may cause major railroad safety accidents. Therefore, the diversity and dynamics of rail defects bring great challenges to rail inspection technology.

The main rail defects detection methods include ultrasonic method, eddy current method, magnetic particle method, etc. (Zhao, 2021). The traditional detection methods need to rely on manual operation, time-consuming, labor-intensive, low efficiency, while it will bring unknown safety hazards to the inspectors.

Machine vision has been paid more and more attention by researchers with the benefits of fast speed, high precision and reliability, and many algorithms for surface defects detection have been generated. Faghih-Roohi et al. (2016) designed 3-layer convolution + maximum pooling layer to improve the speed of defects detection, and the accuracy of rail defects recognition can reach 92.00%, but the method only defects are detected and no classification is performed. Yuan et al. (2016) used the Otsu method to improve it by weighting the target variance of Otsu with the probability of occurrence of the target as the weight, so that the segmentation threshold close to the left edge of the single-mode histogram and the valley of the bimodal histogram, and the defects detection rate reach 93%, but the image segmentation algorithm cannot reach the real-time requirements. Shang et al. (2018) used a convolutional neural network (CNN) based on Inception-v3 to distinguish between normal and defective rail images. The model has a simple structure and faster processing speed, achieving a recognition accuracy of 92.08%, but the method is mainly effective for the detection of scar defects. Wang et al. (2018), Ni et al. (2021), and Ghafoor et al. (2022) analyzed the image features of rail defects, removed interference noise by image filtering, and then trained the model to improve the detection of surface defects, but the image enhancement algorithm is not universal and the image processing is time-consuming. Han et al. (2021) presented a multi-level feature fusion model for rail surface defects detection, which fuses the image features of different receptive field of multiple levels for target detection and enhances the accuracy of detection results and decreases the missing detection rate of small area defects, but the method detects too few types of defects and is not applicable to the detection of multiple complex defects of the rail. In summary, the above research is more concerned with the detection of defects, no classification recognition of defects, and there are problems such as image recognition methods are not universal, the speed of image processing cannot meet the defects detection of rail.

Therefore, according to the typical defect characteristics and defect types of rail, the defects are classified into four types of scars, peeling, wear and cracks, and a visual detection method combining image enhancement and deep learning is used to detect, identify and classify these four types of defects. In terms of the image processing, the captured images are firstly extracted from the rail region, then the defects edge information is enhanced with the improved Retinex algorithm, then the background modeling difference method is used to remove the background interference, and finally the defects are extracted with the adaptive thresholding. The improved Retinex algorithm and the background modeling difference method are more parameterized, and the effect on the detection speed of defects is not significant. In terms of deep learning, the Res2Net structure and attention mechanism are introduced to enhance feature extraction and improve the YOLOv4 network structure to enhance the detection rate of small-sized defects. The improved model enhances the accuracy of the four typical defects on the rail surface and ensures the detection speed.

2. Image enhancement algorithm for rail defects

The rail surface defects are highly susceptible to interference from lighting changes and textured backgrounds in the process of acquisition, making defects detection and recognition very difficult. To make the rail defects can be better detected and classified, the rail defects images are enhanced from four steps of rail region extraction, defects edge enhancement, background modeling difference and threshold segmentation, and the processing flow is shown in **Figure 1**, which solves the influence of unfavorable factors during rail surface defects segmentation.

2.1. Rail region extraction

To reduce the influence of textured backgrounds on rail defects detection, the column histogram minimum method (Xu et al., 2022) is first used to segment the target rail region from the original image. The steps of the column histogram algorithm are as follows:

(1) Calculate the sum of grayscale values for each column Si.
(2) Search for the minimum value min of ($S_{i\ +\ d}$-th) at fixed rail width intervals d.
(3) The i-th column corresponding to the minimum value min is the leftmost position of the corresponding rail.
(4) The position of the rightmost rail is the (i+d)-th column.

2.2. Improved Retinex image enhancement algorithm

Due to environmental interference, the captured rail image has low contrast, which affects the extraction of image defect features. In addition, the two defects, wear and crack, are similar to the background, and the texture features are not obvious, which will bring great challenges to the feature extraction of the image. Therefore, the image needs to be processed to enhance the contrast of the edge contour, which helps the segmentation of this image.

Retinex is an adaptive image enhancement method (Yu et al., 2017). The theory states that the brightness of an object depends on the ambient light and the reflection of the surface of the object on the light. The reflective component is the essence of the object. The object image can be recovered by simply removing the irradiated component. The Multi-Scale Retinex (MSR) (Zhu et al., 2021a) can achieve better results by adding a weighted average of multiple scales, and its expressions are as follows:

$$R_{MSR}(x, y) = \sum_{n=1}^{N} W_n\{\log[I(x, y)] - \log[I(x, y) \cdot G_n(x, y)]\} \qquad (1)$$

Where, N is the total number of scales, generally taken as 3, Wn for the scale coefficient, and meet $\sum_{\mathrm{n}=1}^{\mathrm{N}} \mathrm{W}_{\mathrm{n}} = 1$, said the number of scales for the Gaussian function. $\mathrm{G}_{\mathrm{n}}\left(\mathrm{x}, \mathrm{y}\right)$ represents the Gaussian amplifier model with the number of scales.

The MSR algorithm uses a linear quantization approach, and the processed data are widely distributed, which will show serious bifurcation and generally make it difficult to obtain satisfactory

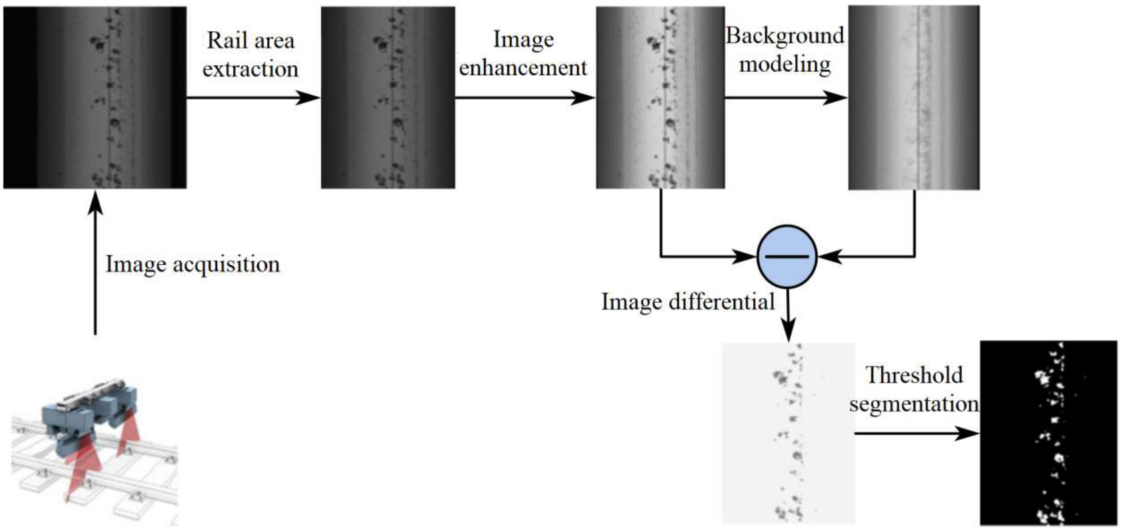


FIGURE 1
Defects detection algorithm based on background difference method.

results. To enhance the edge information of rail defects, the MSR algorithm is improved from the way of quantization. The mean value and mean squared deviation are introduced, and then a parameter controlling the image dynamics is added to realize the contrast adjustment to solve the problem of serious two-level differentiation of the data and thus the unsatisfactory image enhancement effect, with the following equation.

$$R(x,y) = \frac{255}{2}\left(1 + \frac{\log[R_{MSR}(x,y) - \mu]}{D \times MSE}\right) \tag{2}$$

Where D is the dynamic adjustment parameter of the image, the value of D is inversely proportional to the contrast of the image, and μ, MSE are the mean and mean squared deviation of the number of channels of R, G, B in log [R_{MSE} (x, y)], respectively, and Value is the value of log [R_{MSE} (x, y)]. After the experiment, the best effect is obtained when the scale number is 3 and D is 2.5 (**Figure 2**).

The results show that the improved Retinex has stronger contrast and more prominent defect edges information than MSR, and less noise than histogram equalization. If the results of MSR are quantified directly, the overall darker images are obtained, which is due to the smaller data range of the original values after logarithmic processing and the small differences between channels, and the linear quantization is much smoother than the logarithmic curve, so the overall effect is darker and the edge information is easily lost. Proposed in this paper achieves good results by changing the quantization of the mean and mean squared deviation to strengthen the defect edges. The average Peak Signal to Noise Ratio (PSNR) per image is calculated to be 15.40, which is a very significant improvement in image quality and is very suitable for the processing of orbital defect images.

2.3. Background difference segmentation algorithm for surface defects

To segment the rail defects from the background image, the defects segmentation based on background difference algorithm is proposed, the idea of background difference method is the process of subtracting the background from the current image so as to get the defects. The background image is obtained by learning the rail video sequence, and the method of extracting the motion foreground in the video sequence based on background difference is mainly divided into three steps (Chel et al., 2020): background modeling, foreground detection, and background update. Among them, the mean method is the simplest in background modeling (Piccardi, 2004), which can quickly and effectively segment moving targets in static scenes with high real-time performance.

Since single image defects segmentation cannot learn the background model from the video sequence, the background difference method in video surveillance cannot be directly used for rail surface defects segmentation. Considering the feature of small variation range of gray value along the rail direction of the image and the real-time requirement, rail surface defects segmentation algorithm based on the mean background difference is proposed.

2.3.1. Background modeling

Define the direction perpendicular to the rail as the x−axis and the rail direction as the y−axis. Calculate the mean value of each column of the image according to the feature of small change of the image along the y−axis, and modeling the background image.

$$I_m(x) = mean\left(I_y(x)\right) \tag{3}$$

Where $I_m(x)$ denotes the x-th column image background modeling and $mean\left(I_y(x)\right)$ is the mean value function.

The algorithm implements static single-image background modeling, and the processing speed is not affected due to the simplicity of modeling, and the background is maximally close to the original image.

2.3.2. Background subtraction

To highlight defects and diminish the effects of illumination variations and reflection unevenness, and subtract the rail

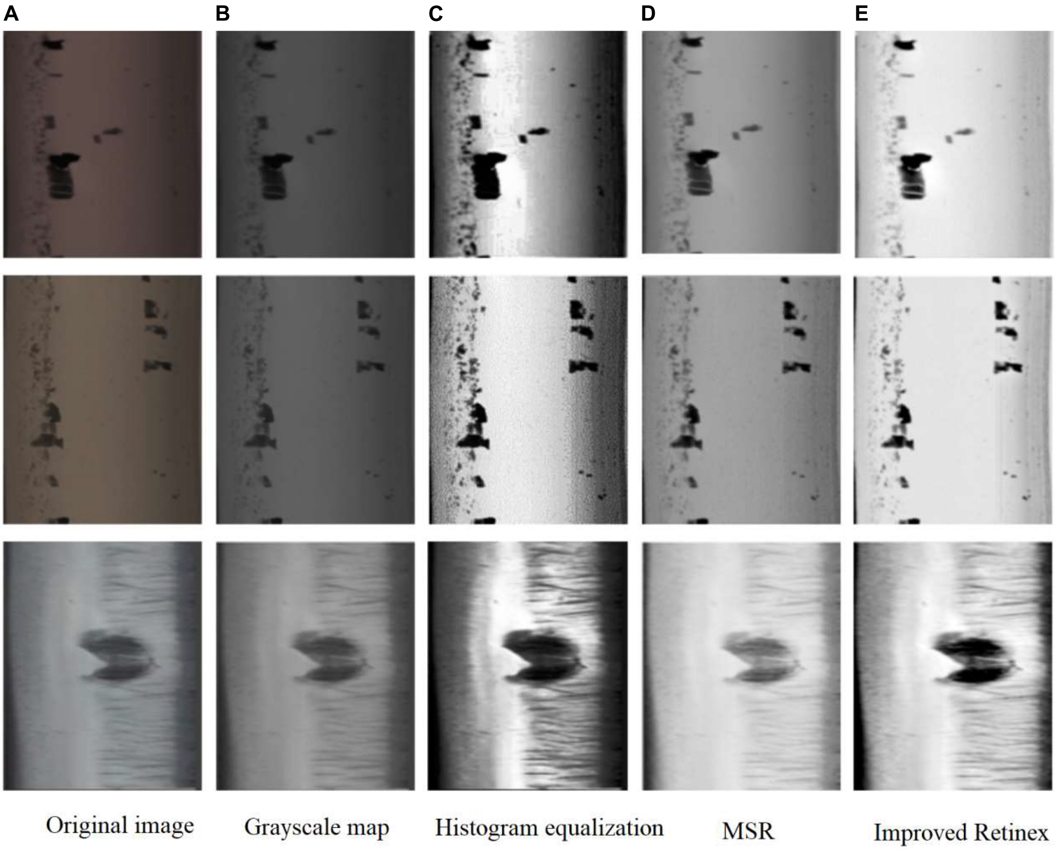


FIGURE 2
Comparison chart of the effect of image enhancement algorithm. **(A)** Original image. **(B)** Grayscale map. **(C)** Histogram equalization. **(D)** MSR. **(E)** Improved Retinex.

image from the background image to get the difference image.

$$\Delta I(x, y) = I_0(x, y) - I_m(x, y), \forall (x, y) \tag{4}$$

where $I_0(x, y)$ is the original image and $I_m(x, y)$ is the modeled background image.

2.3.3. Adaptive thresholding segmentation

To segment the defective regions in the differential images, Niblack thresholds are defined (Zhou et al., 2013).

$$th = \mu_{\Delta I} + C \cdot \delta_{\Delta I} \tag{5}$$

Where $\mu_{\Delta I}$ and $\delta_{\Delta I}$ are the mean and variance of ΔI, respectively, and the control factor C is a constant. Following Chebyshev's formulas, ratio of data with more than C times the Standard Deviation (SD) from the mean is at most $1/C^2$ in any dataset. For this purpose, the value of C can be determined based on the ratio of the target defects to the total image. Since the differential image has the property of zero mean, Equation 5 can be simplified as follows.

$$th = C \cdot \delta_{\Delta I} \tag{6}$$

After experiments, the segmentation effect is best when $C = 3$. The method can segment the defects well according to the obtained threshold th for the image. The processed ones are shown in **Figure 3**.

3. Improved YOLOv4 model for rail defects detection

YOLOv4 has a high performance in recognizing large and medium-sized, significantly separated targets (Bochkovskiy et al., 2020), but the detection accuracy is not high for small-sized targets and targets with small background differences. In the dataset used in this paper, most of the Scar and Peeling defects are small in size, and the foreground background differences of Wear and crack defects are small, which are not ideal for the recognition of defects directly with the YOLOv4 network. Accordingly, the network structure and feature extraction aspects are optimized based on the YOLOv4 network to adapt it to the detection and recognition of orbital defects.

3.1. Rail defects feature extraction method

3.1.1. Introduction of Res2Net

Aiming at the problem of small size and little detail information of rail defects, Res2Net structure and attention mechanism are introduced to enhance the feature extraction of defects.

The ResNet residual blocks in the YOLOv4 network structure are replaced with the Res2Net structure, as shown in **Figure 4**. This structure not only increases the receptive field of each network layer, but also enhances the ability of multi-size feature extraction and enables effective detection of small-size defects.

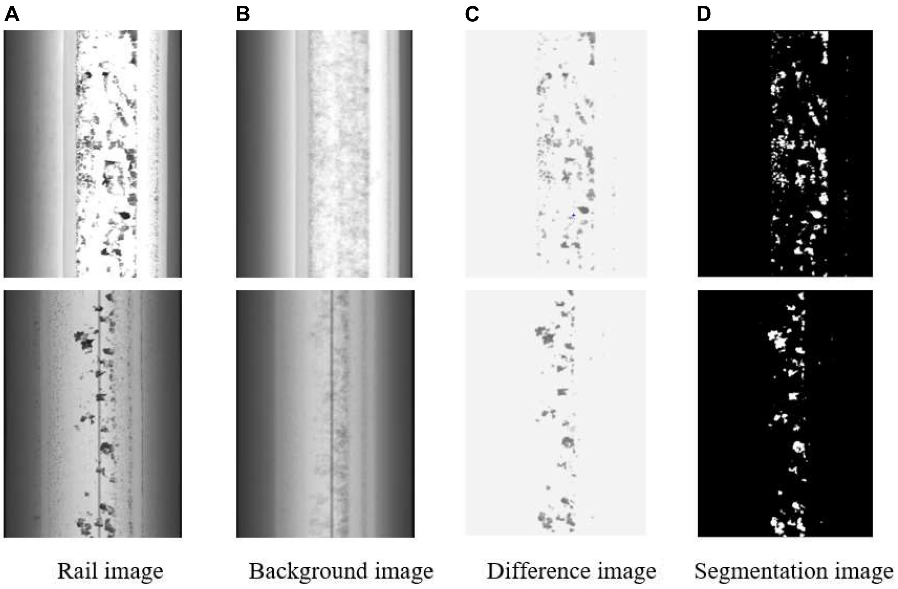


FIGURE 3
Splitting effect of the rail images. **(A)** Rail image. **(B)** Background image. **(C)** Difference image. **(D)** Segmentation image.

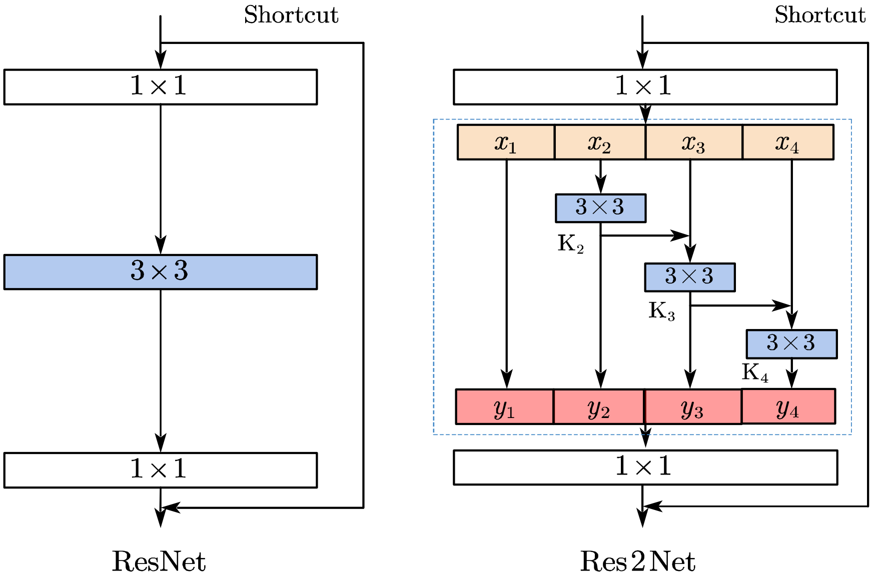


FIGURE 4
Structure of ResNet and Res2Net.

In the Res2Net structure, each output can increase the receptive field, where y2 can get a 3x3 receptive field, can y3 get a 5x5 receptive field, and y4 can get a larger 7x7 receptive field, so each Res2Net can obtain a combination of features with different receptive field sizes. Thus, the structure can both increase the receptive field of each network layer, and fuse multi-scale features. It is very effective for the small-sized targets (Gao et al., 2021).

3.1.2. CBAM attention mechanism

To enhance the attention to the effective feature information and to improve the region weight of rail defects, an attention mechanism is added to the model. Convolutional Block Attention Module (CBAM) (Woo et al., 2018) is a lightweight attention module based on CNN, and is shown in **Figure 5**. It integrates the Channel Attention Module (CAM) (Ilyas et al., 2021) and the Spatial Attention Module (SAM) (Hu et al., 2020) to generate the corresponding feature map mapping to increase the weight of the defects region in the feature map, which in turn makes the model pay more focus to the features of the defects location and reduces the influence of background and uneven spatial distribution on the detection of rail defects.

3.1.2.1. In channel attention

The rail defect features are max-pooled and average-pooled, respectively, to obtain 2 "$1 \times 1 \times C''$ channel descriptions, and then they are sent into a 2-layer shared fully-connected layer, and the two output features are summed up to obtain a weight coefficient after

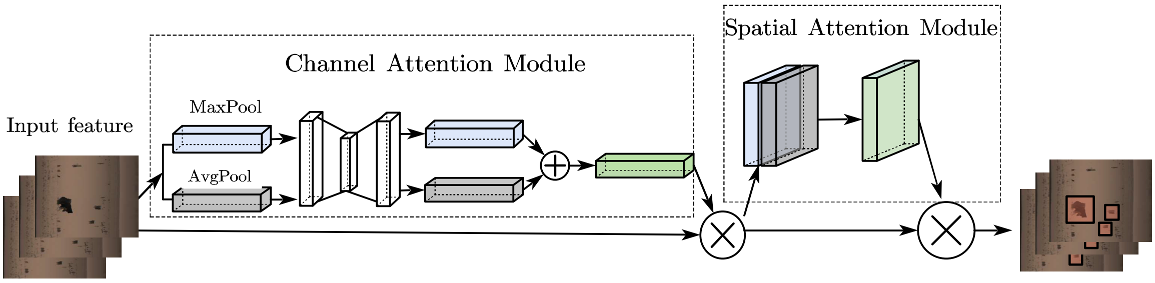


FIGURE 5
CBAM overall structure diagram.

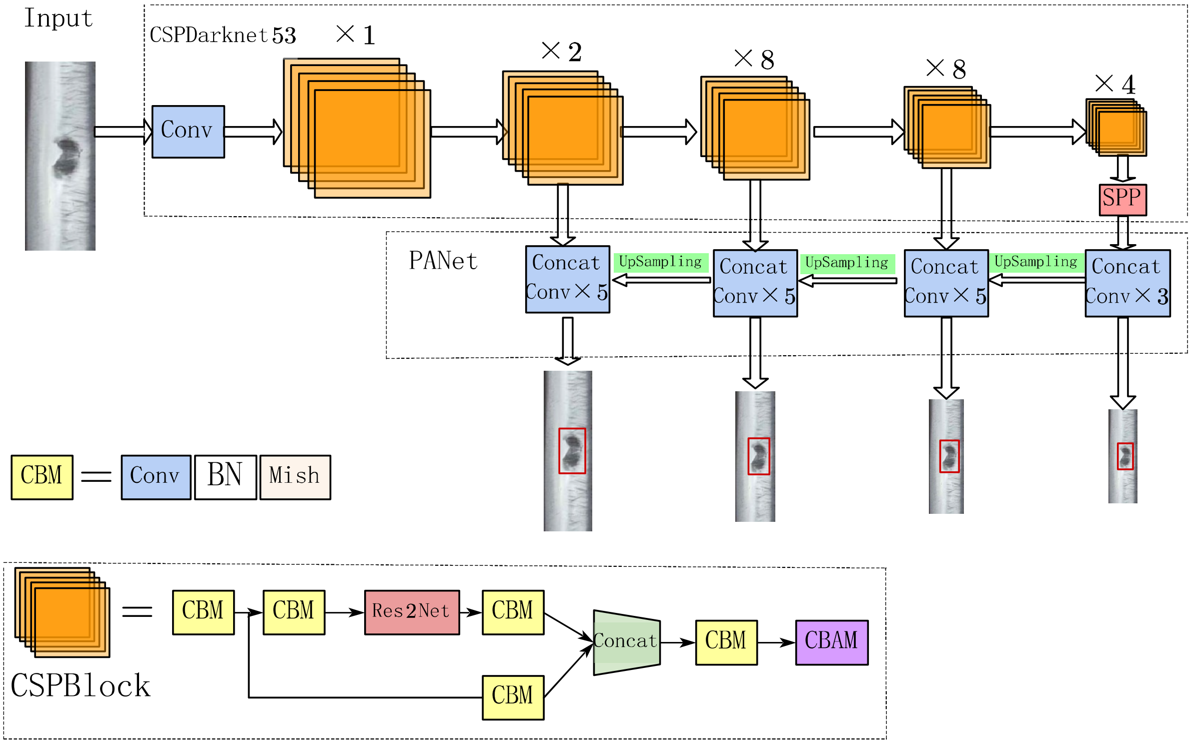


FIGURE 6
Improved YOLOv4 network structure diagram.

the activation function, and eventually the new features multiplied by the weight coefficients and the original features are used as input for the SAM.

3.1.2.2. In SAM

Global average pooling as well as global maximum pooling operations are performed on channels to produce 2 feature maps represent different information. After merging them, feature fusion is proceeded by 7 × 7 convolution with a larger receptive field, and lastly the operation is used to generate a weight map, which is then superimposed on the original input feature map to obtain a final rail defects feature map.

The feature map of CBAM is the same size as the feature map of original image, only the feature elements have changed, focusing more on the edge location information of the defects image, reducing the impact of background on detection accuracy and reducing the rate of wrong and missed detection. It can help the network to extract features better and deeper, and further improve the network's ability to learn rail defects.

3.2. Design of defects recognition network

3.2.1. Network structure

The PANet structure used in YOLOv4 can fuse the semantic information of different feature layers and is suitable for detecting targets of different sizes. However, the number of rail surface defects is high and the proportion of pixels in the image is low, and the original PANet structure still lacks effective detection for tiny defect targets. Therefore, on the basis of the original feature layer, Continue to fuse shallow and deep features to increase the feature detection scale and form a new feature detection layer.

Adding new feature detection layers leads to an increase in the number of network structure parameters, and the bottom-up path enhancement structure contributes less to the detection of small area defects. Therefore, the bottom-up path enhancement structure in PANet is removed in order to reduce parameter redundancy and ensure sufficient detection speed. Meanwhile, to help the network extract features better and deeper, the residual structure in the CSPBlock block is replaced with the Res2Net structure; to further improve the network's ability to learn rail defects, the CBAM structure is added to the CSPBlock block. The improved PANet structure is shown in **Figure 6**. The improved structure not only inherits the feature fusion effect of the original structure, but also can obtain more shallow features while reducing the network parameters, so the feature extraction effect of small area defects of the rail is better.

3.2.2. Anchor frame clustering

Since a new feature detection layer is added, the number and size of anchor frames are not suitable for this network, so it needs to be re-clustered. K-means is used in YOLOv4 network, and the clustering effect is largely determined by selecting the initial cluster center. To ensure a relatively good clustering effect, K-means++ is adopted to re-cluster the anchor frames. The method of clustering is as follows:

1. Randomly select a sample from the rail defects dataset as the initial cluster center v_j.
2. Secondly, calculate the distance between each sample x_i and v_j in the dataset and select the shortest of them.
3. Then calculate the probability of each data sample being selected as the next clustering center, and select the sample with the greatest probability distance as the new clustering center.
4. Repeat steps 2 and 3 until all k clustering centers have been identified.
5. Cluster the k initialized cluster centers obtained, assign each sample to the cluster center with the smallest distance from each other, and update the cluster centers, and repeat the step until the cluster centers unchanged.

The clustering results are shown in **Table 1**.

From **Table 1**, it can see that most of the anchor frames are very different from each other, except for the first three groups of anchor frames, which do not vary much. Compared with k-means randomly selecting the cluster center, k-means++ selects the cluster center by the idea of "the farther the cluster center are from each other, the better," which converges the data faster and achieves good results while reducing the computation time.

4. Experiment and analysis

Evaluation metrics for training and performance are first established, and then the current mainstream deep learning-based target detection algorithms are compared with the algorithms of this paper in terms of accuracy and speed metrics. The computer configuration is a 64-bit Windows 10 system with 32G of RAM, CPU model i9-10980XE, and GPU model is RTX3090. In the training process, the bitch_size is set to 16, the initial learning rate is 0.001, the learning rate is decayed, the final learning rate is 0.00001, and iterations is set in 1,000. A 416 × 416 resolution input is taken for training, the detection threshold is set to 0.5, and the Dropout method is used to prevent overfitting.

TABLE 1 *A priori* box clustering results.

Clustering algorithm	Prior box
Entry 1K-means	(12,14), (15,23), (17,44), (40,26), (41,93), (48,49), (33,151), (63,78), (85,45), (61,125), (74,223), (134,82)
K-means++	(11,13), (16,21), (18,42), (37,27), (51,51), (78,41), (33,152), (40,85), (65,80), (60,123), (71,222), (118,77)

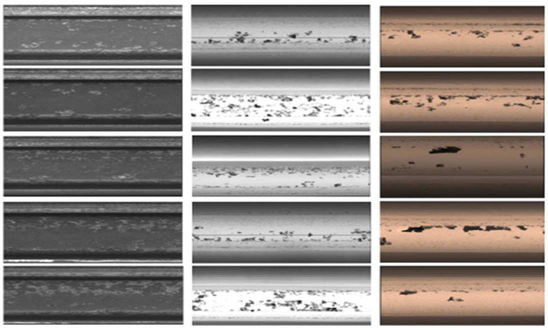


FIGURE 7
Image samples collected by three different types of line-scan CCDs. The figure has been licensed by Gan et al. (2017).

4.1. Dataset and evaluation index

The experimental dataset were obtained from Rail beam factory of Panzhihua Iron and Steel (Group) Company and network datasets, where the self-acquired dataset were used for training and the RSDDs (Gan et al., 2017) network datasets were used for validation. For the image acquisition experiments, color/grayscale images of heavy rails of 60 kg/m were obtained using three different types of line-scan CCD cameras, and a total of 2,124 images of rails with high imaging quality were selected, of which 956 were defective, and image samples are shown are shown in **Figure 7**.

To unify the experimental dataset, all acquired images are first segmented on the rail surface, and then the images are resized to 400*800 pixels, and finally the dataset is expanded by flip transform, brightness transform, random cropping, geometric scaling, etc., 4,000 images of rail surface defects dataset are generated, including 1,000 images each of cracks, scars, wear and peeling. Randomly select 80% as the training set, and 20% as the test set. **Figure 8** shows the typical samples of the four defects and their expansions.

This paper introduces four evaluation indexes: Recall Rate (R_c or R), Precision Rate (P_r or P), F1 Value and Average Inspection Time. Rail surface damage detection is related to the safety of railroad transport, and both R and P indexes are particularly important, while F1 value can visualize the importance of R and P.

$$R = \frac{TP}{TP + TN} \tag{7}$$

$$P = \frac{TP}{TP + FP} \tag{8}$$

$$F1 = \frac{2PR}{P + R} \tag{9}$$

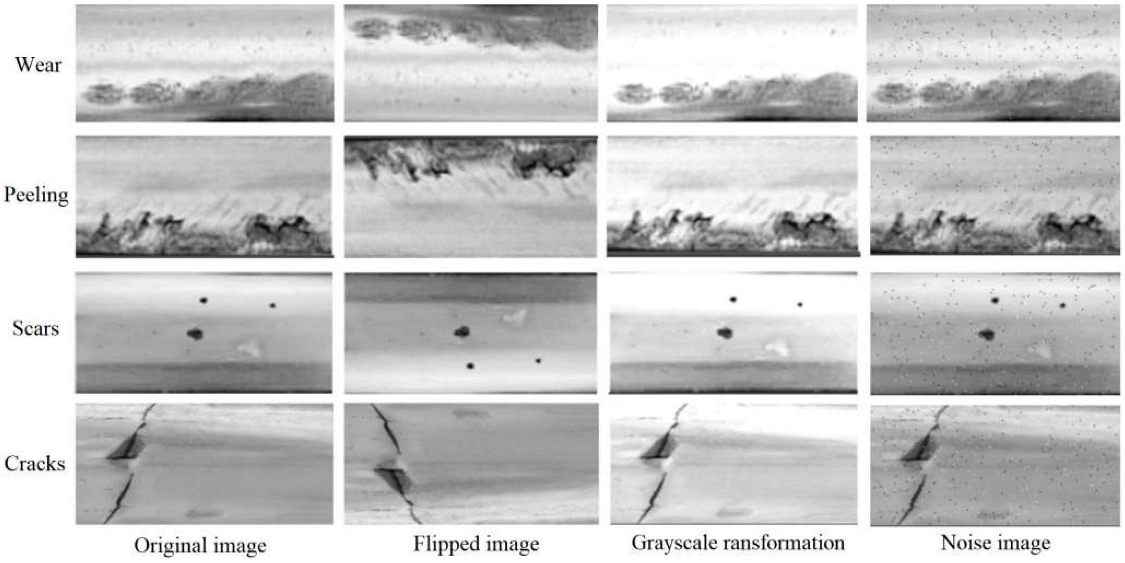


FIGURE 8
Expansion of the dataset.

TABLE 2 Comparison of the detection performance of different algorithms for the self-collected dataset.

Detection algorithm	Cracks		Scars		Wear		Peeling		F1	T/ms
	P/%	R/%	P/%	R/%	P/%	R/%	P/%	R/%		
Faster R-CNN	91.2	93.1	89.8	90.3	88.5	86.9	90.7	91.6	0.903	41.2
SSD	86.3	88.6	84.1	87.2	80.7	77.9	84	86.3	0.844	82.0
YOLOv3	85.8	87.5	84.3	85.8	79.2	76.5	86.9	88.2	0.843	46.0
YOLOV4	88.4	90.4	86.1	85.3	84.1	83.9	89.1	91.2	0.873	55.0
YOLOv5	90.1	91.7	88.6	88.9	85.3	84.1	89.2	90.5	0.885	49.0
YOLOv6	93.6	92.3	92.4	92.8	88.7	89.2	90.4	91.7	0.914	40.0
Algorithm in this paper	94.8	93.7	94.0	93.6	89.7	88.4	92.2	93.6	0.925	68.0

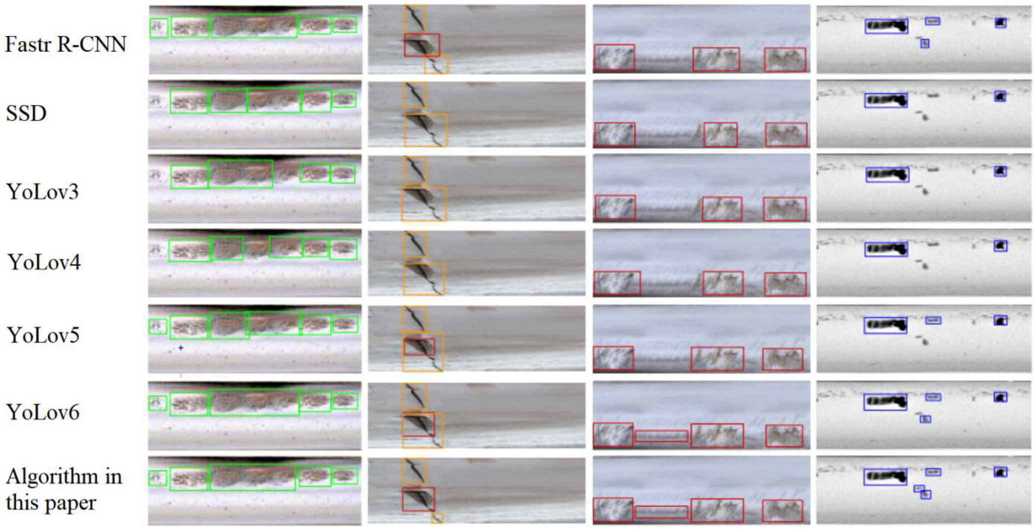


FIGURE 9
Comparison chart of the effect of detection.

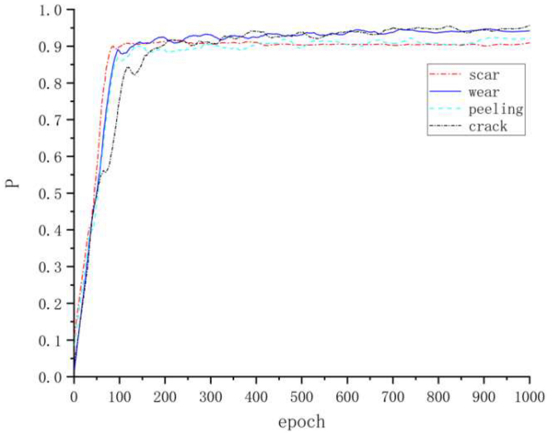


FIGURE 10
Detection accuracy of the four defects.

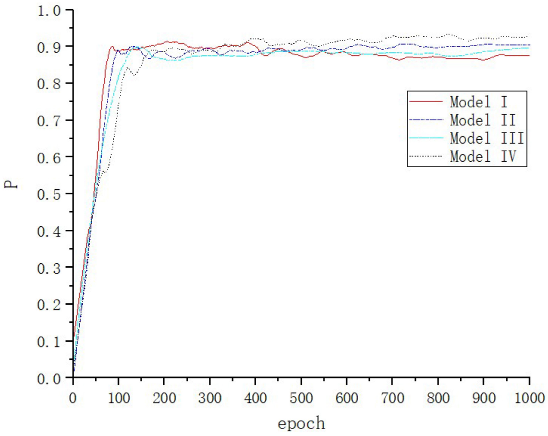


FIGURE 11
Accuracy change curve of each model in ablation experiment.

Where: *TP*: Positive samples predicted to be positive class, *FP*: Negative samples predicted to be positive class, *TN*: Negative samples predicted to be negative class.

4.2. Algorithm performance analysis

Training and test experiments were conducted on several detection algorithms [Faster R-CNN (Sekar and Perumal, 2021), SSD (Liu et al., 2016), YOLOv3 (Redmon and Farhadi, 2021), YOLOv4 (Bochkovskiy et al., 2020), YOLOv5 (Zhu et al., 2021b), YOLOv6 (Li et al., 2022)] and the improved algorithms in this paper, and after the network training and parameter tuning, the network convergence, and then the data results were tallied according to the evaluation metrics.

The data in Table 2 show that the detection algorithm of this paper has the highest R_c and P_r for four defects: cracks, scars, wear, and peeling. Relative to other mainstream algorithms, the improved YOLOv4 algorithm has an F1 value that is 2.2% higher than Faster R-CNN, 8.1% higher than SSD, 8.2% higher than YOLOv3, 5.2% higher than YOLOv4, 4.0% higher than YOLOv5, and 1.1% higher than YOLOv6. All 3 metrics are better than other mainstream detection algorithms. Compared with the original YOLOv4 network, the accuracy of the improved network reaches 94.8% for cracks, 6.4% higher than before the improvement; 94.0% for scars, 7.9% higher than before the improvement; 89.7% for wear, 5.6% higher than before the improvement; and 92.2% for spalling, 3.1% higher than before the improvement. The accuracy rates of the four typical defects are 1.2, 1.6, 1.0, and 1.8% higher than YOLOv6, respectively, which is a significant improvement. In addition, although this algorithm increases the detection layer and adds the attention mechanism resulting in increased parameters, the removal of the bottom-up path structure in the PANet reduces a large number of parameters, and the image pre-processing of the background difference method is concise and effective. The average detection time per image is 0.068 s (68 ms), which is very close to that of YOLOv4, YOLOv5, and YOLOv6, and can meet the system real-time requirements while ensuring the effect of rail defects detection. Mapping the inspection results back to the original image, the effect comparison chart is shown in Figure 9, where the green box is for wear defects, the orange box is for crack defects, the red box is for peeling defects, and the blue box is for scar defects.

From Figure 9, this algorithm can recognize defects of small size and defects with small background differences very well, and the recognition effects are all better than other mainstream algorithms, and Figure 10 shows the accuracy of four kinds of defects.

To continue to verify the effectiveness of this algorithm, the algorithm is tested on the publicly available dataset RSDDs, comparing the method of this paper with improved Cascade R-CNN proposed by Luo et al. (2021), improved YOLOv5 proposed by Guo et al. (2022) and multi-layer feature fusion network proposed by Han et al. (2021), the defects detection accuracy and the average detection time of a single image are shown in Table 3.

The accuracy of this algorithm for defect detection on the RSDDs rail dataset reaches 98.96%, all of which are better than the methods used by the other three. The average detection time per image is 68 ms, which is significantly better than Luo's method and very close to Li and Han's methods, and fully satisfies the real-time performance of rail defects detection. The results show that this method is more suitable for performing the task of rail surface defects detection.

TABLE 3 Comparison of detection performance of different algorithms for RSDDs dataset.

Literature sources	Network structure	AP/%	T/ms
Luo et al. (2021)	Improved cascade R-CNN	98.75	146.3
Guo et al. (2022)	Improved YOLOv5	91.80	54.8
Han et al. (2021)	Multi-layer feature fusion network	96.72	59.8
Algorithms in this paper	Improved YOLOv4	98.96	68.0

TABLE 4 Results of ablation experiments.

Network model	P_r/%	R_c/%	F1 value	T/ms
Model I	87.70	86.93	0.873	55
Model II	90.45	88.92	0.897	59
Model III	89.11	88.67	0.889	63
Model IV	92.68	92.33	0.925	68

4.3. Ablation experiments

The algorithm uses several improved strategies based on YOLOv4, and to verify its effectiveness, ablation experiments were designed for comparative analysis.

Model I: YOLOv4 network. Model II: The model obtained by replacing the Residual Block structure in the feature extraction part of YOLOv4 with the Res2Net module, and then adding the CBAM attention mechanism. Model III: Adding the detection layer and removing the top-down structure in PANet. Model IV is the model of this paper. Each network model is trained for 1,000 cycles (**Figure 11**).

In this figure, the loss values of each network model in the ablation experiments decrease rapidly within the first 50 iterations of the training process, and then gradually converge.

As seen in **Table 4**, Model II makes improvements to feature extraction, and increasing the weight of defects location and increasing the perceptual field to better extract the small defect features of the rails, with a 2.75% improvement in P_r, 1.99% improvement in R_c, and 2.40% improvement in F1 value over the YOLOV4 network, effectively improving the detection performance of small size defects. The model III network structure performs multi-scale feature fusion to enhance the accuracy of defects localization, which improves P_r by 1.41%, R_c by 1.74%, and F1 value by 1.60% over the YOLOV4 network, but the detection time of a single image increases by 8 ms, which is due to the increase of detection layers, resulting in the calculation of a large number of additional parameters. The fusion of the above two improved methods into the benchmark network at the same time can further improve the accuracy of rail defects localization and identification, which improves P_r by 4.98%, R_c by 5.40%, and value by 5.20% over the YOLOV4 network. This verifies the validity of the improved method for rail surface defects detection.

5. Discussion

For the problem of small defects size and complex background of rail. The detection algorithm for rail surface defects is proposed. The improved YOLOv4 defects detection algorithm not only inherits the feature fusion effect of the original structure, but also can obtain more shallow features while reducing the network parameters and improving the feature extraction capability of small targets. The average processing speed of a single image is only 13 ms higher than YOLOv4, which is also very close to the detection speed of YOLOv6. Efficient and accurate detection of rail defects is achieved, where the recognition accuracy of 4 defects, namely, cracks, scars, wear and peeling, reaches 94.8, 94.0, 89.7, and 92.2%, respectively. Their P_r, R_c and F1 values are higher than other mainstream target detection algorithms. The detection algorithm ensures high detection accuracy while guaranteeing detection speed, and is more suitable for performing rail surface defect detection tasks.

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 authors.

Author contributions

ZM organized the database. RC performed the statistical analysis and wrote the first draft of the manuscript. All authors contributed to conception and design of the study, wrote sections of the manuscript, revised the manuscript, and read and approved the submitted version.

Funding

This research was funded by National Natural Science Foundation of China (grant number 61901068), Chongqing Natural Science Foundation (grant numbers KJQN201901150, KJQN202001131, and cstc2021jcyj-msxmX0525), and Chongqing Graduate Student Innovation Foundation (grant numbers gzlcx20222041 and CYS21467).

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

Bochkovskiy, A., Wang, C. Y., and Liao, H. (2020). YOLOv4: optimal speed and accuracy of object detection. *arXiv* [preprint]. doi: 10.48550/arXiv.2004.10934.

Chel, H., Bora, P. K., and Ramchiary, K. K. (2020). A fast technique for hyper-echoic region separation from brain ultrasound images using patch based thresholding and cubic B-spline based contour smoothing. *Ultrasonics* 111:106304. doi: 10.1016/j.ultras.2020.106304

Faghih-Roohi, S., Hajizadeh, S., Núñez, A., Babuska, R., and De Schutter, B. (2016). "Deep convolutional neural networks for detection of rail surface defects," in *Proceeding of the 2016 international joint conference on neural networks (IJCNN)*, 2584–2589. doi: 10.1109/IJCNN.2016.7727522

Gan, J. R., Li, Q. Y., Wang, J. Z., and Yu, H. M. (2017). A hierarchical extractor-based visual rail surface inspection system. *IEEE Sens. J.* 17, 7935–7944. doi: 10.1109/JSEN.2017.2761858

Gao, S., Cheng, M. M., Zhao, K., Zhang, X., Yang, M., and Torr, P. (2021). Res2Net: a new multi-scale backbone architecture. *IEEE Trans. Pattern Analy. Mach. Int.* 43, 652–662. doi: 10.1109/TPAMI.2019.2938758

Ghafoor, I., Tse, P. W., Munir, N., and Trappey, A. J. C. (2022). Non-contact detection of railhead defects and their classification by using convolutional neural network. *Optik* 253:168607. doi: 10.1016/j.ijleo.2022.168607

Guo, Z. X., Wang, C. S., Yang, G., Huang, Z. Y., and Li, G. (2022). MSFT-YOLO: improved YOLOv5 based on transformer for detecting defects of steel surface. *Sensors* 22:3467. doi: 10.3390/s22093467

Han, Q., Liu, J. B., Feng, Q. B., Wang, S. C., and Dai, P. (2021). Damage detection method for rail surface based on multi-level feature fusion. *China Railway Sci.* 42, 41–49.

Hu, J., Shen, L., Albanie, S., Sun, G., and Wu, E. (2020). Squeeze-and-excitation networks. *IEEE Trans. Pattern Analy. Mach. Int.* 42, 2011–2023. doi: 10.1109/TPAMI.2019.2913372

Ilyas, N., Lee, B., and Kim, K. (2021). HADF-crowd: a hierarchical attention-based dense feature extraction network for single-image crowd counting. *Sensors* 21:3486. doi: 10.3390/s21103483

Li, C. Y., Li, L. L., Jiang, H. L., Wang, K. H., Geng, Y. F., Li, L., et al. (2022). YOLOv6: a single-stage object detection framework for industrial applications. *arXiv* [preprint]. doi: 10.48550/arXiv.2209.02976

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C., et al. (2016). "SSD: single shot multibox detextor," in *Computer vision – ECCV 2016. ECCV 2016. Lecture notes in computer science*, Vol. 9905, eds B. Leibe, J. Matas, N. Sebe, and M. Welling (Cham: Springer). doi: 10.1007/978-3-319-46448-0_2

Luo, H., Li, J., and Jia, C. (2021). Rail surface defect detection based on image enhancement and improved cascade R-CNN. *Laser Optoelect. Prog.* 58, 324–335. doi: 10.3788/LOP202158.2212001

Ni, X., Liu, H., Ma, Z., Wang, C., and Liu, J. (2021). Detection for rail surface defects via partitioned edge feature. *IEEE Trans. Int. Trans. Syst.* 23, 5806–5822. doi: 10.1109/TITS.2021.3058635

Piccardi, M. (2004). "Background subtraction techniques: a review," in *Proceeding of the 2004 IEEE international conference on systems, man and cybernetics (IEEE Cat. No.04CH37583)*, 3099–3104.

Popović, Z., Lazarević, L., Mićić, M., and Brajović, L. (2022). Critical analysis of RCF rail defects classification. *Trans. Res. Proc.* 63, 2550–2561. doi: 10.1016/j.trpro.2022.06.294

Redmon, J., and Farhadi, A. (2021). YOLOv3: an incremental improvement. *arxiv* [perprint]. doi: 10.48550/arXiv.1804.02767

Sekar, A., and Perumal, V. (2021). Automatic road crack detection and classification using multi-tasking faster RCNN. *J. Int. Fuzzy Syst.* 41, 6615–6628. doi: 10.3233/JIFS-210475

Shang, L., Yang, Q., Wang, J., Li, S., and Lei, W. (2018). "Detection of rail surface defects based on CNN image recognition and classification," in *Proceeding of the 2018 20th International Conference on Advanced Communication Technology (ICACT)*, 45–51. doi: 10.23919/ICACT.2018.8323642

Wang, Y. D., Zhu, L. Q., Shi, H. M., Fang, E. Q., and Yang, Z. (2018). Vision detection of tunnel cracks based on local image texture calculation. *J. China Railway Soc.* 40, 82–90.

Woo, S., Park, J., Lee, J. Y., and Kweon, I. S. (2018). "CBAM: convolutional block attention module," in *Computer vision – ECCV 2018. ECCV 2018. Lecture notes in computer science*, Vol. 11211, eds V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss (Cham: Springer), doi: 10.1007/978-3-030-01234-2_1

Xu, P., Zeng, H., Qian, T., and Liu, L. (2022). Research on defect detection of high-speed rail based on multi-frequency excitation composite electromagnetic method. *Measurement* 187:110351. doi: 10.1016/j.measurement.2021.110351

Yu, X. Y., Luo, X. Y., Lyu, G. H., and Luo, S. W. (2017). "A novel retinex based enhancement algorithm considering noise," in *Proceeding of the 2017 IEEE/ACIS 16th international conference on computer and information science (ICIS)*, (China), 649–654. doi: 10.1109/TIP.2018.2810539

Yuan, X. C., Wu, L. S., and Chen, H. W. (2016). Rail image segmentation based on Otsu threshold method. *Optics Precis. Eng.* 24, 1772–1781. doi: 10.3788/OPE.20162407.1772

Zhao, Z. (2021). Review of non-destructive testing methods for defect detection of ceramics. *Ceramics Int.* 47, 4389–4397. doi: 10.1016/j.ceramint.2020.10.065

Zhou, M., Wu, Z., Chen, D., and Zhou, Y. (2013). "An improved vein image segmentation algorithm based on SLIC and niblack threshold method," in *proceeding of the 2013 International conference on optical instruments and technology: optoelectronic imaging and processing technology*, (Beijing), 90450D. doi: 10.1117/12.2037345

Zhu, R. N., Guo, Z. Q., and Zhang, X. L. (2021a). Forest 3D reconstruction and individual tree parameter extraction combining close-range photo enhancement and feature matching. *Remote Sens.* 13:1633. doi: 10.3390/rs13091633

Zhu, X. K., Lyu, S. C., Wang, X., and Zhao, Q. (2021b). "TPH-YOLOv5: improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proceeding of the 2021 IEEE/CVF international conference on computer vision workshops (ICCVW)*, (Montreal, BC), 2778–2788. doi: 10.1109/ICCVW54120.2021.00312

TYPE Original Research
PUBLISHED 24 April 2023
DOI 10.3389/fnbot.2023.1143032

A comparative analysis of near-infrared image colorization methods for low-power NVIDIA Jetson embedded systems

Shengdong Shi[1,2], Qian Jiang[1,2], Xin Jin[1,2]*, Weiqiang Wang[1,2], Kaihua Liu[1,2], Haiyang Chen[1,2], Peng Liu[3], Wei Zhou[1,2] and Shaowen Yao[1,2]

[1]Engineering Research Center of Cyberspace, Yunnan University, Kunming, Yunnan, China, [2]School of Software, Yunnan University, Kunming, China, [3]Guangxi Power Grid Co., Ltd., Nanning, China

OPEN ACCESS

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

REVIEWED BY
Zhiqin Zhu,
Chongqing University of Posts and Telecommunications, China
Yu Liu,
Hefei University of Technology, China
Xiaotong He,
Jilin University, China

*CORRESPONDENCE
Xin Jin
✉ xinjin@ynu.edu.cn

RECEIVED 12 January 2023
ACCEPTED 04 April 2023
PUBLISHED 24 April 2023

CITATION
Shi S, Jiang Q, Jin X, Wang W, Liu K, Chen H, Liu P, Zhou W and Yao S (2023) A comparative analysis of near-infrared image colorization methods for low-power NVIDIA Jetson embedded systems.
Front. Neurorobot. 17:1143032.
doi: 10.3389/fnbot.2023.1143032

The near-infrared (NIR) image obtained by an NIR camera is a grayscale image that is inconsistent with the human visual spectrum. It can be difficult to perceive the details of a scene from an NIR scene; thus, a method is required to convert them to visible images, providing color and texture information. In addition, a camera produces so much video data that it increases the pressure on the cloud server. Image processing can be done on an edge device, but the computing resources of edge devices are limited, and their power consumption constraints need to be considered. Graphics Processing Unit (GPU)-based NVIDIA Jetson embedded systems offer a considerable advantage over Central Processing Unit (CPU)-based embedded devices in inference speed. For this study, we designed an evaluation system that uses image quality, resource occupancy, and energy consumption metrics to verify the performance of different NIR image colorization methods on low-power NVIDIA Jetson embedded systems for practical applications. The performance of 11 image colorization methods on NIR image datasets was tested on three different configurations of NVIDIA Jetson boards. The experimental results indicate that the Pix2Pix method performs best, with a rate of 27 frames per second on the Jetson Xavier NX. This performance is sufficient to meet the requirements of real-time NIR image colorization.

KEYWORDS

near-infrared image, image colorization, Jetson, performance evaluation, embedded systems

1. Introduction

In surveillance and vehicle driving scenes (Ni et al., 2022), color image sensors are preferred because their images are close to human visual perception. However, visible images have obvious limitations related to lighting conditions (Yu et al., 2022) and the color of an object's surface (Liao et al., 2022). However, NIR sensors are usually used in night vision and low-illumination scenes because they provide more useful information than visual sensors (Jin et al., 2017). An NIR image is a shaded gray image, which is not in line with human visual habits; so, it is preferable to colorize it, enhancing its color and texture information. Colorized images can improve an observer's ability to assess a scene and increase the efficiency of target detection. The problem of image colorization lies in generating a plausible visible image from only an NIR image (Sun et al., 2019). Thus, NIR image colorization aims to generate a reasonable visible image from an NIR image while preserving the texture in the NIR domain so that the coloring of the converted visible image looks natural.

In the common gray image colorization domain, chromaticity is the only feature that needs to be calculated because the input gray image provides brightness levels. However, the colorized results of NIR images are usually fuzzy and lack high-frequency scene details. Therefore, it is necessary to test the common gray image colorization methods to determine whether they are suitable for NIR image colorization on embedded systems (Liang et al., 2021).

Deep learning models usually require many computing resources (Qin et al., 2020; Fortino et al., 2021), which are deployed on the cloud server. The large amount of data collected by video surveillance equipment needs to be processed by the cloud server (Ma et al., 2018) so that the deep learning model of image processing is affected by network delay (Zhang et al., 2020) or shutdown. Since a deep learning model can be deployed on edge devices that process data in real-time, there is no need to connect the cloud computing platform to process the data from an edge of the network (Han et al., 2020). This would reduce latency and bandwidth costs, improving availability and protecting data privacy and security (Shi et al., 2016). For example, many researchers deploy target detection (Zhao et al., 2019) and visual tracking (Cao et al., 2022) to the edge device for testing and striving for real-time processing.

There has been considerable research that evaluated the effectiveness of various image processing methods (Jin et al., 2017; Liu et al., 2020; Huang et al., 2022). However, most colorization techniques have not been tested for edge devices, and there is no widely recognized system for evaluating these methods on edge devices. However, image colorization has many potential applications on edge devices (Liu et al., 2022). Our study designed an evaluation system to examine the performance of current methods on edge devices. Eleven image colorization methods were tested for NIR image datasets on the Jetson AGX Xavier, Jetson Xavier NX, and Jetson Nano devices. Seven indexes were selected in analyzing the experimental results, and the results using each index were tabulated for evaluating the performance of each method on an edge device.

The contributions of this work are as follows:

We analyzed current image colorization methods to provide guidance in their practical application.

We deployed and tested image colorization methods on three different edge devices and analyzed their resource utilization and energy consumption.

This work inferred general rules and determined key points requiring attention in evaluating the performance of test methods. These were based on the performance of current image colorization methods on edge devices, focusing on resource occupancy, energy consumption, and image quality metrics.

Section 1 summarizes the status of current research on NIR image colorization and the deployment of models on edge devices. Section 2 introduces the structure and operation of the proposed evaluation system and explains why the tested models were chosen. The edge devices used and the evaluation metrics are also described in detail. Experiments on three edge devices and the RGB-NIR scene dataset (Brown and Süsstrunk, 2011) are described in Section 3. Section 4 presents the conclusions of our work and possible directions of future development in this research.

2. Materials and methods

In this work, we designed a system for evaluating the performance of an image colorization method on edge devices, as shown in **Figure 1**. We selected 11 classical image colorization methods based on their network structures, and we briefly introduce these models' structures here. We trained these models using the RGB-NIR scene dataset (Brown and Süsstrunk, 2011) on a server equipped with an RTX3060 GPU to obtain the corresponding model weight files. Then, according to the development of current embedded devices, Nvidia Jetson series edge devices were selected. The Jetson AGX Xavier, Jetson Xavier NX, and Jetson Nano offer high, middle, and low performance levels, respectively. When configuring the software environment of the edge device, we chose the system with the same version number from NVIDIA, which ensures that the software environment for the three edge devices is as similar as possible. According to the environmental requirements of different models, we configured the running environment for each device and compiled the ARM Python package suitable for the particular device. Then, we uploaded the model weight files from the server to each edge device. To better compare the various methods' performance on edge devices, we selected seven evaluation metrics for the experiment. Finally, we analyzed the experimental data and summarized the results of the experiments presented in this paper.

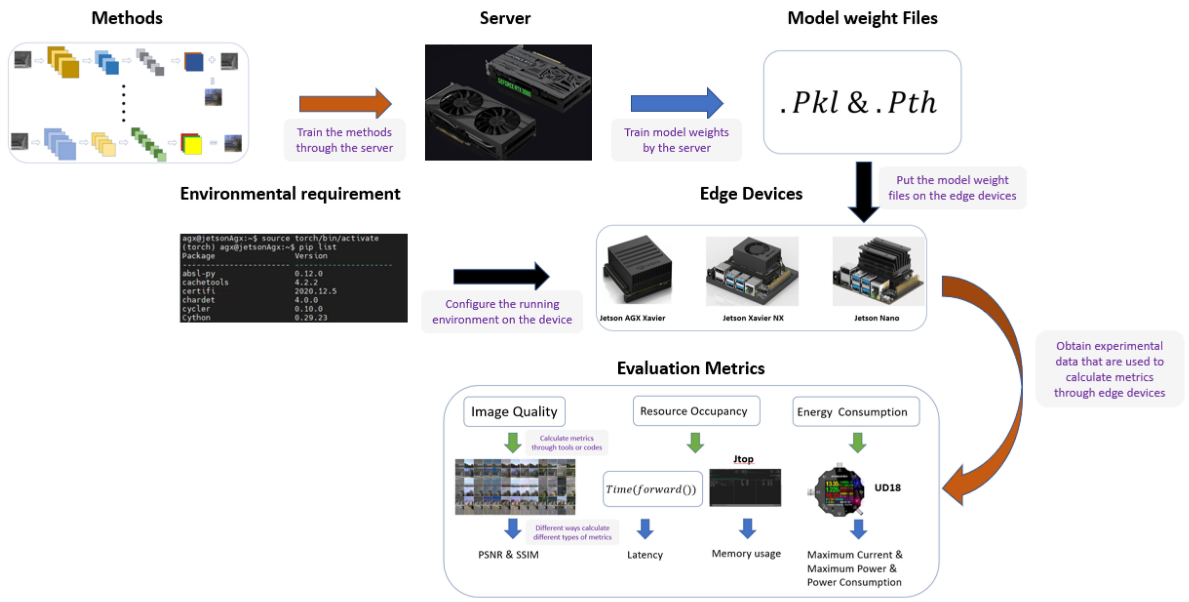
2.1. Image colorization methods

In recent years, methods based on convolutional neural networks (CNNs) have been used extensively in computer vision. ResNet (He et al., 2016) and deep convolution generated adversarial networks (DCGANs) (Radford et al., 2015) are two types of neural networks that have become popular recently. Finding meaningful information in the image is an essential problem in machine vision and image processing research. Attention mechanisms have also attracted the interest of researchers in image processing (Zhu et al., 2023). Many image colorization methods have been proposed based on these structures (Huang et al., 2022).

2.1.1. Convolutional neural network

CICZ (Zhang et al., 2016) is an automatic image colorization method that transforms the colorizing problem into a classification problem by quantifying the color space and combining the method of category balancing, as shown in **Figure 2**. The encoder-decoder structure is adopted. The L channel of the grayscale image is input to predict the a and b channels of the image, and then, the colorized result is obtained.

ELGL (Iizuka et al., 2016) is a fully automatic image colorization method that combines global information and local features, as shown in **Figure 3**. The method first extracts shared low-level features from the image and then uses these features to obtain global image features and middle-level image features. Next,


Methods
Train the methods through the server
Server
Train model weights by the server
Model weight Files
.Pkl & .Pth
Put the model weight files on the edge devices
Environmental requirement
Configure the running environment on the device
Edge Devices
Jetson AGX Xavier
Jetson Xavier NX
Jetson Nano
Obtain experimental data that are used to calculate metrics through edge devices
Evaluation Metrics
Image Quality
Calculate metrics through tools or codes
Different ways calculate different types of metrics
PSNR & SSIM
Resource Occupancy
Time(forward())
Jtop
Latency
Memory usage
Energy Consumption
UD18
Maximum Current & Maximum Power & Power Consumption

```
agx@jetsonAgx:~$ source torch/bin/activate
(torch) agx@jetsonAgx:~$ pip list
Package                  Version
------------------------ --------------------
absl-py                  0.12.0
cachetools               4.2.2
certifi                  2020.12.5
chardet                  4.0.0
cycler                   0.10.0
Cython                   0.29.23
```

FIGURE 1
Architecture of the evaluation system.

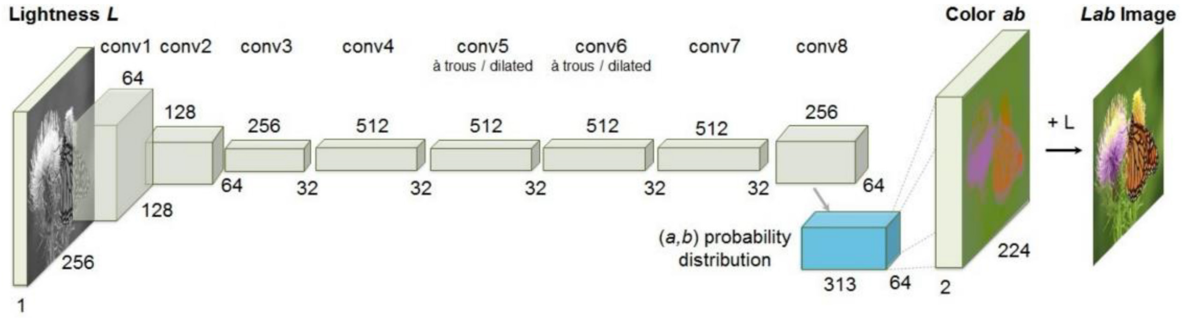


FIGURE 2
Architecture of CICZ (Zhang et al., 2016).

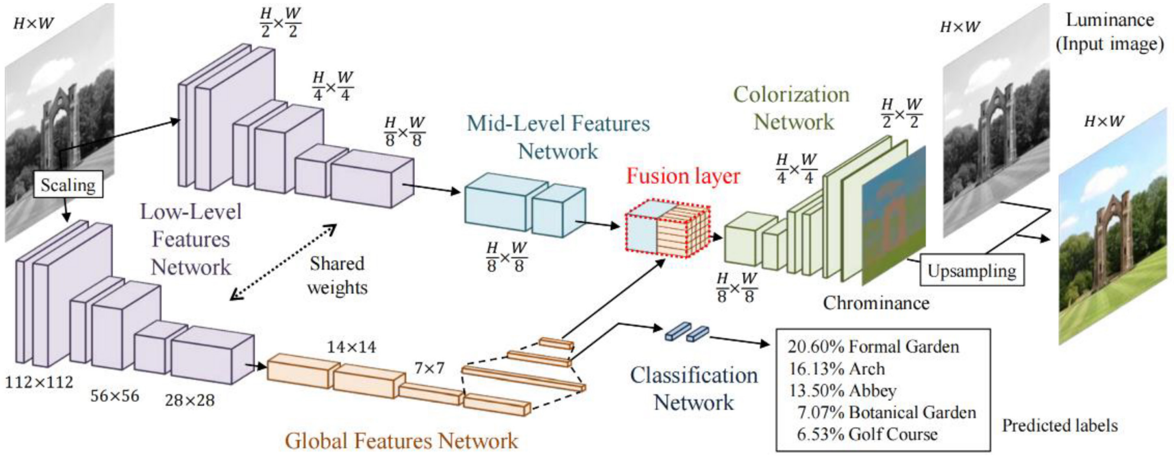


FIGURE 3
Architecture of ELGL (Iizuka et al., 2016).

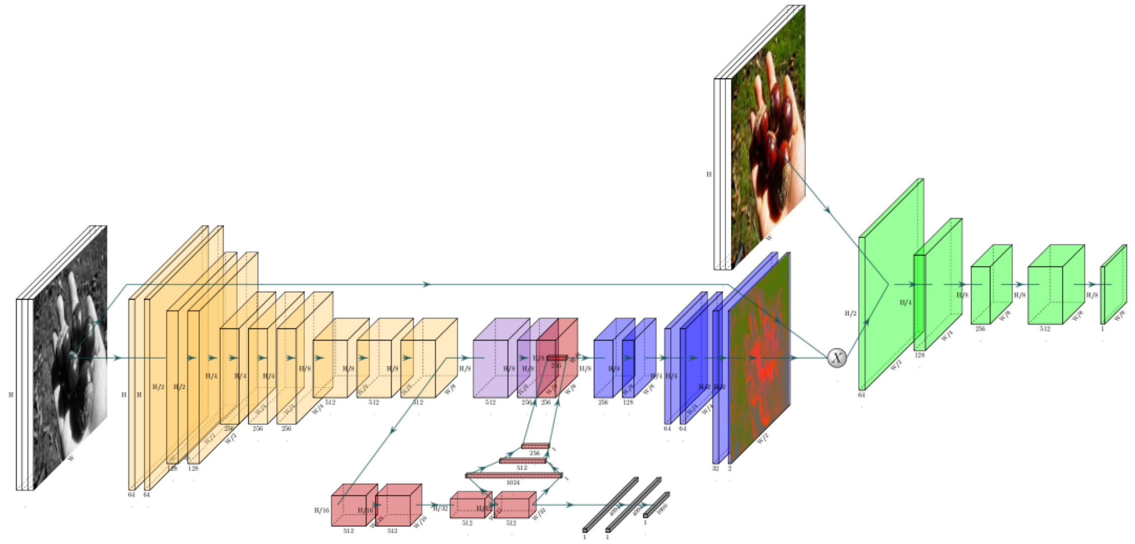


FIGURE 4
Architecture of ChromaGAN (Vitoria et al., 2020).

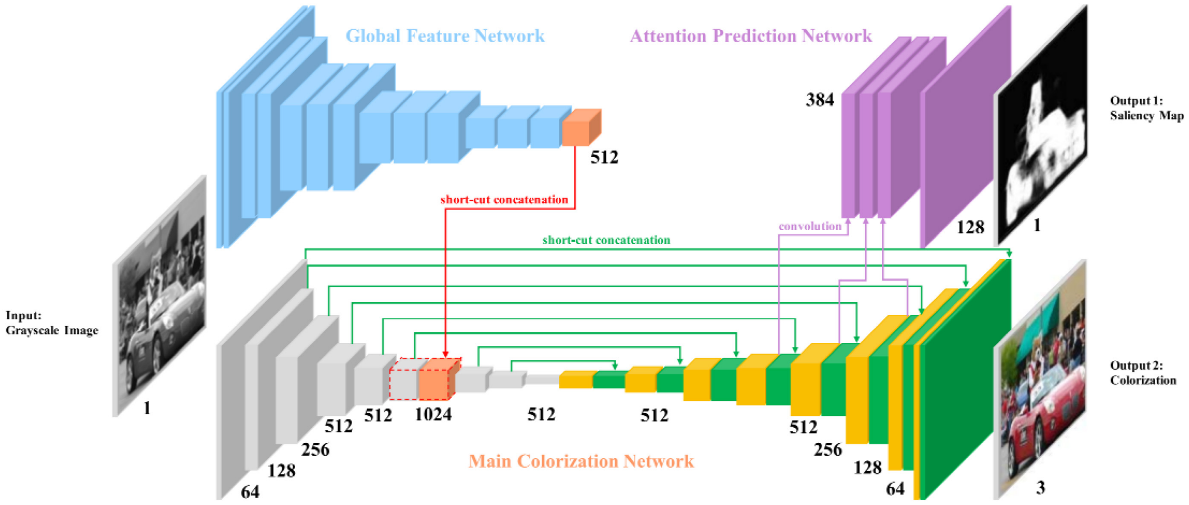


FIGURE 5
Architecture of SCGAN (Zhao et al., 2020)'s generator.

the shallow and global features are fused through the fusion layer, which inputs the result to the colorization network and outputs the final chrominance information.

2.1.2. Wasserstein generated adversarial network

ChromaGAN (Vitoria et al., 2020) is an adversarial learning colorization method that infers the chromaticity of a given grayscale image according to semantic clues. In the adversarial network-based method, a three-term loss function combining color, perceptual information, and semantic category distribution was proposed. A self-supervised strategy is used to train the model. The discriminator is based on Markovian architecture [PatchGAN (Isola et al., 2017)]. **Figure 4** shows the method's block diagram.

SCGAN (Zhao et al., 2020) is an automatic saliency map-guided colorization method with a generative adversarial network. It combines predictive colorizing and saliency maps to minimize semantic confusion and color bleeding in the colorized image, as shown in **Figure 5**. The global features of the pre-trained VGG-16-Gray network were embedded in the color encoder. Branches of the color decoder are used to predict saliency maps as proxy targets. Then, the method uses two hierarchical discriminators to distinguish between the generated colorized result and saliency maps, as shown in **Figure 6**.

2.1.3. Conditional generated adversarial network

Pix2Pix (Isola et al., 2017) is based on the idea of a conditional generated adversarial network (CGAN). Generator G uses the U-Net structure. The input contour map x is encoded and decoded into a real image. The discriminator D uses the condition discriminator PatchGAN proposed by the author himself. The

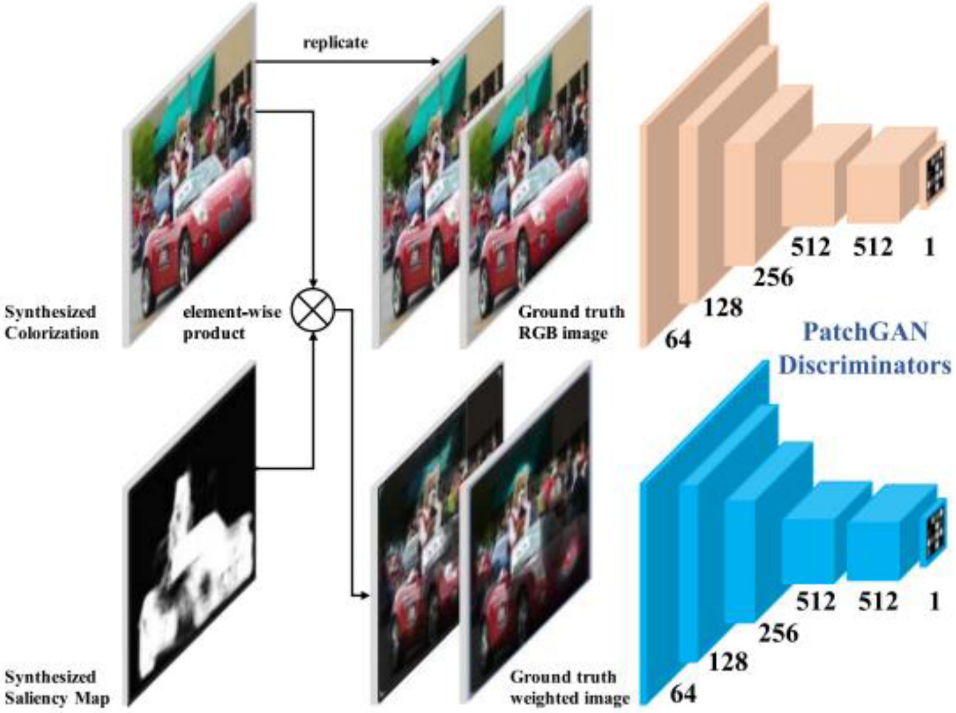


FIGURE 6
Architecture of SCGAN (Zhao et al., 2020)'s discriminator.

function of discriminator D is to judge the generated image as false and the real image as true under the condition of the contour map *x*. Figure 7 shows the structure of Pix2Pix.

MemoPainter (Yoo et al., 2019) is a novel storage memory-enhanced colorizing model that obtains the given color information in the training set with the memory network by querying to guide colorizing. This model can generate high-quality colorized images from limited data and proposes a novel threshold triplet loss, which can complete unsupervised training of storage networks under classless labels. MemoPainter's architecture is shown in Figure 8.

TIC-CGAN (Kuang et al., 2020) uses a detail-preserving coarse-to-fine generator to learn transformation mapping, as shown in Figure 9. The method proposes a composite loss function that integrates content, adversarial, perceptual, and total variation loss. Content loss is used to restore global image information, and the other three losses synthesize local realistic textures.

2.1.4. Cycle-consistent adversarial network

CycleGAN (Zhu et al., 2017) is an unsupervised GAN. Its main idea is to train two pairs of generator-discriminator models (two mapping functions G: X—> Y and F: Y—> X) to convert images from one domain to another. In this process, two cycle-consistency losses are introduced to ensure that the generator does not convert an image from one domain to another that is entirely unrelated to the original image. The architecture of CycleGAN is shown in Figure 10.

RecycleGAN (Bansal et al., 2018) is an unsupervised data-driven method for video redirection that combines spatial and temporal information and adversarial loss for content translation and style retention for video redirection. The method proves that under different conditions, the use of time information provides more constraints for optimizing the transformation from one domain to another, which helps to obtain better local minima. The combination of temporal and spatial constraints helps to learn the style characteristics of a given domain. The difference in design between this method and CycleGAN (Zhu et al., 2017) and Pix2Pix (Isola et al., 2017) is shown in Figure 11.

PearlGAN (Luo et al., 2022) is a GAN based on top-down attention and gradient alignment. First, a top-down guided attention module and an elaborate attentional loss reduce semantic coding ambiguity during translation. Then, the model introduces a structured gradient alignment loss to encourage edge consistency between transmissions. The internal structure of PearlGAN is shown in Figure 12.

I2V-GAN (Li et al., 2021) is an infrared-to-visible video conversion method that generates fine-grained and spatiotemporally consistent visible video from a given unpaired infrared video, as shown in Figure 13. The model utilizes adversarial constraints to generate a synthetic frame similar to the real frame and then introduces the circular consistency of perceptual loss for effective content transformation and style preservation. Finally, it utilizes the similarity constraints between and within domains to enhance the content and motion consistency of space and time-space at the fine-grained level.

2.2. Edge devices

While the Raspberry Pi offers low power consumption and energy-saving performance, NVIDIA Jetson platforms

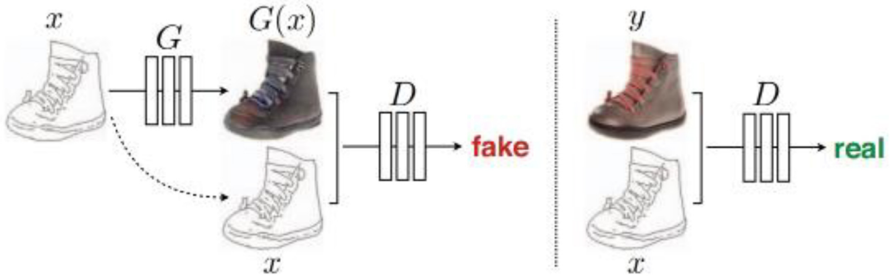


FIGURE 7
Architecture of Pix2Pix (Isola et al., 2017).

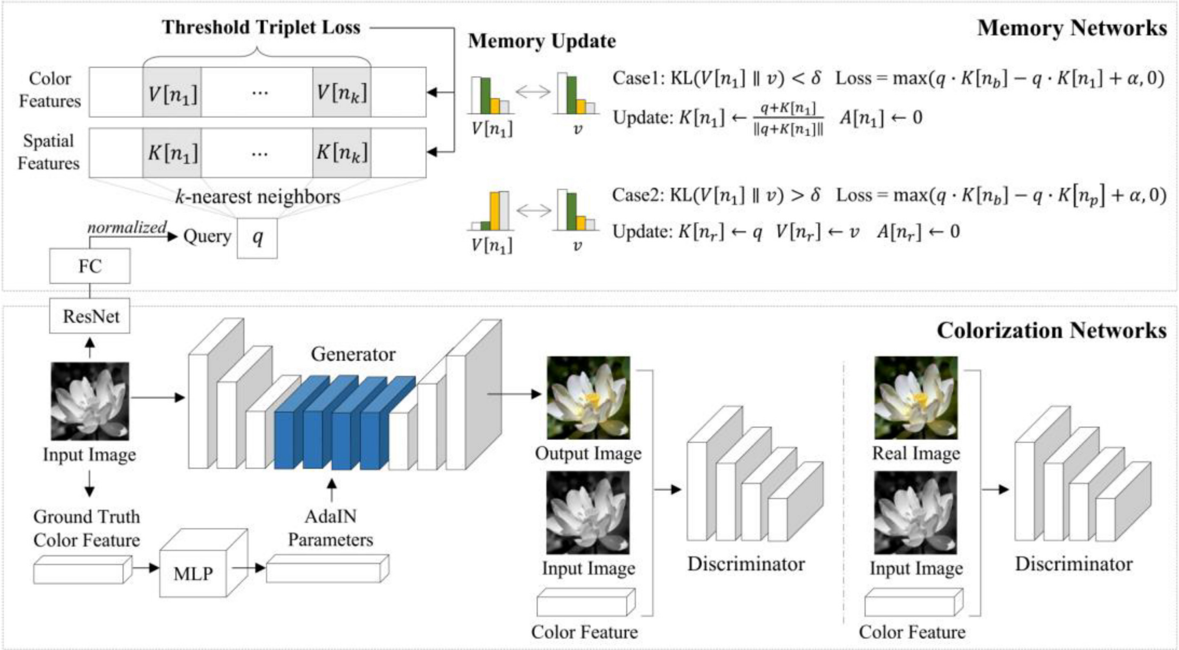

Threshold Triplet Loss
Memory Update
Memory Networks
Color Features
$V[n_1]$
···
$V[n_k]$
Spatial Features
$K[n_1]$
···
$K[n_k]$
k-nearest neighbors
normalized
Query
q
FC
ResNet
$V[n_1]$
v
Case1: $\mathrm{KL}(V[n_1] \parallel v) < \delta$ Loss = $\max(q \cdot K[n_b] - q \cdot K[n_1] + \alpha, 0)$
Update: $K[n_1] \leftarrow \frac{q+K[n_1]}{\|q+K[n_1]\|}$ $A[n_1] \leftarrow 0$
$V[n_1]$
v
Case2: $\mathrm{KL}(V[n_1] \parallel v) > \delta$ Loss = $\max(q \cdot K[n_b] - q \cdot K[n_p] + \alpha, 0)$
Update: $K[n_r] \leftarrow q$ $V[n_r] \leftarrow v$ $A[n_r] \leftarrow 0$
Colorization Networks
Generator
Input Image
Ground Truth Color Feature
MLP
AdaIN Parameters
Output Image
Input Image
Color Feature
Discriminator
Real Image
Input Image
Color Feature
Discriminator

FIGURE 8
Architecture of MemoPainter (Yoo et al., 2019).

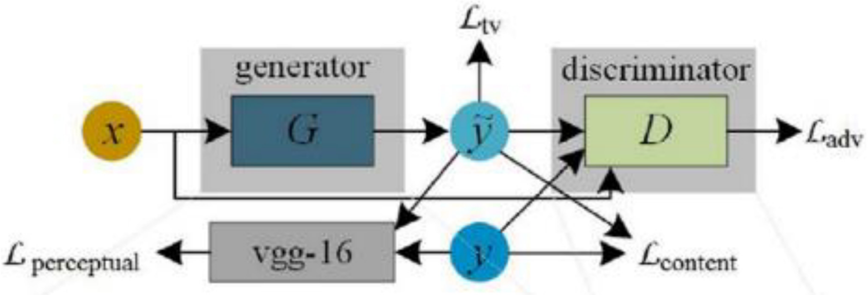

$\mathcal{L}_{\mathrm{tv}}$
generator
discriminator
x
G
$\tilde{y}$
D
$\mathcal{L}_{\mathrm{adv}}$
$\mathcal{L}_{\mathrm{perceptual}}$
vgg-16
y
$\mathcal{L}_{\mathrm{content}}$

FIGURE 9
Architecture of TIC-CGAN (Kuang et al., 2020).

have a higher GPU speed, leading to better deep learning inference performance. The security and reliability of the NVIDIA Jetson series make it possible to deploy deep learning models in harsh environments; hence, the Jetson series of edge devices have been used in many industrial fields. The Jetson platform is compatible with the Jet Pack software development kit, which includes libraries for deep learning, such as computer vision and accelerated computing. By using

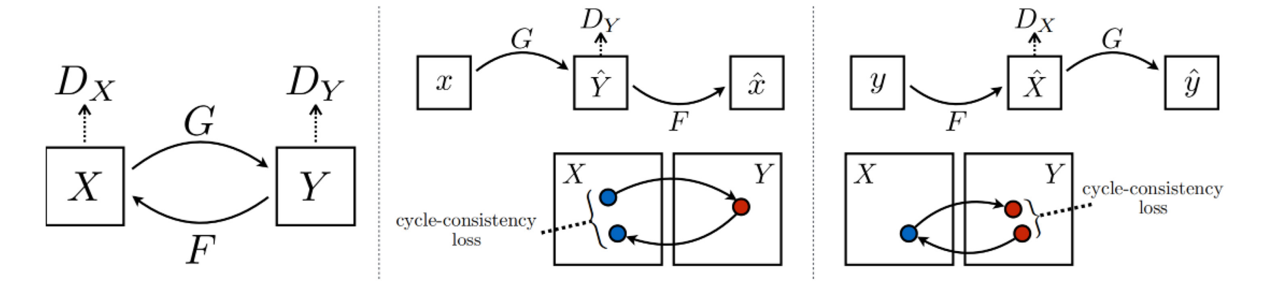


FIGURE 10
Architecture of CycleGAN (Zhu et al., 2017).

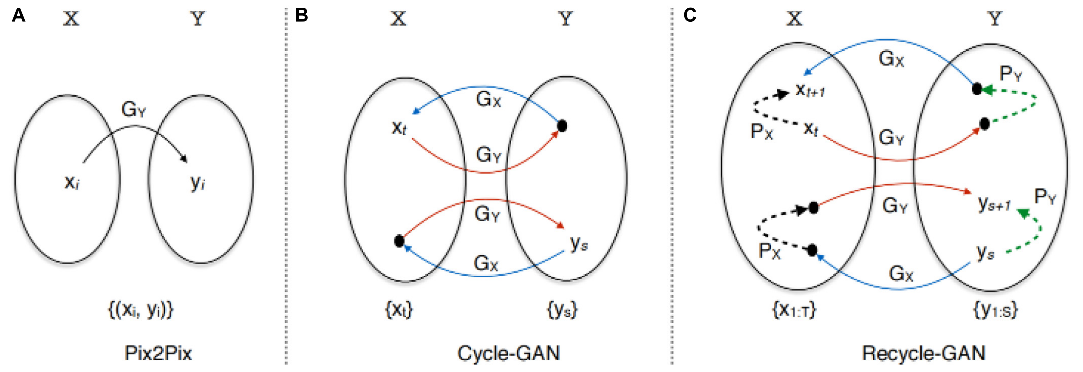


FIGURE 11
Architecture of RecycleGAN (Bansal et al., 2018).

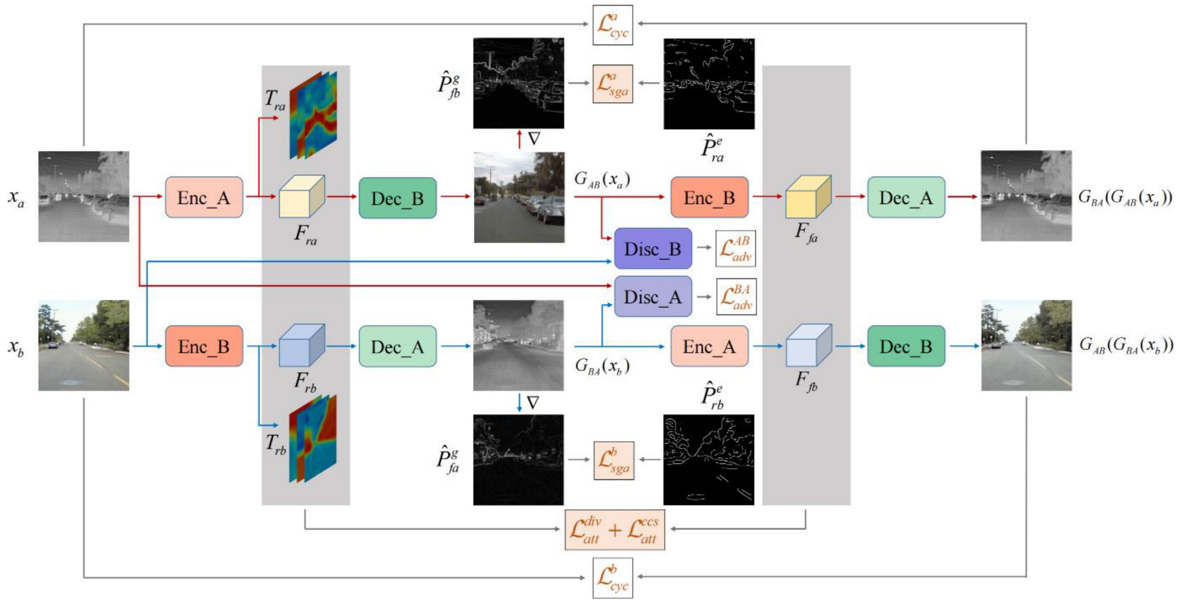


FIGURE 12
Architecture of PearlGAN (Luo et al., 2022).

the same version of the NVIDIA official system, we can maintain consistency in the experimental environment to a certain degree. Thus, we test the performance of different models on Jetson AGX Xavier, Jetson Xavier NX, and Jetson Nano devices that belong to three edge devices of high, middle, and low-performance levels. It is appropriate to compare the performance of different models under the constraints of different hardware conditions.

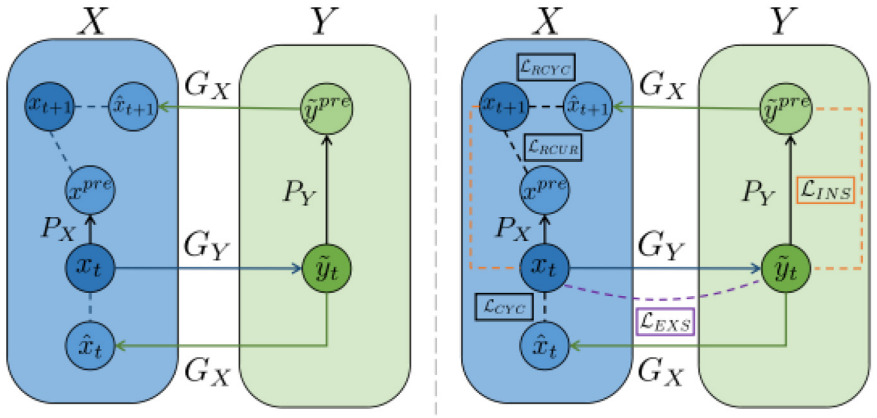


FIGURE 13
Architecture of I2V-GAN (Li et al., 2021).

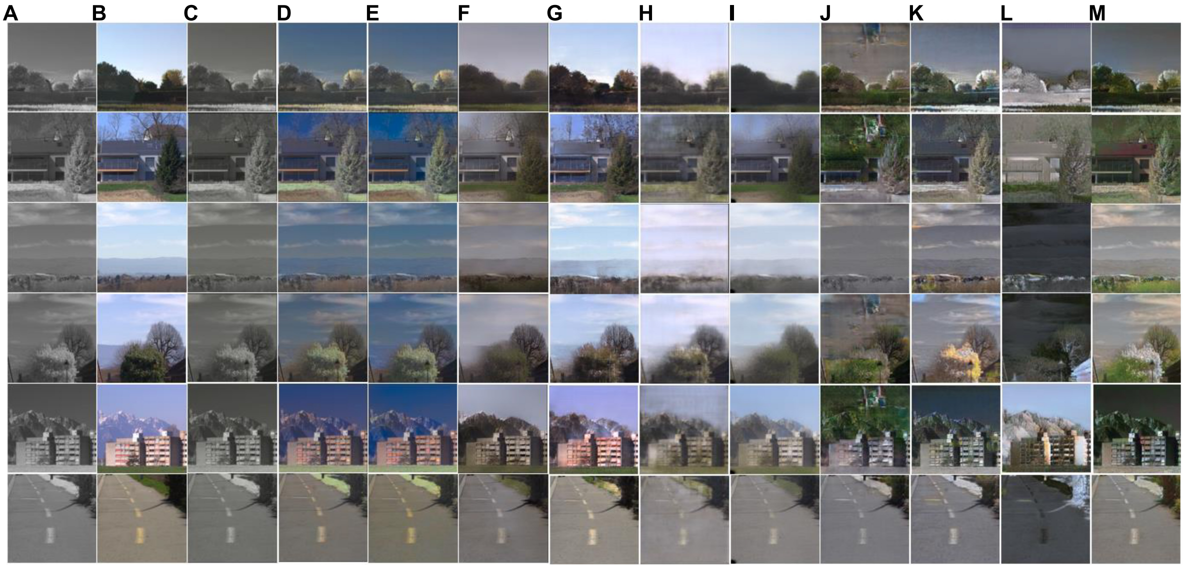


FIGURE 14
Subjective comparison of near-infrared (NIR) image colorization effects of different models on RTX3060. **(A)** Input; **(B)** label; **(C)** CICZ (Zhang et al., 2016); **(D)** ELGL (Iizuka et al., 2016); **(E)** ChromaGAN (Vitoria et al., 2020); **(F)** SCGAN (Zhao et al., 2020); **(G)** Pix2Pix (Isola et al., 2017); **(H)** MemoPainter (Yoo et al., 2019); **(I)** TIC-CGAN (Kuang et al., 2020); **(J)** CycleGAN (Zhu et al., 2017); **(K)** RecycleGAN (Bansal et al., 2018); **(L)** PearlGAN (Luo et al., 2022); **(M)** I2V-GAN (Li et al., 2021).

2.2.1. Jetson AGX Xavier

Jetson AGX Xavier is a 30 W GPU workstation from NVIDIA that was launched in December 2018. Its CPU is eight-core ARM NVIDIA Carmel, the GPU is NVIDIA Volta architecture with 512 NVIDIA CUDA cores, and the memory is 32 GB LRDDR4x. Jetson AGX Xavier provides good memory bandwidth and computing performance. It has a computing speed of up to 32 TOPS (30 W) in deep learning and computer vision tasks. For image processing tasks, real-time effects can be achieved on some models (Mazzia et al., 2020; Jeon et al., 2021).

2.2.2. Jetson Xavier NX

Jetson Xavier NX is a mid-end product launched by NVIDIA in November 2019. Its CPU is 6-core ARM NVIDIA Carmel, the GPU is NVIDIA Volta architecture with 384 NVIDIA CUDA cores, and the memory is eight GB LRDDR4x. Due to the Volta architecture, it has a server-level performance of up to 21 TOPS (15 W) or 14 TOPS (10 W). For image processing tasks, Jetson Xavier NX already offers the performance requirements of most models (Jeon et al., 2021).

2.2.3. Jetson Nano

Jetson Nano is an entry-level product launched by NVIDIA in March 2019. Its CPU is four-core ARM A57, the GPU is NVIDIA Maxwell architecture with 128 NVIDIA CUDA cores, and the memory is four GB LRDDR4, which supports switching between 5 W and 10 W modes. The Jetson Nano has the lowest performance in the series at only 0.5 TFLOPS, but it also has the lowest price and power consumption, making it more suitable for use in less-demanding edge scenes. The Jetson Nano is unsuitable for infrared image colorization, mainly

TABLE 1 Evaluation of different image colorization models based on Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) on Jetson AGX Xavier, Jetson Xavier NX, Jetson Nano, and RTX3060 devices.

Devices	RTX3060	AGX	NX	Nano
Method	PSNR/SSIM			
CICZ	14.249/0.565	14.265/0.565	14.265/0.565	14.265/0.565
ELGL	14.834/0.572	14.806/0.572	14.834/0.572	14.834/0.572
ChromaGAN	14.902/0.569	14.902/0.570	14.696/0.564	14.902/0.570
SCGAN	16.714/0.621	16.714/0.621	16.714/0.621	16.714/0.621
Pix2Pix	22.140/0.580	22.139/0.580	22.122/0.579	22.122/0.580
MemoPainter	18.645/0.535	18.645/0.535	18.645/0.535	–
TIC-CGAN	20.589/0.642	20.590/0.642	20.590/0.642	20.590/0.642
CycleGAN	14.139/0.535	14.139/0.535	14.139/0.535	14.139/0.535
RecycleGAN	14.083/0.474	14.087/0.472	14.098/0.472	14.087/0.471
PearlGAN	13.548/0.475	13.536/0.474	13.536/0.474	13.536/0.474
I2V-GAN	13.637/0.485	13.614/0.485	13.599/0.484	13.631/0.484

TABLE 2 Evaluation of different image colorization models based on latency and Frames Per Second (FPS) on Jetson AGX Xavier, Jetson Xavier NX, and Jetson Nano.

Devices	AGX	NX	Nano	AGX	NX	Nano
Method	Latency (s)			FPS		
CICZ	0.157	0.263	0.831	6.370	3.801	1.203
ELGL	0.021	0.043	0.191	46.823	23.397	5.244
ChromaGAN	0.035	0.072	0.270	28.387	13.805	3.703
SCGAN	0.159	0.303	1.221	6.305	3.301	0.819
Pix2Pix	0.022	0.036	0.146	44.986	27.426	6.863
MemoPainter	0.063	0.109	–	15.806	9.164	–
TIC-CGAN	0.044	0.081	0.412	22.731	12.288	2.427
CycleGAN	0.022	0.038	0.157	44.900	26.395	6.351
RecycleGAN	0.250	0.433	2.291	4.006	2.309	0.436
PearlGAN	0.147	0.252	1.058	6.799	3.968	0.945
I2V-GAN	0.226	0.388	2.051	4.433	2.574	0.488

playing a comparative role in the experiment (Mazzia et al., 2020).

2.3. Evaluation metrics

In research on deploying deep learning methods in edge devices, the allocation of computing resources is a crucial concern. The choice of resources varies depending on the specific scenario. Computing resources such as CPU, GPU, and memory are considered for computing-sensitive tasks (Toczé and Nadjm-Tehrani, 2018). Storage and communication resources such as IO, hard disk, spectrum, and bandwidth are considered for data-sensitive tasks (Toczé and Nadjm-Tehrani, 2018).

The evaluation metrics selected in this work include Peak Signal to Noise Ratio (PSNR), Structural Similarity (SSIM), latency, memory usage, maximum current, maximum power, and power consumption. We have considered CPU occupancy, but in the actual test process, the occupancy rate is difficult to evaluate as a metric because of its multi-core architecture.

TABLE 3 Evaluation of different image colorization models based on RAM and maximum current (Imax) on Jetson AGX Xavier, Jetson Xavier NX, and Jetson Nano.

Devices	AGX	NX	Nano	AGX	NX	Nano
Method	RAM (GB)			Imax (A)		
CICZ	15.150	4.440	2.520	1.370	0.650	1.460
ELGL	16.010	4.640	3.050	1.840	0.860	1.510
ChromaGAN	16.530	4.810	2.950	1.970	0.890	1.650
SCGAN	15.170	4.380	2.740	1.980	0.910	1.710
Pix2Pix	6.200	3.780	2.470	1.630	0.800	1.610
MemoPainter	8.160	5.230	–	1.680	0.770	–
TIC-CGAN	6.110	4.750	2.790	1.730	0.820	1.590
CycleGAN	6.560	4.060	2.840	1.650	0.810	1.600
RecycleGAN	5.930	3.530	2.530	1.770	0.840	1.610
PearlGAN	5.850	2.830	1.960	1.700	0.850	1.690
I2V-GAN	5.730	3.220	2.350	1.730	0.840	1.540

2.3.1. Image quality

Peak Signal to Noise Ratio is generally used between the maximum signal and background noise. Usually, after image processing, the processed image x_1 will be different from the original image x_2. To measure the quality of the processed image, we usually refer to the PSNR value to measure whether a processing program is satisfactory. PSNR's formula is shown in Equation 1. $MAX^2_{x_1}$ represents the maximum pixel value of the processed image x_1. The size of the processed image x_1 and original image x_2 is m*n. PSNR can be calculated as follows:

$$PSNR = 10^{*}\log_{10}\left(\frac{MAX^2_{x_1}}{\frac{1}{m^{*}n}{}^{*}\sum_{i=1}^{m}\sum_{j=1}^{n}\left[x_2(i,j)-x_1(i,j)\right]^2}\right). \quad (1)$$

Structural Similarity is a metric that considers luminance, contrast, and structure. The SSIM value of two images is calculated using the original image x_2 and the processed image x_1. SSIM can measure the degree of distortion and the similarity between the two images. SSIM ranges from −1 to 1. When two images are the same, the SSIM value is 1. SSIM's formula is shown in Equation 2. $l(x_2, x_1)$ represents the luminance contrast function. $c(x_2, x_1)$ represents the contrast function. $s(x_2, x_1)$ represents the structural contrast function. μ_{x_2} and μ_{x_1} represent the averages of x_2 and x_1, respectively. σ_{x_2} and σ_{x_1} represent the variances of x_2 and x_1, respectively. $\sigma_{x_2x_1}$ represents the covariances of x_2 and x_1. θ_1, θ_2, and θ_3 are designed with three constants to avoid zero denominators. SSIM is given by

$$SSIM\left(x_2, x_1\right) = \left[l\left(x_2, x_1\right)\right]^{\alpha}\left[c\left(x_2, x_1\right)\right]^{\beta}\left[s\left(x_2, x_1\right)\right]^{\gamma}, where \quad (2)$$

$$l\left(x_2, x_1\right) = \frac{2\mu_{x_2}\mu_{x_1}+\theta_1}{\mu^2_{x_2}+\mu^2_{x_1}+\theta_1}, \quad (3)$$

$$c\left(x_2, x_1\right) = \frac{2\sigma_{x_2}\sigma_{x_1}+\theta_2}{\sigma^2_{x_2}+\sigma^2_{x_1}+\theta_2}, and \quad (4)$$

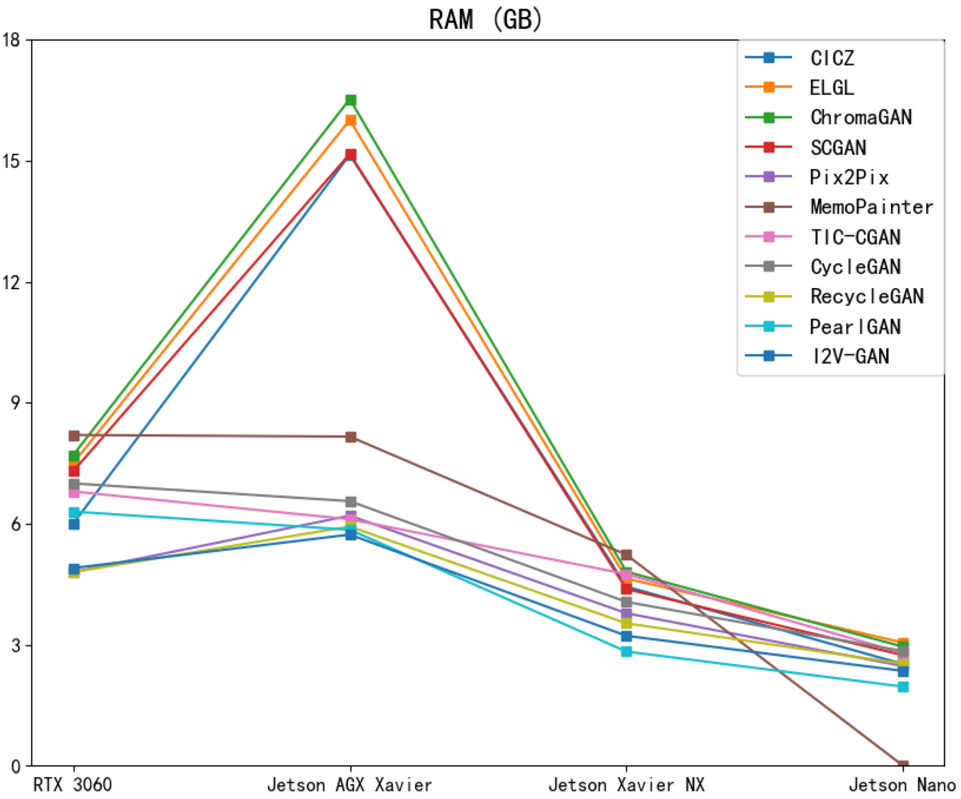


FIGURE 15
Performance of different image colorization models on Jetson AGX Xavier, Jetson Xavier NX, Jetson Nano, and RTX3060 devices for RAM metrics.

$$s(x_2, x_1) = \frac{\sigma_{x_2 x_1} + \theta_3}{\sigma_{x_2}\sigma_{x_1} + \theta_3}. \quad (5)$$

2.3.2. Resource occupancy

By comparing the latency of different models, a suitable model is selected to deploy in different industrial application scenarios (Cao et al., 2022). Meanwhile, the memory of edge devices is a scarce resource because multiple models with different purposes may need to be deployed. By clarifying the memory usage of different models, we can select a suitable model without affecting the deployment of other models.

Latency refers to the average time consumed per image when the model colorizes the image continuously. Because the time consumed is the same as different models have the same operation when reading and saving images, we only calculate the time consumed in generating the colorized image [*forward*()]. We tested 20 NIR images 100 times to calculate the accurate latency and then averaged them. The formula to calculate latency is shown in Equation 6. *a* represents the number of different images used to calculate the latency. *b* represents the number of times the same image runs *forward*(). *Time*() represents the time calculation function. Frames Per Second (FPS) is also used in this article to represent inference speed, as shown in Equation 7. The formulas are as follows:

$$Latency = \frac{1}{a}\sum\nolimits_{i=1}^{a} \frac{Time(forward()*b)}{b} \text{ and} \quad (6)$$

$$FPS = \frac{1}{Latency}. \quad (7)$$

Memory usage refers to the occupied memory monitored by the system process Jtop during the model test. We use RAM to denote the occupied memory in the experiment, including the video memory of the Jetson device, which is also calculated as a part of memory. In contrast, the video memory of the server with RTX 3,060 is calculated separately; so, when comparing the results, the sum of the memory usage and the video memory usage of the server is calculated. To test the accurate memory usage of the colorizing image, we continue to colorize the image for 180 s. The test results are the increase in memory usage from reading an NIR image to outputting a colorized image.

2.3.3. Energy consumption

In laboratory studies, we usually do not consider the energy consumed by the model operation. In the actual application scenario, users take the energy consumption problem seriously. Therefore, recording the model's energy consumption when deployed on edge devices makes sense.

The maximum current (Imax) refers to the maximum current recorded. The maximum power (Pmax) is the product of the maximum current and voltage. Power consumption (PC) refers to the total power consumption of the model running on the edge device for a certain time. The UD18 detector measures these three metrics during the model test. To test the accurate data, we need only to test the function of colorizing images and can continue

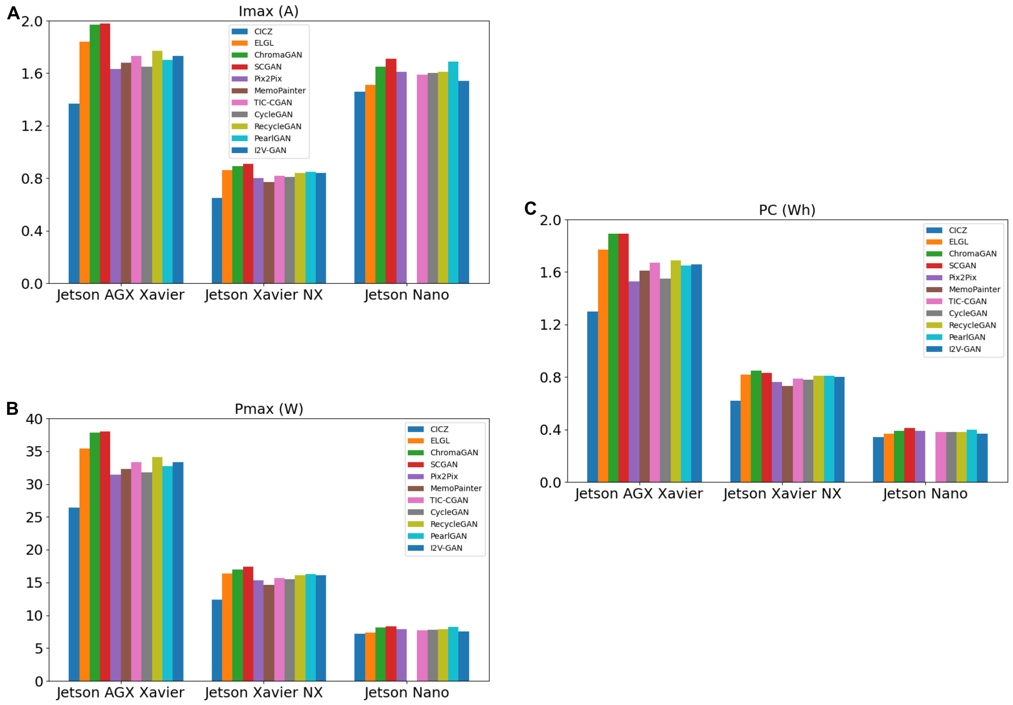


FIGURE 16
Performance of different image colorization models on Jetson AGX Xavier, Jetson Xavier NX, and Jetson Nano for maximum current (Imax) **(A)**, maximum power (Pmax) **(B)**, and power consumption (PC) **(C)**.

to colorize the image for 180 s. It measures the process of end-to-end inference, from reading an NIR image to outputting a colorized image.

3. Results and analysis

We trained the different methods using a computer with an AMD Ryzen7 5800H 3.2 GHz CPU and one NVIDIA Geforce RTX 3060 GPU. We compared the following methods [CICZ (Zhang et al., 2016), ELGL (Iizuka et al., 2016), ChromaGAN (Vitoria et al., 2020), SCGAN (Zhao et al., 2020), Pix2Pix (Isola et al., 2017), MemoPainter (Yoo et al., 2019), TIC-CGAN (Kuang et al., 2020), CycleGAN (Zhu et al., 2017), RecycleGAN (Bansal et al., 2018), PearlGAN (Luo et al., 2022), and I2V-GAN (Li et al., 2021)] on three different edge devices based on the selected metrics.

TABLE 4 Evaluation of different image colorization models based on maximum power (Pmax) and power consumption (PC) on Jetson AGX Xavier, Jetson Xavier NX, and Jetson Nano.

Devices	AGX	NX	Nano	AGX	NX	Nano
Method	Pmax (W)			PC (Wh)		
CICZ	26.400	12.400	7.200	1.300	0.620	0.340
ELGL	35.400	16.400	7.400	1.770	0.820	0.370
ChromaGAN	37.800	17.000	8.100	1.890	0.850	0.390
SCGAN	38.000	17.400	8.300	1.890	0.830	0.410
Pix2Pix	31.400	15.300	7.900	1.530	0.760	0.390
MemoPainter	32.300	14.600	–	1.610	0.730	–
TIC-CGAN	33.300	15.700	7.700	1.670	0.790	0.380
CycleGAN	31.800	15.500	7.800	1.550	0.780	0.380
RecycleGAN	34.100	16.100	7.900	1.690	0.810	0.380
PearlGAN	32.700	16.300	8.200	1.650	0.810	0.400
I2V-GAN	33.300	16.100	7.500	1.660	0.800	0.370

3.1. Experimental dataset

We used the RGB-NIR scene dataset (Brown and Süsstrunk, 2011), which contains 477 image pairs with a resolution of 1,024 × 680 captured from nine scene categories. Image scene categories were villages, fields, forests, indoors, mountains, ancient buildings, streets, cities, and water. The image pairs in this dataset are coarsely registered using a global calibration method; so, pixel-level registration could not be guaranteed. We cropped each of the nine types of scene images to 256 × 256 and did a mirror flip. Then, we selected two types of scene images, fields, and streets, to merge as the training set and test set of the experiment, for a total of 5,616 RGB-NIR image pairs. Among them, 5,460 image pairs were used as the training set and 156 were used as the test set.

3.2. Experimental environment

The basic configuration of the operating environment of the edge device is the same. The system is Ubuntu 18.04 for ARM, the Jet Pack version is 4.5, the CUDA version is 10.2, the cuDNN version is 8.0.0, the OpenCV version is 4.1.1, and the TensorRT version is 7.1.3. The selected configuration is currently more stable because different system versions and dependent environments impact device performance.

3.3. Subjective assessment

As shown in Figure 14, different methods perform quite differently on the RGB-NIR scene dataset used in this work. Pix2Pix has the best image effect, closest to the visible image, as shown in Figure 14G. TIC-CGAN's performance is slightly blurrier than that of Pix2Pix. The image effect of the MemoPainter is different from the color of the visible image, and the image effect of SCGAN is dark. The subjective evaluation of the models based on Cycle-Consistent Adversarial Networks (CycleGAN, RecycleGAN, PearlGAN, I2V-GAN) is poor, especially in the images shown in Figure 14L. The reason is that the number of training sets is small, and the network cannot learn representative features. The CICZ does not learn helpful information on the datasets used in this work, resulting in subjective evaluation close to NIR images, as shown in Figure 14C. ELGL and ChromaGAN directly combine the L-channels of the NIR image during colorization to preserve details but with severe color deviations.

3.4. Objective assessment

3.4.1. Image quality

We found that in the 11 models tested, the results of their image quality metrics on different edge devices were the same with only a few subtle differences; so, we only compared the test results on the RT3060 device. As shown in Table 1, from the image quality metrics, PSNR and SSIM, Pix2Pix, and TIC-CGAN have the best results, followed by MemoPainter. Part of the reason for the poor performance of CNN methods is that they combine L-channels, the brightness of NIR images when they finally generate the colorized images. This results in a significant difference between them and visible images.

3.4.2. Resource occupancy

As shown in Table 2, the latency and the inference speed of the 11 compared models vary significantly across different edge devices. We found that, as the performance of the devices decreases, the ratio of latency difference to each other also narrows. ELGL (46.8 FPS), Pix2Pix (45.0 FPS), CycleGAN (44.9 FPS), ChromaGAN (28.4 FPS), and TIC-CGAN (22.7 FPS) achieve real-time colorization on the Jetson AGX Xavier. Pix2Pix (27.4 FPS), CycleGAN (26.4 FPS), and ELGL (23.4 FPS) can achieve real-time colorization on the Jetson Xavier NX. The fastest on the Jetson Nano is Pix2Pix (6.8 FPS), followed by CycleGAN (6.3 FPS). We found that the fastest model to run on high-performance devices does not necessarily represent the fastest model to run on low-performance devices. Combined with the data in Figure 14, we believe that the running speed of a model with larger memory usage may be significantly affected when the memory resources are limited.

The initial memory usage of the server is 7.2 GB, the initial memory usage of Jetson AGX Xavier is 0.72 GB, the initial memory usage of Jetson Xavier NX is 0.56 GB, and the initial memory usage of Jetson Nano is 0.52 GB. The RAM values in Table 3 are the measured values minus the initial memory usage. As shown in Table 3, Figure 15, when the model is deployed on edge devices with sufficient running memory, it will occupy more than those with limited memory. This phenomenon may be related to the memory invocation principle of the PyTorch framework. I2V-GAN consumes the least memory on Jetson AGX Xavier. PearlGAN consumes the least memory on Jetson Xavier NX and Jetson Nano. MemoPainter cannot be run on Jetson Nano due to excessive memory usage.

3.4.3. Energy consumption

As shown in Figures 16B, C, the comparison model's performance of the maximum power and total power consumption has the same trend. Since both Jetson AGX Xavier and Jetson Xavier NX are rated at 19 V and Jetson Nano is rated at 5 V, the maximum current of the model on Jetson Nano is higher than that on Jetson Xavier NX when the performance is limited, as shown in Figure 16A and Table 3. CICZ has the smallest energy consumption per unit time when it runs on the three edge devices, as shown in Table 4. The total power consumption is the power consumption in a certain period rather than the power consumption of each inference. Therefore, when selecting a model on an edge device with limited energy, we had to consider both the model's latency (or FPS) and energy consumption metrics.

3.4.4. Equilibrium assessment

From the results shown in Table 5, we believe that, if a model is suitable for running on edge devices, it requires a balance between the quality of colorized results and the inference speed. In general, on the edge device (Jetson Xavier NX), Pix2Pix can achieve real-time NIR image colorization requirements and has good image quality, as shown in Figure 17A. TIC-CGAN is slightly inferior in terms of latency. The performance differences between RecycleGAN, PearlGAN, and I2V-GAN are insignificant, as shown in Figure 17B.

TABLE 5 Evaluation of different image colorization models based on Peak Signal to Noise Ratio (PSNR), Structural Similarity (SSIM), and Frames Per Second (FPS) on Jetson Xavier NX.

Method	PSNR (%)	SSIM (%)	FPS (%)
CICZ	0.645	0.976	0.139
ELGL	0.671	0.988	0.853
ChromaGAN	0.664	0.974	0.503
SCGAN	0.756	1.072	0.120
Pix2Pix	1.000	1.000	1.000
MemoPainter	0.843	0.924	0.334
TIC-CGAN	0.931	1.109	0.448
CycleGAN	0.639	0.923	0.962
RecycleGAN	0.637	0.815	0.084
PearlGAN	0.612	0.819	0.145
I2V-GAN	0.615	0.836	0.094

This (%) represents the ratio of the model and Pix2Pix on the corresponding metric.

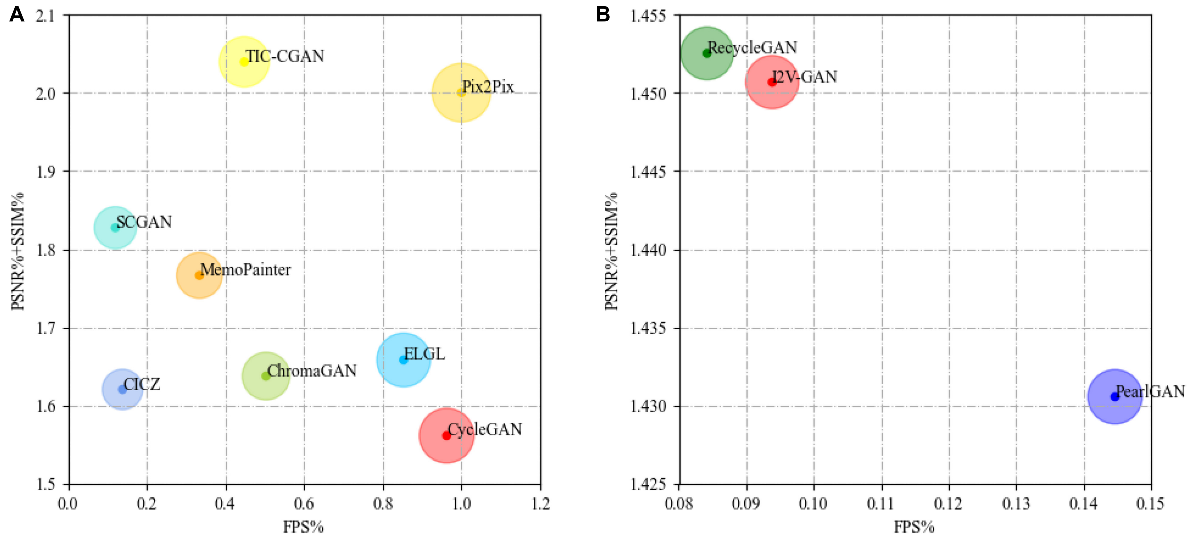


FIGURE 17

Comparison of different image colorization models [**(A)**: CICZ (Zhang et al., 2016), ELGL (Iizuka et al., 2016), ChromaGAN (Vitoria et al., 2020), SCGAN (Zhao et al., 2020), Pix2Pix (Isola et al., 2017), MemoPainter (Yoo et al., 2019), TIC-CGAN (Kuang et al., 2020), CycleGAN (Zhu et al., 2017); **(B)**: RecycleGAN (Bansal et al., 2018), PearlGAN (Luo et al., 2022), I2V-GAN (Li et al., 2021)]. The size of the circle represents the combined weight of the values on X-axis and Y-axis. The larger the circle, the better the performance.

4. Conclusion

The Jetson series is a widely used embedded system. Limits on hardware resources and energy consumption restrict the deployment of current deep learning models on edge devices. In this study, an evaluation system was designed to test the performance of NIR image colorization methods on edge devices on the RGB-NIR scene dataset (Brown and Süsstrunk, 2011). From the experimental results, we summarize several conclusions for reference and provide suggestions for future work:

1. We found that the data were very close by comparing the results of the image quality metrics of the same model on the server and the edge devices. When considering image quality metrics of methods, researchers only needed to refer to the results on the server.
2. Among the 11 methods, the image quality metrics of Pix2Pix and TIC-CGAN were the best on the RGB-NIR scene dataset (Brown and Süsstrunk, 2011).
3. The latency of each model varied significantly across different edge devices. As device performance decreased, the proportion of the latency differences among the models also changed.
4. Of the 11 methods, ELGL had the smallest latency on Jetson AGX Xavier. On Jetson Xavier NX and Jetson Nano, Pix2Pix had the smallest latency.
5. When deployed on an edge device with enough running memory, the model will occupy more memory than the memory-limited device. The memory usage may be related to the memory allocation policy of the deep learning framework.
6. The RecycleGAN, PearlGAN, and I2V-GAN had smaller memory usage on edge devices than the others. Since we used only the generator to create colorized results for model testing, researchers who wish to optimize a model's memory usage can refer to these models' generator structures.
7. Of the 11 methods, CICZ had the smallest energy consumption per unit of time, while the maximum current and maximum power were the smallest. Meanwhile, the difference in energy consumption among other models was lower than the difference between CICZ and them. For optimizing energy consumption, researchers can refer to the structure of CICZ.
8. Combining the testing results of image quality and latency metrics, it can be concluded that Pix2Pix and TIC-CGAN could serve as a basis for further optimization of NIR image colorization on edge devices.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://www.epfl.ch/labs/ivrl/research/downloads/rgb-nir-scene-dataset/.

Author contributions

SS proposed the theory, conducted the experiment, and wrote the manuscript. XJ proposed the general idea of this theory. QJ and XJ supervised this work and revised the manuscript. WW, KL, and HC participated in the design and testing of the experimental process. PL, WZ, and SY discussed the theory. All authors contributed to the article and approved the submitted version.

Funding

This study was supported by the National Natural Science Foundation of China (Nos. 62101481, 62002313, 62261060, 62162067, 62101480), Basic Research Project of Yunnan Province (Nos. 202301AW070007, 202301AU070210, 202201AU070033, 202201AT070112, 202001BB050076, and 202005AC160007), Major Scientific and Technological Project of Yunnan Province (No. 202202AD080002), the Fund Project of Yunnan Province Education Department (No. 2022j0008), the Open Project of Engineering Research Center of Cyberspace in 2021–2022 (No. KJAQ202112012), Key Laboratory in Software Engineering of Yunnan Province (No.2020SE408), and Research and Application of Object Detection Based on Artificial Intelligence.

Acknowledgments

The authors acknowledge XJ, Xiang Cheng, and Guoliang Yao from Yunnan University for their useful suggestions and discussions and Ming Feng and Youwei He from Yunnan University for their generous assistance in the experiments.

Conflict of interest

PL was employed by Guangxi Power Grid 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

Bansal, A., Ma, S., Ramanan, D., and Sheikh, Y. (2018). "Recycle-gan: unsupervised video retargeting," in *Proceedings of the European conference on computer vision (ECCV)*, (Berlin: Springer). doi: 10.1007/978-3-030-01228-1_8

Brown, M., and Süsstrunk, S. (2011). *Multi-Spectral SIFT for Scene Category Recognition*. Piscataway, NJ: IEEE. doi: 10.1109/CVPR.2011.5995637

Cao, Z., Huang, Z., Pan, L., Zhang, S., Liu, Z., and Fu, C. (2022). "TCTrack: temporal contexts for aerial tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (Piscataway, NJ: IEEE). doi: 10.1109/CVPR52688.2022.01438

Fortino, G., Zhou, M. C., Hassan, M. M., Pathan, M., and Karnouskos, S. (2021). Pushing Artificial intelligence to the edge: emerging trends, issues and challenges. *Eng. Appl. Artif. Intell.* 103:104298. doi: 10.1016/j.engappai.2021.104298

Han, D., Liu, Y., and Ni, J. (2020). Research on multinode collaborative computing offloading algorithm based on minimization of energy consumption. *Wirel. Commun. Mob. Comput.* 2020:8858298. doi: 10.1155/2020/8858298

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (Piscataway, NJ: IEEE), 770–778. doi: 10.1109/CVPR.2016.90

Huang, S., Jin, X., Jiang, Q., and Liu, L. (2022). Deep learning for image colorization: current and future prospects. *Eng. Appl. Artif. Intell.* 114:105006. doi: 10.1016/j.engappai.2022.105006

Iizuka, S., Simo-Serra, E., and Ishikawa, H. (2016). Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Trans. Graphics* 35, 1–11. doi: 10.1145/2897824.2925974

Isola, P., Zhu, J. Y., Zhou, T., and Efros, A. A. (2017). "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (Piscataway, NJ: IEEE). doi: 10.1109/CVPR.2017.632

Jeon, J., Jung, S., Lee, E., Choi, D., and Myung, H. (2021). Run your visual-inertial odometry on NVIDIA jetson: benchmark tests on a micro aerial vehicle. *IEEE Robot. Autom. Lett.* 6, 5332–5339. doi: 10.1109/LRA.2021.3075141

Jin, X., Jiang, Q., Yao, S., Zhou, D., Nie, R., Hai, J., et al. (2017). A survey of infrared and visual image fusion methods. *Infrared Phys. Technol.* 85, 478–501. doi: 10.1016/j.infrared.2017.07.010

Kuang, X., Zhu, J., Sui, X., Liu, Y., Liu, C., Chen, Q., et al. (2020). Thermal infrared colorization via conditional generative adversarial network. *Infrared Phys. Technol.* 107:103338. doi: 10.1016/j.infrared.2020.103338

Li, S., Han, B., Yu, Z., Liu, C. H., Chen, K., and Wang, S. (2021). "I2v-gan: unpaired infrared-to-visible video translation," in *Proceedings of the 29th ACM International Conference on Multimedia*, (New York, NY: ACM). doi: 10.1145/3474085.3475445

Liang, W., Ding, D., and Wei, G. (2021). An improved DualGAN for near-infrared image colorization. *Infrared Phys. Technol.* 116:103764. doi: 10.1016/j.infrared.2021.103764

Liao, H., Jiang, Q., Jin, X., Liu, L., Liu, L., Lee, S. J., et al. (2022). MUGAN: thermal infrared image colorization using mixed-skipping UNet and generative adversarial network. *IEEE Trans. Intell. Vehicles* 1–16. doi: 10.1109/TIV.2022.3218833

Liu, L., Jiang, Q., Jin, X., Feng, J., Wang, R., Liao, H., et al. (2022). CASR-net: a color-aware super-resolution network for panchromatic image. *Eng. Appl. Artif. Intell.* 114:105084. doi: 10.1016/j.engappai.2022.105084

Liu, Y., Wang, L., Cheng, J., Li, C., and Chen, X. (2020). Multi-focus image fusion: a survey of the state of the art. *Information Fusion* 64, 71–91. doi: 10.1016/j.inffus.2020.06.013

Luo, F., Li, Y., Zeng, G., Peng, P., Wang, G., and Li, Y. (2022). *Thermal Infrared Image Colorization for Nighttime Driving Scenes with Top-Down Guided Attention. IEEE Transactions on Intelligent Transportation Systems*. Piscataway, NJ: IEEE. doi: 10.1109/TITS.2022.3145476

Ma, X., Xu, S., An, F., and Lin, F. (2018). A novel real-time image restoration algorithm in edge computing. *Wirel. Commun. Mob. Comput.* 2018:3610482. doi: 10.1155/2018/3610482

Mazzia, V., Khaliq, A., Salvetti, F., and Chiaberge, M. (2020). Real-time apple detection system using embedded systems with hardware accelerators: an edge AI application. *IEEE Access* 8, 9102–9114. doi: 10.1109/ACCESS.2020.2964608

Ni, J., Shen, K., Chen, Y., Cao, W., and Yang, S. X. (2022). An improved deep network-based scene classification method for self-driving cars. *IEEE Trans. Instrum. Meas.* 71, 1–14. doi: 10.1109/TIM.2022.3146923

Qin, Z., Qiu, X., Ye, J., and Wang, L. (2020). User-edge collaborative resource allocation and offloading strategy in edge computing. *Wirel. Commun. Mob. Comput.* 2020, 1–12. doi: 10.1155/2020/8867157

Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv [Preprint]*

Shi, W., Cao, J., Zhang, Q., Li, Y., and Xu, L. (2016). Edge computing: vision and challenges. *IEEE Internet Things J.* 3, 637–646. doi: 10.1109/JIOT.2016.2579198

Sun, T., Jung, C., Fu, Q., and Han, Q. (2019). Nir to rgb domain translation using asymmetric cycle generative adversarial networks. *IEEE Access* 7, 112459–112469. doi: 10.1109/ACCESS.2019.2933671

Toczé, K., and Nadjm-Tehrani, S. (2018). A taxonomy for management and optimization of multiple resources in edge computing. *Wirel. Commun. Mobile Comput.* 2018:7476201. doi: 10.1155/2018/7476201

Vitoria, P., Raad, L., and Ballester, C. (2020). "Chromagan: adversarial picture colorization with semantic class distribution," in *Proceedings of the IEEE/CVF Winter*

Conference on Applications of Computer Vision, (Piscataway, NJ: IEEE). doi: 10.1109/WACV45572.2020.9093389

Yoo, S., Bahng, H., Chung, S., Lee, J., Chang, J., and Choo, J. (2019). "Coloring with limited data: Few-shot colorization via memory augmented networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (Piscataway, NJ: IEEE). doi: 10.1109/CVPR.2019.01154

Yu, B., Chen, Y., Cao, S. Y., Shen, H. L., and Li, J. (2022). Three-channel infrared imaging for object detection in haze. *IEEE Trans. Instrum. Meas.* 71, 1–13. doi: 10.1109/TIM.2022.3164062

Zhang, P., Zhang, A., and Xu, G. (2020). Optimized task distribution based on task requirements and time delay in edge computing environments. *Eng. Appl. Artif. Intell.* 94:103774. doi: 10.1016/j.engappai.2020.103774

Zhang, R., Isola, P., and Efros, A. A. (2016). "Colorful image Colorization," in *Proceedings of the European Conference on Computer Vision*. Cham: Springer. doi: 10.1007/978-3-319-46487-9_40

Zhao, M., Cheng, L., Yang, X., Feng, P., Liu, L., and Wu, N. (2019). TBC-net: a real-time detector for infrared small target detection using semantic constraint. *arXiv [Preprint]*

Zhao, Y., Po, L. M., Cheung, K. W., Yu, W. Y., and Rehman, Y. A. U. (2020). SCGAN: saliency map-guided colorization with generative adversarial network. *IEEE Trans. Circuits Syst. Video Technol.* 31, 3062–3077. doi: 10.1109/TCSVT.2020.3037688

Zhu, J. Y., Park, T., Isola, P., and Efros, A. A. (2017). "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, (Piscataway, NJ: IEEE). doi: 10.1109/ICCV.2017.244

Zhu, Z., He, X., Qi, G., Li, Y., Cong, B., and Liu, Y. (2023). Brain tumor segmentation based on the fusion of deep semantics and edge information in multimodal MRI. *Information Fusion* 91, 376–387. doi: 10.1016/j.inffus.2022.10.022

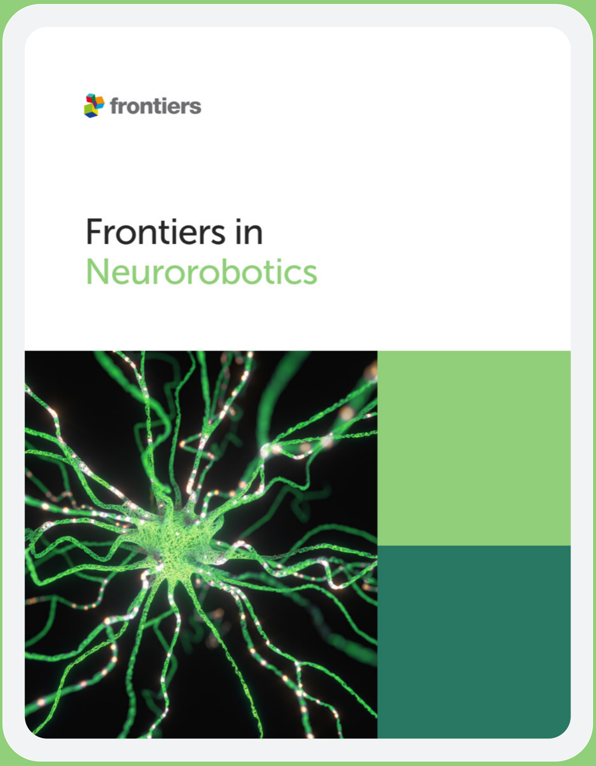
Frontiers in
Neurorobotics

Investigates embodied autonomous neural systems and their impact on our lives

Part of the most cited neuroscience series, this journal advances understanding of neurorobotics - from prosthetic devices to brain machine interfaces, and wearable systems to home appliances.

Discover the latest Research Topics

See more →



frontiers

Frontiers in
Neurorobotics

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact