Computers and Electrical Engineering 123 (2025) 110238

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Multi-focus image fusion with visual state space model and dual adversarial learning

Xinzhe Xie [a,b], Buyu Guo [b,c,*], Peiliang Li [a,b], Shuangyan He [a,b], Sangjun Zhou [a,b]

[a] Ocean College, Zhejiang University, Zhoushan, 316021, PR China
[b] Hainan Institute, Zhejiang University, Sanya, 572025, PR China
[c] Donghai Laboratory, Zhoushan, 316021, PR China

ARTICLE INFO

Keywords:
Deep learning
Multi-focus image fusion
Adversarial learning
Fine-tuning technology
Visual state space model

ABSTRACT

In recent years, the two-stage multi-focus image fusion (MFF) method, which utilizes neural networks to first generate decision maps and then calculate the fused image, has witnessed significant advancements. However, after supervised training, many networks become overly reliant on semantic information, making it challenging to discern whether homogeneous regions and flat regions are in focus or not, as these regions lack distinct blur cues. To alleviate this issue, this paper proposes a multi-focus image fusion network named BridgeMFF by applying a visual state space model and developing a general fine-tuning technique named BridgeTune, which bridges the semantic and texture gap via dual adversarial learning. By fine-tuning the entire network in an adversarial manner, decision maps are generated to synthesize clear and blurred images with probability density distributions closely approximating real ones, thereby implicitly learning local spatial patterns and statistical properties of pixel values. Extensive experiments demonstrate that the proposed BridgeMFF achieves superior fusion quality, especially in challenging homogeneous regions. Moreover, BridgeMFF has the smallest model size (0.05M) and fastest processing speed (0.09s per image pair), enabling real-time fusion applications. The codes are available at https://github.com/Xinzhe99/BridgeMFF.

1. Introduction

The primary objective of MFF is to combine images captured at different focus settings into a single image where all objects are in focus and clear. Recent MFF methods based on deep convolutional neural networks (CNN) have achieved high performance through rich labeled data [1,2]. However, these networks typically exhibit lower performance in flat and homogeneous image regions due to the lack of distinct blur cues [3].

In MFF, the two input images are identical in content except for their blurred regions. Our key insight is that the distinction between source images primarily lies in low-level texture features rather than high-level semantic features. In homogeneous regions, semantic features are ambiguous while useful texture features are often overwhelmed by deep semantic representations, limiting the network's ability to effectively discriminate focus conditions.

To verify this, we analyze networks of varying depths and visualize their feature responses using Grad-Cam [4]. As shown in Fig. 1, shallow layers maintain more uniform attention to focused regions compared to deep layers. This disparity intensifies as network depth increases — in the deepest layer, the network fails to attend to homogeneous regions like the yellow dog area.

* Corresponding author at: Donghai Laboratory, Zhoushan, 316021, PR China.
E-mail addresses: xiexinzhe@zju.edu.cn (X. Xie), guobuyuwork@163.com (B. Guo), lipeiliang@zju.edu.cn (P. Li), hesy103@163.com (S. He), 22234117@zju.edn.cn (S. Zhou).

https://doi.org/10.1016/j.compeleceng.2025.110238
Received 24 June 2024; Received in revised form 10 February 2025; Accepted 1 March 2025

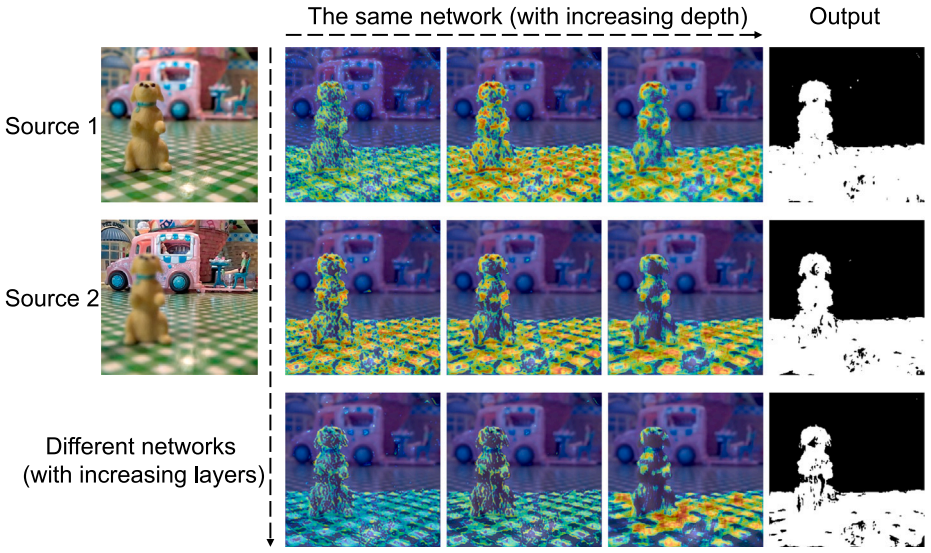

Fig. 1. Comparison of the performance of fully convolutional networks of different depths on the MFF task and comparison of activation heatmaps at different depths of the same network.

Fig. 2. Illustration of dual adversarial training. First column: input image pairs. Top row: accurate decision maps lead to synthesized images that fool both discriminators (✓). Bottom row: inaccurate decision maps result in synthesized images that fail to fool the discriminators (×).

Our analysis reveals the strong correlation between MFF and texture features, explaining the effectiveness of skip connections in this task [5,6]. Skip connections help preserve low-level texture features that would otherwise be diminished in deeper layers.

A similar phenomenon exists in defocus blur detection (DBD). SG [7] revealed that DBD relies more heavily on low-level texture features compared to semantic-driven tasks like salient object segmentation. They addressed this by designing a self-supervised Generative Adversarial Network (GAN) with dual discriminators, achieving strong generalization without labeled data.

Inspired by this insight and considering the similar role of texture features in MFF (as demonstrated in Fig. 1), we adapt SG's principles to our task. Our key observation is that blurred/clear regions can be transferred to completely blurry/clear images without affecting their overall characteristics. Based on this, we propose a dual adversarial fine-tuning method for MFF. As shown in Figs. 2 and 4, our approach uses a generator to segment focused and unfocused regions from source images. These regions are then pasted onto reference clear and blurry images to create synthesized images (S_{c1}, S_{c2}, S_{b1}, S_{b2}). Two discriminators (D_c, D_b) learn to distinguish these synthesized images from real ones.

Our method takes a fundamentally different approach from previous works that focus on enhancing local feature extraction [8,9]. Instead of directly processing texture features, we align the probability distributions of synthesized and real images, allowing the network to implicitly learn local texture patterns through adversarial training. With paired multi-focus inputs, we can simultaneously align the distributions of synthesized clear images with real clear images, and synthesized blurred images with real blurred images, enabling more effective texture learning. Furthermore, since our approach only introduces additional loss terms during fine-tuning without modifying network architectures, it can be readily integrated into existing decision map-based fusion networks.

Our main contributions are three-fold:

- We propose BridgeTune, a dual adversarial fine-tuning approach that bridges the semantic and texture gap to address the challenge of homogeneous region detection in MFF decision maps. BridgeTune can be readily integrated into existing networks.
- We develop BridgeMFF, a visual state space model-based fusion network that leverages global receptive field and dynamic weighting for effective MFF.
- Extensive experiments demonstrate that our method achieves state-of-the-art performance in both fusion quality and computational efficiency, with the smallest model size (0.05M), lowest computational complexity (0.06G FLOPs), and fastest processing speed (0.09 s per image pair) among existing methods.

The remainder of this paper is organized as follows. We review related work in Section 2. Section 3 presents our methodology. We provide experimental results in Section 4 and present discussions including limitations and future directions in Section 5. Section 6 concludes the paper.

2. Related works

2.1. Multi-focus image fusion

2.1.1. Conventional multi-focus image fusion methods

Prior MFF methods can be broadly categorized into transform domain-based and spatial domain-based approaches.

Transform domain-based methods leverage various transforms for fusion, including pyramids [10,11], wavelets [12,13], Discrete Cosine Transform (DCT) [14], Non-Subsampled Contourlet Transform (NSCT) [15], and Sparse Representation (SR) [16,17]. While these methods enable effective multi-scale and frequency analysis, they face several limitations: pyramid methods suffer from detail loss and halo artifacts; wavelet approaches struggle with directional features; DCT shows weak boundary performance; NSCT incurs high computational cost; and SR methods require extensive parameter tuning and are noise-sensitive [5].

Spatial domain-based methods perform fusion through weighted averaging based on focus assessment. Block-based approaches [18,19] introduce blocking artifacts, region-based methods [20] suffer from segmentation errors, and pixel-based techniques using SIFT or defocus estimation [21,22] are computationally intensive. These methods, while intuitive, rely heavily on image gradients and often fail in homogeneous regions [2], leading to suboptimal fusion results.

2.1.2. Deep learning-based image fusion methods

Deep learning-based MFF methods can be divided into decision map-based and end-to-end approaches. Decision map-based methods first predict focus maps then select pixels accordingly. CNN [23] pioneered this direction using convolutional networks. Subsequent works improved performance through various strategies: MSFIN [6] and GEU-Net [5] leverage multi-scale features; SESF [24] enables unsupervised fusion via high-level feature maps; SMFuse [25] introduces self-supervised mask optimization; MFIF-GAN [26] utilizes adversarial learning with α-matte modeling; TPP [27] employs three-channel networks with thresholding; recent works [2,28] incorporate transformers for global modeling. ZMFF [29] achieves zero-shot fusion using Deep Image Prior [30].

End-to-end methods directly regress fused images, showing better robustness to complex scenes and edge artifacts. While this approach may slightly compromise image fidelity, it has been widely adopted in general fusion frameworks like IFCNN [31], U2Fusion [32], SDNet [33], MFF-GAN [34], SwinFusion [35], and the recent diffusion-based FusionDiff [36].

The proposed BridgeTune targets decision map-based methods. Unlike existing approaches that rely on post-processing (e.g., CRF or morphological operations) to refine decision maps [6,26], our method achieves similar refinement through adversarial training, effectively preventing holes in homogeneous regions without explicit post-processing steps.

2.1.3. Problems in previous works and research gaps

Prior methods tackle focus detection in homogeneous regions through three main approaches: multi-scale analysis [5,6], feature fusion [37], and post-processing refinement [24]. While these strategies have been adopted by both traditional and deep learning methods, they face inherent limitations: hand-crafted solutions are parameter-sensitive, and CNN-based methods struggle with the local nature of convolutions. As shown in Fig. 9, even state-of-the-art methods produce significant errors in homogeneous regions.

To address this challenge, we introduce two key innovations. First, we propose a dual adversarial learning scheme that aligns probability distributions between synthesized and real images, enabling implicit texture learning [7]. Second, we incorporate a visual state space model [38] that achieves global receptive field for effective homogeneous region detection.

2.2. Visual state space model

While CNNs and Transformers dominate computer vision, they face inherent limitations: CNNs struggle with long-range dependencies, and Transformers incur quadratic complexity. State Space Models (SSMs) have emerged as a promising alternative, offering linear complexity while capturing global interactions. Mamba [39] enhanced SSMs with time-varying parameters, demonstrating superior efficiency over Transformers. Recent works (VMamba [40], Vision Mamba [41]) successfully adapted SSMs for vision tasks. VM-UNet [42] further showed their effectiveness in medical image segmentation, while UltraLight VM-UNet [38] achieved comparable performance with only 0.18% of VM-UNet's parameters.

Recent findings [43] suggest that visual state space models excel at long-sequence modeling. Since MFF can be formulated as a binary segmentation task of focus/defocus regions, it inherently involves long-range dependencies. We therefore propose BridgeMFF, built upon UltraLight VM-UNet [38], leveraging its global effective receptive field (Fig. 3) for accurate focus detection while maintaining high computational efficiency.

Fig. 3. Information flow of Vision Mamba models. This kind of models integrate pixels from top-left, bottom-right, top-right, and bottom-left with $O(N)$ complexity.

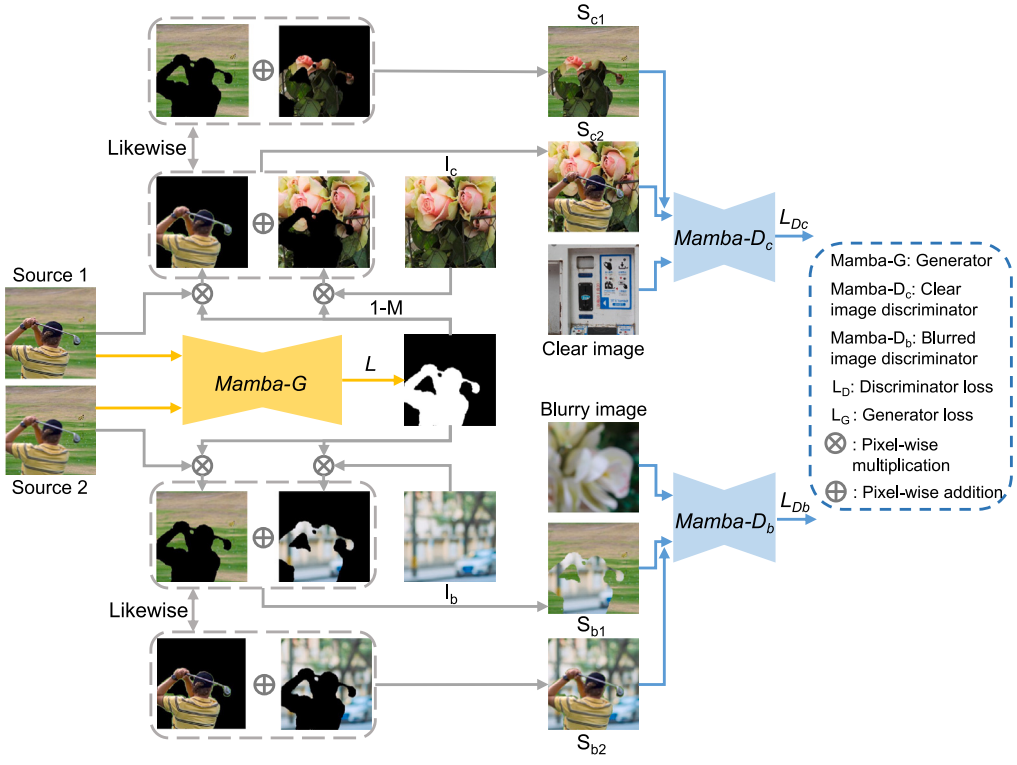

Fig. 4. Architecture illustration of the proposed fine-tuning approach BridgeTune. The approach consists of two parts: (1) Utilizing the masks generated by the generator Mamba-G, the clear and blurred regions from the two input images are respectively pasted to real completely clear and completely blurry images, generating two synthesized clear images and two synthesized blurred images (gray path); (2) A dual adversarial training strategy that forces the generator to produce masks that synthesize two clear images and two blurred images, which can simultaneously fool their corresponding discriminators (blue path).

3. Methods

3.1. Fine-tuning approach

As illustrated in Fig. 4, the proposed fine-tuning approach BridgeTune consists of two main components: (1) A three-network architecture comprising a generator Mamba-G and two discriminators (Mamba-D_c, Mamba-D_b). Specifically, Mamba-G generates focus masks for both supervised and adversarial training, while Mamba-D_c and Mamba-D_b discriminate between real and synthesized images in clear and blurry domains, respectively. (2) A synthesis pipeline where masks from Mamba-G are used to generate two pairs of images: clear images (S_{c1}, S_{c2}) for training Mamba-D_c and blurred images (S_{b1}, S_{b2}) for training Mamba-D_b.

We first leverage ground truth masks to guide Mamba-G in focus/defocus region discrimination through supervised learning. This process enables the network to capture high-level semantic information, facilitating complex scene understanding and foreground–background separation. We then incorporate a dual adversarial learning strategy where feedback from Mamba-D_c and Mamba-D_b

implicitly enhances the network's sensitivity to low-level textural features. During adversarial training, the generated mask M from Mamba-G is employed to transfer focused and unfocused regions from source images onto reference clear image I_c and blurry image I_b, producing synthesized image pairs (S_{c1}, S_{c2}) and (S_{b1}, S_{b2}). The synthesis process can be formulated as:

$$S_{c1} = M \otimes \mathrm{I}_1 \oplus (1 - M) \otimes \mathrm{I}_c \tag{1}$$

$$S_{c2} = (1 - M) \otimes \mathrm{I}_2 \oplus M \otimes \mathrm{I}_c \tag{2}$$

$$S_{b1} = (1 - M) \otimes \mathrm{I}_1 \oplus M \otimes \mathrm{I}_b \tag{3}$$

$$S_{b2} = M \otimes \mathrm{I}_2 \oplus (1 - M) \otimes \mathrm{I}_b \tag{4}$$

where $\otimes$ denotes pixel-wise multiplication, and $\oplus$ denotes pixel-wise addition. The terms I_1 and I_2 represent the input multi-focus images, while I_c and I_b represent a completely clear image and a completely blurry image, respectively.

Inspired by SG [7], we train the discriminator Mamba-D_c by maximizing the following loss:

$$\begin{aligned} L_{D_c} = {} & E_{I_c \sim P_c}\left[\log\left(D_c\left(I_c\right)\right)\right] \\ & + E_{T \sim P_t}\left[\log\left(1 - D_c\left(\mathrm{S}_{c1}\right)\right)\right] \\ & + E_{T \sim P_t}\left[\log\left(1 - D_c\left(\mathrm{S}_{c2}\right)\right)\right] \end{aligned} \tag{5}$$

Similarly, we maximize the following loss to train the discriminator Mamba-D_b:

$$\begin{aligned} L_{D_b} = {} & E_{I_b \sim P_b}\left[\log\left(D_b\left(I_b\right)\right)\right] \\ & + E_{T \sim P_t}\left[\log\left(1 - D_b\left(\mathrm{S}_{b1}\right)\right)\right] \\ & + E_{T \sim P_t}\left[\log\left(1 - D_b\left(\mathrm{S}_{b2}\right)\right)\right] \end{aligned} \tag{6}$$

where D_c and D_b are Mamba-D_c and Mamba-D_b respectively. P_c and P_b represent the probability density distribution of the completely clear image and the completely blurry image respectively. P_t represents the probability density distribution of the training image pair.

Specifically, Mamba-D_c needs to discriminate whether the input image is a real completely clear image or a clear image synthesized using the mask generated by Mamba-G. Mamba-D_b needs to discriminate whether the input image is a real completely blurry image or a blurred image synthesized using the mask generated by Mamba-G. Meanwhile, Mamba-G aims to produce masks that can simultaneously fool both Mamba-D_c and Mamba-D_b into being unable to distinguish between real and synthesized images, by minimizing the following loss [7]:

$$\begin{aligned} L_G = {} & E_{T \sim P_t}[\log\left(1 - D_c\left(\mathrm{S}_{c1}\right)\right)] \\ & + E_{T \sim P_t}\left[\log\left(1 - D_c\left(\mathrm{S}_{c2}\right)\right)\right] \\ & + E_{T \sim P_t}\left[\log\left(1 - \mathrm{D}_b\left(\mathrm{S}_{b1}\right)\right)\right] \\ & + E_{T \sim P_t}\left[\log\left(1 - \mathrm{D}_b\left(\mathrm{S}_{b2}\right)\right)\right] \end{aligned} \tag{7}$$

Therefore, the overall adversarial loss function can be written as:

$$L_{GAN} = L_G + L_{D_c} + L_{D_b}$$

In this work, we use a combination of focal loss [44] and dice loss [45] as the loss function L_{SUP} for supervised learning, the focal loss L_{Focal} is:

$$L_{Focal} = -\left(1 - p_s\right)^{\gamma} \log\left(p_s\right) \tag{8}$$

where γ is a hyperparameter that controls the sample weights. A larger value of γ leads to lower weights for easy samples and higher weights for difficult samples. And, the p_s is as follows:

$$p_s = \begin{cases} p & \text{if } y_{true} = 1 \\ 1 - p & \text{otherwise} \end{cases} \tag{9}$$

In this function, y_{true} is the label. The advantage of focal loss over common binary cross-entropy loss lies in its ability to dynamically adjust the importance of different regions in the image, thereby emphasizing the training focus on challenging homogeneous and flat areas while decreasing the influence of easily classified regions.

Additionally, we employ a dice loss to mitigate the adverse effects caused by the imbalance between the areas of clear and blurred regions in the input images, the dice loss can be written as:

$$L_{Dice} = 1 - \frac{2\sum_i y_{true} y_{pred}}{\sum_i \left({y_{true}}^2 + {y_{pred}}^2\right)} \tag{10}$$

where y_{pred} are the predicted value. So, the L_{SUP} is as follows:

$$L_{SUP} = \lambda L_{Focal} + L_{Dice} \tag{11}$$

where λ is the weight coefficient to balance the numerical scales between focal loss and dice loss. In this work, we set $\lambda = 20$. Therefore, the overall loss function L in the fine-tuning process can be written as:

$$L = \alpha L_{GAN} + L_{SUP} \tag{12}$$

In Eq. (12), α serves as a weight coefficient balancing supervised and adversarial losses, which is set to 1e–4 in this paper. A larger α strengthens the adversarial loss during training, thereby enhancing texture sensitivity while reducing semantic dependency.

The dual adversarial learning scheme adaptively guides the network to effectively separate clear and blurred regions through mask generation, producing realistic synthesized images that challenge both discriminators. While the discriminators learn to differentiate between synthesized and real images, Mamba-G receives feedback solely based on the transfer regions defined by its generated masks. This mechanism establishes a connection between the probability distributions of the transfer regions and those of real images, implicitly emphasizing local textural properties in the decision process. Consequently, it alleviates the over-reliance on high-level semantic features and helps preserve low-level texture information. Furthermore, by tuning the loss weights, we can precisely control this balance, achieving an optimal trade-off between semantic and texture information utilization.

Algorithm 1 Training Process of BridgeMFF

1: **Stage 1:** Initial Training
2: **for** epoch = 1 to N1 **do**
3: **for** each batch (I_1, I_2, M_{gt}) in training set **do**
4: $M_{pred} \leftarrow$ Mamba-$G(I_1, I_2)$ ▷ Generate decision map
5: Calculate L_{SUP} using Eq. (11) ▷ Supervised loss
6: Update Mamba-G by minimizing L_{SUP}
7: **end for**
8: **end for**
9: **Stage 2:** BridgeTune Fine-tuning
10: **for** epoch = 1 to N2 **do**
11: **for** each batch $(I_1, I_2, M_{gt}, I_c, I_b)$ in training set **do**
12: $M_{pred} \leftarrow$ Mamba-$G(I_1, I_2)$ ▷ Generate decision map
13: $S_{c1} \leftarrow M_{pred} \odot I_1 + (1 - M_{pred}) \odot I_c$
14: $S_{c2} \leftarrow (1 - M_{pred}) \odot I_2 + M_{pred} \odot I_c$
15: $S_{b1} \leftarrow (1 - M_{pred}) \odot I_1 + M_{pred} \odot I_b$
16: $S_{b2} \leftarrow M_{pred} \odot I_2 + (1 - M_{pred}) \odot I_b$
17: $L_{D_c} \leftarrow \log(D_c(I_c)) + \log(1 - D_c(S_{c1})) + \log(1 - D_c(S_{c2}))$
18: $L_{D_b} \leftarrow \log(D_b(I_b)) + \log(1 - D_b(S_{b1})) + \log(1 - D_b(S_{b2}))$
19: Update Mamba-D_c by maximizing L_{D_c}
20: Update Mamba-D_b by maximizing L_{D_b}
21: $L_{GAN} \leftarrow -\log(D_c(S_{c1})) - \log(D_c(S_{c2})) - \log(D_b(S_{b1})) - \log(D_b(S_{b2}))$
22: Calculate L_{SUP} using Eq. (11)
23: $L_{total} \leftarrow L_{SUP} + \alpha L_{GAN}$ ▷ α balances two losses
24: Update Mamba-G by minimizing L_{total}
25: **end for**
26: **end for**

Algorithm 1 outlines the complete training process of BridgeMFF, which consists of two main stages:
Stage 1: Initial Training (Lines 1–8) The first stage focuses on supervised learning to establish basic fusion capabilities:

- Line 2 initiates the first training phase for N1 epochs (set to 100 in our implementation)
- Lines 3–6 describe the main training loop for each batch:
  - Line 4: The generator Mamba-G processes input image pairs (I_1, I_2) to produce decision maps
  - Line 5: Supervised loss L_{SUP} is calculated by comparing predicted maps with ground truth
  - Line 6: Network parameters are updated to minimize the supervised loss

Stage 2: BridgeTune Fine-tuning (Lines 9–26) The second stage implements our dual adversarial learning strategy:

- Line 10 begins the fine-tuning phase for N2 epochs (set to 50 in our implementation)
- Lines 11–25 detail the fine-tuning process for each batch:
  - Line 12: Generate initial decision maps using Mamba-G
  - Lines 13-16: Synthesize four images using the generated decision maps:
    - S_{c1}, S_{c2}: Clear domain images (Line 13–14)
    - S_{b1}, S_{b2}: Blurred domain images (Line 15–16)
  - Lines 17-20: Update discriminators:

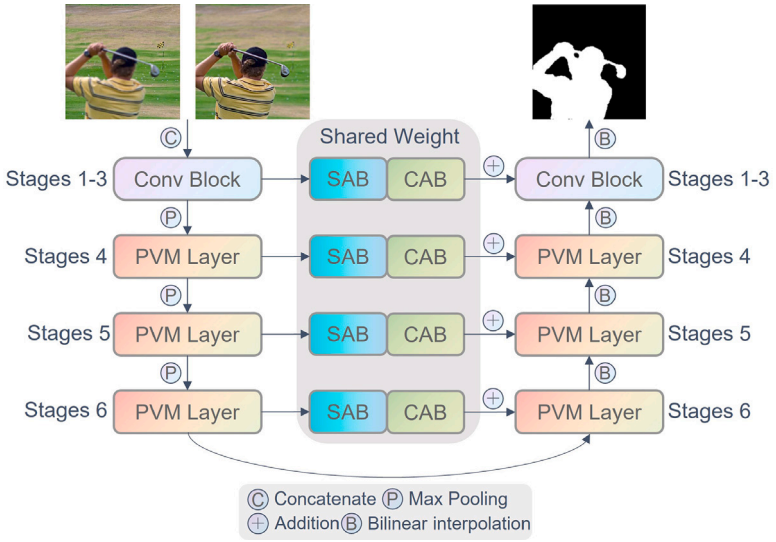

Fig. 5. The network structure of the Mamba-G based on UltraLight VM-UNet [38].

- Line 17: Calculate clear domain discriminator loss L_{D_c}
- Line 18: Calculate blurred domain discriminator loss L_{D_b}
- Lines 19-20: Update both discriminators to maximize their respective losses

– Lines 21-24: Update generator:

- Line 21: Calculate adversarial loss L_{GAN}
- Line 22: Calculate supervised loss L_{SUP}
- Line 23: Combine losses with weighting factor α
- Line 24: Update Mamba-G to minimize the total loss

The algorithm effectively combines supervised learning with adversarial training, where the initial stage establishes basic fusion capabilities, and the fine-tuning stage enhances the network's ability to handle challenging regions through dual adversarial learning. The hyperparameter α (Line 23) balances the contribution of adversarial and supervised losses, with an empirically determined value of 1e–4 achieving optimal results.

3.2. Network

We implement both the generator Mamba-G and discriminators (Mamba-D_b, Mamba-D_c) based on UltraLight VM-UNet [38], leveraging its key advantages: linear complexity, global receptive field, and dynamic weighting. The linear complexity enables efficient processing of high-resolution images, while the global receptive field facilitates comprehensive scene understanding for focus discrimination. Additionally, the dynamic weighting mechanism adaptively modulates attention across regions with varying focus characteristics and scene complexities.

As illustrated in Fig. 5, Mamba-G adopts a 6-layer U-shaped architecture with encoder–decoder branches and skip connections. The first three layers employ Conv Blocks for shallow feature extraction, where each block consists of a 3×3 convolution followed by max pooling. These layers capture low-level patterns, edges, and textures, progressively learning compact and transferable representations through pooling operations. The deeper features (layers 4–6) are extracted through Parallel Vision Mamba Layers (PVM Layer), which comprise four parallel VSS Blocks [40]. This design enables effective modeling of contextual dependencies and global information encoding. The PVM layer operation can be formulated as [38]:

$$Y_1^{C/4}, Y_2^{C/4}, Y_3^{C/4}, Y_4^{C/4} = Split\left[LN\left(X_{in}^{C}\right)\right] \tag{13}$$

$$vss\_Y_i^{C/4} = VSS\left(Y_i^{C/4}\right) + \theta \cdot Y_i^{C/4} \quad i = 1, 2, 3, 4 \tag{14}$$

$$X_{\text{out}} = Cat\left(vss\_Y_1^{C/4}, vss\_Y_2^{C/4}, vss\_Y_3^{C/4}, vss\_Y_4^{C/4}\right) \tag{15}$$

$$Out = \text{Projection}\left[LN\left(X_{out}\right)\right] \tag{16}$$

where LN denotes LayerNorm, $Split$ performs feature splitting, VSS represents the VSS Block operation, θ controls the residual connection strength, Cat performs feature concatenation, and $Projection$ maps features to the desired dimension. Here, X_{in}^{C} represents

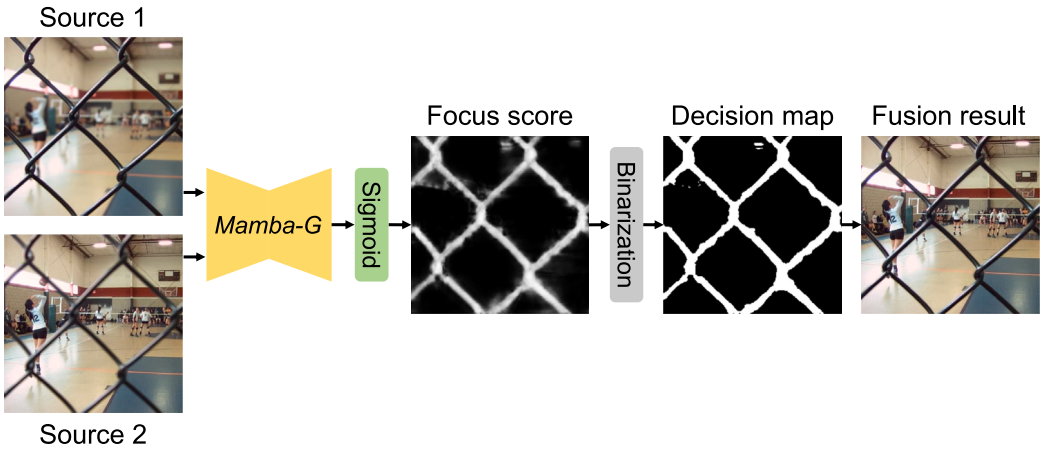

Fig. 6. The pipeline of the fusion in inference phase.

the input feature with C channels, $Y_i^{C/4}$ denotes the ith split feature with $C/4$ channels, and $vss\_Y_i^{C/4}$ represents the output of VSS Block for the ith split feature.

The decoder mirrors the encoder structure, incorporating Channel Attention Bridges (CAB) and Spatial Attention Bridges (SAB) in skip connections for effective multi-scale feature fusion [46]. Both discriminators (Mamba-D_b, Mamba-D_c) share the same architecture as Mamba-G.

3.3. Fusion scheme

During inference (see Fig. 6), the generator Mamba-G processes source images (I_1, I_2) to produce initial focus scores, which are normalized to [0,1] through a Sigmoid function, yielding a focus score map S. The final decision map D for fusion is obtained through binarization:

$$D(x,y) = \begin{cases} 1 & S(x,y) \geq 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

At last, the fused image I_F can be obtained using the following equation:

$$I_F = (I_1 \otimes D) \oplus (I_2 \otimes (1 - D)) \tag{18}$$

4. Experiments

4.1. Implementation details

Preparation of training data: The synthesis method of the dataset is similar to SwinMFF [47]. We utilized 15,572 high-quality images from the DUTS [48] salient object segmentation dataset to synthesize the training dataset, with 10,553 images used for the training set and 5019 images for the validation set. All images were resized to 256 × 256 pixels. Firstly, we binarized the ground truth images in this dataset to generate a mask, which was then used to apply Gaussian blur to the corresponding regions of the source images, synthesizing the required multi-focus image pairs. The Gaussian filter kernel size ranged from 3 to 21 randomly. The code used to synthesize the dataset will be provided in the code repository. Besides, the all-in-focus and all-blurred images used for BridgeTune fine-tuning can be accessed through the link https://github.com/shangcai1/SG.

Hyperparameter settings: We implemented our method in PyTorch, and the model was trained on a platform equipped with two Nvidia A6000 GPUs and an Intel(R) Xeon(R) Platinum 8375C CPU clocked at 2.90 GHz. We first conducted supervised training with a batch size of 32 using the AdamW optimizer (betas = (0.9, 0.999)) with an initial learning rate of 1e–3, and a cosine annealing learning rate decay strategy for 100 epochs, without introducing adversarial training at this stage. After completing the supervised training, we additionally introduced adversarial training to fine-tune the entire model with a batch size of 16, using the same optimizer and learning rate decay strategy with an initial learning rate of 6e−4 for 50 epochs to obtain the final model. To balance the numerical scales and respective weights of the supervised loss and adversarial loss, the coefficient α in the loss function L was set to 1e–4.

How to use BridgeTune: Given that the proposed fine-tuning approach necessitates a specific training framework, we recommend researchers utilize our provided training pipeline when implementing BridgeTune. For optimal training stability and to prevent generator-discriminator capability mismatches, we recommend maintaining architectural consistency across all three networks. Our released codebase includes both pre-training and fine-tuning implementations, requiring only the replacement of the network architecture within the framework while maintaining the training paradigm.

4.2. Experimental setting

4.2.1. Evaluation datasets

In this paper, we primarily evaluate our method on two widely-used multi-focus image fusion datasets:

- **Lytro dataset** [49]: Contains 20 pairs of real-world multi-focus images captured by a light-field camera. Each image pair consists of two images of the same scene with different focus settings, making it ideal for evaluating fusion performance on naturally occurring focus variations.
- **MFI-WHU dataset** [34]: Comprises 120 pairs of synthetic multi-focus images. The images are generated by applying Gaussian blur to clear images according to predefined masks, providing a large-scale benchmark for evaluating fusion algorithms under controlled conditions.

Additionally, we conduct supplementary experiments on the MFFW dataset [50] which features strong defocus blur, and the Road-MF dataset [51] which contains real-world traffic scenes, to further validate our method's generalization capability.

4.2.2. Methods for comparison

We compared our method with 14 state-of-the-art deep learning-based methods, including:

End-to-end methods: the general image fusion network based on the convolutional neural network (IFCNN-MAX) [31], the unified unsupervised image fusion network (U2Fusion) [32], the versatile squeeze-and-decomposition image fusion network (SDNet) [33], the gradient-constrained unsupervised generative adversarial network (MFF-GAN) [34], the transformer-based fusion network (SwinFusion) [35], and the diffusion model-based network (FusionDiff) [36].

Decision map-based methods: the fusion network based on deep convolutional neural networks (CNN) [23], the ensemble of CNN-based method (ECNN) [52], the deep regression pair learning method (DRPL) [53], the method combining traditional spatial frequency operators (SESF) [24], the adversarial generative network-based method (MFIF-GAN) [26], the multi-scale feature interaction network (MSFIN) [6], the gradient-aware cascade network (GACN) [54], and the zero-shot method based on deep image prior (ZMFF) [29].

To ensure fair comparison of network performance without the influence of post-processing operations on visual comparison, all results presented in qualitative comparison for decision map-based methods are either direct outputs from the network or binarized results, without involving any other post-processing operations. In the quantitative comparison, the results of CNN, MFIF-GAN, MSFIN, GACN, and SESF are post-processed, while the other methods do not involve post-processing operations.

4.2.3. Evaluation metrics

We employ six widely-used metrics to comprehensively evaluate the fusion performance. For all metrics, larger values indicate better fusion quality. The metrics are:

- **Edge-based Similarity Metric** $Q^{AB/F}$ [55]. The metric is defined as:

$$Q^{AB/F} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(Q^{A,F}(i,j)w^{A}(i,j) + Q^{B,F}(i,j)w^{B}(i,j))}{\sum_{i=1}^{M}\sum_{j=1}^{N}(w^{A}(i,j) + w^{B}(i,j))} \tag{19}$$

  where $Q^{X,F}(i,j)$ represents the edge information preservation value, computed as $Q^{X,F}(i,j) = Q_g^{X,F}(i,j)Q_\alpha^{X,F}(i,j)$. Here, $Q_g^{X,F}(i,j)$ and $Q_\alpha^{X,F}(i,j)$ denote the edge strength and orientation preservation values at location (i,j), respectively.
- **Normalized Mutual Information Metric** Q_{MI} [56]. This metric is defined as:

$$Q_{MI} = 2\left[\frac{MI_{A,F}}{H(A)+H(F)} + \frac{MI_{B,F}}{H(B)+H(F)}\right] \tag{20}$$

  where $H(\cdot)$ represents the entropy of the image, and $MI_{X,F}$ denotes the mutual information between source image X and the fused image F.
- **Phase Congruency-based Metric** Q^{P} [57]. This metric captures image salient features through phase congruency:

$$Q^{P} = (P_p)^{\alpha}(P_M)^{\beta}(P_m)^{\gamma} \tag{21}$$

  where P_p, P_M, and P_m represent the correlation coefficients between the fused and source images.
- **Structural Similarity Metrics** Q_S and Q_W [58]. Based on the universal image quality index (UIQI):

$$Q_S = \sum_{w\in W} c(w)\max\{\lambda(w)Q_0(A,F|w), (1-\lambda(w))Q_0(B,F|w)\} \tag{22}$$

$$Q_W = \sum_{w\in W} c(w)[\lambda(w)Q_0(A,F|w) + (1-\lambda(w))Q_0(B,F|w)] \tag{23}$$

  where $Q_0(A,F|w)$ represents the UIQI map computed within a sliding window w, and $\lambda(w)$ is determined by the local saliency of each source image. Q_S is a selective weighted average that better reflects the fusion quality when source images are complementary, while Q_W considers the weighted average of both source images' contributions.

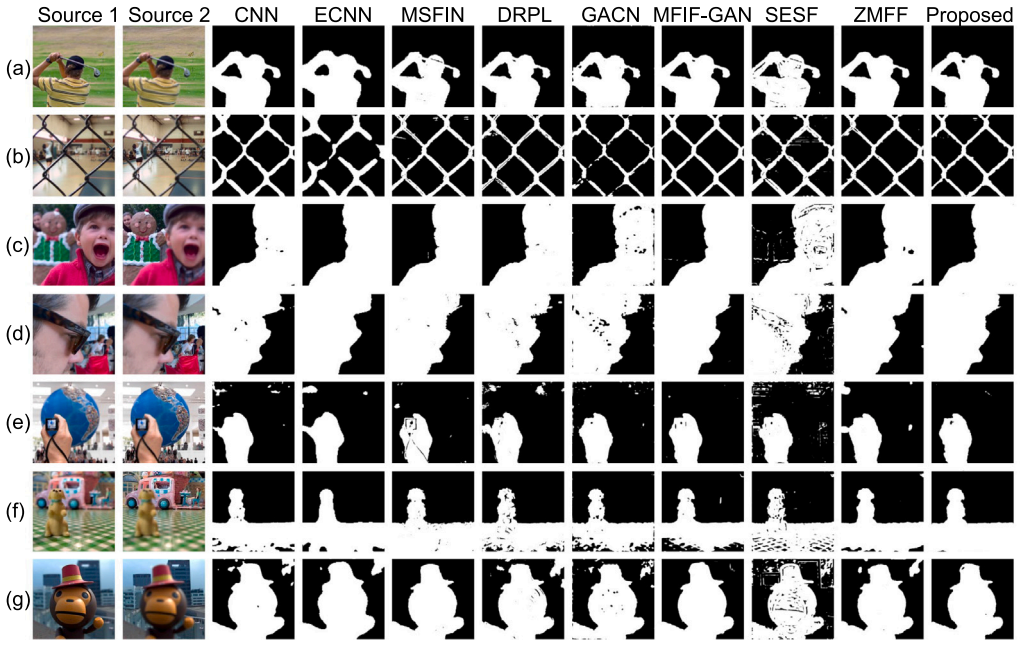

Fig. 7. Comparisons of decision maps for different methods on the Lytro [49] dataset. From left to right: source image 1, source image 2, CNN [23], ECNN [52], MSFIN [6], DRPL [53], GACN [54], MFIF-GAN [26], SESF [24], ZMFF [29], and the proposed method.

- **Human Perception-inspired Metric** Q_{CB} [59]. This metric simulates the human visual system through:

$$Q_{GQM}(i,j) = \lambda_A(i,j)Q_{AF}(i,j) + \lambda_B(i,j)Q_{BF}(i,j) \tag{24}$$

where $Q_{AF}(i,j)$ and $Q_{BF}(i,j)$ measure the contrast information preservation.

All quantitative evaluations were performed using single-channel grayscale images to ensure consistency, and the average score over all testing examples in each dataset is reported.

4.3. Qualitative and quantitative comparison on Lytro dataset

4.3.1. Qualitative comparison

We analyze the decision map quality across different MFF networks in Fig. 7. In Fig. 7(a), only MSFIN, MFIF-GAN, and our BridgeMFF successfully identify the defocused background between the golf club and right arm. However, MFIF-GAN underestimates the region size, while MSFIN exhibits significant errors on the club. For the grid-patterned ground in Fig. 7(f), which comprises multiple small homogeneous regions, only MSFIN and our method achieve accurate detection, while others, particularly the spatial frequency-based SESF, show notable errors. Similarly, in the homogeneous sky region (Fig. 7(g)), only MSFIN and our method correctly determine the focus condition. Across all test cases in Fig. 7, our method demonstrates superior accuracy in decision map generation, especially for challenging homogeneous regions.

Fig. 8 presents fusion results on the Lytro dataset, with challenging regions highlighted and magnified (red dashed boxes). In the first example, end-to-end methods (IFCNN, U2Fusion, SDNet, SwinFusion) effectively preserve lawn textures, while among decision map-based approaches, only our BridgeMFF and MFIF-GAN achieve comparable quality. In the second example, while end-to-end methods successfully fuse focused regions, several decision map-based methods (CNN, ECNN, DRPL) produce incorrect results. Our BridgeMFF maintains high accuracy in both cases, demonstrating the robustness of its decision map generation.

4.3.2. Quantitative comparison

Table 1 presents a comprehensive quantitative evaluation on the Lytro dataset. Our method achieves state-of-the-art performance, ranking first in five out of six metrics ($Q^{AB/F}$, Q_{MI}, Q^P, Q_W, and Q_{CB}) and second in Q_S. These results demonstrate our method's superior capability in preserving mutual information, gradient features, visual information fidelity, and structural similarity. The quantitative improvements align well with our qualitative observations on the Lytro [49] dataset.

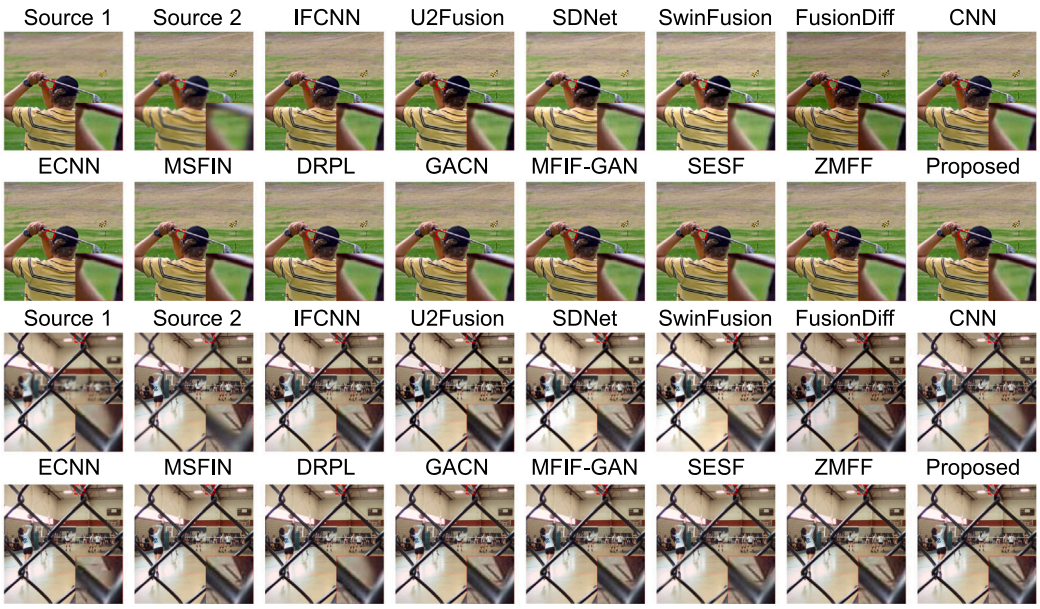

Fig. 8. Comparisons of fusion results for different methods on the Lytro [49] dataset: IFCNN-MAX [31], U2Fusion [32], SDNet [33], SwinFusion [35], FusionDiff [36], CNN [23], ECNN [52], MSFIN [6], DRPL [53], GACN [54], MFIF-GAN [26], SESF [24], ZMFF [29], and the proposed method.

Table 1
Quantitative comparison of different methods on the Lytro [49] dataset.

Methods	$Q^{AB/F}\uparrow$	$Q_{MI}\uparrow$	$Q_S\uparrow$	$Q^P\uparrow$	$Q_W\uparrow$	$Q_{CB}\uparrow$
End-to-end methods						
IFCNN-MAX (2020) [31]	0.6784	0.8863	0.8108	0.2962	0.9013	0.5986
U2Fusion (2020) [32]	0.6190	0.7803	0.8237	0.2994	0.8909	0.5159
SDNet (2021) [33]	0.6441	0.8464	0.8352	0.3072	0.8934	0.5739
MFF-GAN (2021) [34]	0.6222	0.7930	0.8067	0.2840	0.8887	0.5399
SwinFusion (2022) [35]	0.6597	0.8404	0.8364	0.3117	0.9011	0.5745
FusionDiff (2024) [36]	0.6744	0.8692	0.8070	0.2900	0.8980	0.5747
Decision map-based methods						
CNN (2017) [23]	0.7019	1.0424	0.8094	0.2968	0.8976	0.6628
ECNN (2019) [52]	0.7030	1.0723	0.8086	0.2945	0.8946	0.6698
DRPL (2020) [53]	0.7574	<u>1.1405</u>	0.9449	0.8435	0.9397	<u>0.8035</u>
SESF (2020) [24]	0.7031	1.0524	0.8083	0.2950	0.8977	0.6657
MFIF-GAN (2021) [26]	0.7029	1.0618	0.8077	0.2960	0.8982	0.6660
MSFIN (2021) [6]	0.7045	1.0601	0.8080	0.2973	0.8990	0.6664
GACN (2022) [54]	<u>0.7581</u>	1.1334	**0.9461**	<u>0.8443</u>	<u>0.9405</u>	0.8024
ZMFF (2023) [29]	0.6635	0.8694	0.8072	0.2890	0.8951	0.6136
Proposed	**0.7586**	**1.1432**	<u>0.9452</u>	**0.8459**	**0.9406**	**0.8062**
Enhancement (%)	+0.07%	+0.24%	−0.10%	+0.19%	+0.01%	+0.34%

4.4. Qualitative and quantitative comparison on MFI-WHU dataset

4.4.1. Qualitative comparison

We further evaluate our method on the MFI-WHU dataset (Fig. 9). While this synthetic dataset with Gaussian blur presents lower fusion complexity compared to the real-world Lytro dataset, we specifically focus on challenging cases from its 120 image pairs. These cases feature either extensive homogeneous regions (snow in Fig. 9(d), sky in Fig. 9(a)(c)(e)(f)(g)) or small focused targets (flagpole in Fig. 9(b)).

Our method demonstrates superior performance across these challenging scenarios. A representative example is Fig. 9(c), where most methods fail to handle the homogeneous sky region due to similar pixel values. In contrast, our approach successfully captures subtle focus variations in these regions, showcasing strong generalization capability.

4.4.2. Quantitative comparison

Table 2 presents quantitative results on the MFI-WHU dataset. Our method achieves superior performance across multiple metrics, ranking first in Q_{MI}, Q_S, Q^P, and Q_{CB}, while maintaining competitive performance in Q_W (second place). Most notably, we demonstrate a substantial improvement in Q^P, surpassing the second-best method GACN by a significant margin of 9.17%.

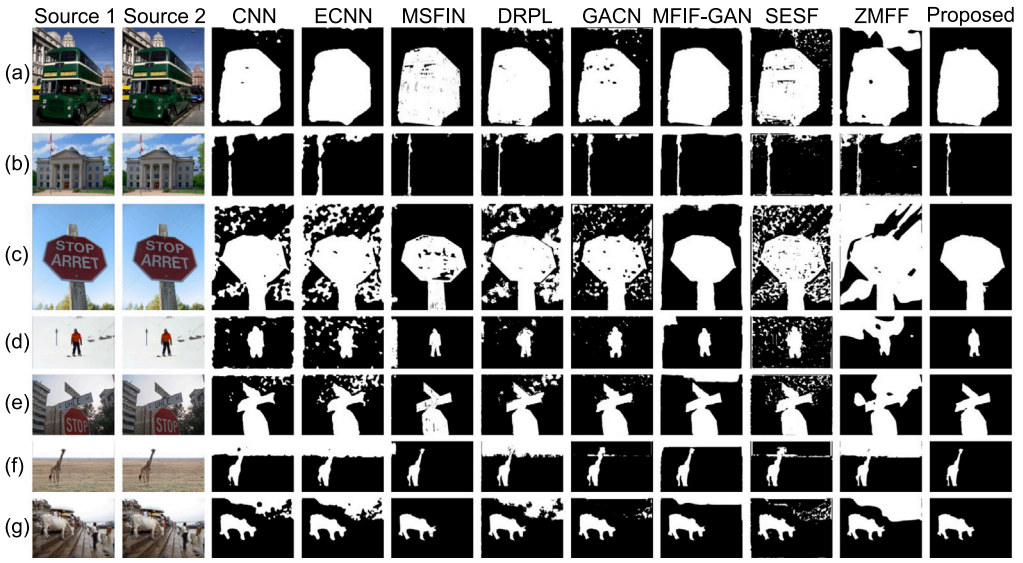

Fig. 9. Comparisons of decision maps for different methods on the MFI-WHU [34] dataset. From left to right: source image 1, source image 2, CNN [23], ECNN [52], MSFIN [6], DRPL [53], GACN [54], MFIF-GAN [26], SESF [24], ZMFF [29], and the proposed method.

Table 2
Quantitative comparison of different methods on the MFI-WHU [34] dataset.

Methods	$Q^{AB/F}\uparrow$	$Q_{MI}\uparrow$	$Q_S\uparrow$	$Q^P\uparrow$	$Q_W\uparrow$	$Q_{CB}\uparrow$
End-to-end methods						
IFCNN-MAX (2020) [31]	0.6936	0.9070	0.9463	0.7414	0.9301	0.7365
U2Fusion (2020) [32]	0.5917	0.7308	0.9111	0.6491	0.8905	0.5905
SDNet (2021) [33]	0.6889	0.9086	0.9478	0.7323	**0.9337**	0.7523
MFF-GAN (2021) [34]	0.6496	0.8244	0.9332	0.6956	0.9227	0.6619
SwinFusion (2022) [35]	0.6777	0.8534	0.9440	0.7209	0.9289	0.7427
FusionDiff (2024) [36]	0.6762	0.8886	0.9169	0.7150	0.9030	0.6998
Decision map-based methods						
CNN (2017) [23]	0.7276	1.1624	0.9486	0.7563	0.9317	0.8269
ECNN (2019) [52]	**0.7314**	1.1827	0.9478	0.7541	0.9303	0.8213
DRPL (2020) [53]	0.7305	1.1672	0.9488	0.7523	0.9317	0.8194
SESF (2020) [24]	0.7267	1.1627	0.9476	0.7552	0.9303	0.8200
MFIF-GAN (2021) [26]	0.7302	1.1368	0.9479	0.7544	0.9310	0.8099
MSFIN (2021) [6]	0.7273	1.1827	0.9484	0.7556	0.9313	0.8247
GACN (2022) [54]	0.7259	1.1685	0.9492	0.7589	0.9317	0.8199
ZMFF (2023) [29]	0.6193	0.6953	0.8964	0.6181	0.8988	0.6732
Proposed	0.7280	**1.1949**	**0.9495**	**0.8285**	0.9317	**0.8287**
Enhancement (%)	−0.47%	+1.03%	+0.07%	+9.17%	−0.21%	+0.22%

4.5. More comparisons

4.5.1. Comparison of fusion performance stability on the Lytro [49] dataset.

We analyze the stability of fusion performance across the Lytro dataset in Fig. 10, which visualizes performance metrics for each image pair under different fusion methods. Our BridgeMFF demonstrates consistent high performance across all 20 pairs, achieving comparable stability with GACN [54] and DRPL [53], while significantly outperforming other methods in Q_S and Q^P metrics.

Notably, both our method and DRPL operate without decision map post-processing, while GACN employs minimal post-processing (guided filtering only). This observation suggests that decision map post-processing might be a key factor affecting fusion stability. While post-processing operations may enhance the visual appearance of decision maps, they introduce subjective factors that do not necessarily improve fusion quality, as noted in DRPL [53]. Our analysis indicates that avoiding such post-processing steps may contribute to more stable and reliable fusion performance.

4.5.2. Qualitative comparison on images with strong defocus blur in the MFFW [50] dataset.

Fig. 11 illustrates the decision maps generated by different methods when handling images with strong defocus blur. Several methods (CNN, ECNN, DRPL, GACN, SESF, and ZMFF) show notable limitations, particularly in background region detection. In contrast, MSFIN, MFIF-GAN, and our BridgeMFF maintain robust performance. Our method demonstrates particular strength in handling high-contrast multi-focus pairs, as evidenced in Fig. 11(d), where it achieves superior fusion quality compared to existing approaches.

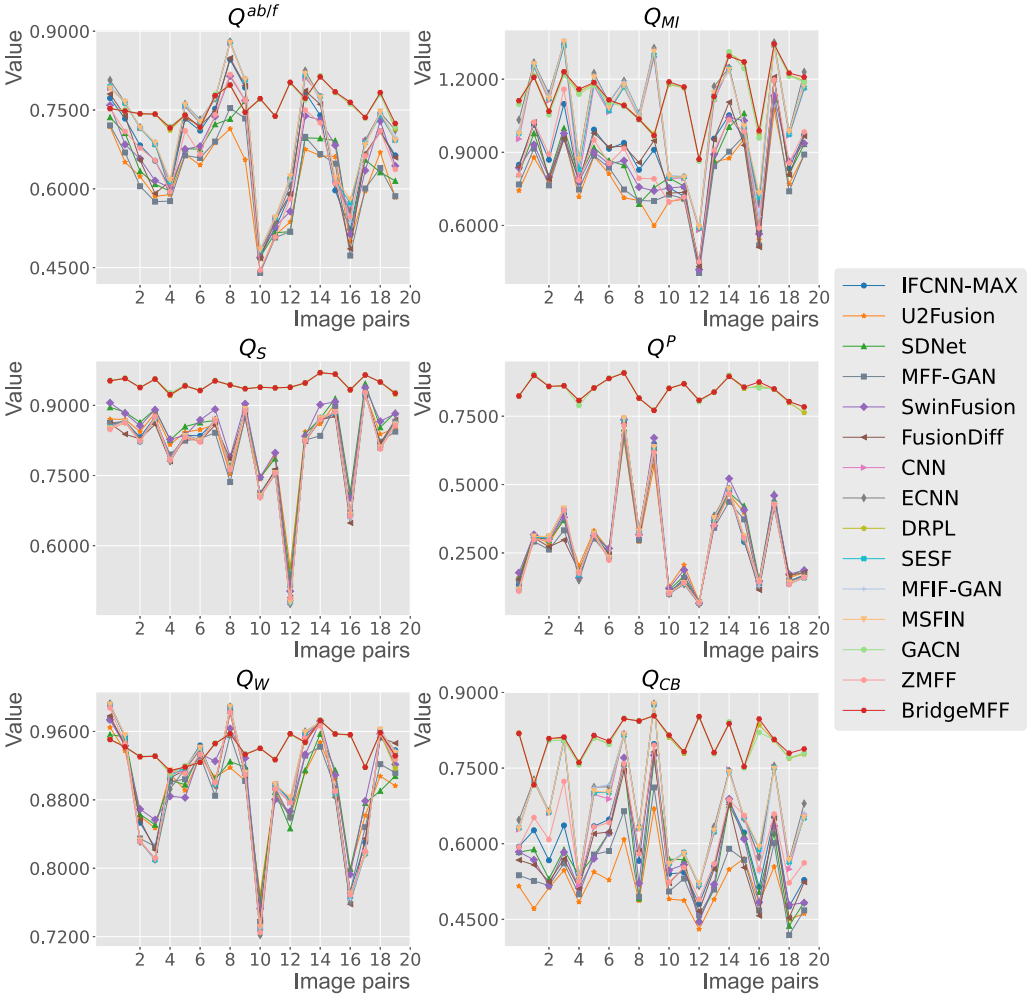

Fig. 10. Comparison of fusion performance stability on the Lytro [49] dataset: IFCNN-MAX [31], U2Fusion [32], SDNet [33], MFF-GAN [34], SwinFusion [35], FusionDiff [36], CNN [23], ECNN [52], DRPL [53], SESF [24], MFIF-GAN [26], MSFIN [6], GACN [54], ZMFF [29], and the proposed method.

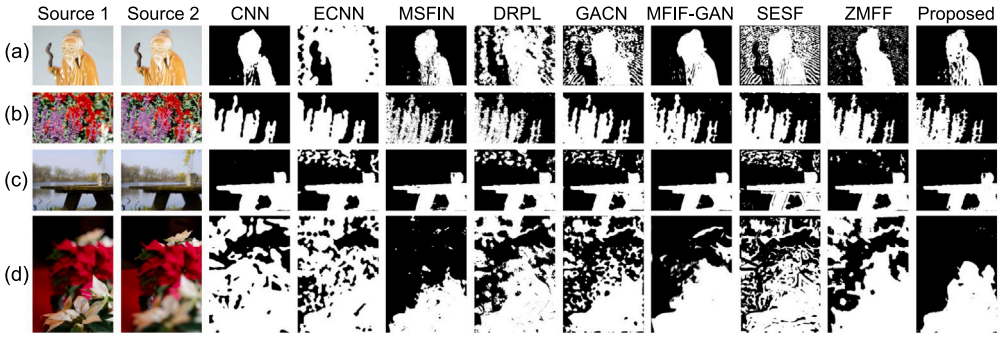

Fig. 11. Comparisons of decision maps for different methods on the MFFW [50] dataset. From left to right: source image 1, source image 2, CNN [23], ECNN [52], MSFIN [6], DRPL [53], GACN [54], MFIF-GAN [26], SESF [24], ZMFF [29], and the proposed method.

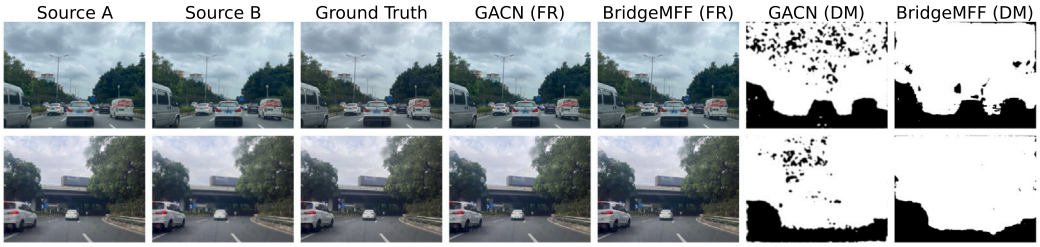

Fig. 12. Visual comparison on Road-MF dataset [51]. From left to right: source images, ground truth, fusion results (FR) of GACN [54] and BridgeMFF, decision maps (DM) of GACN and BridgeMFF.

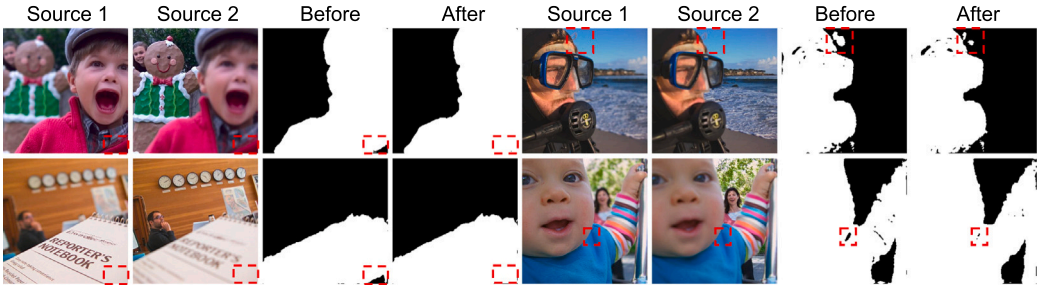

Fig. 13. Comparison of decision maps output by the proposed BridgeMFF before and after using the dual adversarial training strategy.

Table 3
Quantitative comparison of BridgeMFF before and after using BridgeTune on the Lytro [49] dataset.

BridgeTune	$Q^{AB/F}\uparrow$	$Q_{MI}\uparrow$	$Q_S\uparrow$	$Q^P\uparrow$	$Q_W\uparrow$	$Q_{CB}\uparrow$
×	0.7554	1.1402	0.9446	0.8367	0.9386	0.8041
✓	0.7586	1.1432	0.9452	0.8459	0.9406	0.8062
Enhancement (%)	+0.42%	+0.26%	+0.06%	+1.10%	+0.21%	+0.26%

4.5.3. Performance in real-world

We further evaluate our method's real-world performance on the challenging Road-MF dataset [51]. Fig. 12 compares our BridgeMFF with GACN, revealing significant differences in fusion quality. While GACN struggles with homogeneous regions, particularly in sky areas, our method successfully maintains both the smoothness of homogeneous regions and the sharpness of focused details (e.g., buildings and roads). These results demonstrate our method's strong generalization capability, extending beyond benchmark datasets to practical scenarios.

4.6. More analysis

4.6.1. Ablation study

Effectiveness of BridgeTune and BridgeMFF. Fig. 13 illustrates the comparison of decision maps generated by BridgeMFF before and after applying BridgeTune on the Lytro dataset. The visualization clearly demonstrates that BridgeTune significantly suppresses detection errors in decision maps, particularly in challenging homogeneous regions, leading to more accurate fusion results. Notably, even without BridgeTune fine-tuning, BridgeMFF already achieves superior performance compared to many existing methods in Table 1, validating the effectiveness of our network architecture and training strategy. The application of BridgeTune further enhances this strong baseline, as evidenced by consistent improvements across all metrics in Table 3, with gains of 1.10% in Q^P and 0.42% in $Q^{AB/F}$.

Generalization to different architectures. BridgeTune is designed as a general-purpose fine-tuning framework applicable to existing decisionmap-based MFF networks. The framework allows flexible substitution of all three networks (Mamba-G, Mamba-D_s, and Mamba-D_c) with alternative architectures, provided the generator produces fusion masks.

To validate BridgeTune's versatility, we experiment with two representative MFF architectures: MSFIN [6] (featuring multi-scale information interaction) and DRPL [53] (a residual network variant for MFF). These networks were specifically selected to: (1) investigate the prevalence of texture information submergence across different architectural paradigms (multi-scale interaction vs. residual connections), and (2) demonstrate the effectiveness of our adversarial fine-tuning strategy in enhancing decision map quality across diverse network designs. For fair comparison, we maintain consistent experimental settings across all networks, including hyperparameters, loss functions, and training datasets. The training process consists of two phases: initial training for 100 epochs

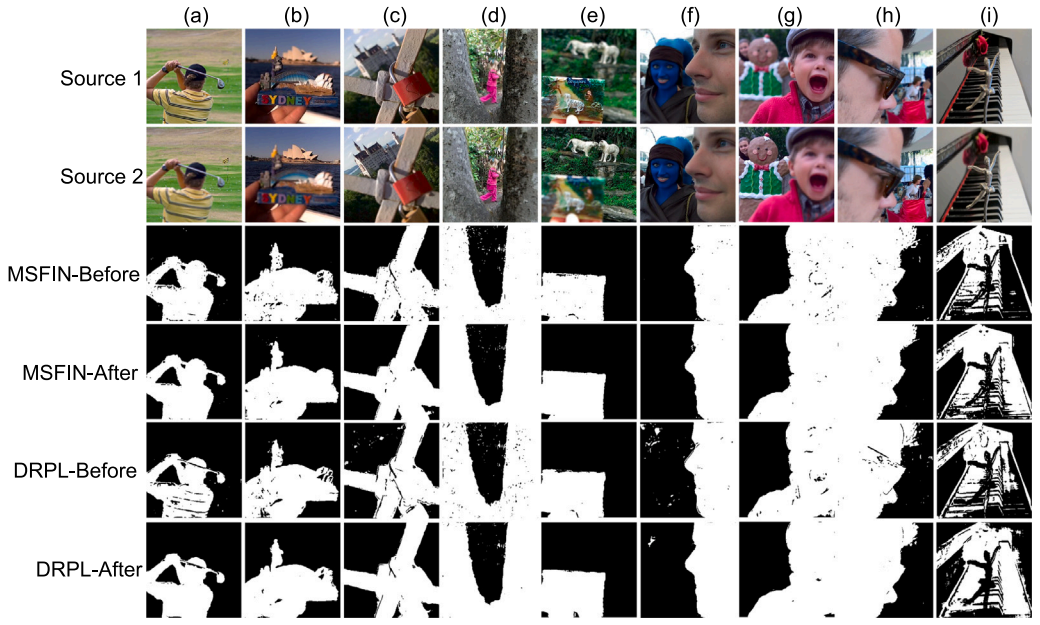

Fig. 14. Comparison of decision maps output by networks of MSFIN [6] and DRPL [53] before and after using BridgeTune.

Table 4
Quantitative comparison of different networks before and after using BridgeTune on the Lytro [49] dataset.

Network	BridgeTune	$Q^{AB/F}\uparrow$	$Q_{MI}\uparrow$	$Q_S\uparrow$	$Q^P\uparrow$	$Q_W\uparrow$	$Q_{CB}\uparrow$
MSFIN [6]	×	0.7537	1.1285	0.9451	0.8353	0.9389	0.8022
	✓	0.7571	1.1362	0.9454	0.8430	0.9404	0.8040
Enhancement (%)		+0.45%	+0.68%	+0.03%	+0.92%	+0.16%	+0.22%
DRPL [53]	×	0.7484	1.1188	0.9442	0.8227	0.9347	0.7977
	✓	0.7546	1.1366	0.9454	0.8351	0.9373	0.8045
Enhancement (%)		+0.83%	+1.59%	+0.13%	+1.51%	+0.28%	+0.85%
BridgeMFF	×	0.7554	1.1402	0.9446	0.8367	0.9386	0.8041
	✓	0.7586	1.1432	0.9452	0.8459	0.9406	0.8062
Enhancement (%)		+0.42%	+0.26%	+0.06%	+1.10%	+0.21%	+0.26%

Table 5
Quantitative comparison of different α values on the Lytro [49] dataset.

α	$Q^{AB/F}\uparrow$	$Q_{MI}\uparrow$	$Q_S\uparrow$	$Q^P\uparrow$	$Q_W\uparrow$	$Q_{CB}\uparrow$
1e−1	0.7573	1.1395	0.9449	0.8446	0.9402	0.8048
1e−2	0.7577	1.1432	0.9451	0.8452	0.9403	0.8058
1e−3	0.7578	1.1425	0.9451	0.8456	0.9404	0.8057
1e−4	**0.7586**	**1.1432**	**0.9452**	**0.8459**	**0.9406**	**0.8062**
1e−5	0.7579	1.1425	0.9449	0.8454	0.9403	0.8056

until convergence, followed by 50 epochs of BridgeTune fine-tuning on the best-performing model. The qualitative and quantitative results before and after BridgeTune are presented in Fig. 14 and Table 4, respectively.

Fig. 14 demonstrates significant error reduction in decision maps post-BridgeTune, particularly in homogeneous regions. A representative example is the piano key area in Fig. 14(i), where both networks initially failed to correctly identify focus conditions but showed marked improvement after BridgeTune application.

While Table 4 shows modest quantitative improvements on the Lytro dataset (enhancements ranging from +0.03% to +1.59%), these metrics warrant careful interpretation. Current evaluation metrics, computed on fused images rather than decision maps due to the absence of ground truth maps, may not fully capture improvements in homogeneous region detection. This limitation arises from two factors: (1) homogeneous regions typically constitute a small portion of the image with similar pixel values regardless of focus condition, and (2) conventional metrics may lack sensitivity to improvements in these regions.

These findings suggest a common limitation across networks: excessive semantic sensitivity after supervised training, particularly detrimental for homogeneous regions. Our method addresses this issue by implicitly moderating network semantic sensitivity, although these improvements may not be fully reflected in conventional evaluation metrics.

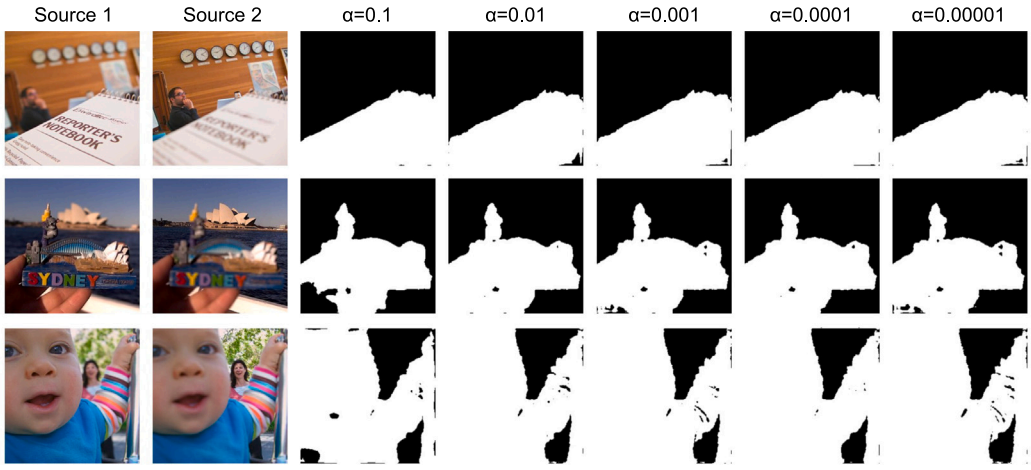

Fig. 15. Qualitative comparison of the effects of different alpha values on the fusion results of image pairs in the Lytro [49] dataset.

Table 6
Comparison of model size, Floating Point Operations (FLOPs) and running time of various methods.

Methods	Model size (M)	FLOPs (G)	Time cost (s)
CNN [23]	8.76	142.23	133.46
ECNN [52]	1.59	14.93	125.53
DRPL [53]	1.07	140.49	0.22
SESF [24]	0.07	4.90	0.26
MFIF-GAN [26]	3.82	693.03	0.32
MSFIN [6]	4.59	26.76	1.10
GACN [54]	0.07	10.89	0.16
ZMFF [29]	4.67	464.53	165.38
Proposed	**0.05**	**0.06**	**0.09**
Reduction (%)	28.57%	98.78%	43.75%

Impact of Hyperparameter α. The hyperparameter α in Eq. (12) controls the network's semantic relevance, with larger values corresponding to reduced semantic sensitivity. While semantic information provides valuable context for decision map generation, texture information is crucial for handling homogeneous and flat regions. Therefore, finding an optimal α value is essential to balance these complementary factors.

Fig. 15 and Table 5 illustrate the impact of different α values on fusion performance. At $\alpha = 1e{-}1$, we observe significant performance degradation due to excessive suppression of semantic information, hindering foreground–background discrimination. As α decreases to $1e{-}4$, fusion performance improves steadily. However, further reduction to $\alpha = 1e{-}5$ renders the adversarial loss term in Eq. (12) negligible, effectively nullifying BridgeTune's influence and resulting in suboptimal performance.

Our experiments indicate that $\alpha = 1e{-}4$ achieves optimal results, maintaining an approximate 1:10 ratio between adversarial (L_{GAN}) and supervised (L_{SUP}) terms in Eq. (12). This ratio effectively balances semantic and texture information processing in the network.

4.6.2. Model efficiency comparison

Table 6 shows the mean runtime of various methods when processing a pair images from the Lytro dataset [49]. For method CNN [23], only the codes of CPU version are available, so its runtime is obtained using CPU processing. The runtimes of other methods are obtained using GPU inference on the same device. As shown in Table 6, our BridgeMFF has the smallest number of parameters and the fastest processing speed, capable of processing over ten image pairs per second on average, thus achieving real-time fusion. Notably, our method achieves a significant reduction of 98.78% in FLOPs compared to the previous state-of-the-art methods while maintaining superior performance. In terms of inference speed, it is 43.75% faster than the previous fastest GACN. Combining the quantitative and qualitative comparative results with other methods from the previous section, it is evident that our BridgeMFF method achieves the best fusion performance and the fastest processing speed, demonstrating promising application prospects. It is worth mentioning that the dual adversarial training strategy proposed in this paper does not introduce any additional network parameters in the inference phase, so it does not affect the inference time.

Table 7
Average rankings of different methods on Lytro [49] and MFI-WHU [34] datasets, sorted by overall average from largest to smallest. Lower numbers indicate better methods.

Method	Lytro	MFI-WHU	Overall average
MFF-GAN [34]	14.50	12.83	13.67
U2Fusion [32]	11.83	14.50	13.17
ZMFF [29]	11.33	14.33	12.83
FusionDiff [36]	11.17	12.17	11.67
SwinFusion [35]	8.33	11.00	9.67
SDNet [33]	10.17	7.83	9.00
IFCNN-MAX [31]	8.00	9.83	8.92
SESF [24]	8.17	6.67	7.42
MFIF-GAN [26]	7.83	6.33	7.08
ECNN [52]	7.83	5.17	6.50
CNN [23]	8.33	4.33	6.33
MSFIN [6]	6.50	4.33	5.42
DRPL [53]	2.67	4.83	3.75
GACN [54]	2.17	4.17	3.17
Proposed	1.17	1.67	1.42

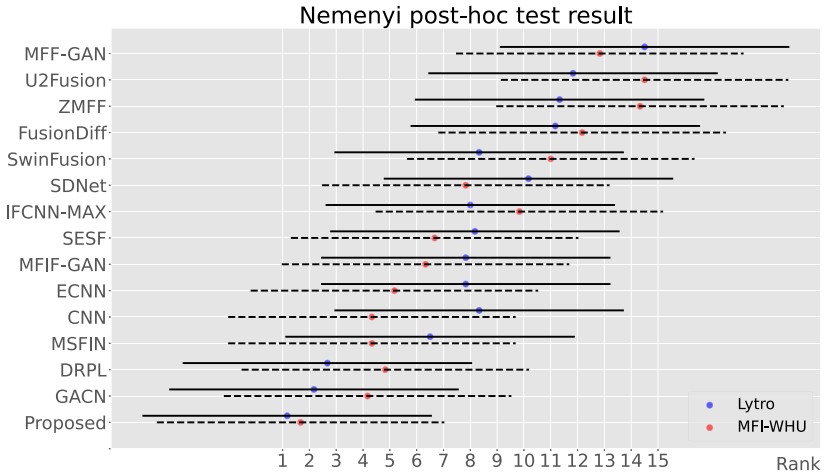

Fig. 16. Results of the Nemenyi post-hoc test: MFF-GAN [34], U2Fusion [32], ZMFF [29], FusionDiff [36], SwinFusion [35], SDNet [33], IFCNN-MAX [31], SESF [24], MFIF-GAN [26], ECNN [52], CNN [23], MSFIN [6], DRPL [53], GACN [54], and the proposed method.

4.6.3. Statistical significance analysis

Table 7 presents a comprehensive performance ranking analysis across the Lytro and MFI-WHU datasets, with lower ranks indicating better performance. Our method demonstrates consistent superiority, achieving top rankings (1.17 and 1.67) on both datasets, while baseline methods like IFCNN-MAX and U2Fusion show consistently inferior performance.

To establish statistical significance, we employ the Nemenyi post-hoc test (Fig. 16). The critical difference diagram reveals two key findings: (1) our method achieves statistically significant improvements over most competitors, particularly U2Fusion, ZMFF, and MFF-GAN; (2) even the strongest baselines (GACN and DRPL with overall rankings of 3.17 and 3.75, respectively) maintain a substantial performance gap from our approach. These results validate both the quantitative superiority and cross-dataset generalization capability of our method.

5. Discussion

While we have demonstrated the effectiveness of BridgeTune on classic networks like MSFIN [6] and DRPL [53], several limitations remain. The primary constraint is that BridgeTune is designed for basic decision map-based fusion pipelines, making it challenging to extend to more sophisticated frameworks (e.g., ZMFF [29], INPA [60]) and inapplicable to end-to-end fusion networks like [36,47]. Additionally, the balance between semantic and texture information is controlled by parameter α; although we empirically identified an effective value, the optimal setting may vary across different scenarios and datasets, potentially requiring multiple experiments for parameter tuning. Furthermore, while BridgeMFF achieves real-time inference, the dual adversarial fine-tuning process involves simultaneous optimization of three networks, demanding substantial GPU memory, which may limit its applicability on resource-constrained training devices. Another limitation is that the current visual state space model requires a CUDA environment, as its core computational modules rely on CUDA acceleration for efficient parallel computing. However, with the advancement of hardware and software ecosystems, support for such models may expand to more platforms in the future.

Moreover, the proposed network achieves state-of-the-art performance with the smallest model size and fastest inference speed among all compared methods. This demonstrates that superior performance can be achieved through the synergy of advanced architecture design (visual state space model) and effective training strategies (dual adversarial fine-tuning), rather than simply scaling up model size. This finding points to a promising direction for future research: combining innovative network architectures with tailored training strategies to achieve both efficiency and effectiveness.

6. Conclusion

In this paper, we address the challenge of poor discrimination in homogeneous regions during multi-focus image fusion, which we attribute to the submergence of low-level texture features by semantic information in deep networks. We propose BridgeTune, a general fine-tuning framework, and BridgeMFF, an efficient fusion network combining visual state space modeling with BridgeTune.

Extensive experiments demonstrate that BridgeTune effectively bridges the semantic-texture gap through dual adversarial learning, showing consistent improvements when integrated with existing networks (improvements of up to 1.59% for DRPL and 0.92% for MSFIN). Our BridgeMFF achieves state-of-the-art performance across multiple datasets while maintaining exceptional efficiency (0.05M parameters, 0.06G FLOPs, and 0.09 s inference time per image pair, representing reductions of 28.57%, 98.78%, and 43.75% respectively compared to previous best methods).

In future work, we plan to explore unsupervised MFF based on our framework. The dual adversarial learning mechanism in BridgeTune, which already demonstrates the ability to learn implicit texture patterns without direct supervision, provides a promising foundation for this direction.

CRediT authorship contribution statement

Xinzhe Xie: Conceptualization, Methodology, Software, Experiment, Writing – original draft, Writing – review & editing. **Buyu Guo:** Conceptualization, Methodology, Investigation, Writing – review & editing, Funding acquisition. **Peiliang Li:** Conceptualization, Validation, Project administration, Supervision, Funding acquisition. **Shuangyan He:** Formal analysis, Resources, Writing – review & editing. **Sangjun Zhou:** Visualization, Validation, Data curation, Software.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Hainan Provincial Joint Project of Sanya Yazhou Bay Science and Technology City (No: 2021JJLH0079), Innovational Fund for Scientific and Technological Personnel of Hainan Province (NO. KJRC2023D19), and the Hainan Provincial Joint Project of Sanya Yazhou Bay Science and Technology City (No. 2021CXLH0020).

Data availability

All data used in this study are publicly available.

References

[1] Fang J, Ning X, Mao T, Zhang M, Zhao Y, Hu S, Wang J. A multi-focus image fusion network combining dilated convolution with learnable spacings and residual dense network. Comput Electr Eng 2024;117:109299.
[2] Duan Z, Luo X, Zhang T. Combining transformers with CNN for multi-focus image fusion. Expert Syst Appl 2024;235:121156.
[3] Zhao F, Zhao W, Lu H, Liu Y, Yao L, Liu Y. Depth-distilled multi-focus image fusion. IEEE Trans Multimed 2021;25:966–78.
[4] Gildenblat J, contributors. PyTorch library for CAM methods. 2021, https://github.com/jacobgil/pytorch-grad-cam.
[5] Xiao B, Xu B, Bi X, Li W. Global-feature encoding U-net (GEU-net) for multi-focus image fusion. IEEE Trans Image Process 2020;30:163–75.
[6] Liu Y, Wang L, Cheng J, Chen X. Multiscale feature interactive network for multifocus image fusion. IEEE Trans Instrum Meas 2021;70:1–16.
[7] Zhao W, Shang C, Lu H. Self-generated defocus blur detection via dual adversarial discriminators. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. CVPR, 2021, p. 6933–42.
[8] Zhou Y, Yang X, Zhang R, Liu K, Anisetti M, Jeon G. Gradient-based multi-focus image fusion method using convolution neural network. Comput Electr Eng 2021;92:107174. http://dx.doi.org/10.1016/j.compeleceng.2021.107174, URL https://www.sciencedirect.com/science/article/pii/S0045790621001762.
[9] Xu H, Fan F, Zhang H, Le Z, Huang J. A deep model for multi-focus image fusion based on gradients and connected regions. IEEE Access 2020;8:26316–27. http://dx.doi.org/10.1109/ACCESS.2020.2971137.
[10] Toet A. Image fusion by a ratio of low-pass pyramid. Pattern Recognit Lett 1989;9(4):245–53.
[11] Burt PJ, Kolczynski RJ. Enhanced image capture through fusion. In: 1993 (4th) international conference on computer vision. IEEE; 1993, p. 173–82.
[12] Lewis JJ, O'Callaghan RJ, Nikolov SG, Bull DR, Canagarajah N. Pixel-and region-based image fusion with complex wavelets. Inf Fusion 2007;8(2):119–30.
[13] Li H, Manjunath B, Mitra SK. Multisensor image fusion using the wavelet transform. Graph Models Image Process 1995;57(3):235–45.
[14] Haghighat MBA, Aghagolzadeh A, Seyedarabi H. Multi-focus image fusion for visual sensor networks in DCT domain. Comput Electr Eng 2011;37(5):789–97. http://dx.doi.org/10.1016/j.compeleceng.2011.04.016, URL https://www.sciencedirect.com/science/article/pii/S0045790611000619. Special Issue on Image Processing.

[15] Zhang Q, Guo B-l. Multifocus image fusion using the nonsubsampled contourlet transform. Signal Process 2009;89(7):1334–46.
[16] Liu Z, Chai Y, Yin H, Zhou J, Zhu Z. A novel multi-focus image fusion approach based on image decomposition. Inf Fusion 2017;35:102–16.
[17] Jiang Y, Wang M. Image fusion with morphological component analysis. Inf Fusion 2014;18:107–18.
[18] Li S, Kwok JT, Wang Y. Combination of images with diverse focuses using the spatial frequency. Inf Fusion 2001;2(3):169–76. http://dx.doi.org/10.1016/S1566-2535(01)00038-0, URL https://www.sciencedirect.com/science/article/pii/S1566253501000380.
[19] Huang W, Jing Z. Multi-focus image fusion using pulse coupled neural network. Pattern Recognit Lett 2007;28(9):1123–32. http://dx.doi.org/10.1016/j.patrec.2007.01.013, URL https://www.sciencedirect.com/science/article/pii/S0167865507000402.
[20] Li M, Cai W, Tan Z. A region-based multi-sensor image fusion scheme using pulse-coupled neural network. Pattern Recognit Lett 2006;27(16):1948–56. http://dx.doi.org/10.1016/j.patrec.2006.05.004, URL https://www.sciencedirect.com/science/article/pii/S0167865506001565.
[21] Liu Y, Liu S, Wang Z. Multi-focus image fusion with dense SIFT. Inf Fusion 2015;23:139–55. http://dx.doi.org/10.1016/j.inffus.2014.05.004, URL https://www.sciencedirect.com/science/article/pii/S1566253514000670.
[22] Aslantas V, Toprak AN. Multi-focus image fusion based on optimal defocus estimation. Comput Electr Eng 2017;62:302–18. http://dx.doi.org/10.1016/j.compeleceng.2017.02.003, URL https://www.sciencedirect.com/science/article/pii/S0045790617302501.
[23] Liu Y, Chen X, Peng H, Wang Z. Multi-focus image fusion with a deep convolutional neural network. Inf Fusion 2017;36:191–207.
[24] Ma B, Zhu Y, Yin X, Ban X, Huang H, Mukeshimana M. Sesf-fuse: An unsupervised deep model for multi-focus image fusion. Neural Comput Appl 2021;33:5793–804.
[25] Ma J, Le Z, Tian X, Jiang J. SMFuse: Multi-focus image fusion via self-supervised mask-optimization. IEEE Trans Comput Imaging 2021;7:309–20.
[26] Wang Y, Xu S, Liu J, Zhao Z, Zhang C, Zhang J. MFIF-GAN: A new generative adversarial network for multi-focus image fusion. Signal Process, Image Commun 2021;96:116295.
[27] Fang L, Zhao J, Pan Z, Li Y. TPP: Deep learning based threshold post-processing multi-focus image fusion method. Comput Electr Eng 2023;110:108736.
[28] Shao X, Jin X, Jiang Q, Miao S, Wang P, Chu X. Multi-focus image fusion based on transformer and depth information learning. Comput Electr Eng 2024;119:109629. http://dx.doi.org/10.1016/j.compeleceng.2024.109629, URL https://www.sciencedirect.com/science/article/pii/S0045790624005561.
[29] Hu X, Jiang J, Liu X, Ma J. ZMFF: Zero-shot multi-focus image fusion. Inf Fusion 2023;92:127–38.
[30] Ulyanov D, Vedaldi A, Lempitsky V. Deep image prior. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 9446–54.
[31] Zhang Y, Liu Y, Sun P, Yan H, Zhao X, Zhang L. IFCNN: A general image fusion framework based on convolutional neural network. Inf Fusion 2020;54:99–118.
[32] Xu H, Ma J, Jiang J, Guo X, Ling H. U2Fusion: A unified unsupervised image fusion network. IEEE Trans Pattern Anal Mach Intell 2020;44(1):502–18.
[33] Zhang H, Ma J. SDNet: A versatile squeeze-and-decomposition network for real-time image fusion. Int J Comput Vis 2021;129(10):2761–85.
[34] Zhang H, Le Z, Shao Z, Xu H, Ma J. MFF-GAN: An unsupervised generative adversarial network with adaptive and gradient joint constraints for multi-focus image fusion. Inf Fusion 2021;66:40–53.
[35] Ma J, Tang L, Fan F, Huang J, Mei X, Ma Y. SwinFusion: Cross-domain long-range learning for general image fusion via swin transformer. IEEE/CAA J Autom Sin 2022;9(7):1200–17.
[36] Li M, Pei R, Zheng T, Zhang Y, Fu W. FusionDiff: Multi-focus image fusion using denoising diffusion probabilistic models. Expert Syst Appl 2024;238:121664.
[37] Zang Y, Zhou D, Wang C, Nie R, Guo Y. UFA-FUSE: A novel deep supervised and hybrid model for multifocus image fusion. IEEE Trans Instrum Meas 2021;70:1–17.
[38] Wu R, Liu Y, Liang P, Chang Q. Ultralight vm-unet: Parallel vision mamba significantly reduces parameters for skin lesion segmentation. 2024, arXiv preprint arXiv:2403.20035.
[39] Gu A, Dao T. Mamba: Linear-time sequence modeling with selective state spaces. 2023, arXiv preprint arXiv:2312.00752.
[40] Liu Y, Tian Y, Zhao Y, Yu H, Xie L, Wang Y, Ye Q, Liu Y. VMamba: Visual state space model. 2024, arXiv preprint arXiv:2401.10166.
[41] Zhu L, Liao B, Zhang Q, Wang X, Liu W, Wang X. Vision mamba: Efficient visual representation learning with bidirectional state space model. 2024, arXiv preprint arXiv:2401.09417.
[42] Ruan J, Xiang S. Vm-unet: Vision mamba unet for medical image segmentation. 2024, arXiv preprint arXiv:2402.02491.
[43] Yu W, Wang X. MambaOut: Do we really need mamba for vision? 2024, arXiv preprint arXiv:2405.07992.
[44] Ross T-Y, Dollár G. Focal loss for dense object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 2980–8.
[45] Milletari F, Navab N, Ahmadi S-A. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 fourth international conference on 3D vision. 3DV, IEEE; 2016, p. 565–71.
[46] Ruan J, Xiang S, Xie M, Liu T, Fu Y. MALUNet: A multi-attention and light-weight unet for skin lesion segmentation. In: 2022 IEEE international conference on bioinformatics and biomedicine. BIBM, IEEE; 2022, p. 1150–6.
[47] Xie X, Guo B, Li P, He S, Zhou S. SwinMFF: toward high-fidelity end-to-end multi-focus image fusion via swin transformer-based network. Vis Comput 2024;1–24.
[48] Wang L, Lu H, Wang Y, Feng M, Wang D, Yin B, Ruan X. Learning to detect salient objects with image-level supervision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 136–45.
[49] Nejati M, Samavi S, Shirani S. Multi-focus image fusion using dictionary-based sparse representation. Inf Fusion 2015;25:72–84.
[50] Xu S, Wei X, Zhang C, Liu J, Zhang J. MFFW: A new dataset for multi-focus image fusion. 2020, arXiv preprint arXiv:2002.04780.
[51] Li X, Li X, Tan H, Li J. SAMF: small-area-aware multi-focus image fusion for object detection. In: ICASSP 2024-2024 IEEE international conference on acoustics, speech and signal processing. ICASSP, IEEE; 2024, p. 3845–9.
[52] Amin-Naji M, Aghagolzadeh A, Ezoji M. Ensemble of CNN for multi-focus image fusion. Inf Fusion 2019;51:201–14.
[53] Li J, Guo X, Lu G, Zhang B, Xu Y, Wu F, Zhang D. DRPL: Deep regression pair learning for multi-focus image fusion. IEEE Trans Image Process 2020;29:4816–31.
[54] Ma B, Yin X, Wu D, Shen H, Ban X, Wang Y. End-to-end learning for simultaneously generating decision map and multi-focus image fusion result. Neurocomputing 2022;470:204–16.
[55] Xydeas CS, Petrovic V, et al. Objective image fusion performance measure. Electron Lett 2000;36(4):308–9.
[56] Qu G, Zhang D, Yan P. Information measure for performance of image fusion. Electron Lett 2002;38(7):1.
[57] Zhao J, Laganiere R, Liu Z. Performance assessment of combinative pixel-level image fusion based on an absolute feature measurement. Int J Innov Comput Inf Control 2007;3(6):1433–47.
[58] Piella G, Heijmans H. A new quality metric for image fusion. In: Proceedings 2003 international conference on image processing (cat. no. 03CH37429). vol. 3, IEEE; 2003, p. III–173.
[59] Chen Y, Blum RS. A new automated quality assessment algorithm for image fusion. Image Vis Comput 2009;27(10):1421–32.
[60] Hu X, Jiang J, Wang C, Liu X, Ma J. Incrementally adapting pretrained model using network prior for multi-focus image fusion. IEEE Trans Image Process 2024.