EFFICIENT IMAGE SUPER-RESOLUTION WITH MULTI-BRANCH MIXER TRANSFORMER

LONG ZHANG✉ AND YI WAN

School of Information Science and Engineering, Lanzhou University, 222 S. Tianshui Rd., Lanzhou, 730000, China.
e-mail: lzhang2019@lzu.edu.cn, wanyi@lzu.edu.cn

(Received September 24, 2024; revised December 17, 2024; accepted January 15, 2025)

ABSTRACT

Deep learning methods have demonstrated significant advancements in single image super-resolution (SISR), with Transformer-based models frequently outperforming CNN-based counterparts in performance. However, due to the self-attention mechanism in Transformers, achieving lightweight models remains challenging compared to CNN-based approaches. In this paper, we propose a lightweight Transformer model termed Multi-Branch Mixer Transformer (MBMT) for SR. The design of MBMT is motivated by two main considerations: while self-attention excels at capturing long-range dependencies in features, it struggles with extracting local features. Secondly, the quadratic complexity of self-attention forms a significant challenge in building lightweight models. To address these problems, we propose a Multi-Branch Token Mixer (MBTM) to extract richer global and local information. Compared to other Transformer-based SR networks, MBTM achieves a balance between capturing global information and reducing the computational complexity of self-attention through its compact multi-branch structure. Specifically, MBTM consists of three parts: shifted window attention, depthwise convolution, and active token mixer. This multi-branch structure handles both long-range dependencies and local features simultaneously, enabling us to achieve excellent SR performance with just a few stacked modules. Experimental results demonstrate that MBMT achieves competitive performance while maintaining model efficiency compared to SOTA methods.

Keywords: active token mixer, multi-branch token mixer, single image super-resolution, transformer.

INTRODUCTION

Single-image super-resolution (SISR) is focused on upscaling low-resolution (LR) images to generate high-resolution (HR) images. The success of Convolutional Neural Networks (CNNs) in the field of SR can be attributed to their robust feature extraction capabilities and end-to-end learning framework. Researchers have achieved significant advancement by training CNN models to learn the mapping from LR to HR images (Dong *et al.*, 2016a;b; Kim *et al.*, 2016a; Ledig *et al.*, 2016; Wang *et al.*, 2018; Tai *et al.*, 2017b; Lim *et al.*, 2017a). Despite the impressive performance of CNN models in generating high-quality SR images, the depth and complexity of these networks result in large model sizes and high computational requirements, presenting obstacles to deployment in resource limited environments. To address these challenges, researchers are actively exploring the design of lightweight super-resolution architectures, aiming to reduce model size and computational costs while maintaining performance standards (Hui *et al.*, 2019; Ahn *et al.*, 2018; Li *et al.*, 2022c; Kong *et al.*, 2022; Liu *et al.*, 2020; Lai *et al.*, 2017; Shi *et al.*, 2016; Tai *et al.*, 2017a; Kim *et al.*, 2016b).

The receptive field of CNN-based SR models is primarily constrained by the limitations in the size and depth of convolutional kernels. These limitations restrict the model's ability to capture long-term dependencies in images, potentially leading to distortion or blurriness, particularly evident in larger-sized images or complex scenes. To tackle this challenge, researchers have recently started exploring the utilization of emerging architectures like Transformers for super-resolution tasks (Vaswani *et al.*, 2017; Dosovitskiy *et al.*, 2021; Wang *et al.*, 2024; Cao *et al.*, 2022; Zhou *et al.*, 2023; Chen *et al.*, 2023). Unlike CNNs, Transformers are not constrained by the limitations of receptive fields, enabling models to effectively capture global information and long-term dependencies, which has the potential to enhance the performance and effectiveness of SR models. Although Transformers have made significant progress in computer vision, the training and inference require significant computational resources, largely attributed to self-attention (SA) (Vaswani *et al.*, 2017), which results in a quadratic growth in computational complexity as the token length increases. Moreover, transformers struggle with handling local information, which consequently limits the model's performance. Based on these reasons, designing lightweight transformer-based SR models remains a challenging problem in the field (Choi *et al.*, 2022; Gao *et al.*, 2022; Zhang *et al.*, 2023; Liang *et al.*, 2021; Wang *et al.*, 2023).

To address the aforementioned issues, this paper proposes a lightweight SR network named Multi-Branch Mixer Transformer (MBMT), which effectively balances global and local feature extraction by incorporating multiple token mixing strategies. In contrast to other conventional Transformer architectures, the proposed MBMT introduces a unique module named multi-branch token mixer (MBTM), which adopts multiple token mixing methods, including self-attention. MBTM represents a more complex token mixing approach, designed to trade off network depth for wider width, thus strengthening the model's learning capability and generalization ability. It consists of three branches: self-attention, depthwise convolution (DWConv)(Howard *et al.*, 2017a), which refers to a 1 × 1 convolution, and active token mixer(ATM) (Wei *et al.*, 2022), which is a method for providing long-range dependencies through pixel reorganization. Each branch is tasked with capturing long-term dependencies, extracting local features, and expanding the global information space, respectively. Compared to other Transformer-based models, MBTM leverages the DWConv branch to effectively extract local information, addressing the limitation of Transformers in capturing fine-grained details. Additionally, the ATM branch expands the global feature space captured by the Self-Attention branch while maintaining a lower computational complexity than Self-Attention. This multi-branch design allows MBTM to achieve performance comparable to deeper Transformer-based models with fewer network layers, effectively addressing the limitations in current SOTA methods. Drawing inspiration from DAT (Chen *et al.*, 2023), we incorporate adaptive spatial and channel attention mechanisms to ensure sufficient interaction among the information extracted from the three branches, hence augmenting MBTM's feature representation capability. Benefiting from the efficiency of MBTM and a shallower network structure, our MBMT significantly enhances SR model performance and reduces complexity, as shown in Fig. 1.

The primary contributions of this paper are enumerated as follows:

1. We propose a novel multi-branch token mixer designed to capture long-term dependencies while simultaneously extracting local information. Additionally, we employ spatial and channel attention mechanisms to interact features at different levels, significantly enhanced the model's super-resolution performance.
2. We introduce ATM within MBTM to achieve richer global information interaction while avoiding excessive computational costs associated with the self-attention mechanism.
3. We conduct extensive experiments to demonstrate that our proposed MBMT achieves superior image restoration quality while maintaining low complexity.

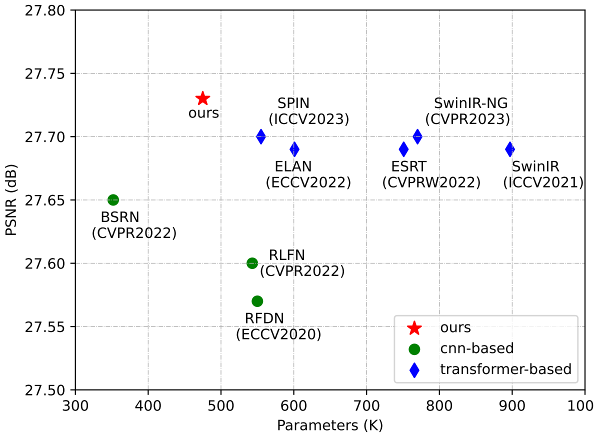


Fig. 1. *Performance vs parameters comparison for BSD100 ×4 dataset. The proposed MBMT achieves better SR quality than previous approaches, with a slimmer model.*

RELATED WORK

EFFICIENT SR METHODS

In the domain of CNN-based SISR, the restoration of high-quality HR images typically relies on complex network architectures and a substantial number of parameters. However, these architectures commonly demand significant computational resources, consequently limiting their practicality in resource-constrained environments. To achieve a balance between model complexity and performance, there is a growing focus on lightweight image super-resolution approaches (Dong *et al.*, 2016b; Hui *et al.*, 2019; Ahn *et al.*, 2018; Kong *et al.*, 2022; Liu *et al.*, 2020; Lai *et al.*, 2017; Shi *et al.*, 2016; Li *et al.*, 2022c). With a straightforward convolutional architecture, SRCNN (Dong *et al.*, 2016a) accomplishes end-to-end SISR, highlighting the pioneering role of convolutional neural networks in super-resolution tasks. FSRCNN (Dong *et al.*, 2016b) significantly improves inference speed while maintaining excellent SR performance by optimizing the network architecture of SRCNN. ESPCN (Shi *et al.*, 2016) innovatively proposes the use of sub-pixel convolutional layers to rearrange LR features into HR images, effectively addressing the limitations associated with pixel-wise operations in conventional upscale approaches. The approach demonstrates remarkable outcomes in terms of both visual fidelity and computational efficiency. The inception of ResNet (He *et al.*, 2016) has stimulated the growth of numerous lightweight SR models with residual structures, driving substantial progress in the field of super-resolution(Zhang *et al.*, 2018; Hui *et al.*, 2019; Liu *et al.*, 2020; Kong *et al.*, 2022; Li *et al.*, 2022c;b). In recent years, there has been a growing trend of transformer-based SR networks (Choi *et al.*, 2022; Chen *et al.*, 2023; Wang *et al.*, 2023; Zhang *et al.*, 2023) achieving comparable or even superior performance compared to traditional CNN methods.

TRANSFORMER FOR SR

The success of the Transformer in natural language processing has paved the way for innovative approaches in image processing tasks through its application into computer vision, achieving performance levels that match or even surpass those of traditional convolutional neural networks (Vaswani *et al.*, 2017; Dosovitskiy *et al.*, 2021; Liu *et al.*, 2021). Employing the Transformer in super-resolution tasks enables the model to leverage its strength in capturing global information, thus contributing to the understanding of image structure and content for the generation of high-quality HR images (Zhou *et al.*, 2023; Cao *et al.*, 2022; Chen *et al.*, 2023; Wang *et al.*, 2024). TTSR (Yang *et al.*, 2020) introduces a texture transformer to compute the correlation between the LR image and the reference (Ref) image. By utilizing both hard-attention and soft-attention modules to learn the joint features between LR and Ref, it effectively transfers texture information from Ref to the HR image. SWinIR (Liang *et al.*, 2021) makes use of the Swin Transformer (Liu *et al.*, 2021) architecture for SISR tasks. It capitalizes on the Transformer's self-attention mechanism to capture long-term dependencies in images, thereby enhancing feature extraction and processing efficiency through hierarchical structures. In comparison to conventional convolutional neural networks, SWinIR demonstrates outstanding performance in various SR tasks. ESRT (Lu *et al.*, 2022) is a lightweight architecture that combines CNN and Transformer. The model employs the feature split module to split long sequences into multiple sub-sequences, enabling attention operations on these sub-sequences to reduce GPU memory consumption. To tackle the limited receptive field challenge in SWinIR, Haram *et al.* (Choi *et al.*, 2022) introduced N-Gram (Li *et al.*, 2022a) context into SWin and proposed NGSWin. NGSWin enlarges the visible region and restores degraded pixels through sliding window self-attention interaction. LBNet (Gao *et al.*, 2022) introduces a recursive mechanism within the Transformer, allowing the model to learn global information effectively without significantly increasing GPU memory consumption and model parameters. DAT (Chen *et al.*, 2023) employs a strategy of alternating spatial and channel self-attention within Transformer blocks. This enables feature aggregation across spatial and channel dimensions, enhancing the model's image representation capabilities and significantly improving image super-resolution performance.

TOKEN MIXER

Within the Transformer model, the input feature is decomposed into a sequence of tokens, where each token interacts with other tokens via a self-attention mechanism. This interaction process is commonly referred to as token mixing (Vaswani *et al.*, 2017; Dosovitskiy *et al.*, 2021). However, self-attention leads to quadratic growth in computational complexity with longer sequences, resulting in increased costs and slower training and inference speeds, requiring a significant increase in computational resources. MLP-like mixer (Tolstikhin *et al.*, 2021; Touvron *et al.*, 2022) demonstrated that exclusively employing MLPs can match Transformer's performance, indicating the potential for substituting self-attention with basic token mixers. Poolformer (Yu *et al.*, 2022) is a Transformer-based model that replaces the self-attention module in Transformer with a simple spatial pooling operation as a token mixer for basic token mixing. This approach enables the model to achieve performance equivalent to Transformer and MLP-like models. Huang *et al.* (Huang *et al.*, 2023) employed Fourier transformation to transform tokens into the frequency domain and conducted adaptive filtering operations on the transformed tokens, enabling lightweight token mixing. ATM (Wei *et al.*, 2022) accomplishes adaptive global information integration by recomposing tokens, demonstrating outstanding performance with limited computational cost.

THE METHOD

In this section, we extensively describe the technical details of the proposed MBMT. Firstly, we provide a comprehensive description of the

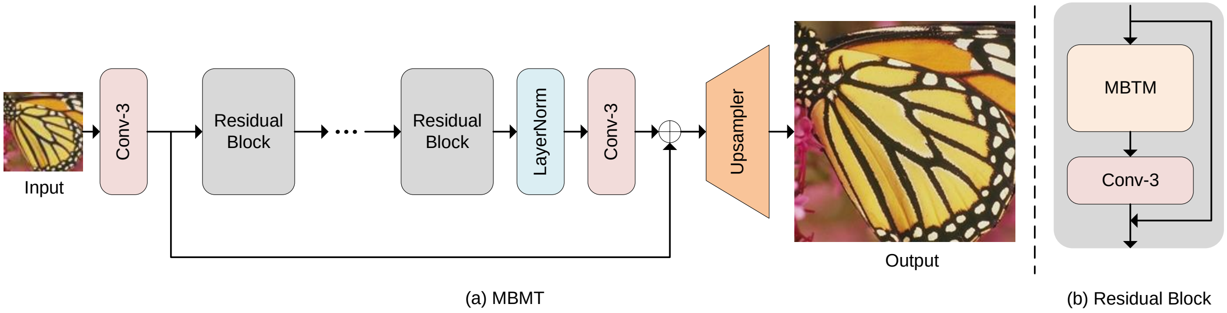


Fig. 2. *The architecture of Multi-Branch Mixer Transformer (MBMT).*

overall architecture of the network in Section 3.1. In Section 3.2, we further explore the details of the core module of multi-branch token mixer (MBTM). Finally, we introduce the structure of active token mixer (ATM) in detail.

NETWORK ARCHITECTURE

The MBMT illustrated in Fig. 2 aims to learn the end-to-end mapping function $\mathscr{F}(\cdot)$ from I_{LR} to I_{HR} images.

$$\hat{I}_{HR} = \mathscr{F}(I_{LR};\theta), \tag{1}$$

where, θ denotes the learnable parameters. The overall structure of the proposed MBMT can be decomposed into three parts: the expanding layer, the feature extraction layer, and the image reconstruction layer.

Expanding layer The expanding layer is mainly utilized to increase the channel dimensions of the input image, thereby introducing additional features and contextual information to enhance the model's performance:

$$F_l = Conv_3(I_{LR}), \tag{2}$$

where, $Conv_3(\cdot)$ denotes a 3×3 convolutional kernel, while F_l represents the low-level features extracted through the expanding layer.

Feature extraction layer To further extract complex and contextually meaningful features from the input low-level features, F_l is fed into the feature extraction layer to generate high-level features F_h:

$$\begin{aligned} F_h &= \mathscr{F}_{ext}(F_l) + F_l, \\ \mathscr{F}_{ext}(\cdot) &= Conv_3\left(LN\left(\mathscr{H}^i_{MBTM}(\cdot)\right)\right), \quad i = 1,\cdots,n. \end{aligned} \tag{3}$$

As demonstrated in the above equation, the feature extraction layer is composed of a feature extraction function $\mathscr{F}_{ext}(\cdot)$ and a residual connection. Where, $\mathscr{H}^i_{MBTM}(\cdot)$ denotes the i-th MBTM, and $LN(\cdot)$ represents layer normalization.

Reconstruction layer Finally, we scale up the feature maps F_h in the image reconstruction layer to restore high-resolution images.

$$\begin{aligned} \hat{I}_{HR} &= \mathscr{F}_{rec}(F_h), \\ \mathscr{F}_{rec}(\cdot) &= UP_s(Conv_3(\cdot)), \end{aligned} \tag{4}$$

where $UP_s(\cdot)$ denotes the sub-pixel convolution (Shi *et al.*, 2016) and s is the upscale factor. In the training process, we adopt L_1 loss (Zhao *et al.*, 2017) as the cost function, the optimization objective can be calculated as:

$$\arg\min_{\theta} \frac{1}{N}\sum_{i=1}^{N} \left|\mathscr{F}\left(I_i^{LR};\theta\right) - I_i^{HR}\right|. \tag{5}$$

MULTI-BRANCH TOKEN MIXER

In this subsection, we will introduce the core component of the proposed MBMT model, the MBTM. The MBTM adopts the general macro architecture of the metaformer (Yu *et al.*, 2022), which can be formulated as follows:

$$\begin{aligned} F_{hidden} &= \mathscr{T}(LN(F_{in})) + F_{in}, \\ F_{out} &= MLP(LN(F_{hidden})) + F_{hidden}, \end{aligned} \tag{6}$$

where, F_{in}, F_{out}, and F_{hidden} respectively denote the input feature, output feature, and hidden layer feature. $\mathscr{T}(\cdot)$ represents multi-branch token mixer, and $MLP(\cdot)$ refers to multi-layer perceptron.

Multi-branch token mixer In computer vision, a token mixer enables interaction among information (tokens) between different spatial positions (patches) within an image to extract global features. Previous works (Yu *et al.*, 2022; Huang *et al.*, 2023) have explored different token mixing approaches, with self-attention mechanism being the most commonly utilized. However, given that stacking

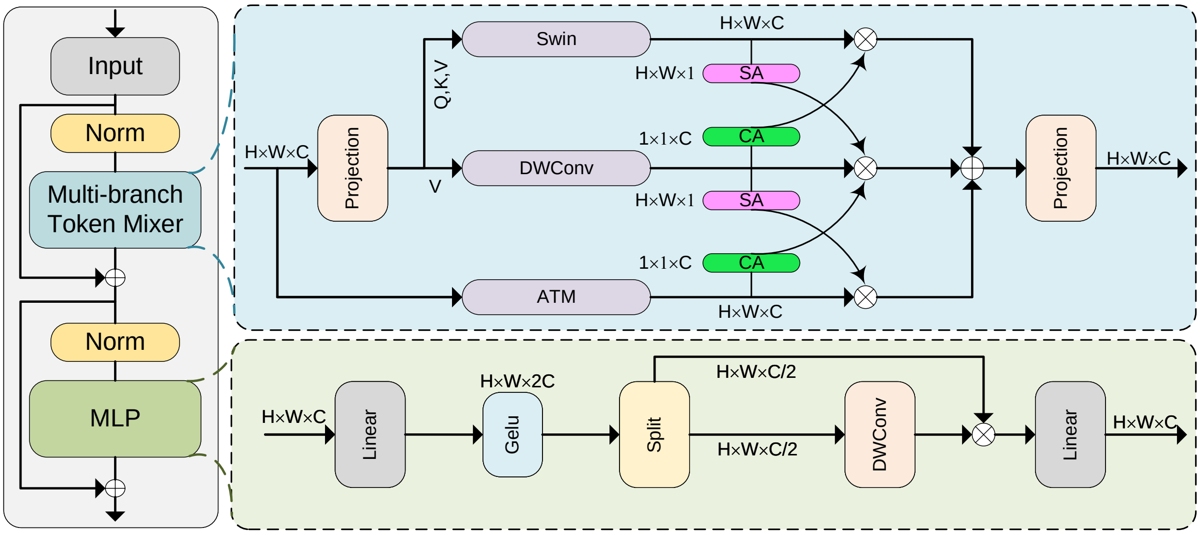


Fig. 3. *The architecture of multi-branch token mixer (MBTM).*

self-attention modules in deeper network may lead to decreased inference efficiency, we have proposed a more efficient token mixing method: the multi-branch token mixer. As shown in Fig. 3, the module consists of three branches, stated as follows:

$$\mathscr{T}(F_{in}) = \mathscr{T}_{SW}(F_{in}) + \mathscr{T}_{DW}(F_{in}) + \mathscr{T}_{ATM}(F_{in}). \quad (7)$$

where, $\mathscr{T}_{SW}(\cdot)$ represents shifted window (Swin) self-attention mixer (Liu *et al.*, 2021), $\mathscr{T}_{DW}(\cdot)$ denotes 3×3 depthwise convolution layer, and $\mathscr{T}_{ATM}(\cdot)$ stands for active token mixer (Wei *et al.*, 2022). The computational formula for the Swin self-attention is given below:

$$\begin{aligned} SW(Q,K,V) &= \mathscr{S}\left(\frac{Q \otimes K^T}{\sqrt{d}} + B\right) \odot M \otimes V, \\ Q,K,V &= \mathscr{P}(F_{in}), \end{aligned} \quad (8)$$

where, Q, K, and V denote the query, key, and value matrices, respectively, with d as the feature dimension. M is the mask matrix, and B is the bias matrix. $\mathscr{P}(\cdot)$ represents the linear projection layer, and $\mathscr{S}(\cdot)$ denotes the softmax function. We also utilized depthwise convolution layer to extract local features:

$$\begin{aligned} DW(V) &= GELU(BN(DWConv_3(V))), \\ V &= \mathscr{P}(F_{in}), \end{aligned} \quad (9)$$

where, $GELU(\cdot)$ denotes Gaussian Error Linear Unit activation function (Hendrycks and Gimpel, 2016), $DWConv_3(\cdot)$ denotes the 3×3 depthwise convolution (Howard *et al.*, 2017b), and $BN(\cdot)$ represents the batch normalization. Additionally, to ensure complete integration of local and global information extracted by different modules, we conduct interaction on features across different branches:

$$\begin{aligned} \mathscr{T}_{SW}(\cdot) &= SW(\cdot) \odot CA(DW(\cdot)), \\ \mathscr{T}_{DW}(\cdot) &= DW(\cdot) \odot SA(SW(\cdot)) \odot CA(ATM(\cdot)), \\ \mathscr{T}_{ATM}(\cdot) &= ATM(\cdot) \odot SA(DW(\cdot)), \end{aligned} \quad (10)$$

where $\odot$ denotes the Hadamard product, $SA(\cdot)$ and $CA(\cdot)$ represent spatial attention and channel attention, respectively. and $ATM(\cdot)$ stands for adaptive token mixer. We will provide detailed explanations of the implementation details of ATM in the Section 3.3.

Multi-layer perceptron In the architecture of metaformer-style models, a feedforward layer is typically appended following the token mixer. The feedforward layer generally employs linear transformations and nonlinear activation functions to enhance the model's capacity for nonlinear learning. Inspired by the DAT network (Chen *et al.*, 2023), we introduce the Spatial-Gate Feed-Forward Network (SGFN) as the feedforward layer. The SGFN adopts the structure of the self-gated activation function (Swish) (Ramachandran *et al.*, 2017) and utilizes depthwise convolution to reduce channel redundancy, thus enhancing the model's inference efficiency. The structure of SGFN is depicted in Fig. 3.

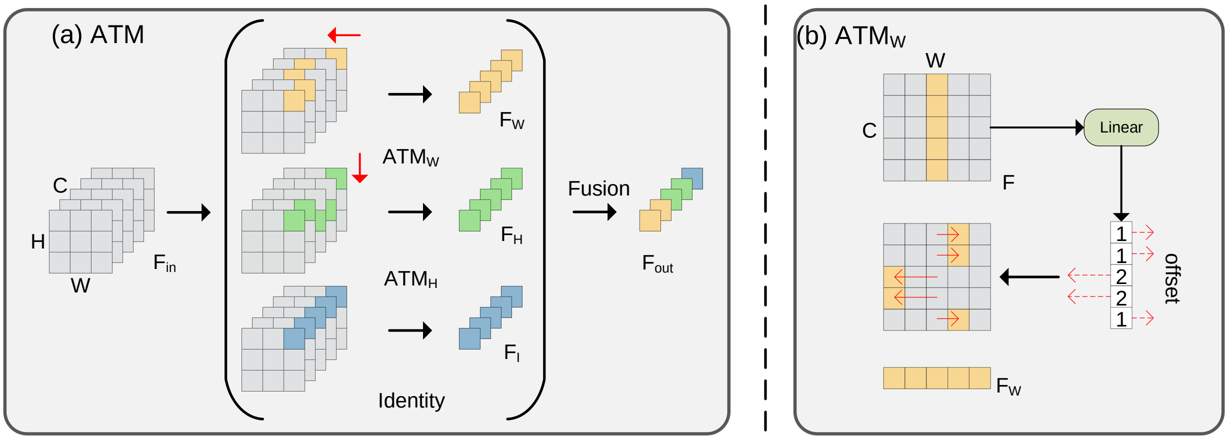


Fig. 4. *(a) The architecture of active token mixer (ATM). (b) Illustration of ATM along the horizontal direction (ATM$_W$).*

ACTIVE TOKEN MIXER

In the token mixer module, we have enriched the feature information and enhanced the overall network performance by broadening the model's width. Given that the token mixing method of self-attention would increase the load on the network, we have adopted a more efficient mixing approach, named active token mixer. The steps for calculating $ATM(\cdot)$ (Wei *et al.*, 2022) are described as follows:

$$\begin{aligned} ATM\left(F_{in}\right) &= \mathscr{P}\left(F_{fusion}\right), \\ F_{fusion} &= \lambda_W \odot F_W + \lambda_H \odot F_H + \lambda_I \odot F_I, \\ F_W, F_H, F_I &= ATM_W\left(F_{in}\right), ATM_H\left(F_{in}\right), \mathscr{P}\left(F_{in}\right), \\ \left[\lambda_W, \lambda_H, \lambda_I\right] &= MLP\left(F_W + F_H + F_I\right). \end{aligned} \tag{11}$$

In the equation above, $ATM_W(\cdot)$ and $ATM_H(\cdot)$ denote the adaptive token mixers along the horizontal and vertical directions, respectively, while $\mathscr{P}$ represents the linear projection layer. F_H, F_W, and F_I represent the recomposed features along the vertical and horizontal directions, as well as the original features. Subsequently, we utilize learned weights $[\lambda_W, \lambda_H, \lambda_I]$ to mix the three sets of features into the fused feature F_{fusion}. Token recomposition, illustrated by $ATM_W(\cdot)$, is performed as follows:

$$\begin{aligned} ATM_W\left(F_{in}^{[i,j,c]}\right) &= F_{in}^{[i,j+o,c]}, \\ ATM_H\left(F_{in}^{[i,j,c]}\right) &= F_{in}^{[i+o,j,c]}, \\ O &= \mathscr{P}\left(F_{in}\right). \end{aligned} \tag{12}$$

Variables i, j, c and o respectively represent the indices of the feature's height, width, channel, and the corresponding offset for the queried feature value. The $O \in \mathbb{R}^{H\times W\times C}$ denotes the query offset matrix.

Complexity Analysis For an input $F_{in} \in \mathbb{R}^{H\times W}$ (assuming the channel dimension C is not considered), the computational complexity of self-attention is calculated as follows:

$$\mathscr{O}(SW) = H\times W\times H\times W = \mathscr{O}(H^2W^2). \tag{13}$$

This results from the pairwise interaction between all spatial locations in the input, leading to a quadratic complexity in both height H and width W.

For the active token mixer (ATM), under the same input, the complexity of generating an offset matrix in the channel dimension is:

$$\mathscr{O}(ATM) = H\times W = \mathscr{O}(HW). \tag{14}$$

This represents the complexity of computing a token mixer that generates offsets based on spatial dimension, which results in a linear complexity with respect to the spatial size $H\times W$.

Compared to self-attention, the computational complexity of ATM is significantly lower. The specific numerical reduction depends on the network structure and implementation details. In practice, the linear complexity of ATM makes it a more efficient choice, especially for large-scale image inputs, where the quadratic complexity of self-attention becomes computationally expensive.

EXPERIMENTS

SETUP

Datasets and Metrics Similar to most supervised models in SISR, we use the DF2K (Agustsson and

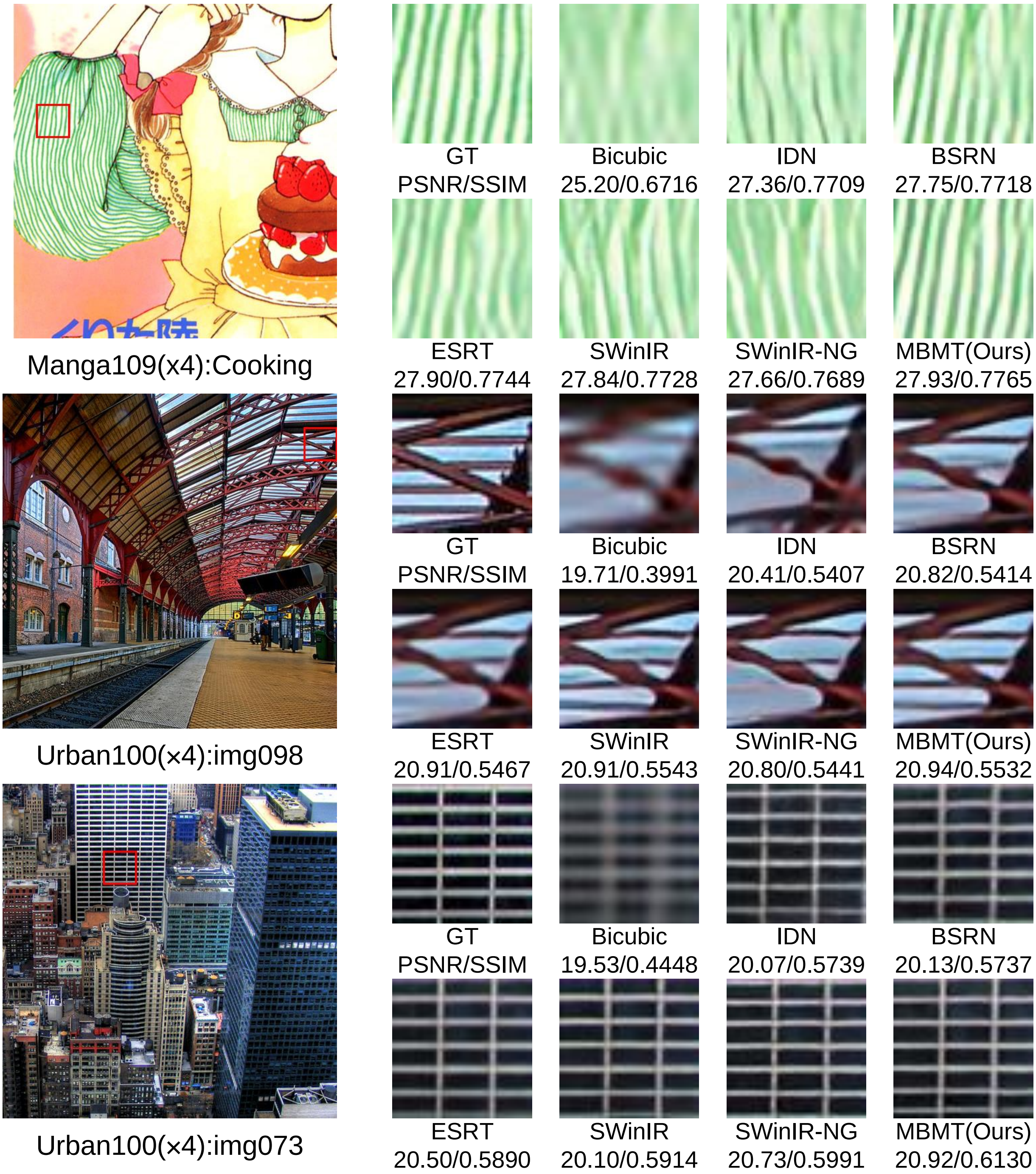


Fig. 5. *Qualitative comparison with state-of-the-art SR($\times 4$) methods on Urban100/Manga109 datasets.*

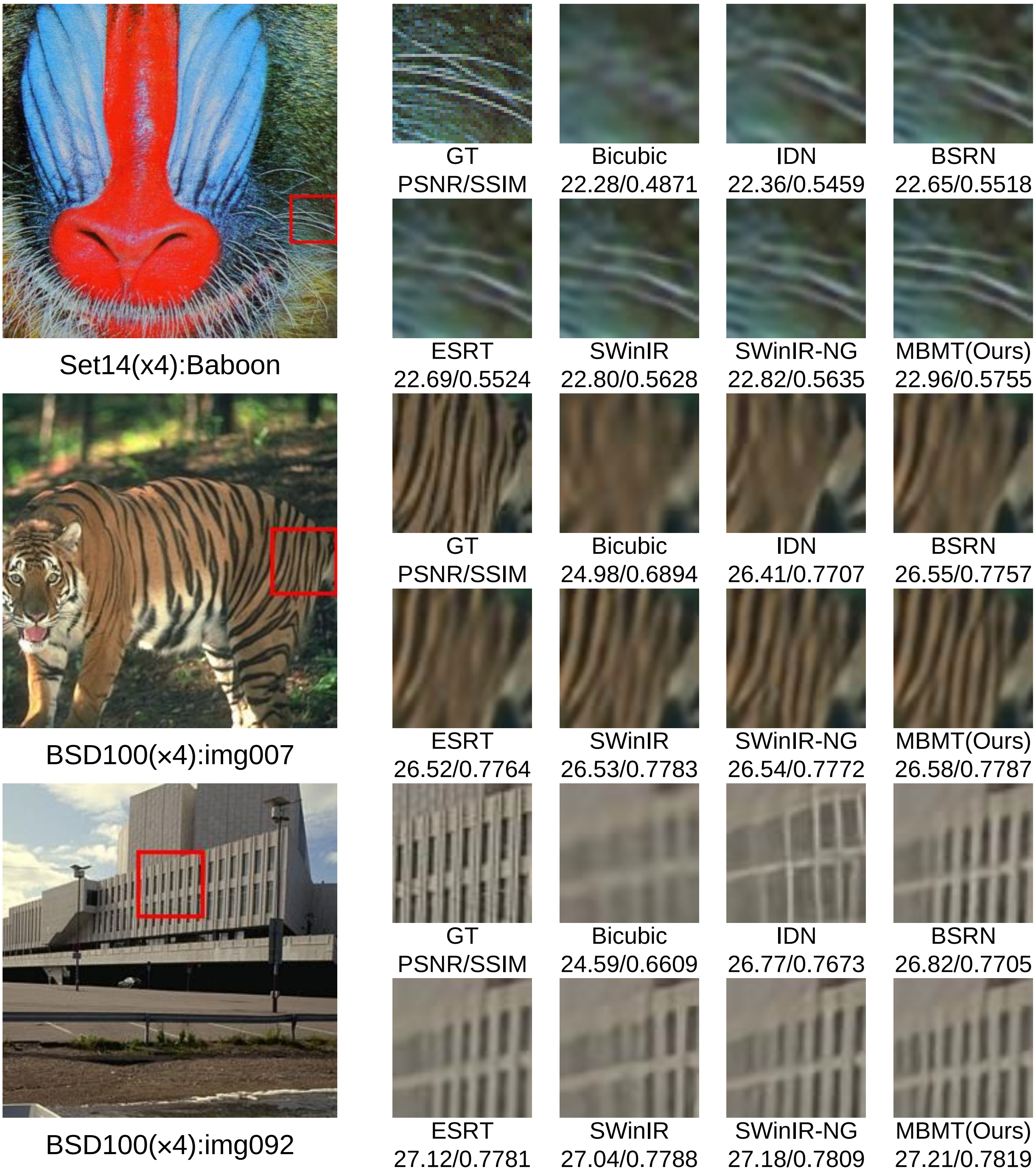
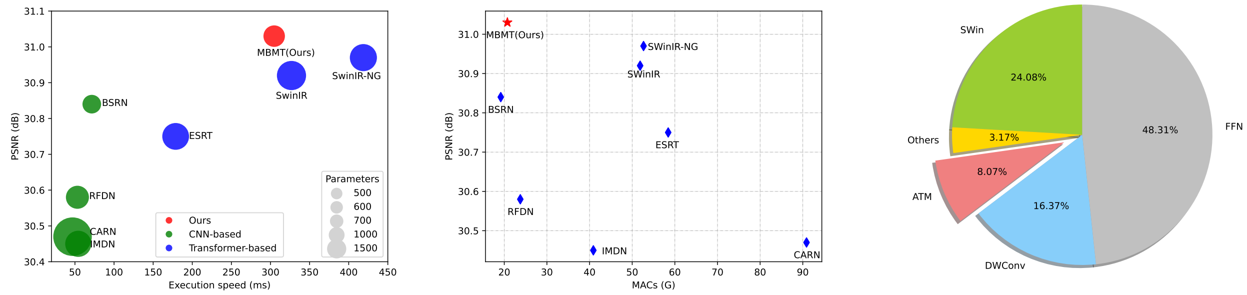


Fig. 6. *Qualitative comparison with state-of-the-art SR($\times$4) methods on Set14/BSD100 datasets.*

Table 1. *Quantitative comparison results of the state-of-the-art methods on public benchmark datasets, with the first and second best results highlighted in* ***Red*** *and* ***Blue*** *respectively. '—' indicates that the item is not included in the original paper.*

Scale	Model	Params[K]	Set5	Set14	BSD100	Urban100	Manga109
			PSNR↑ / SSIM↑	PSNR↑ / SSIM↑	PSNR↑ / SSIM↑	PSNR↑ / SSIM↑	PSNR↑ / SSIM↑
×2	VDSR(Kim *et al.*, 2016a)	666	37.53 / 0.9587	33.03 / 0.9124	31.90 / 0.8960	30.76 / 0.9140	37.22 / 0.9750
	CARN(Ahn *et al.*, 2018)	1592	37.76 / 0.9590	33.52 / 0.9166	32.09 / 0.8978	31.92 / 0.9256	38.36 / 0.9765
	IDN(Hui *et al.*, 2018)	553	37.83 / 0.9600	33.30 / 0.9148	32.08 / 0.8985	31.27 / 0.9196	– / –
	IMDN(Hui *et al.*, 2019)	694	38.00 / 0.9605	33.63 / 0.9177	32.19 / 0.8996	32.17 / 0.9283	38.88 / 0.9774
	RFDN(Liu *et al.*, 2020)	534	38.05 / 0.9606	33.68 / 0.9184	32.16 / 0.8994	32.12 / 0.9278	38.88 / 0.9773
	RLFN(Kong *et al.*, 2022)	527	38.07 / 0.9607	33.72 / 0.9187	32.22 / 0.9000	32.34 / 0.9299	– / –
	BSRN(Li *et al.*, 2022c)	332	38.10 / 0.9610	33.74 / 0.9193	32.24 / 0.9006	32.34 / 0.9303	39.14 / 0.9782
	SWinIR-light(Liang *et al.*, 2021)	878	38.14 / 0.9611	33.86 / 0.9206	32.31 / 0.9012	32.76 / 0.9340	39.12 / 0.9783
	ESRT(Lu *et al.*, 2022)	677	38.03 / 0.9600	33.75 / 0.9184	32.25 / 0.9001	32.58 / 0.9318	39.12 / 0.9774
	ELAN-light(Zhang *et al.*, 2022)	582	38.17 / 0.9611	33.94 / 0.9207	32.30 / 0.9012	32.76 / 0.9340	39.11 / 0.9782
	SWinIR-NG(Choi *et al.*, 2022)	–	– / –	– / –	– / –	– / –	– / –
	SPIN(Zhang *et al.*, 2023)	497	38.20 / 0.9615	33.90 / 0.9215	32.31 / 0.9015	32.79 / 0.9340	39.18 / 0.9784
	MBMT(Ours)	455	38.18 / 0.9617	33.85 / 0.9198	32.31 / 0.9014	32.79 / 0.9343	39.29 / 0.9787
×4	VDSR(Kim *et al.*, 2016a)	666	31.35 / 0.8838	28.01 / 0.7674	27.29 / 0.7251	25.18 / 0.7524	28.83 / 0.8870
	CARN(Ahn *et al.*, 2018)	1592	32.13 / 0.8937	28.60 / 0.7806	27.58 / 0.7349	26.07 / 0.7837	30.47 / 0.9084
	IDN(Hui *et al.*, 2018)	553	31.82 / 0.8903	28.25 / 0.7730	27.41 / 0.7297	25.41 / 0.7632	– / –
	IMDN(Hui *et al.*, 2019)	715	32.21 / 0.8948	28.58 / 0.7811	27.56 / 0.7353	26.04 / 0.7838	30.45 / 0.9075
	RFDN(Liu *et al.*, 2020)	550	32.24 / 0.8952	28.61 / 0.7819	27.57 / 0.7360	26.11 / 0.7858	30.58 / 0.9089
	RLFN(Kong *et al.*, 2022)	543	32.24 / 0.8952	28.62 / 0.7813	27.60 / 0.7364	26.17 / 0.7877	– / –
	BSRN(Li *et al.*, 2022c)	352	32.10 / 0.8966	28.73 / 0.7847	27.65 / 0.7387	26.27 / 0.7908	30.84 / 0.9123
	SWinIR-light(Liang *et al.*, 2021)	897	32.44 / 0.8976	28.77 / 0.7858	27.69 / 0.7406	26.47 / 0.7980	30.92 / 0.9151
	ESRT(Lu *et al.*, 2022)	751	32.19 / 0.8947	28.69 / 0.7833	27.69 / 0.7379	26.39 / 0.7962	30.75 / 0.9100
	ELAN-light(Zhang *et al.*, 2022)	601	32.43 / 0.8975	28.78 / 0.7858	27.69 / 0.7406	26.54 / 0.7982	30.92 / 0.9150
	SWinIR-NG(Choi *et al.*, 2022)	770	32.44 / 0.8978	28.80 / 0.7863	27.70 / 0.7407	26.47 / 0.7977	30.97 / 0.9147
	SPIN(Zhang *et al.*, 2023)	555	32.48 / 0.8983	28.80 / 0.7862	27.70 / 0.7415	26.55 / 0.7998	30.98 / 0.9156
	MBMT(Ours)	475	32.44 / 0.8983	28.80 / 0.7863	27.73 / 0.7416	26.57 / 0.8012	31.03 / 0.9141



(a) (b) (c)

Fig. 7. *Analysis of model performance and complexity. (a) Comparison of running time between different models. (b) Comparison of MACs between different models. (c) Pie chart of FLOPs distribution among different modules in the MBMT. The running times, FLOPs, and MACs are calculated on the Manga109 dataset.*

Timofte, 2017; Lim *et al.*, 2017b) dataset to train our model in the experiments. This dataset adopts bicubic downscaling to obtain corresponding low resolution images, with a total of 3550 images, of which 3450 images for training, 100 images for validation. We use peak signal-to-noise ratio (PSNR) and structure similarity index (SSIM) (Wang *et al.*, 2004) as benchmarks to evaluate the performance of our model and compare it to other models on five standard benchmark datasets: Set5 (Bevilacqua *et al.*, 2012), Set14 (Zeyde *et al.*, 2010), BSD100 (Martin *et al.*, 2001), Urban100 (Huang *et al.*, 2015), Manga109 (Matsui *et al.*, 2015).

Implementation details The proposed MBMT comprises 8 Multi-branch token mixers (MBTMs). Within each MBTM, the number of self-attention heads, channel dimensions, and dimension expansion factor are set to 6, 64, and 2, respectively. In MTMB, the offset generation layer interval is set to 1, and the dimension expansion factor is 2. We randomly crop images to 192 × 192 as the input to the model, using a mini-batch size of 16. To enhance model performance, we apply horizontal flip and random rotations (90°, 180°, 270°) for data augmentation. The dataset is enlarged by a factor of 100 times the original size. During the model training, we set the

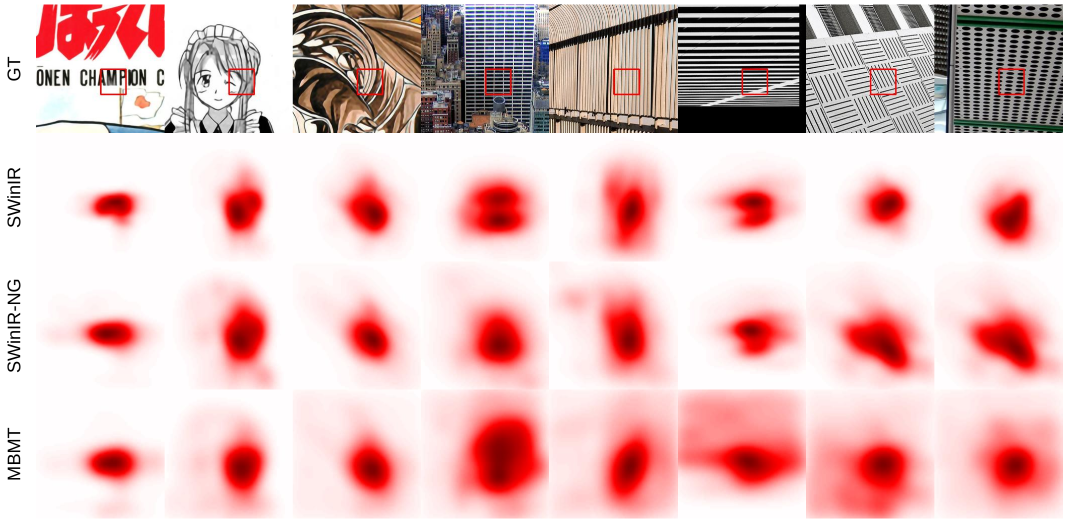


Fig. 8. *Attribution analysis for different transformed-based SR models. Top row: GT images with a highlighted red box. Bottom three rows: Heatmaps showing pixel contributions to the red box in the SR process.*

initial learning rate to 1×10^{-3}, with a total of 2×10^{6} iterations. We employ the Cosine Annealing scheduler to dynamically adjust the learning rate throughout the training stage. Our model is implemented using the PyTorch framework and trained on an NVIDIA GeForce RTX 2070 SUPER GPU.

BENCHMARK RESULTS

As shown in Table 1, we compared MBMT with state-of-the-art lightweight SR approaches. This comparative analysis includes model parameters and the performance of 2× and 4× SR results across five benchmark testsets. The outcomes demonstrate the outstanding performance of MBMT in terms of both model efficiency and multi-scale super-resolution capabilities. Compared to SWinIR-NG (Choi *et al.*, 2022) and SPIN (Zhang *et al.*, 2023), the proposed MBMT achieves average PSNR gains of 0.04 and 0.01 dB on the testsets for 4× SR. Despite transformed-based models generally outperforming CNN-based SR models, they often come with larger model sizes. In contrast, our model achieves impressive SR performance while maintaining a smaller model size compared to some CNN-based counterparts. In Figs. 5 and 6, we provide a visual comparison of our proposed MBMT with other models on the testset for 4× SR results. The figure demonstrates MBMT's superior ability in restoring fine details and textures compared to other methods.

Comparison on Model Complexity We conducted comparative experiments on different models, including comparisons of model parameters, running time, and multiply-accumulate operations (MACs). To ensure fairness in our comparisons, we focused on SR methods that provide pre-trained models. Figure 7a depicts a comparison of parameters and execution time across different SR models on the Manga109 dataset. It is apparent that transformer-based models typically demonstrate slower execution speeds compared to CNN-based SR models. Importantly, among transformer-based models, MBMT exhibits certain advantages. The computational complexity analysis in Fig. 7b indicates that our MBMT model maintains similar levels of MACs to CNN-based SR models. However, our model excels in terms of super-resolution (SR) quality, demonstrating that it achieves superior performance even with limited computational resources. This highlights the effectiveness of our approach in balancing both efficiency and high-quality results. Specifically, the reduced computational complexity of our model makes it particularly suitable for real-time applications where resource constraints are critical, such as in mobile devices, edge computing, or remote sensing tasks. In these scenarios, the model's ability to process high-resolution images with minimal memory usage and faster inference times allows for practical deployment without compromising on performance.

Attribution analysis We applied Local Attribution Maps (LAM) (Gu and Dong, 2021) to conduct attribution analysis on three Transformer-based models: SWinIR, SWinIR-NG, MBMT. As illustrated in Fig. 8, the red regions in the heatmap indicate the network's attention to specific pixel areas during the SR. These pixels are crucial for generating the GT image within the red box. It is evident from the figure that the proposed MBMT exhibits a markedly larger receptive field in comparison to SWinIR and SWinIR-NG. This observation suggests that MBMT is capable of leveraging a wider range of pixels for image reconstruction, consequently leading to enhanced SR performance.

ABLATION STUDY

Table 2. *Ablation study of multi-branch token mixer on model performance. All models were trained from scratch with the same hyperparameters. FLOPs calculated on* 1280×720 *GT images.*

Scale	SWin	DWConv	ATM	Params[K]	FLOPs[G]	Mem[M]	DIV2K100 PSNR/SSIM
×4		✓	✓	465	33	2160.05	30.2472/0.8332
×4	✓		✓	460	36	2886.34	30.5133/0.8401
×4	✓	✓		289	40	2502.36	30.4742/0.8391
×4	✓	✓	✓	475	43	2915.93	30.5320/0.8404

Table 3. *Ablation study of active token mixer on model performance. All models were trained from scratch with the same hyperparameters. MACs calculated on* 1280×720 *GT images.*

Scale	ATM_H	ATM_W	Identity	Params[K]	MACs[G]	DIV2K100 PSNR/SSIM
×4		✓	✓	438	20.73	30.5246/0.8401
×4	✓		✓	438	20.73	30.5175/0.8398
×4	✓	✓		438	19.07	30.5253/0.8401
×4	✓	✓	✓	475	20.73	30.5320/0.8404

Effectiveness of multi-branch token mixer To analyze the effectiveness of the multi-branch token mixer structure, we performed separate ablation experiments on each individual branch and validated the models using the high-resolution dataset DIV2K100. Table 2 demonstrates that the SWin+DWConv+ATM multi-branch structure attained the highest SR performance. Among the three branches, SWin has the most significant impact on SR, followed by ATM, with DWConv making the smallest contribution. Despite SWin's smaller parameters, its quadratic computational complexity due to the self-attention mechanism leads to higher FLOPs, consequently impacting the model's efficiency. While ATM has a larger parameter count, ATM's computational load mainly focuses on offset calculations, resulting in lower FLOPs. As depicted in Fig. 7c, SWin, ATM, and DWConv account for 24.08%, 8.07%, and 16.37% of the total FLOPs in the model, respectively. Considering their impact on SR performance, it indicates the importance of the ATM branch in achieving lightweight SR.

Effectiveness of active token mixer Table 3 shows the results of ablative experiments conducted on the ATM to assess the influence of identity features, as well as adaptive token mixers along both horizontal and vertical dimensions, on the performance of ×4 SR. Overall, the three factors have roughly equal influence. Although the parameters and model complexity of adaptive token mixers along the horizontal and vertical dimensions are identical, the influence in the horizontal dimension on SR performance is slightly greater than in the vertical dimension. In comparison, identity features have a slight advantage in model size, model complexity, and SR performance. Combining the conclusions from Table 2, we can infer that the ATM significantly contributes to the model's lightweight structure.

CONCLUSION

In this paper, we propose a lightweight SR model, MBMT, which achieves the goal of simultaneously extracting global and local features through a multi-branch structure design. This innovative design significantly reduces network depth while slightly increasing network width, paving the way for a lightweight super-resolution Transformer. Extensive experimental results demonstrate that MBMT achieves SOTA performance in both super-resolution quality and model complexity across benchmark datasets. However, considering that the self-attention branch still has a greater impact on model performance compared to other branches, designing a more efficient token mixer remains a significant challenge. As a potential future direction, replacing the active token mixer with a more efficient token mixer, such as the State Space Model (SSM) (Gu and Dao, 2024), could further reduce complexity (with linear complexity) while maintaining performance.

REFERENCES

Agustsson E, Timofte R (2017). Ntire 2017 challenge on single image super-resolution: Dataset and study. In: Proc CVPR IEEE.

Ahn N, Kang B, Sohn KA (2018). Fast, accurate, and lightweight super-resolution with cascading residual network. In: Lect Notes Comput Sc.

Bevilacqua M, Roumy A, Guillemot C, Morel A (2012). Low-complexity single image super-resolution based on nonnegative neighbor embedding. In: BMVC.

Cao J, Liang J, Zhang K, Li Y, Zhang Y, Wang W, Van Gool L (2022). Reference-based image super-resolution with deformable attention transformer. In: Lect Notes Comput Sc.

Chen Z, Zhang Y, Gu J, Kong L, Yang X, Yu F (2023). Dual aggregation transformer for image super-resolution. In: IEEE I Conf Comp Vis.

Choi H, Lee JS, Yang J (2022). N-gram in swin transformers for efficient lightweight image super-resolution. Proc CVPR IEEE :2071–81.

Dong C, Loy CC, He K, Tang X (2016a). Image super-resolution using deep convolutional networks. IEEE T Pattern Anal 38:295–307.

Dong C, Loy CC, Tang X (2016b). Accelerating the super-resolution convolutional neural network. In: Lect Notes Comput Sc.

Dosovitskiy A, Beyer L, Houlsby N (2021). An image is worth 16x16 words: Transformers for image recognition at scale. ICLR .

Gao G, Wang Z, Li J, Li W, Yu Y, Zeng T (2022). Lightweight bimodal network for single-image super-resolution via symmetric cnn and recursive transformer. In: Int Joint Conf Artif.

Gu A, Dao T (2024). Mamba: Linear-time sequence modeling with selective state spaces. In: COLM.

Gu J, Dong C (2021). Interpreting super-resolution networks with local attribution maps. In: Proc CVPR IEEE.

He K, Zhang X, Ren S, Sun J (2016). Deep residual learning for image recognition. In: Proc CVPR IEEE.

Hendrycks D, Gimpel K (2016). Gaussian error linear units (gelus). arXiv Learning .

Howard AG, Zhu M, Adam H (2017a). Mobilenets: Efficient convolutional neural networks for mobile vision applications. ArXiv abs/1704.04861.

Howard AG, Zhu M, Adam H (2017b). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv170404861 .

Huang JB, Singh A, Ahuja N (2015). Single image super-resolution from transformed self-exemplars. In: Proc CVPR IEEE.

Huang Z, Zhang Z, Lan C, Zha ZJ, Lu Y, Guo B (2023). Adaptive frequency filters as efficient global token mixers. In: IEEE I Conf Comp Vis.

Hui Z, Gao X, Yang Y, Wang X (2019). Lightweight image super-resolution with information multi-distillation network. In: ACM MM.

Hui Z, Wang X, Gao X (2018). Fast and accurate single image super-resolution via information distillation network. In: Proc CVPR IEEE.

Kim J, Lee JK, Lee KM (2016a). Accurate image super-resolution using very deep convolutional networks. In: Proc CVPR IEEE.

Kim J, Lee JK, Lee KM (2016b). Deeply-recursive convolutional network for image super-resolution. In: Proc CVPR IEEE.

Kong F, Li M, Liu S, Liu D, He J, Bai Y, Chen F, Fu L (2022). Residual local feature network for efficient super-resolution. In: Proc CVPR IEEE.

Lai WS, Huang JB, Ahuja N, Yang MH (2017). Deep laplacian pyramid networks for fast and accurate super-resolution. In: Proc CVPR IEEE.

Ledig C, Theis L, Huszar F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z (2016). Photo-realistic single image super-resolution using a generative adversarial network. IEEE Comp Soc .

Li H, Cai D, Xu J, Watanabe T (2022a). Residual learning of neural text generation with n-gram language model. In: ACL.

Li Y, Zhang K, Timofte R, Van Gool L, *et al.* (2022b). Ntire 2022 challenge on efficient super-resolution: Methods and results. In: Proc CVPR IEEE.

Li Z, Liu Y, Chen X, Cai H, Gu J, Qiao Y, Dong C (2022c). Blueprint separable residual network for efficient image super-resolution. In: Proc CVPR IEEE.

Liang J, Cao J, Sun G, Zhang K, Van Gool L, Timofte R (2021). Swinir: Image restoration using swin transformer. In: IEEE I Conf Comp Vis.

Lim B, Son S, Kim H, Nah S, Lee KM (2017a). Enhanced deep residual networks for single image super-resolution. In: Proc CVPR IEEE.

Lim B, Son S, Kim H, Nah S, Lee KM (2017b). Enhanced deep residual networks for single image super-resolution. In: Proc CVPR IEEE.

Liu J, Tang J, Wu G (2020). Residual feature distillation network for lightweight image super-resolution. In: Lect Notes Comput Sc.

Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In: IEEE I Conf Comp Vis.

Lu Z, Li J, Liu H, Huang C, Zhang L, Zeng T (2022). Transformer for single image super-resolution. In: Proc CVPR IEEE.

Martin D, Fowlkes C, Tal D, Malik J (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: IEEE I Conf Comp Vis, vol. 2.

Matsui Y, Ito K, Aramaki Y, Fujimoto A, Ogawa T, Yamasaki T, Aizawa K (2015). Sketch-based manga retrieval using manga109 dataset. Multimed Tools Appl 76:21811–38.

Ramachandran P, Zoph B, Le QV (2017). Swish: a self-gated activation function. arXiv Neural and Evolutionary Computing .

Shi W, Caballero J, Huszár F, Totz J, Aitken AP, Bishop R, Rueckert D, Wang Z (2016). Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proc CVPR IEEE.

Tai Y, Yang J, Liu X (2017a). Image super-resolution via deep recursive residual network. In: Proc CVPR IEEE.

Tai Y, Yang J, Liu X, Xu C (2017b). Memnet: A persistent memory network for image restoration. In: IEEE I Conf Comp Vis.

Tolstikhin IO, Houlsby Neiland Lucic M, Dosovitskiy A (2021). Mlp-mixer: An all-mlp architecture for vision. In: Adv Neur In, vol. 34.

Touvron H, Bojanowski P, Verbeek J, *et al.* (2022). Resmlp: Feedforward networks for image classification with data-efficient training. IEEE T Pattern Anal 45:5314–21.

Vaswani A, Shazeer NM, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017). Attention is all you need. In: Neu Inf Pro.

Wang H, Chen X, Ni B, Liu Y, Liu J (2023). Omni aggregation networks for lightweight image super-resolution. Proc CVPR IEEE :22378–87.

Wang X, Yu K, Wu S, Gu J, Liu Y, Dong C, Qiao Y, Loy CC (2018). Esrgan: Enhanced super-resolution generative adversarial networks. In: Lect Notes Comput Sc.

Wang Y, Liu Y, Zhao S, Li J, Zhang L (2024). Camixersr: Only details need more “attention”. In: Proc CVPR IEEE.

Wang Z, Bovik A, Sheikh H, Simoncelli E (2004). Image quality assessment: from error visibility to structural similarity. IEEE T Image Process 13:600–12.

Wei G, Zhang Z, Lan C, Lu Y, Chen Z (2022). Activemlp: An mlp-like architecture with active token mixer. In: AAAI.

Yang F, Yang H, Fu J, Lu H, Guo B (2020). Learning texture transformer network for image super-resolution. In: Proc CVPR IEEE.

Yu W, Luo M, Zhou P, Si C, Zhou Y, Wang X, Feng J, Yan S (2022). Metaformer is actually what you need for vision. In: Proc CVPR IEEE.

Zeyde R, Elad M, Protter M (2010). On single image scale-up using sparse-representations. In: ICCS.

Zhang A, Ren W, Liu Y, Cao X (2023). Lightweight image super-resolution with superpixel token interaction. In: IEEE I Conf Comp Vis.

Zhang X, Zeng H, Guo S, Zhang L (2022). Efficient long-range attention network for image super-resolution. In: Lect Notes Comput Sc.

Zhang Y, Tian Y, Kong Y, Zhong B, Fu Y (2018). Residual dense network for image super-resolution. In: Proc CVPR IEEE.

Zhao H, Gallo O, Frosio I, Kautz J (2017). Loss functions for image restoration with neural networks. TCI 3:47–57.

Zhou Y, Li Z, Guo CL, Bai S, Cheng MM, Hou Q (2023). Srformer: Permuted self-attention for single image super-resolution. In: IEEE I Conf Comp Vis.