SRFormer: Permuted Self-Attention for Single Image Super-Resolution

Yupeng Zhou[1] Zhen Li[1] Chun-Le Guo[1] Song Bai[2] Ming-Ming Cheng[1] Qibin Hou[1*]

[1]VCIP, School of Computer Science, Nankai University
[2]ByteDance, Singapore
{ypzhousdu, andrewhoux}@gmail.com

Abstract

Previous works have shown that increasing the window size for Transformer-based image super-resolution models (e.g., SwinIR) can significantly improve the model performance but the computation overhead is also considerable. In this paper, we present SRFormer, a simple but novel method that can enjoy the benefit of large window self-attention but introduces even less computational burden. The core of our SRFormer is the permuted self-attention (PSA), which strikes an appropriate balance between the channel and spatial information for self-attention. Our PSA is simple and can be easily applied to existing super-resolution networks based on window self-attention. Without any bells and whistles, we show that our SRFormer achieves a 33.86dB PSNR score on the Urban100 dataset, which is 0.46dB higher than that of SwinIR but uses fewer parameters and computations. We hope our simple and effective approach can serve as a useful tool for future research in super-resolution model design. Our code is available at https://github.com/HVision-NKU/SRFormer.

1. Introduction

Single image super-resolution (SR) endeavors to recover a high-quality image from its degraded low-resolution counterpart. The pursuit of efficient and proficient super-resolution algorithms has been a hot research topic in computer vision, which has a variety of applications [2, 25, 63]. Since the pioneer works [9, 26, 30, 38, 52, 82], CNN-based methods have been mainstream for image super-resolution for a long time. These methods mostly take advantage of residual learning [26, 30, 32, 38, 55, 80], dense connections [58, 67, 87], or channel attention [72, 86] to construct network architectures, making substantial contributions to the advancement of super-resolution models.

*Corresponding author.

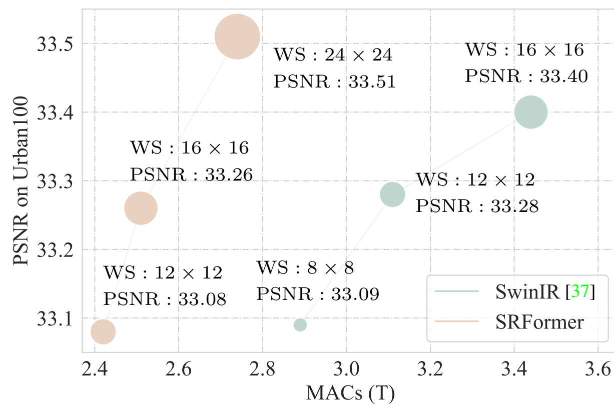

Figure 1. Performance comparison between SwinIR and our SRFormer when training 200k iterations with different window sizes (WS) for 200k iterations. Our SRFormer enjoys a large window size of 24 × 24 with even fewer computations but higher PSNR scores.

Despite the success made by CNN-based models in super-resolution, recent works [5, 37, 79, 85] have shown that Transformer-based models perform better. They observe that the ability of self-attention to build pairwise relationships is a more efficient way to produce high-quality super-resolution images than convolutions. One typical work among them should be SwinIR [37] which introduces Swin Transformer [41] to image super-resolution, greatly improving the state-of-the-art CNN-based models on various benchmarks. Later, a variety of works, such as SwinFIR [79], ELAN [85], and HAT [6], further develop SwinIR and use Transformers to design different network architectures for SR.

The aforementioned methods reveal that properly enlarging the windows for the shifted window self-attention in SwinIR can result in clear performance gain (see Figure 1). However, the computational burden is also an important issue as the window size goes larger. In addition, Transformer-based methods utilize self-attention and require networks of larger channel numbers compared to previous CNN-based methods [26, 86, 87]. To explore efficient and effective super-resolution algorithms, a straightforward question should be: How would the performance go if we reduce the channel number and meanwhile increase the window size?

Motivated by the question mentioned above, in this paper, we present permuted self-attention (PSA), an efficient way to build pairwise relationships within large windows (e.g., 24×24). The intention is to enable more pixels to get involved in the attention map computation and at the same time introduce no extra computational burden. To this end, we propose to shrink the channel dimensions of the key and value matrices and adopt a permutation operation to convey part of the spatial information into the channel dimension. In this way, despite the channel reduction, there is no loss of spatial information, and each attention head is also allowed to keep a proper number of channels to produce expressive attention maps [60]. In addition, we also improve the original feed-forward network (FFN) by adding a depth-wise convolution between the two linear layers, which we found helps in high-frequency component recovery.

Given the proposed PSA, we construct a new network for SR, termed SRFormer. We evaluate our SRFormer on five widely-used datasets. Benefiting from the proposed PSA, our SRFormer can clearly improve its performance on almost all five datasets. Notably, for $\times 2$ SR, our SRFormer trained on only the DIV2K dataset [38] achieves a 33.86 PSNR score on the challenging Urban100 dataset [20]. This result is much higher than those of the recent SwinIR (33.40) and ELAN (33.44). A similar phenomenon can also be observed when evaluating on the $\times 3$ and $\times 4$ SR tasks. In addition, we perform experiments using a light version of our SRFormer. Compared to previous lightweight SR models, our method also achieves better performance on all benchmarks.

To sum up, our contributions can be summarized as follows:

- We propose a novel permuted self-attention for image super-resolution, which can enjoy large-window self-attention by transferring spatial information into the channel dimension. By leveraging it, we are the first to implement 24x24 large window attention mechanism at an acceptable time complexity in SR.
- We build a new transformer-based super-resolution network, dubbed SRFormer, based on the proposed PSA and an improved FFN from the frequency perspective (ConvFFN). Our SRFormer achieves state-of-the-art performance in classical, lightweight, and real-world image SR tasks.

2. Related Work

In this section, we briefly review the literature on image super-resolution. We first describe CNN-based methods and then transition to the recent popular Transformer-based models.

2.1. CNN-Based Image Super-Resolution

Since SRCNN [9] first introduced CNN into image super-resolution (SR), a large number of CNN-based SR models have emerged. DRCN [27] and DRRN [55] introduce recursive convolutional networks to increase the depth of the network without increasing the parameters. Some early CNN-based methods [9, 27, 55, 56] attempt to interpolate the low-resolution (LR) as input, which results in a computationally expensive feature extraction. To accelerate the SR inference process, FSRCNN [10] extracts features at the LR scale and conducts an upsampling operation at the end of the network. This pipeline with pixel shuffle upsampling [52] has been widely used in later works [37, 85, 86]. LapSRN [29] and DBPN [19] perform upsampling during extracting feature to learn the correlation between LR and HR. There are also some works [30, 65, 67, 83] that use GAN [14] to generate realistic textures in reconstruction. MemNet [56], RDN [87], and HAN [49] efficiently aggregate the intermediate features to enhance the quality of the reconstructed images. Non-Local attention [64] has also been extensively explored in SR to better model the long-range dependencies. Methods of this type include CS-NL [48], NLSA [47], SAN [7], IGNN [89], etc.

2.2. Vision Transformers

Transformers recently have shown great potential in a variety of vision tasks, including image classification [11, 59, 62, 74, 75], object detection [4, 12, 16, 54, 73], image caption [33, 69, 84], semantic segmentation [53, 71, 88], image restoration [5, 17, 37, 76], etc. Among these, the most typical work should be Vision Transformer (ViT) [11] which proves Transformers can perform better than convolutional neural networks on feature encoding. The application of Transformers in low-level vision mainly includes two categories: generation [8, 24, 31, 78] and restoration. Further, the restoration tasks can also be divided into two categories: video restoration [13, 39, 40, 42, 51] and image restoration [5, 18, 68, 70, 76].

As an important task of image restoration, image super-resolution needs to preserve the structural information of the input, which poses a great challenge when devising Transformer-based models. IPT [5] is a large pre-trained model based on the Transformer encoder and decoder structure and has been applied to super-resolution, denoising, and deraining. Based on the Swin Transformer encoder [41], SwinIR [37] performs self-attention on an 8×8 local window in feature extraction and achieves extremely powerful performance. ELAN [85] simplifies the architecture of SwinIR and uses self-attention computed in different window sizes to collect the correlations between long-range

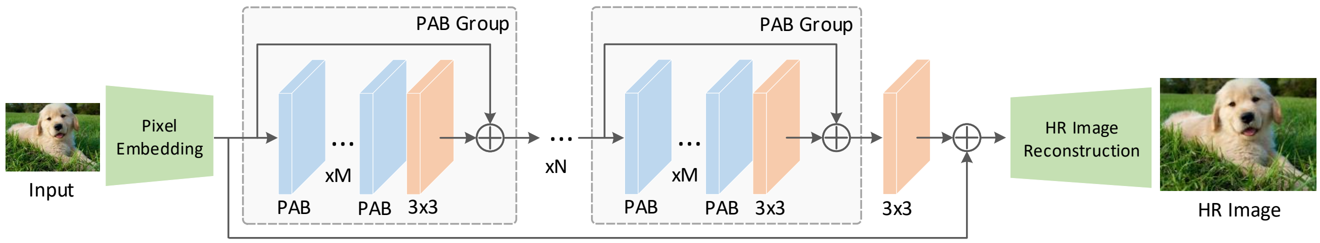

Figure 2. Overall architecture of SRFormer. The pixel embedding module is a 3×3 convolution to map the input image to feature space. The HR image reconstruction module contains a 3×3 convolution and a pixel shuffle operation to reconstruct the high-resolution image. The middle feature encoding part has N PAB groups, followed by a 3×3 convolution.

pixels.

Our SRFormer is also based on Transformer. Different from the aforementioned methods that directly leverage self-attention to build models, our SRFormer mainly aims at the self-attention itself. Our intention is to study how to compute self-attention in a large window to improve the performance of SR models without increasing the parameters and computational cost.

3. Method

3.1. Overall Architecture

The overall architecture of our SRFormer is shown in Fig. 2, consisting of three parts: a pixel embedding layer G_P, a feature encoder G_E, and a high-resolution image reconstruction layer G_R. Following previous works [37, 85], the pixel embedding layer G_P is a single 3×3 convolution that transforms the low-resolution RGB image $I \in \mathbb{R}^{H \times W \times 3}$ to feature embeddings $F_P \in \mathbb{R}^{H \times W \times C}$. F_P will then be sent into the feature encoder G_E with a hierarchical structure. It consists of N permuted self-attention groups, each of which is with M permuted self-attention blocks followed by a 3×3 convolution. A 3×3 convolution is added at the end of the feature encoder, yielding F_E. The summation results of F_E and F_P are fed into G_R for high-resolution image reconstruction, which contains a 3×3 convolution and a sub-pixel convolutional layer [52] to reconstruct high-resolution images. We compute the L1 loss between the high-resolution reconstructed image and ground-truth HR image to optimize our SRFormer.

3.2. Permuted Self-Attention Block

The core of our SRFormer is the permuted self-attention block (PAB), which consists of a permuted self-attention (PSA) layer and a convolutional feed-forward network (ConvFFN).

Permuted self-attention. As shown in Fig. 3(b), given an input feature map $\mathbf{X}_{in} \in \mathbb{R}^{H \times W \times C}$ and a tokens reduction factor r, we first split $\mathbf{X}_{in}$ into N non-overlapping square windows $\mathbf{X} \in \mathbb{R}^{NS^2 \times C}$, where S is the side length of each window. Then, we use three linear layers L_Q, L_K, L_V to get $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$:

$$\mathbf{Q}, \mathbf{K}, \mathbf{V} = L_Q(\mathbf{X}), L_K(\mathbf{X}), L_V(\mathbf{X}) \tag{1}$$

Here, $\mathbf{Q}$ keeps the same channel dimension to $\mathbf{X}$ while L_K and L_V compress the channel dimension to C/r^2, yielding $\mathbf{K} \in \mathbb{R}^{NS^2 \times C/r^2}$ and $\mathbf{V} \in \mathbb{R}^{NS^2 \times C/r^2}$. After that, to enable more tokens to get involved in the self-attention calculation and avoid the increase of the computational cost, we propose to permute the spatial tokens in $\mathbf{K}$ and $\mathbf{V}$ to the channel dimension, attaining permuted tokens $\mathbf{K}_p \in \mathbb{R}^{NS^2/r^2 \times C}$ and $\mathbf{V}_p \in \mathbb{R}^{NS^2/r^2 \times C}$.

We use $\mathbf{Q}$ and the shrunken $\mathbf{K}_p$ and $\mathbf{V}_p$ to perform the self-attention operation. In this way, the window size for $\mathbf{K}_p$ and $\mathbf{V}_p$ will be reduced to $\frac{S}{r} \times \frac{S}{r}$ but their channel dimension is still unchanged to guarantee the expressiveness of the attention map generated by each attention head [60]. The formulation of the proposed PSA can be written as follows:

$$\mathrm{PSA}(\mathbf{Q}, \mathbf{K}_p, \mathbf{V}_p) = \mathrm{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}_p^T}{\sqrt{d_k}} + \mathbf{B}\right)\mathbf{V}_p \tag{2}$$

where $\mathbf{B}$ is an aligned relative position embedding that can be attained by interpolating the original one defined in [41] since the window size of $\mathbf{Q}$ does not match that of $\mathbf{K}_p$. $\sqrt{d_k}$ is a scalar as defined in [11]. Note that the above equation can easily be converted to the multi-head version by splitting the channels into multiple groups.

Our PSA transfers the spatial information to the channel dimension. It ensures the following two key design principles: i) We do not downsample the tokens first as done in [62, 71] but allow each token to participate in the self-attention computation independently. This enables more representative attention maps. ii) PSA can be conducted in a large window (e.g., 24×24) using even fewer computations than SwinIR with 8×8 window while attaining better performance. For a $h \times w \times c$ feature, the original WSA [41] divides it into $\frac{h}{W} \times \frac{w}{W}$ windows. Each window involves three linear projections for K, Q, V, and a linear projection after attention, resulting in $\Omega(4W^2C^2)$. Additionally, the attention calculation requires $\Omega(2W^4C)$:

$$\Omega(\mathrm{WSA}) = 4hwC^2 + 2W^2hwC$$

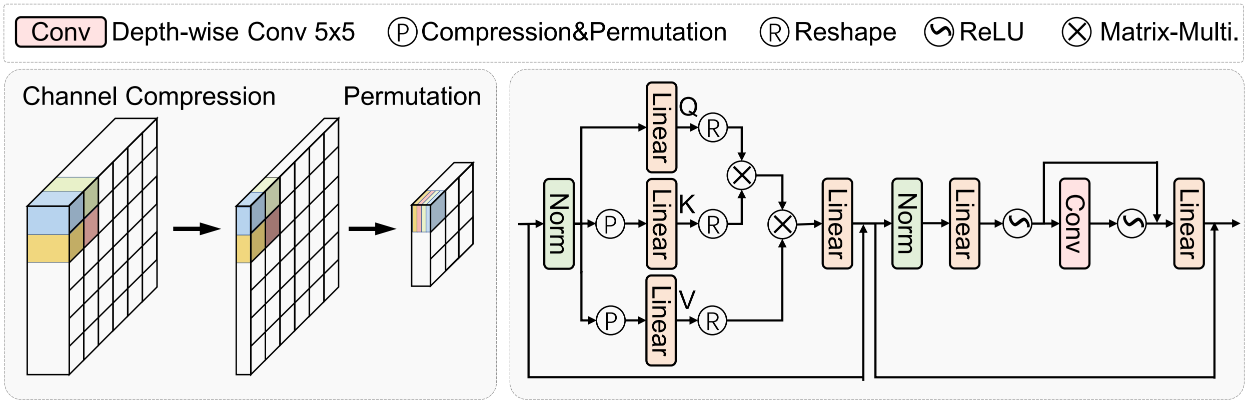

Figure 3. (a) We propose to reduce the channel numbers and transfer the spatial information to the channel dimension to avoid spatial information loss. (b) The structure of our Permuted Self-Attention Block(PAB).

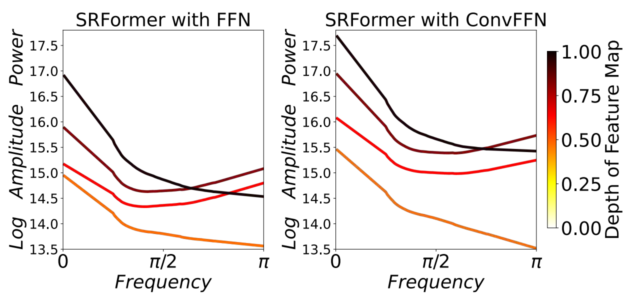

Figure 4. Power spectrum of the intermediate feature maps produced by our SRFormer with FFN and ConvFFN. Lines in darker color correspond to features from deeper layers.

In contrast, our PSA reduces the computations of linear projection of K, V and attention calculation to $1/r^2$:

$$\Omega(\mathrm{PSA}) = 2hw\frac{C^2}{r^2} + 2hwC^2 + 2W^2\frac{h}{r}\frac{w}{r}C$$

ConvFFN. Previous works have demonstrated that self-attention can be viewed as a low-pass filter [50, 61]. To better restore high-frequency information, a 3×3 convolution is often added at the end of each group of Transformers as done in SwinIR [37]. Different from SwinIR, in our PAB, we propose to add a local depthwise convolution branch between the two linear layers of the FFN block to assist in encoding more details. We name the new block as ConvFFN. We empirically found that such an operation increases nearly no computations but can compensate for the loss of high-frequency information caused by self-attention shown in Fig. 4. We simply calculate the power spectrum of the feature maps produced by our SRFormer with FFN and ConvFFN. By comparing the two figures, we can see that ConvFFN can clearly increase high-frequency information, and hence yields better results as listed in Tab. 1.

3.3. Large-Window Self-Attention Variants

To provide guidance for the design of large-window self-attention and demonstrate the advantage of our PSA, here, we introduce another two large-window self-attention variants. The quantitative comparisons and analysis can be found in our experiment section.

Token Reduction. The first way to introduce large-window self-attention and avoid the increase in computational cost is to reduce the number of tokens as done in [71]. Let r and S be a reduction factor and the window size. Given an input $\mathbf{X} \in \mathbb{R}^{NS^2 \times C}$, we can adopt a depthwise convolutional function with kernel size $r \times r$ and stride r to reduce the token numbers of $\mathbf{K}$ and $\mathbf{V}$ in each window to $(\frac{S}{r})^2$, yielding $\mathbf{Q}_r \in \mathbb{R}^{NS^2 \times C}$ and $\mathbf{K}_r, \mathbf{V}_r \in \mathbb{R}^{NS^2/r^2 \times C}$. $\mathbf{Q}_r$ and $\mathbf{K}_r$ are used to compute the attention scores $\mathbf{A} \in \mathbb{R}^{S^2 \times S^2/r^2}$. Computing the matrix multiplication between $\mathbf{A}$ and $\mathbf{V}_r$ yields the output with the same number of tokens to $\mathbf{X}$.

Token Sampling. The second way to achieve large-window self-attention is to randomly sample T^2 $(0 \leq T \leq S)$ tokens from each window according to a given sampling ratio t for the key $\mathbf{K}$ and value $\mathbf{V}$. Given the input $\mathbf{X} \in \mathbb{R}^{NS^2 \times C}$, $\mathbf{Q}$ shares the same shape with $\mathbf{X}$ but the shapes of $\mathbf{K}$ and $\mathbf{V}$ are reduced to $NT^2 \times C$. In this way, as long as T is fixed, the computational cost increases linearly as the window size gets larger. A drawback of token sampling is that randomly selecting a portion of tokens loses structural information of content, which is essentially needed for image SR.

4. Experiments

In this section, we conduct experiments on both the classical, lightweight, and real-world image SR tasks, compare our SRFormer with existing state-of-the-art methods, and do ablation analysis of the proposed method.

Method	Window size	Params	MACs	SET5 [3]		SET14 [77]		B100 [45]		Urban100 [20]		Manga109 [46]	
				PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
SwinIR [37]	8×8	11.75M	2868G	38.24	0.9615	33.94	0.9212	32.39	0.9023	33.09	0.9373	39.34	0.9784
	12×12	11.82M	3107G	38.30	0.9617	34.04	0.9220	32.42	0.9026	33.28	0.9381	39.44	0.9788
	16×16	11.91M	3441G	38.32	0.9618	34.00	0.9212	32.44	0.9030	33.40	0.9394	39.53	0.9791
SRFormer w/o **ConvFFN**	12×12	9.97M	2381G	38.23	0.9615	34.00	0.9216	32.37	0.9023	32.99	0.9367	39.30	0.9786
	16×16	9.99M	2465G	38.25	0.9616	33.98	0.9209	32.38	0.9022	33.09	0.9371	39.42	0.9789
	24×24	10.06M	2703G	38.30	0.9618	34.08	0.9225	32.43	0.9030	33.38	0.9397	39.44	0.9786
SRFormer	12×12	10.31M	2419G	38.22	0.9614	34.08	0.9220	32.38	0.9025	33.08	0.9372	39.13	0.9780
	16×16	10.33M	2502G	38.31	0.9617	34.10	0.9217	32.43	0.9026	33.26	0.9385	39.36	0.9785
	24×24	10.40M	2741G	38.33	0.9618	34.13	0.9228	32.44	0.9030	33.51	0.9405	39.49	0.9788

Table 1. Ablation study on the window size. We report results on the original SwinIR, SRFormer without ConvFFN, and our full SRFormer. Note that the parameters and MACs of SRFormer with 24×24 window are fewer than SwinIR with 8×8 window. Larger windows can result in better performance.

ConvFFN	Urban100 [20]		Manga109 [46]	
	PSNR	SSIM	PSNR	SSIM
w/o Depth-wise Conv	33.38	0.9397	39.44	0.9786
3×3 Depth-wise Conv	33.42	0.9398	39.34	0.9787
5×5 Depth-wise Conv	**33.51**	**0.9405**	**39.49**	**0.9788**

Table 2. Ablation study on ConvFFN for $\times 2$ SR. From the results on Urban100 and Manga109, we can see that using 5×5 depthwise convolution yields the best results. This indicates that local details are also essential for Transformer-based models.

Method	Params	MACs	S	r	PSNR	SSIM
SwinIR [37]	11.75M	2868G	8	-	33.09	0.9373
Token Reduction	11,78M	2471G	16	2	33.09	0.9372
Token Reduction	11.85M	2709G	24	2	33.24	0.9387
Token Sampling	11.91M	2465G	16	2	32.38	0.9312
Token Sampling	12.18M	2703G	24	2	32.34	0.9305
PSA	9.99M	2465G	16	2	33.09	0.9371
PSA	10.06M	2703G	24	2	33.38	0.9397

Table 3. $\times 2$ SR performance comparison among SwinIR [37], our proposed PSA, and the two variants on Urban100 [20]. The results reported here are based on the best model trained on DIV2K for 200k iterations. For token sampling, $r = S/T$. PSA performs better than another two variants.

4.1. Experimental Setup

Datasets and Evaluation. The choice of training datasets keeps the same as the comparison models. In classical image SR, we use DIV2K [38] and DF2K (DIV2K [38] + Flickr2K [57]) to train two versions SRFormer. In lightweight image SR, we use DIV2K [38] to train our SRFormer-light. In real-world SR, We use DF2K and OST [66]. For testing, we mainly evaluate our method on five benchmark datasets, including Set5 [3], Set14 [77], BSD100 [45], Urban100 [20], and Manga109 [46]. Self-ensemble strategy is introduced to further improve performance, named SRFormer+. The experimental results are evaluated in terms of PSNR and SSIM values, which are calculated on the Y channel from the YCbCr space.

Implementation Details. In the classical image SR task, we set the PAB group number, PAB number, channel number, and attention head number to 6, 6, 180, and 6, respectively. When training on DIV2K [38], the patch size, window size S, and reduction factor r are set to 48×48, 24, and 2, respectively. When training on DF2K [38, 57], they are 64×64, 22, and 2, respectively. For the lightweight image SR task, we set the PAB group number, PAB number, channel number, windows size S, reduction factor r, and attention head number to 4, 6, 60, 16, 2, and 6, respectively. The training patch size of SRFormer-light is 64×64. We randomly rotate images by 90°, 180°, or 270° and randomly flip images horizontally for data augmentation. We adopt the Adam [28] optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.99$ to train the model for 500k iterations. The initial learning rate is set as 2×10^{-4} and subsequently halved at the $\{250\text{k}, 400\text{k}, 450\text{k}, 475\text{k}\}$-th iterations.

4.2. Ablation Study

Impact of window size in PSA. Permuted self-attention provides an efficient and effective way to enlarge window size. To investigate the impact of different window sizes on model performance, we conduct three group experiments and report in Table 1. The first group is the vanilla SwinIR [37] with 8×8, 12×12, and 16×16 window sizes. In the second group, we do not use the ConvFFN but only the PSA in our SRFormer and set the window size to 12×12, 16×16, and 24×24, respectively, to observe the performance difference. In the third group, we use our full SRFormer with 12×12, 16×16, and 24×24 as window size to explore the performance change. The results show that a larger window size yields better performance improvement for all three groups of experiments. In addition, the parameters and MACs of our SRFormer with 24×24 window are

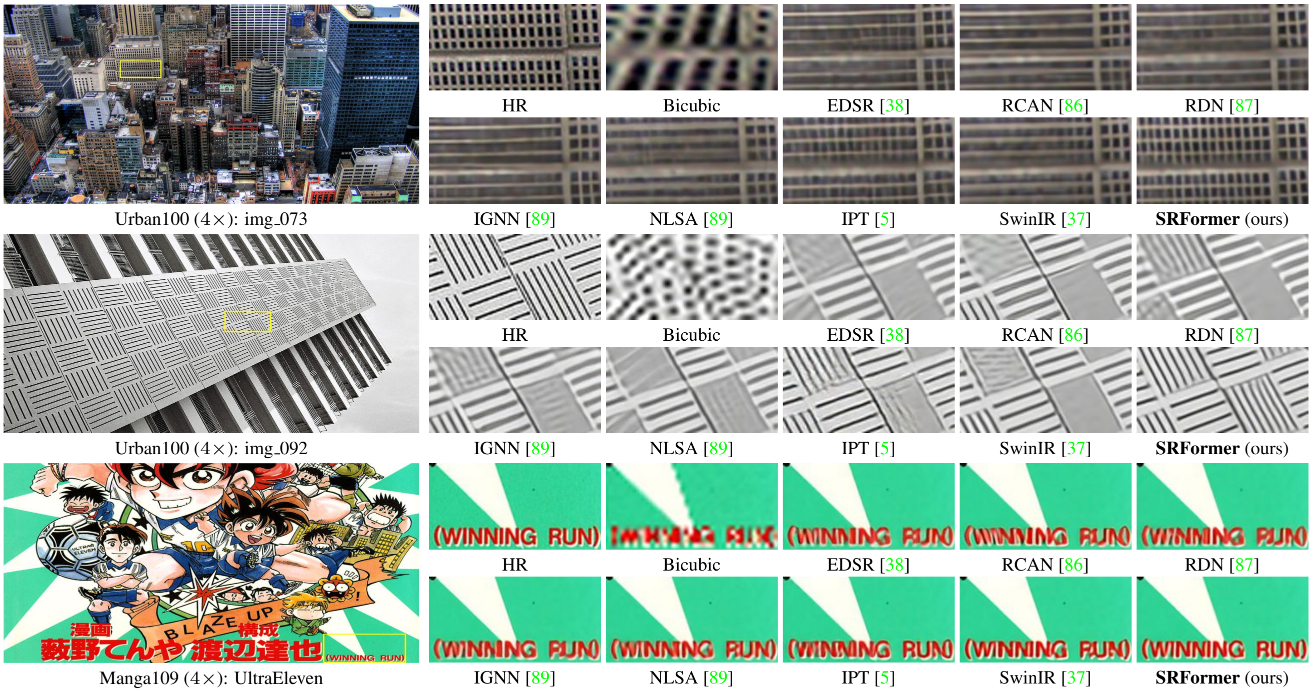

Figure 5. Qualitative comparison with recent state-of-the-art **classical image SR** methods on the $\times 4$ SR task.

even fewer than the original SwinIR with 8×8 window. To balance the performance and MACs, we set window size as 24×24 in SRFormer and 16×16 in SRFormer-light.

Impact of kernel size of ConvFFN. We introduce ConvFFN in Sec. 3.2, which aims to encode more local information without increasing too many computations. In order to explore which kernel size can bring the best performance improvement, we attempt to use 3×3 depth-wise convolution and 5×5 depth-wise convolution and report the results in Table 2. Given that the depth-wise convolution has little effect on the number of parameters and MACs, we do not list them in the table. Thus, we use 5×5 depth-wise convolution in our ConvFFN since it leads to the best results.

Large-window self-attention variants. In Sec. 3.3, we introduce another two large-window self-attention variants. We summarize the results in Table 3. Though token reduction can slightly improve SwinIR when using a large window, the number of parameters does not decrease and the performance gain is lower than ours. We argue that it is because directly applying downsampling operations to the key and value results in spatial information loss. For token sampling, the performance is even worse than the original SwinIR. We believe the reason is that dropping out some tokens severely breaks the image content structure.

4.3. Classical Image Super-Resolution

For the classical image SR task, we conduct quantitative comparison and qualitative comparison with a series of state-of-the-art CNN-based and Transformer-based SR methods: RCAN [86], RDN [87], SAN [7], IGNN [89], HAN [49], NLSA [47], SRFBN [36], IPT [5], SwinIR [37], EDT [34], and ELAN [85].

Quantitative comparison. The quantitative comparison of the methods for classical image SR is shown in Table 4. For a fair comparison, the number of parameters and MACs of SRFormer are lower than SwinIR [37] (10.52M and 697G vs 11.90M and 747G for upscaling a low-resolution image to 1280×720 in $\times 4$ SR). The remarkable achievement of SRFormer is readily apparent as it attains the best performance across nearly all five datasets for all scale factors. Since calculating self-attention within large windows can allow more information to be aggregated over a large area, our SRFormer demonstrates notable superiority when applied to high-resolution test sets like Urban100 and Manga109. Especially, for the $\times 2$ SR training with DIV2K, our SRFormer achieves a 33.86dB PSNR score on the Urban100 dataset, which is 0.46dB higher than SwinIR but uses fewer parameters and computations. The performance boost gets even bigger when introducing ensemble strategy as SRFormer+. The aforementioned results strongly support the effectiveness and efficiency of our SRFormer.

Scale	Method	Training Dataset	SET5 [3]		SET14 [77]		B100 [45]		Urban100 [20]		Manga109 [46]	
			PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
×2 SR	**EDSR** [38]	DIV2K	38.11	0.9602	33.92	0.9195	32.32	0.9013	32.93	0.9351	39.10	0.9773
	RCAN [86]	DIV2K	38.27	0.9614	34.12	0.9216	32.41	0.9027	33.34	0.9384	39.44	0.9786
	SAN [7]	DIV2K	38.31	0.9620	34.07	0.9213	32.42	0.9028	33.10	0.9370	39.32	0.9792
	IGNN [89]	DIV2K	38.24	0.9613	34.07	0.9217	32.41	0.9025	33.23	0.9383	39.35	0.9786
	HAN [49]	DIV2K	38.27	0.9614	34.16	0.9217	32.41	0.9027	33.35	0.9385	39.46	0.9785
	NLSA [47]	DIV2K	38.34	0.9618	34.08	0.9231	32.43	0.9027	33.42	0.9394	39.59	0.9789
	SwinIR [37]	DIV2K	38.35	0.9620	34.14	0.9227	32.44	0.9030	33.40	0.9393	39.60	0.9792
	ELAN [85]	DIV2K	38.36	0.9620	34.20	0.9228	32.45	0.9030	33.44	0.9391	39.62	**0.9793**
	SRFormer (ours)	DIV2K	**38.45**	**0.9622**	**34.21**	**0.9236**	**32.51**	**0.9038**	**33.86**	**0.9426**	**39.69**	0.9786
	SRFBN [36]	DF2K	38.11	0.9609	33.82	0.9196	32.29	0.9010	32.62	0.9328	39.08	0.9779
	IPT [5]	ImageNet	38.37	-	34.43	-	32.48	-	33.76	-	-	-
	SwinIR [37]	DF2K	38.42	0.9623	34.46	0.9250	32.53	0.9041	33.81	0.9427	39.92	0.9797
	EDT [34]	DF2K	38.45	0.9624	34.57	0.9258	32.52	0.9041	33.80	0.9425	39.93	0.9800
	SRFormer (ours)	DF2K	38.51	0.9627	34.44	0.9253	32.57	0.9046	34.09	0.9449	40.07	0.9802
	SRFormer+ (ours)	DF2K	**38.58**	**0.9628**	**34.60**	**0.9262**	**32.61**	**0.9050**	**34.29**	**0.9457**	**40.19**	**0.9805**
×3 SR	**EDSR** [38]	DIV2K	34.65	0.9280	30.52	0.8462	29.25	0.8093	28.80	0.8653	34.17	0.9476
	RCAN [86]	DIV2K	34.74	0.9299	30.65	0.8482	29.32	0.8111	29.09	0.8702	34.44	0.9499
	SAN [7]	DIV2K	34.75	0.9300	30.59	0.8476	29.33	0.8112	28.93	0.8671	34.30	0.9494
	IGNN [89]	DIV2K	34.72	0.9298	30.66	0.8484	29.31	0.8105	29.03	0.8696	34.39	0.9496
	HAN [49]	DIV2K	34.75	0.9299	30.67	0.8483	29.32	0.8110	29.10	0.8705	34.48	0.9500
	NLSA [47]	DIV2K	34.85	0.9306	30.70	0.8485	29.34	0.8117	29.25	0.8726	34.57	0.9508
	SwinIR [37]	DIV2K	34.89	0.9312	30.77	0.8503	29.37	0.8124	29.29	0.8744	34.74	0.9518
	ELAN [85]	DIV2K	34.90	0.9313	30.80	0.8504	29.38	0.8124	29.32	0.8745	34.73	0.9517
	SRFormer (ours)	DIV2K	**34.94**	**0.9318**	**30.81**	**0.8518**	**29.41**	**0.8142**	**29.52**	**0.8786**	**34.78**	**0.9524**
	SRFBN [36]	DF2K	34.70	0.9292	30.51	0.8461	29.24	0.8084	28.73	0.8641	34.18	0.9481
	IPT [5]	ImageNet	34.81	-	30.85	-	29.38	-	29.49	-	-	-
	SwinIR [37]	DF2K	34.97	0.9318	30.93	0.8534	29.46	0.8145	29.75	0.8826	35.12	0.9537
	EDT [34]	DF2K	34.97	0.9316	30.89	0.8527	29.44	0.8142	29.72	0.8814	35.13	0.9534
	SRFormer (ours)	DF2K	35.02	0.9323	30.94	0.8540	29.48	0.8156	30.04	0.8865	35.26	0.9543
	SRFormer+ (ours)	DF2K	**35.08**	**0.9327**	**31.04**	**0.8551**	**29.53**	**0.8162**	**30.21**	**0.8884**	**35.45**	**0.9550**
×4 SR	**EDSR** [38]	DIV2K	32.46	0.8968	28.80	0.7876	27.71	0.7420	26.64	0.8033	31.02	0.9148
	RCAN [86]	DIV2K	32.63	0.9002	28.87	0.7889	27.77	0.7436	26.82	0.8087	31.22	0.9173
	SAN [7]	DIV2K	32.64	0.9003	28.92	0.7888	27.78	0.7436	26.79	0.8068	31.18	0.9169
	IGNN [89]	DIV2K	32.57	0.8998	28.85	0.7891	27.77	0.7434	26.84	0.8090	31.28	0.9182
	HAN [49]	DIV2K	32.64	0.9002	28.90	0.7890	27.80	0.7442	26.85	0.8094	31.42	0.9177
	NLSA [47]	DIV2K	32.59	0.9000	28.87	0.7891	27.78	0.7444	26.96	0.8109	31.27	0.9184
	SwinIR [37]	DIV2K	32.72	0.9021	28.94	0.7914	27.83	0.7459	27.07	0.8164	31.67	0.9226
	ELAN [85]	DIV2K	32.75	0.9022	28.96	0.7914	27.83	0.7459	27.13	0.8167	31.68	0.9226
	SRFormer (ours)	DIV2K	**32.81**	**0.9029**	**29.01**	**0.7919**	**27.85**	**0.7472**	**27.20**	**0.8189**	**31.75**	**0.9237**
	SRFBN [36]	DF2K	32.47	0.8983	28.81	0.7868	27.72	0.7409	26.60	0.8015	31.15	0.9160
	IPT [5]	ImageNet	32.64	-	29.01	-	27.82	-	27.26	-	-	-
	SwinIR [37]	DF2K	32.92	0.9044	29.09	0.7950	27.92	0.7489	27.45	0.8254	32.03	0.9260
	EDT [34]	DF2K	32.82	0.9031	29.09	0.7939	27.91	0.7483	27.46	0.8246	32.03	0.9254
	SRFormer (ours)	DF2K	32.93	0.9041	29.08	0.7953	27.94	0.7502	27.68	0.8311	32.21	0.9271
	SRFormer+ (ours)	DF2K	**33.09**	**0.9053**	**29.19**	**0.7965**	**28.00**	**0.7511**	**27.85**	**0.8338**	**32.44**	**0.9287**

Table 4. Quantitative comparison of our SRFormer with recent state-of-the-art **classical image SR** methods on five benchmark datasets. For a fair comparison, the parameters and MACs of SRFormer are **lower than** SwinIR. The best performance is highlighted and the second is underlined.

Qualitative comparison. We show qualitative comparisons with other methods in Figure 5. From the first two examples of Figure 5, one can clearly observe that SRFormer can restore more crisp and detailed textures as well as edges. In contrast, other models' results suffer blurry or low-quality. For the third example, our SRFormer is the only model that clearly restores every letter. The qualitative comparison reveals the fact that our SRFormer excels in restoring better high-resolution images than compared methods.

4.4. Lightweight Image Super-Resolution

To demonstrate our model's scalability and further proof of SRFormer's efficiency and effectiveness, we train SRFormer-light and compare it with a compilation of state-of-the-art lightweight SR methods: EDSR-baseline [38], CARN [1], IMDN [21], LAPAR-A [35], LatticeNet [44], ESRT [43], SwinIR-light [37], and ELAN [85].

Quantitative comparison. The quantitative comparisons

	Method	Training Dataset	Params	MACs	SET5 [3]		SET14 [77]		B100 [45]		Urban100 [20]		Manga109 [46]	
					PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
×2 SR	**EDSR-baseline** [38]	DIV2K	1370K	316G	37.99	0.9604	33.57	0.9175	32.16	0.8994	31.98	0.9272	38.54	0.9769
	CARN [1]	DIV2K	1592K	222.8G	37.76	0.9590	33.52	0.9166	32.09	0.8978	31.92	0.9256	38.36	0.9765
	IMDN [21]	DIV2K	694K	158.8G	38.00	0.9605	33.63	0.9177	32.19	0.8996	32.17	0.9283	38.88	0.9774
	LAPAR-A [35]	DF2K	548K	171G	38.01	0.9605	33.62	0.9183	32.19	0.8999	32.10	0.9283	38.67	0.9772
	LatticeNet [44]	DIV2K	756K	169.5G	38.15	0.9610	33.78	0.9193	32.25	0.9005	32.43	0.9302	-	-
	ESRT [43]	DIV2K	751K	-	38.03	0.9600	33.75	0.9184	32.25	0.9001	32.58	0.9318	39.12	0.9774
	SwinIR-light [37]	DIV2K	910K	244G	38.14	0.9611	33.86	0.9206	32.31	0.9012	32.76	0.9340	39.12	0.9783
	ELAN [85]	DIV2K	621K	203G	38.17	0.9611	**33.94**	0.9207	32.30	0.9012	32.76	0.9340	39.11	0.9782
	SRFormer-light	DIV2K	853K	236G	**38.23**	**0.9613**	**33.94**	**0.9209**	**32.36**	**0.9019**	**32.91**	**0.9353**	**39.28**	**0.9785**
×3 SR	**EDSR-baseline** [38]	DIV2K	1555K	160G	34.37	0.9270	30.28	0.8417	29.09	0.8052	28.15	0.8527	33.45	0.9439
	CARN [1]	DIV2K	1592K	118.8G	34.29	0.9255	30.29	0.8407	29.06	0.8034	28.06	0.8493	33.50	0.9440
	IMDN [21]	DIV2K	703K	71.5G	34.36	0.9270	30.32	0.8417	29.09	0.8046	28.17	0.8519	33.61	0.9445
	LAPAR-A [35]	DF2K	594K	114G	34.36	0.9267	30.34	0.8421	29.11	0.8054	28.15	0.8523	33.51	0.9441
	LatticeNet [44]	DIV2K	765K	76.3G	34.53	0.9281	30.39	0.8424	29.15	0.8059	28.33	0.8538	-	-
	ESRT [43]	DIV2K	751K	-	34.42	0.9268	30.43	0.8433	29.15	0.8063	28.46	0.8574	33.95	0.9455
	SwinIR-light [37]	DIV2K	918K	111G	34.62	0.9289	30.54	0.8463	29.20	0.8082	28.66	0.8624	33.98	0.9478
	ELAN [85]	DIV2K	629K	90.1G	34.61	0.9288	30.55	0.8463	29.21	0.8081	28.69	0.8624	34.00	0.9478
	SRFormer-light	DIV2K	861K	105G	**34.67**	**0.9296**	**30.57**	**0.8469**	**29.26**	**0.8099**	**28.81**	**0.8655**	**34.19**	**0.9489**
×4 SR	**EDSR-baseline** [38]	DIV2K	1518K	114G	32.09	0.8938	28.58	0.7813	27.57	0.7357	26.04	0.7849	30.35	0.9067
	CARN [1]	DIV2K	1592K	90.9G	32.13	0.8937	28.60	0.7806	27.58	0.7349	26.07	0.7837	30.47	0.9084
	IMDN [21]	DIV2K	715K	40.9G	32.21	0.8948	28.58	0.7811	27.56	0.7353	26.04	0.7838	30.45	0.9075
	LAPAR-A [35]	DF2K	659K	94G	32.15	0.8944	28.61	0.7818	27.61	0.7366	26.14	0.7871	30.42	0.9074
	LatticeNet [44]	DIV2K	777K	43.6G	32.30	0.8962	28.68	0.7830	27.62	0.7367	26.25	0.7873	-	-
	ESRT [43]	DIV2K	751K	-	32.19	0.8947	28.69	0.7833	27.69	0.7379	26.39	0.7962	30.75	0.9100
	SwinIR-light [37]	DIV2K	930K	63.6G	32.44	0.8976	28.77	0.7858	27.69	0.7406	26.47	0.7980	30.92	0.9151
	ELAN [85]	DIV2K	640K	54.1G	32.43	0.8975	28.78	0.7858	27.69	0.7406	26.54	0.7982	30.92	0.9150
	SRFormer-light	DIV2K	873K	62.8G	**32.51**	**0.8988**	**28.82**	**0.7872**	**27.73**	**0.7422**	**26.67**	**0.8032**	**31.17**	**0.9165**

Table 5. Quantitative comparison of our SRFormer-light with recent state-of-the-art **lightweight image SR** methods on five benchmark datasets. The best performance among all the model is highlighted.

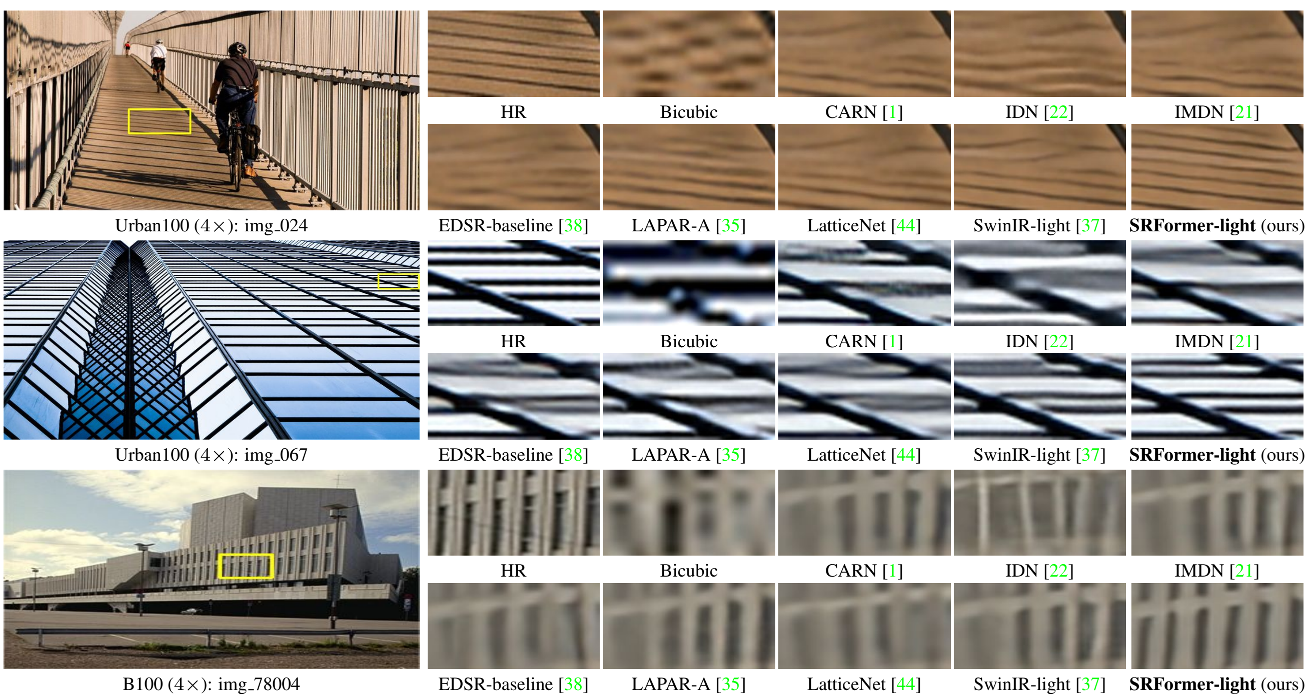

Figure 6. Qualitative comparison of our SRFormer-light with recent **lightweight image SR** methods on the ×4 SR task.

of lightweight image SR models are shown in Table 5. Following previous works [1, 44], we report the MACs by upscaling a low-resolution image to 1280×720 resolution on all scales. We can see that our SRFormer-light achieves the best performance across all five benchmark datasets, regardless of the scale factors. Notably, Our model surpasses

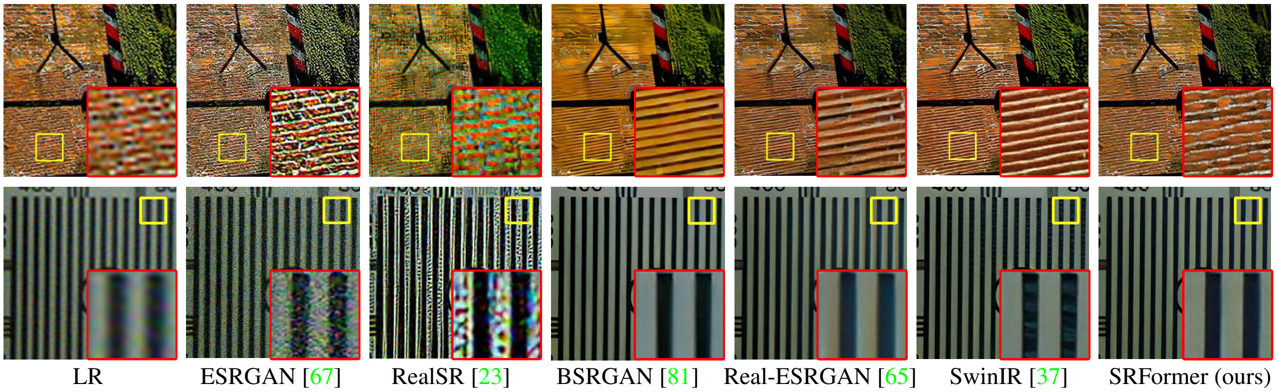

Figure 7. Qualitative comparisons with recent state-of-the-art methods on the $\times 4$ **real-world image SR** task.

SwinIR-light [37] by a substantial margin, up to 0.20 dB PSNR scores on the Urban100 dataset and 0.25 dB PSNR scores on the Manga109 dataset with even fewer parameters and MACs. The results indicate that despite the simplicity, our permuted self-attention is a more effective way to encode spatial information.

Qualitative comparison. We also conduct a qualitative comparisons between our SRFormer and state-of-the-art lightweight image super-resolution models, as illustrated in Figure 6. Notably, across all examples in Figure 6, SRFormer-light succeeds in restoring primary structures with minimal blurring and artifacts. This strongly demonstrates that the light version of SRFormer also outperforms alternative models.

4.5. Real-World Image Super-Resolution

Since the ultimate goal of image SR is to address the rich real-world degradation and generate visually pleasing images, we follow SwinIR [37] and BSRGAN [81] to retrain our SRFormer by using multiple degradations and show results in Fig. 7. SRFormer still produces more realistic and visually appealing textures without artifacts when faced with real-world images, which demonstrates the robustness of our method.

4.6. LAM Comparison

To observe the range of utilized pixels for SR reconstruction, we compare our model with SwinIR using the interpretability analysis tool LAM [15], as shown in Fig. 8. LAM shows the range of pixels used by the super-resolution network when inferring a certain part of the HR image. Based on the extremely large attention window, SRFormer infers SR images with a significantly wider range of pixels than SwinIR [37]. The experimental results are strongly consistent with our motivation and demonstrate the superiority of our method from the interpretability perspective.

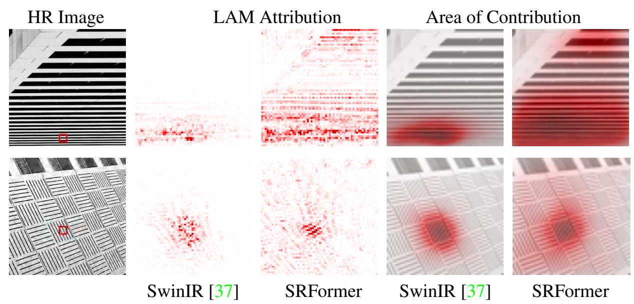

Figure 8. LAM results of SwinIR [37] and SRFormer on multiple challenging examples. We can see that SRFormer can perform SR reconstruction based on a particularly wide range of pixels, which demonstrates the superiority of our method from the interpretability perspective.

5. Conclusion

In this paper, we propose PSA, an efficient self-attention mechanism which can efficiently build pairwise correlations within large windows. Based on our PSA, we design a simple yet effective Transformer-based model for single image super-resolution, called SRFormer. Due to the extremely large attention window and high-frequency information enhancement, SRFormer achieves state-of-the-art performance on classical, lightweight, and real-world SR tasks. We hope our permuted self-attention can be a paradigm of large window self-attention and serve as a useful tool for future research in super-resolution model design.

Acknowledgments. This research was supported by NSFC (No. 62276145, 62176130), the Fundamental Research Funds for the Central Universities (Nankai University, 070-63223049, 070-63233089), CAST through Young Elite Scientist Sponsorship Program (No. YESS20210377). Computation is supported by the Supercomputing Center of Nankai University (NKSC).

References

[1] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *Eur. Conf. Comput. Vis.*, 2018. 7, 8

[2] Saeed Anwar, Salman Khan, and Nick Barnes. A deep journey into super-resolution: A survey. *ACM Computing Surveys*, 2020. 1

[3] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *BMVC*, 2012. 5, 7, 8

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Eur. Conf. Comput. Vis.*, 2020. 2

[5] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 1, 2, 6, 7

[6] Xiangyu Chen, Xintao Wang, Jiantao Zhou, and Chao Dong. Activating more pixels in image super-resolution transformer. *arXiv:2205.04437*, 2022. 1

[7] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2, 6, 7

[8] Yingying Deng, Fan Tang, Weiming Dong, Chongyang Ma, Xingjia Pan, Lei Wang, and Changsheng Xu. Stytr2: Image style transfer with transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[9] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Eur. Conf. Comput. Vis.*, 2014. 1, 2

[10] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *Eur. Conf. Comput. Vis.*, 2016. 2

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. Learn. Represent.*, 2020. 2, 3

[12] Yuxin Fang, Bencheng Liao, Xinggang Wang, Jiemin Fang, Jiyang Qi, Rui Wu, Jianwei Niu, and Wenyu Liu. You only look at one sequence: Rethinking transformer in vision through object detection. In *Adv. Neural Inform. Process. Syst.*, 2021. 2

[13] Zhicheng Geng, Luming Liang, Tianyu Ding, and Ilya Zharkov. Rstt: Real-time spatial temporal transformer for space-time video super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[14] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv:1406.2661*, 2014. 2

[15] Jinjin Gu and Chao Dong. Interpreting super-resolution networks with local attribution maps. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 9

[16] Yuchao Gu, Lijuan Wang, Ziqin Wang, Yun Liu, Ming-Ming Cheng, and Shao-Ping Lu. Pyramid constrained self-attention network for fast video salient object detection. In *Association for the Advancement of Artificial Intelligence*, 2020. 2

[17] Yuchao Gu, Xintao Wang, Liangbin Xie, Chao Dong, Gen Li, Ying Shan, and Ming-Ming Cheng. Vqfr: Blind face restoration with vector-quantized dictionary and parallel decoder. In *Eur. Conf. Comput. Vis.*, 2022. 2

[18] Chun-Le Guo, Qixin Yan, Saeed Anwar, Runmin Cong, Wenqi Ren, and Chongyi Li. Image dehazing transformer with transmission-aware 3d position embedding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[19] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 2

[20] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015. 2, 5, 7, 8

[21] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *ACM Int. Conf. Multimedia*, 2019. 7, 8

[22] Zheng Hui, Xiumei Wang, and Xinbo Gao. Fast and accurate single image super-resolution via information distillation network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 8

[23] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2020. 9

[24] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. Transgan: Two transformers can make one strong gan. *arXiv:2102.07074*, 2021. 2

[25] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 1

[26] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 1

[27] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 2

[28] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014. 5

[29] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 2

[30] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken,

Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 1, 2

[31] Kwonjoon Lee, Huiwen Chang, Lu Jiang, Han Zhang, Zhuowen Tu, and Ce Liu. Vitgan: Training gans with vision transformers. *arXiv:2107.04589*, 2021. 2

[32] Juncheng Li, Faming Fang, Kangfu Mei, and Guixu Zhang. Multi-scale residual network for image super-resolution. In *Eur. Conf. Comput. Vis.*, 2018. 1

[33] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning (ICML)*, 2022. 2

[34] Wenbo Li, Xin Lu, Jiangbo Lu, Xiangyu Zhang, and Jiaya Jia. On efficient transformer and image pre-training for low-level vision. *arXiv:2112.10175*, 2022. 6, 7

[35] Wenbo Li, Kun Zhou, Lu Qi, Nianjuan Jiang, Jiangbo Lu, and Jiaya Jia. Lapar: Linearly-assembled pixel-adaptive regression network for single image super-resolution and beyond. In *Adv. Neural Inform. Process. Syst.*, 2020. 7, 8

[36] Zhen Li, Jinglei Yang, Zheng Liu, Xiaomin Yang, Gwanggil Jeon, and Wei Wu. Feedback network for image super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 6, 7

[37] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Int. Conf. Comput. Vis. Worksh.*, 2021. 1, 2, 3, 4, 5, 6, 7, 8, 9

[38] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2017. 1, 2, 5, 6, 7, 8

[39] Chengxu Liu, Huan Yang, Jianlong Fu, and Xueming Qian. Learning trajectory-aware transformer for video super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[40] Rui Liu, Hanming Deng, Yangyi Huang, Xiaoyu Shi, Lewei Lu, Wenxiu Sun, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fuseformer: Fusing fine-grained information in transformers for video inpainting. In *Int. Conf. Comput. Vis.*, 2021. 2

[41] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Int. Conf. Comput. Vis.*, 2021. 1, 2, 3

[42] Liying Lu, Ruizheng Wu, Huaijia Lin, Jiangbo Lu, and Jiaya Jia. Video frame interpolation with transformer. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[43] Zhisheng Lu, Hong Liu, Juncheng Li, and Linlin Zhang. Efficient transformer for single image super-resolution. *arXiv:2108.11084*, 2021. 7, 8

[44] Xiaotong Luo, Yuan Xie, Yulun Zhang, Yanyun Qu, Cuihua Li, and Yun Fu. Latticenet: Towards lightweight image super-resolution with lattice block. In *Eur. Conf. Comput. Vis.*, 2020. 7, 8

[45] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Int. Conf. Comput. Vis.*, 2001. 5, 7, 8

[46] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, 2017. 5, 7, 8

[47] Yiqun Mei, Yuchen Fan, and Yuqian Zhou. Image super-resolution with non-local sparse attention. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2, 6, 7

[48] Yiqun Mei, Yuchen Fan, Yuqian Zhou, Lichao Huang, Thomas S Huang, and Honghui Shi. Image super-resolution with cross-scale non-local attention and exhaustive self-exemplars mining. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

[49] Ben Niu, Weilei Wen, Wenqi Ren, Xiangde Zhang, Lianping Yang, Shuzhen Wang, Kaihao Zhang, Xiaochun Cao, and Haifeng Shen. Single image super-resolution via a holistic attention network. In *Eur. Conf. Comput. Vis.*, 2020. 2, 6, 7

[50] Namuk Park and Songkuk Kim. How do vision transformers work? In *Int. Conf. Learn. Represent.*, 2022. 4

[51] Jingjing Ren, Qingqing Zheng, Yuanyuan Zhao, Xuemiao Xu, and Chen Li. Dlformer: Discrete latent transformer for video inpainting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[52] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 1, 2, 3

[53] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Int. Conf. Comput. Vis.*, 2021. 2

[54] Zhiqing Sun, Shengcao Cao, Yiming Yang, and Kris M Kitani. Rethinking transformer-based set prediction for object detection. In *Int. Conf. Comput. Vis.*, 2021. 2

[55] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 1, 2

[56] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *Int. Conf. Comput. Vis.*, 2017. 2

[57] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2017. 5

[58] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *Int. Conf. Comput. Vis.*, 2017. 1

[59] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *arXiv:2012.12877*, 2020. 2

[60] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Int. Conf. Comput. Vis.*, 2021. 2, 3

[61] Peihao Wang, Wenqing Zheng, Tianlong Chen, and Zhangyang Wang. Anti-oversmoothing in deep vision transformers via the fourier domain analysis: From theory to practice. *arXiv:2203.05962*, 2022. 4

[62] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Int. Conf. Comput. Vis.*, 2021. 2, 3

[63] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2019. 1

[64] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 2

[65] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *Int. Conf. Comput. Vis.*, 2021. 2, 9

[66] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 5

[67] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Eur. Conf. Comput. Vis. Worksh.*, 2018. 1, 2, 9

[68] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[69] Mingrui Wu, Xuying Zhang, Xiaoshuai Sun, Yiyi Zhou, Chao Chen, Jiaxin Gu, Xing Sun, and Rongrong Ji. Difnet: Boosting visual information flow for image captioning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[70] Rui-Qi Wu, Zheng-Peng Duan, Chun-Le Guo, Zhi Chai, and Chongyi Li. Ridcp: Revitalizing real image dehazing via high-quality codebook priors. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2023. 2

[71] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Adv. Neural Inform. Process. Syst.*, 2021. 2, 3, 4

[72] Yue Yang and Yong Qi. Image super-resolution via channel attention and spatial graph convolutional network. *Pattern Recognition*, 2021. 1

[73] Bowen Yin, Xuying Zhang, Qibin Hou, Bo-Yuan Sun, Deng-Ping Fan, and Luc Van Gool. Camoformer: Masked separable attention for camouflaged object detection. *arXiv:2212.06570*, 2022. 2

[74] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis E.H. Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Int. Conf. Comput. Vis.*, 2021. 2

[75] Li Yuan, Qibin Hou, Zihang Jiang, Jiashi Feng, and Shuicheng Yan. Volo: Vision outlooker for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2022. 2

[76] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[77] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *Curves and Surfaces*, 2010. 5, 7, 8

[78] Bowen Zhang, Shuyang Gu, Bo Zhang, Jianmin Bao, Dong Chen, Fang Wen, Yong Wang, and Baining Guo. Styleswin: Transformer-based gan for high-resolution image generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[79] Dafeng Zhang, Feiyu Huang, Shizhuo Liu, Xiaobing Wang, and Zhezhu Jin. Swinfir: Revisiting the swinir with fast fourier convolution and improved training for image super-resolution. *arXiv:2208.11247*, 2022. 1

[80] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021. 1

[81] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *Int. Conf. Comput. Vis.*, 2021. 9

[82] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 1

[83] Wenlong Zhang, Yihao Liu, Chao Dong, and Yu Qiao. Ranksrgan: Generative adversarial networks with ranker for image super-resolution. In *Int. Conf. Comput. Vis.*, 2019. 2

[84] Xuying Zhang, Xiaoshuai Sun, Yunpeng Luo, Jiayi Ji, Yiyi Zhou, Yongjian Wu, Feiyue Huang, and Rongrong Ji. Rstnet: Captioning with adaptive attention on visual and non-visual words. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[85] Xindong Zhang, Hui Zeng, Shi Guo, and Lei Zhang. Efficient long-range attention network for image super-resolution. *arXiv:2203.06697*, 2022. 1, 2, 3, 6, 7, 8

[86] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Eur. Conf. Comput. Vis.*, 2018. 1, 2, 6, 7

[87] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 1, 2, 6

[88] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[89] Shangchen Zhou, Jiawei Zhang, Wangmeng Zuo, and Chen Change Loy. Cross-scale internal graph neural network for image super-resolution. In *Adv. Neural Inform. Process. Syst.*, 2020. 2, 6, 7