# Dynamic Window Transformer for Image Super-Resolution

Zheng Xie, Zhongxun Wang*, Tianci Qin, Zhexuan Han, and Ruoyu Zhou

School of Physics and Electronics Information, Yantai University, Yantai, China
{zhengxie, zhouruoyu}@s.ytu.edu.cn

**Abstract.** Image super-resolution (SR) reconstruction is a critical task in image processing that aims to generate high-resolution (HR) images from low-resolution (LR) inputs. Recently, Swin-Transformer-based models have become mainstream in this field due to their efficient handling of computational complexity and scalability, where window-based mechanisms are employed to effectively extract local features and window interaction strategies are utilized to enhance global information integration. However, existing Swin-Transformer-based SR models employ the fixed-window strategy, confining attention in fixed areas. In this paper, we present Dynamic Window Transformer (DWT), a simple but novel method that can use windows of various shapes to effectively extract diverse features and achieve efficient global dependency modelling by utilizing image anisotropy. The core of our DWT is Dynamic-Window Self-Attention (DWSA), which dynamically selects the optimal window for different inputs to perform self-attention. We evaluate our model on various popular benchmark datasets and compare it with other state-of-the-art (SOTA) lightweight models. For example, our DWT achieves a PSNR of 26.56 dB on the Urban100 dataset, which is 0.09 dB higher than the SOTA model SwinIR.

**Keywords:** Super-Resolution · Dynamic-Window Self-Attention · Transformer.

## 1 Introduction

Single image super-resolution (SR) is a long-standing low-level vision problem that aims to reconstruct a high-resolution (HR) image from its degraded low-resolution (LR) counterpart. In recent years, CNN-based methods [9,19,22,31,33, 34, 36, 46] have nearly achieved dominance in the field of image super-resolution due to their powerful ability to extract high-frequency details from images. These methods largely adopt techniques such as residual learning [17, 19, 22, 32], dense connections [39, 46], or channel attention [23] to develop network architectures, significantly advancing the progress of SR models.

Recently, inspired by the great success of Transformer [24,30,35,36] in natural language processing, researchers have been using Transformer to replace CNN and achieving impressive performance in high-level vision tasks [11, 12, 35, 41].

Meanwhile, with the rapid progress of this approach, some researchers have also explored the potential of Transformer in high-resolution scenarios and proposed many Transformer-based models for low-level vision task [5,21,37,40] such as SR [21,29,44]. The benchmark of them should be SwinIR [21] which first applies Swin Transformer [25] in SR, outperforming the state-of-the-art CNN-based models on various benchmarks.

However, two observations limit the quality of SR images produced by SwinIR: 1) Because the attention mechanism is implemented by capturing global information of the source image, the overall reconstructed image is smoother, while some local details are difficult to detect. This has little effect on high-resolution images, but significantly reduces the perceived quality of small-size images. 2) Because image features often display diverse patterns and structures, it can be challenging to effectively capture these features using the fixed window partitioning strategy.

Despite SwinIR [21] has proven that increasing the window size can result in clear performance gain, the computation overhead is also considerable. Building upon ELAN [44] introduces a multi-scale window strategy, achieving competitive performance compared to previous state-of-the-art SR models. However, its reconstructed images continue to exhibit blurring and ghosting, even in regions with repetitive content or simple structures. The aforementioned methods reveal that using the fixed window partitioning strategy limits the ability to effectively capture and adapt diverse patterns and structures in image features.

This paper introduces a content-adaptive window partitioning strategy, namely dynamic window partitioning strategy. We apply this strategy to the deep feature extraction module in SwinIR and get a new model specifically designed for the SR task, called Dynamic Window Transformer (DWT). The core innovation of DWT is the replacement of the standard Window Multi-Head Self-Attention (WMSA) module of SwinIR with the Dynamic-Window Self-attention (DWSA) module. This DWSA module consists of multiple experts and a routing network, where each expert is responsible for performing self-attention using a specific window configuration. Additionally, to ensure the ability of this model to capture long-range features, we implement a shifting scheme between consecutive layers that use the same window.

Our contributions are summarized as follows:

1. We propose a dynamic window partitioning strategy that enhances the extraction of local feature information. This approach is better tailored to accommodate various image structures and patterns, which is crucial for improving super-resolution (SR) performance.
2. We apply the new strategy to SwinIR and develop a new image super-resolution model. In a lightweight framework, this model achieves state-of-the-art (SOTA) performance on several popular benchmark datasets.

## 2 Related work

**CNN-based SR.** CNNs have achieved remarkable success in the image SR task. SRCNN [8] was the pioneering work that introduced CNNs to the field of SR. The network structure is notably simple, employing only three convolutional layers. Subsequently, CNN-based methods have evolved with a focus on intricate architectural designs to improve performance. VDSR [17] introduces residual learning in the network of VGG-19 to build a network with 20 layers. FSRCNN [10] adopts a post-upsampling strategy to accelerate the CNN model. RCAN [45] proposes a residual-in-residual structure to train a model over 400 layers. IMDN [16] proposes a n information multi distillation network to achieve improved time performance. LAPAR [20] and LatticeNet [26] enhance traditional CNN architectures for image SR tasks by incorporating linear regression methods and butterfly structures, respectively. Subsequently, numerous models such as SAN [7] and RFDN [23] have introduced various attention mechanisms across spatial or channel dimensions. While these models achieve noteworthy outcomes, constructing the network for improved feature aggregation demands a substantial number of parameters.

**Transformer-based SR.** Transformer is proposed in natural language processing and has been adapted to multiple high-level vision tasks, such as image classification [4], and image detection [14, 26]. Vision Transformer [26, 41]used Transformer into the field of Computer Vision for the first time and worked out promising result. The IPT [5] integrates the Transformer architecture specifically during the feature map processing phase, marking a novel adaptation for low-level visual tasks. Swin-Transformer [25] is improved on the basis of Vision Transformer by introducing sliding window. The CSWin [11] utilizes a unique cross-shaped window self-attention mechanism to efficiently handle both global and local processing demands across common vision tasks like image classification, object detection, and semantic segmentation. SwinIR [21] follows the design of Swin-Transformer [25], which adopts the fixed-size window partitioning strategy to divide the feature map into non-overlapping small fixed-size windows (*i.e.*, $8 \times 8$) and then calculate separately for each window, while utilizing the shifted window mechanism to connect other windows. ELAN [44] further develops the SwinIR, which adopts the fixed multi-size window partitioning strategy to split the input feature into non-overlapped groups in the channel dimension and then calculate SA on these groups using different window sizes (*i.e.*, $4 \times 4$, $8 \times 8$, and $16 \times 16$), while utilizing a convolution to concatenate and merge these groups. However, existing Swin-Transformer-based SR models use fixed window strategy, confining the SA in fixed areas and consequently limiting their capacity for effective feature extraction.

## 3 Method

We propose the dynamic window Transformer for image SR, which excels in adapting to diverse image structures and patterns for enhanced high-resolution

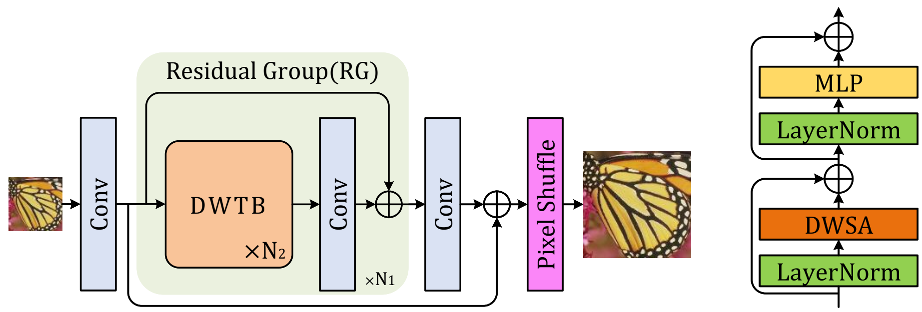


**Fig. 1:** (a) The architecture of our DWT. (b) Illustration of DWTB

image reconstruction. In this section, we first introduce the architecture of dynamic window Transformer (DWT) and then focus on the core design of the dynamic window Transformer block (DWTB), including dynamic-window self-attention (DWSA).

### 3.1 DWT Architecture

As shown in Fig. 1a, our proposed DWT consists of three modules: shallow feature extraction, deep feature extraction, and reconstruction part. Given a low-resolution (LR) image $I_{LQ} \in \mathbb{R}^{H\times W\times C_{in}}$ ($H, W, C_{in}$ are the image height, width, and input channel number), we first leverages a convolutional layer as the shallow feature extraction to get the low-level feature $X_0 \in \mathbb{R}^{H\times W\times C}$ as

$$X_0 = H_{SF}(I_{LQ}), \tag{1}$$

where $H_{SF}(\cdot)$ is the shallow feature extraction module, and $C$ is the feature channel number. Then the $X_0$ is transformed into the deep feature $X_{DF} \in \mathbb{R}^{H\times W\times C}$ as

$$X_{DF} = H_{DF}(X_0), \tag{2}$$

where $H_{DF}(\cdot)$ is the deep feature extraction module. The $X_0$ is used for the deep feature extraction module, which is composed of $N_1$ residual groups (RGs) and a convolutional layer as

$$\begin{aligned} X_i = H_{RG_i}(X_{i-1}), i \in \{1, 2, \ldots, N_1\},\\ X_{DF} = H_{CONV}(X_{N_1}), \end{aligned} \tag{3}$$

where $X_1, ..., X_{N_1}$ are intermediate features, $H_{RG_i}(\cdot)$ denotes the $i$-th RG, and $H_{CONV}(\cdot)$ is the last convolutional layer. Then each RG contains $N_2$ dynamic window Transformer blocks (DWTBs) along with a convolutional layer before the residual connection, which embeds distinctive characteristics of the CNN

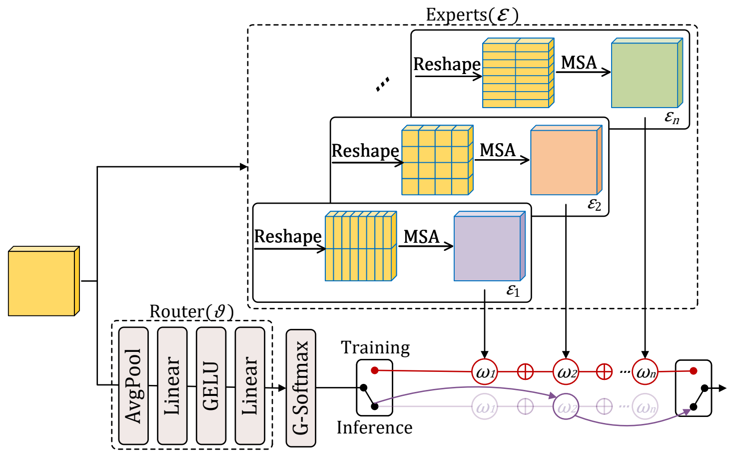


**Fig. 2:** Illustration of the dynamic-window self-attention (DWSA) as a core of DWTB

such as translation invariance and locality into the output from the Transformer blocks, and ensures training stability as

$$\begin{aligned} X_{i,j} &= H_{DWTB_{i,j}}(X_{i,j-1}), j \in \{1, 2, \dots, N_2\}, \\ X_{i,out} &= H_{CONV_i}(X_{N_2}) + X_{i,0}, \end{aligned} \tag{4}$$

where $X_{i,0}$ is the input feature of the $i$-th RG, $X_{i,1}, ..., X_{i,N_2}$ are intermediate features of DWTBs, $H_{DWTB_{i,j}}$ is the $j$-th DWTB in the $i$-th RG, and $H_{CONV_i}(\cdot)$ is the convolutional layer in the $i$-th RG.

Finally, the $X_0$ and $X_{DF}$ are fused through a residual connection, and processed by the reconstruction module to generate the high-resolution (HR) image $I_{HR} \in \mathbb{R}^{H \times W \times C_{out}}$, where $C_{out}$ is the output dimension. For the implementation of reconstruction module, we use the PixelShuffleDirect [31] to upsample the feature. The entire network process is formulated as

$$I_{HR} = H_{RC}(X_0 + X_{DF}), \tag{5}$$

where $H_{RC}(\cdot)$ is the reconstruction module.

### 3.2 Dynamic Window Transformer Block

The dynamic window Transformer block (DWTB) is the key part of our DWT, which is more flexible and adaptive for image SR tasks than previous Transformer structures [21, 40]. We pay more attention on dynamic-window self-attention (DWSA) in detail below.

**Dynamic-Window Self-Attention.** As illustrated in Fig. 2, we introduce a dynamic-window self-attention (DWSA) module, which can use the best and most suitable window configuration for self-attention based on different inputs

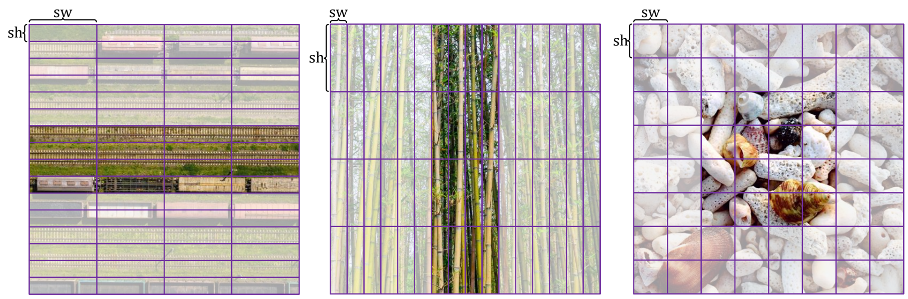


**Fig. 3:** Illustration of the self-attention using different window configurations. The $sh$ and $sw$ represent the height and width of the window.

and different network depths. DWSA consists of multiple experts, each implements the multi-head self-attention using a specific window configuration. Additionally, we employ a dynamic approach using routing network systematically explores the experts.

The square window uniformly extracts features from all directions due to the symmetry, while the rectangular window can capture different features in horizontal or vertical directions of each pixel. This makes the rectangular window particularly effective in handling landscapes and architectural images that extend longitudinally or horizontally, especially those with distinct directional and repetitive textures. As illustrated in Fig. 3, to better adapt to different images, DWSA uses window configurations including the vertical rectangle window ($sh < sw$), the square window ($sh = sw$), and the horizontal rectangle window ($sh > sw$), where $sh$ and $sw$ represent the height and width of the window. We denote these different windows, which vary in shape but are identical in area, as $W_1, \ldots, W_n$.

Specifically, from a layer-normalized input $X \in \mathbb{R}^{H \times W \times C}$, we reshape $X$ into non-overlapping $sh \times sw$ windows as $X \in \mathbb{R}^{L \times sh \times sw \times C}$ , where $L = \frac{H \times W}{sh \times sw}$ is the total number of windows. We perform multi-head self-attention computations in parallel on these windows. Then, after reshaping and merging them in the order of division, we can get the attention feature $\varepsilon_i \in \mathbb{R}^{H \times W \times C}$ of $X$.The $i$-th expert process is formulated as

$$\varepsilon_i = W_i - \mathrm{MSA}(X), \tag{6}$$

where $\varepsilon_i$ is the output of multi-head self-attention using the $i$-th window configuration, and $i \in \{1, \ldots, n\}$. Especially, to further conserve computing resources during training, we first generate the projection matrix for $X$, before the window partitioning.

However, manually determining the optimal window settings may not fully exploit all available information for modulation, leading to underutilized model capacity. Thus, we employ a dynamic approach using routing network ($\vartheta$) systematically explores, based on the overall characteristics of the input and the depth of the network. The architecture of the routing network is illustrated in

Fig. 2. For a layer-normalized input of size $H \times W \times C$, the process initiates adaptive average pooling to reshape the input to $1 \times 1 \times C$. This reshaped input is then sequentially processed through multiple linear layers and activation functions, transforming it into an n-dimensional space, where n is the number of window types. The Gumbel-SoftMax method is subsequently utilized to evaluate and fine-tune the output scores, facilitating a nuanced decision-making process among various window configurations beyond simple probability maximization. The routing network process is formulated as

$$\begin{aligned} \omega &= \text{Gumbel-Softmax}(\pi) \\ &= \text{Concat}\left[\frac{\exp(\log(\pi_i)+g_i)/\tau)}{\sum_{i=1}^{n}\exp(\log(\pi_i)+g_i)/\tau)}\right], i \in \{1,2,\dots,n\}, \end{aligned} \tag{7}$$

where $\pi$ are router outputs, $\omega$ are weights, $g_i$ represents noise sampled from a standard Gumbel distribution, and $\tau$ is the temperature parameter used to adjust the distribution. This approach enhances the robustness and adaptability of the model in selecting window modes. During the training phase, the model starts with a higher temperature parameter to encourage exploration, gradually lowering the temperature as training progresses to promote more definitive choices. In the testing phase, a fixed low temperature is used to ensure the accuracy in selections.

Finally, the output $Y$ of the DWSA is as

$$Y = \text{DWSA}(X) = \sum_{i}^{n} \vartheta(X)\varepsilon_i(X), \tag{8}$$

where $\vartheta(\cdot)$ and $\varepsilon(\cdot)$ denote the learned routing function and the output of the $i$-th expert. During training, our method leverages all experts. However, during inference, only the selected best expert is used for computation, further enhancing efficiency. Additionally, it is a lightweight attention network that contributes minimally to the increase in model complexity. We also provide the pseudocode for the proposed DWSA in Algorithm 1.

**Dynamic Window Transformer Block (DWTB).** As shown in Fig. 1b, the output of DWTB is formulated as

$$\begin{aligned} X &= \text{DWSA}(\text{LN}(X_{in})) + X_{in}, \\ X_{out} &= \text{MLP}(\text{LN}(X)) + X, \end{aligned} \tag{9}$$

where $X_{in}$ and $X_{out}$ are the input and output feature maps of DWTB. The Multi-layer perceptron (MLP) that has two fully connected layers with GELU non-linearity between them is used for further feature transformations. To improve the stability and convergence speed of the model training, we add LayerNorm (LN) layers before both the MLP and DWSA modules. Additionally, residual connections are applied to both modules to further accelerate the convergence of the network model.

However, the shifting operation should be performed before dividing the entire feature map. This process involves shifting the entire feature map by half the window size along its height and width directions.

### 3.3 Optimization Objective.

Following prior works [10,21,38,46], we train the model by minimizing a standard $L_1$ loss between model prediction $\hat{I}_{HR}$ and HR label $I_{HR}$ as follows:

$$l = \left\| I_{HR} - \hat{I}_{HR} \right\|_1 \tag{10}$$

**Algorithm 1** Dynamic-Window Self-Attention

**Input:** Input feature $x$
**Parameters:** $n$ Experts $\varepsilon$, Router $\vartheta$, window configurations $w_i$ with $i \in \{1, \ldots, n\}$, best expert $k$
**Output:** Final output $y$

Compute router outputs: $\pi = \vartheta(x)$
Normalize weights: $\omega = \text{Gumbel-Softmax}(\pi)$
Select best expert: $\omega_k = \text{best}(\omega, k)$
**if** training **then**
  **for** each $e \in \varepsilon$ **do**
    $y'_e = \mathrm{W_i}\text{-MAS}(x)$
  **end for**
  Compute final output: $y = \sum_{i=1}^{n} \omega_i \cdot y_e^i$
**else**
  Compute final output: $y = \omega_k \cdot y_e^k$
**end if**
**return** Final output $y$

## 4 Experiments

### 4.1 Experimental Setup

**Datasets and Metrics.** Following previous work [21, 44], DIV2K [22] is used as training dataset. For testing, we adopt five standard benchmark datasets: Set5 [3], Set14 [42], BSD100 [27], Urban100 [15] and Manga109 [28]. PSNR and SSIM are adopted to evaluate the SR performance on the Y channel of the transformed YCbCr space.

### 4.2 Experimental Results

To evaluate the performance of our model, we compare the proposed DWT model with other SOTA lightweight image super-resolution models, such as CARN [1], IMDN [16], RFDN-L [23], LAPAR [20], LatticeNet [26], SwinIR [21], ElAN [44], DiVNet [2], NGSwin [6], and STSN [13].

**Implementation Details.** During training, we augment the data with random horizontal flips and 90/270-degree rotations. LR images are generated by

**Table 1:** Quantitative comparison (average PSNR/SSIM) with other state-of-the-art models for lightweight image super-resolution on benchmark datasets. Best and 2nd best performance are in red and blue colors, respectively.

| Method | Scale | Params(K) | Flops(G) | PSNR/SSIM | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | Set5 | Set14 | BSD100 | Urban100 | Manga109 |
| Bicubic | | - | - | 33.66/0.9299 | 30.24/0.8688 | 29.56/0.8431 | 26.88/0.8403 | 30.80/0.9339 |
| CARN [1] | | 1,592 | 222.8 | 37.76/0.9590 | 33.52/0.9166 | 32.09/0.8978 | 31.92/0.9256 | 38.36/0.9765 |
| IMDN [16] | | 694 | 158.8 | 38.00/0.9605 | 33.63/0.9177 | 32.19/0.8996 | 32.17/0.9283 | 38.88/0.9774 |
| RFDN-L [23] | | 626 | 145.8 | 38.08/0.9606 | 33.67/0.9190 | 32.18/0.8996 | 32.24/0.9290 | 38.95/0.9773 |
| LAPAR [20] | | 548 | 171.5 | 38.01/0.9605 | 33.62/0.9183 | 32.19/0.8999 | 32.10/0.9283 | 38.67/0.9772 |
| LatticeNet [26] | | 756 | 169.5 | 38.15/0.9610 | 33.78/0.9193 | 32.25/0.9005 | 32.43/0.9302 | -/- |
| SwinIR [21] | ×2 | 878 | 195.6 | 38.14/0.9611 | 33.86/0.9206 | 32.31/0.9012 | 32.76/0.9340 | 39.12/0.9783 |
| ElAN [44] | | 582 | 168.4 | 38.17/0.9611 | 33.94/0.9207 | 32.30/0.9012 | 32.77/0.9340 | 39.11/0.9782 |
| DiVNet [2] | | 902 | 189 | 38.16/0.9612 | 33.80/0.9195 | 32.29/0.9012 | 32.60/0.9325 | 39.08/0.9775 |
| NGSwin [6] | | 998 | 140 | 38.05/0.9610 | 33.79/0.9199 | 32.27/0.9008 | 32.53/0.9324 | 38.97/0.9777 |
| STSN [13] | | 888.7 | 198.4 | 38.17/0.9611 | 33.78/0.9199 | 32.30/0.9013 | 32.68/0.9336 | 39.13/0.9778 |
| DWT(Ours) | | 882 | 197 | 38.18/0.9612 | 33.96/0.9211 | 32.34/0.9017 | 32.81/0.9344 | 39.13/0.9783 |
| Bicubic | | - | - | 30.39/0.8682 | 27.55/0.7742 | 27.21/0.7385 | 24.46/0.7349 | 26.95/0.8556 |
| CARN [1] | | 1,592 | 118.8 | 34.29/0.9255 | 30.29/0.8407 | 29.06/0.8034 | 28.06/0.8493 | 33.50/0.9440 |
| IMDN [16] | | 703 | 71.5 | 34.36/0.9270 | 30.32/0.8417 | 29.09/0.8046 | 28.17/0.8519 | 33.61/0.9445 |
| RFDN-L [23] | | 633 | 65.6 | 34.47/0.9280 | 30.35/0.8421 | 29.11/0.8053 | 28.32/0.8547 | 33.78/0.9458 |
| LAPAR [20] | | 594 | 114.4 | 34.36/0.9267 | 30.34/0.8421 | 29.11/0.8054 | 28.15/0.8523 | 33.51/0.9441 |
| LatticeNet [26] | | 765 | 76.3 | 34.53/0.9281 | 30.39/0.8424 | 29.15/0.8059 | 28.33/0.8538 | -/- |
| SwinIR [21] | ×3 | 886 | 87.2 | 34.62/0.9289 | 30.54/0.8463 | 29.20/0.8082 | 28.66/0.8624 | 33.98/0.9478 |
| ElAN [44] | | 590 | 75.7 | 34.61/0.9288 | 30.55/0.8463 | 29.21/0.8081 | 28.69/0.8624 | 34.00/0.9478 |
| DiVNet [2] | | 949 | 89 | 34.60/0.9285 | 30.47/0.8447 | 29.19/0.8073 | 28.58/0.8603 | 33.94/0.9468 |
| NGSwin [6] | | 1007 | 66.6 | 34.52/0.9282 | 30.53/0.8456 | 29.19/0.8078 | 28.52/0.8603 | 33.89/0.9470 |
| STSN [13] | | 895.5 | 100.6 | 34.62/0.9290 | 30.55/0.8466 | 29.22/0.8090 | 28.59/0.8621 | 34.01/0.9478 |
| DWT(Ours) | | 890 | 99.4 | 34.64/0.9291 | 30.57/0.8469 | 29.25/0.8093 | 28.72/0.8637 | 34.02/0.9479 |
| Bicubic | | - | - | 28.42/0.8104 | 26.00/0.7027 | 25.96/0.6675 | 23.14/0.6577 | 24.89/0.7866 |
| CARN [1] | | 1,592 | 90.9 | 32.13/0.8937 | 28.60/0.7806 | 27.58/0.7349 | 26.07/0.7837 | 30.47/0.9084 |
| IMDN [16] | | 715 | 40.9 | 32.21/0.8948 | 28.58/0.7811 | 27.56/0.7353 | 26.04/0.7838 | 30.45/0.9075 |
| RFDN-L [23] | | 643 | 37.4 | 32.28/0.8957 | 28.61/0.7818 | 27.58/0.7363 | 26.20/0.7883 | 30.61/0.9096 |
| LAPAR [20] | | 659 | 94.8 | 32.15/0.8944 | 28.61/0.7818 | 27.61/0.7366 | 26.14/0.7871 | 30.42/0.9074 |
| LatticeNet [26] | | 777 | 43.6 | 32.30/0.8962 | 28.68/0.7830 | 27.62/0.7367 | 26.25/0.7873 | -/- |
| SwinIR [21] | ×4 | 897 | 49.6 | 32.44/0.8976 | 28.77/0.7858 | 27.69/0.7406 | 26.47/0.7980 | 30.92/0.9151 |
| ElAN [44] | | 601 | 43.2 | 32.43/0.8975 | 28.78/0.7858 | 27.68/0.7406 | 26.54/0.7982 | 30.92/0.9150 |
| DiVNet [2] | | 939 | 48.8 | 32.41/0.8973 | 28.70/0.7844 | 27.65/0.7391 | 26.42/0.7958 | 30.73/0.9119 |
| NGSwin [6] | | 1019 | 36.4 | 32.33/0.8963 | 28.78/0.7859 | 27.66/0.7396 | 26.45/0.7963 | 30.80/0.9128 |
| STSN [13] | | 905 | 51 | 32.46/0.8976 | 28.76/0.7860 | 27.70/0.7406 | 26.39/0.7971 | 30.93/0.9142 |
| DWT(Ours) | | 901 | 50.6 | 32.46/0.8977 | 28.79/0.7862 | 27.73/0.7411 | 26.56/0.7988 | 30.93/0.9151 |

bicubic downsampling [43] from HR images. To ensure the meaningfulness of the added components, all settings are kept the same as in the lightweight SwinIR. The model is configured with a RG number of 4, a DWTB number of 4, a window configuration group of ((4,16), (8,8), (16,4)), a channel number of 60, and an attention head number of 6. During training, the $\tau$ of the routing network is initialized to 0.72 with a decay rate of 0.999998 per iteration, while $\tau$ is fixed at 0.03 during testing. This work employs the AdamW optimizer [18] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and the learning rate is initialized to $5 \times 10^{-4}$. The learning rate is reduced by half at 250, 400, 450 and 475 epochs. The model is trained for 500 epochs on GPU NVIDIA RTX 3090s. For ×2, ×3, and ×4 upscaling image super-resolution tasks, we use batch sizes of 64, respectively.

**Quantitative results.** In Tab. 1, the quantitative comparisons of different lightweight methods are presented on five benchmark datasets. With a similar

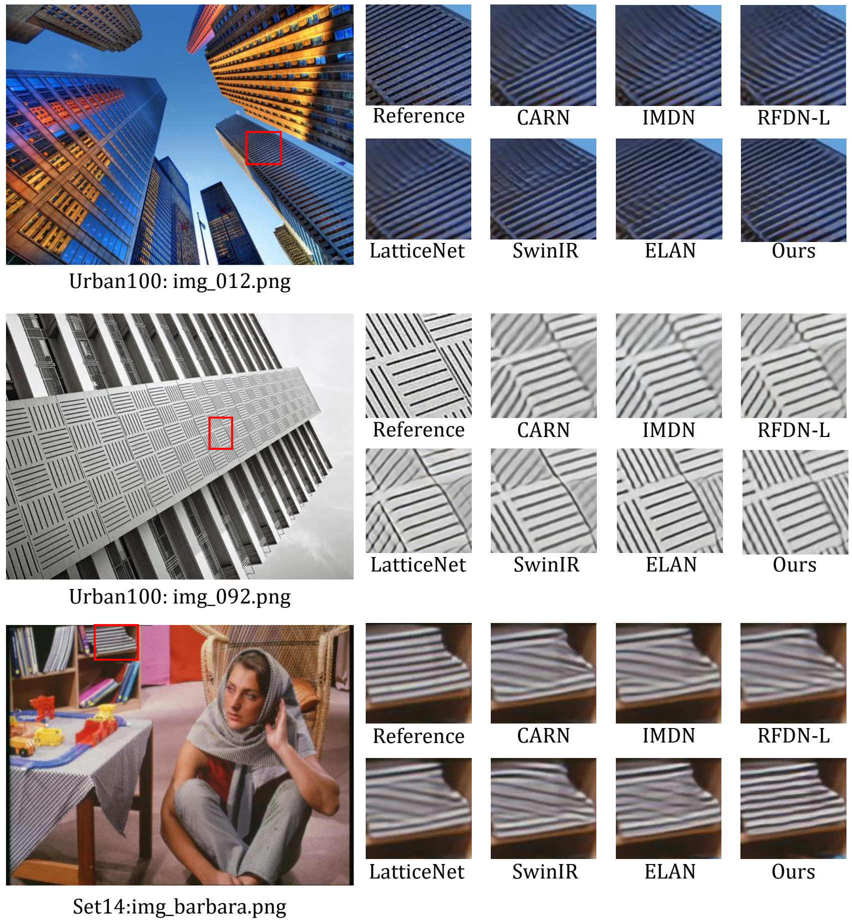


**Fig. 4:** Visual comparison for $\times 4$ SR methods. The patches for comparison are marked with red boxes. (lightweight image SR)

model size, the performance of our DWT surpasses existing methods with a notable margin on all benchmarks. In particular, compared to other transformer architectures with comparable parameters like SwinIR [21] and ELAN [44], as well as recent SR models like DiVNet [2], NGSwin [6], and STSN [13], the proposed DWT obtains the best performance.

**Visual comparison.** In Fig. 4, we also provide a visual comparison of different lightweight SR methods at $\times 4$ scale. We can observe that the HR images constructed by DWT contain more fine-grained details, while other methods generate blurred edges or artifacts in complicated areas. For example, in the images below, our method is able to more accurately restore the detailed textures of building facades and book covers, whereas all other methods show poorer restoration effects.

**Table 2:** Ablation study. (a) Sq: square window ($sw \times sh$) self-attention; Re: rectangle window ($sw \times sh$) self-attention; (b) D-G: dynamic window-groups self-attention. The ($sw \times sh$) of window-groups ($2 \times 32$, $32 \times 2$), ($4 \times 16$, $16 \times 4$), ($2 \times 32$, $8 \times 8$, $32 \times 2$), ($4 \times 16$, $8 \times 8$, $16 \times 4$), ($2 \times 32$, $4 \times 16$, $16 \times 4$, $32 \times 2$) and ($2 \times 32$, $4 \times 16$, $8 \times 8$, $16 \times 4$, $32 \times 2$) are represented by D-G1, D-G2, D-G3, D-G4, D-G5, D-G6, respectively. (c) $\alpha$ is the decay rate of the temperature parameter $\tau$ in the routing network.

| Network | FLOPs | PSNR | |
|---|---|---|---|
| | | BSD100 | Urban100 |
| Re($2 \times 32$) | 49.61G | 27.67 | 26.46 |
| Re($4 \times 16$) | 49.61G | 27.69 | 26.50 |
| Sq($8 \times 8$) | 49.61G | 27.69 | 26.47 |
| Re($16 \times 4$) | 49.61G | 27.70 | 26.50 |
| Re($32 \times 2$) | 49.61G | 27.67 | 26.46 |

(a)

| Network | FLOPs | PSNR | |
|---|---|---|---|
| | | BSD100 | Urban100 |
| D-G1 | 50.13G | 27.68 | 26.48 |
| D-G2 | 50.13G | 27.72 | 26.53 |
| D-G3 | 50.67G | 27.71 | 26.51 |
| D-G4 | 50.67G | 27.73 | 26.56 |
| D-G5 | 51.18G | 27.72 | 26.53 |
| D-G6 | 51.70G | 27.71 | 26.56 |

(b)

| $\alpha$ | 1 | 0.999998 | 0.99998 | 0.9998 |
|---|---|---|---|---|
| PSNR | 27.63/26.44 | 27.73/26.56 | 27.69/26.50 | 27.66/26.46 |

(c)

### 4.3 Ablation Study

For the ablation study, we first trained the image super-resolution (SR x4) model on the DIV2K [22] dataset using a lightweight configuration. Subsequently, we tested the model on the BSD100 [27] and Urban100 [15] datasets, with the test results shown in Tab. 2. FLOPs are calculated based on an input size of $320 \times 180$.

**The Impact of Window Shape.** Tab. 2 shows that when processing datasets with natural and urban themes, particularly Urban100 [15] dataset which contains many directional and repetitive texture features, using conventional rectangular windows (*e.g.*, $4 \times 16$ or $16 \times 4$) for local self-attention is more effective than using square windows ($8 \times 8$). Without increasing computational complexity (*e.g.*, FLOPs), rectangular windows can improve performance on the Urban100 dataset from 26.47dB to 26.50dB. We also observe that extreme rectangular windows (*e.g.*, $2 \times 32$ or $32 \times 2$) perform poorly on both datasets. This is likely because overly stretched windows are too narrow on one side to capture sufficient information and may even introduce noise. These results indicate that the shape of the window is crucial to performance.

**Dynamic-Window Self-Attention.** Tab. 2 shows that the dynamic window attention mechanism significantly outperforms the traditional fixed window attention mechanism in image restoration tasks. For example, when comparing Re ($4 \times 16$), Re ($16 \times 4$), the performance of D-G2 increases to 27.72dB and 26.53dB on the two datasets, respectively, with only about a 1.9% increase in FLOPs. This small computational increment indicates that the adaptive strategy has a very limited impact on model efficiency. Additionally, by comparing different combinations of the same number of window types (*e.g.*, D-G1: 26.48 vs. D-G2: 26.53), as well as the number of window types (*e.g.*, D-G1: 26.48 vs. D-G4: 26.56), we can observe that both the number and combination of window types significantly affect performance. Despite its higher complexity, D-G6 does not perform as well as D-G4. This discrepancy may arise because the dimensions such as (2,32) and (32,2) are so narrow that they fail to capture sufficient information, possibly introducing noise instead.

**The routing network.** Tab. 2 The results were tested on BSD100/Urban100 datasets. It shows that the optimal performance is achieved when $\alpha$=0.999998. Moreover, both excessive exploration ($\alpha$=1) and premature cessation of exploration ($\alpha$=0.9998) negatively impact the performance of our model. This demonstrates the importance of a balanced exploration strategy.

## 5 Conclusion

In this paper, we propose a novel Swin-Transformer-based model named the Dynamic Window Transformer (DWT) for image super-resolution (SR). Our DWT excels at modeling both local and contextual information, surpassing other Swin-Transformer approaches in terms of detail preservation and visual consistency with the original image. The core component of our DWT is the Dynamic-Window Self-Attention (DWSA), which dynamically selects the optimal window to perform self-attention based on the input. Compared to static window-based self-attention, DWSA significantly improves detail extraction capability with negligible extra computational cost. Furthermore, by incorporating the shifted window mechanism, our model achieves efficient global dependency modeling by exploiting image anisotropy. Experimental results demonstrate the efficacy of the proposed DWT, confirming its effectiveness and indicating its potential for further advancements in SR tasks. However, our method currently supports only window configurations of regular shapes, such as squares and rectangles, and does not accommodate more complex shapes. In future work, we aim to extend our method to support a wider variety of window shapes, thereby enhancing the flexibility and applicability of our model.

**Acknowledgments.** This work was supported in part by the Science and Technology-Based Small and Medium-Sized Enterprise Innovation Capacity Enhancement Project of Shandong Province under Grant 2023TSGC0823.

**Disclosure of Interests.** The authors have no competing interests to declare that are relevant to the content of this article.

## References

1. Ahn, N., Kang, B., Sohn, K.A.: Fast, accurate, and lightweight super-resolution with cascading residual network. In: Proceedings of the European conference on computer vision (ECCV). pp. 252–268 (2018)

2. Behjati, P., Rodriguez, P., Fernández, C., Hupont, I., Mehri, A., Gonzalez, J.: Single image super-resolution based on directional variance attention network. Pattern Recognition **133**, 108997 (2023)
3. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding (2012)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
5. Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., Gao, W.: Pre-trained image processing transformer. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12299–12310 (2021)
6. Choi, H., Lee, J., Yang, J.: N-gram in swin transformers for efficient lightweight image super-resolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2071–2081 (2023)
7. Dai, T., Cai, J., Zhang, Y., Xia, S.T., Zhang, L.: Second-order attention network for single image super-resolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11065–11074 (2019)
8. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13. pp. 184–199. Springer (2014)
9. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence **38**(2), 295–307 (2015)
10. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14. pp. 391–407. Springer (2016)
11. Dong, X., Bao, J., Chen, D., Zhang, W., Yu, N., Yuan, L., Chen, D., Guo, B.: Cswin transformer: A general vision transformer backbone with cross-shaped windows. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12124–12134 (2022)
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
13. Gendy, G., Sabor, N., Hou, J., He, G.: A simple transformer-style network for lightweight image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1484–1494 (2023)
14. Gu, Y., Wang, L., Wang, Z., Liu, Y., Cheng, M.M., Lu, S.P.: Pyramid constrained self-attention network for fast video salient object detection. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 10869–10876 (2020)
15. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5197–5206 (2015)
16. Hui, Z., Gao, X., Yang, Y., Wang, X.: Lightweight image super-resolution with information multi-distillation network. In: Proceedings of the 27th acm international conference on multimedia. pp. 2024–2032 (2019)
17. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1646–1654 (2016)

18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
19. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4681–4690 (2017)
20. Li, W., Zhou, K., Qi, L., Jiang, N., Lu, J., Jia, J.: Lapar: Linearly-assembled pixel-adaptive regression network for single image super-resolution and beyond. Advances in Neural Information Processing Systems **33**, 20343–20355 (2020)
21. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1833–1844 (2021)
22. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 136–144 (2017)
23. Liu, J., Tang, J., Wu, G.: Residual feature distillation network for lightweight image super-resolution. In: Computer vision–ECCV 2020 workshops: Glasgow, UK, August 23–28, 2020, proceedings, part III 16. pp. 41–55. Springer (2020)
24. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
25. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
26. Luo, X., Xie, Y., Zhang, Y., Qu, Y., Li, C., Fu, Y.: Latticenet: Towards lightweight image super-resolution with lattice block. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16. pp. 272–289. Springer (2020)
27. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings eighth IEEE international conference on computer vision. ICCV 2001. vol. 2, pp. 416–423. IEEE (2001)
28. Matsui, Y., Ito, K., Aramaki, Y., Fujimoto, A., Ogawa, T., Yamasaki, T., Aizawa, K.: Sketch-based manga retrieval using manga109 dataset. Multimedia tools and applications **76**, 21811–21838 (2017)
29. Mei, Y., Fan, Y., Zhou, Y.: Image super-resolution with non-local sparse attention. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3517–3526 (2021)
30. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018)
31. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1874–1883 (2016)
32. Tai, Y., Yang, J., Liu, X.: Image super-resolution via deep recursive residual network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3147–3155 (2017)
33. Timofte, R., Agustsson, E., Van Gool, L., Yang, M.H., Zhang, L.: Ntire 2017 challenge on single image super-resolution: Methods and results. In: Proceedings of the

IEEE conference on computer vision and pattern recognition workshops. pp. 114–125 (2017)
34. Tong, T., Li, G., Liu, X., Gao, Q.: Image super-resolution using dense skip connections. In: Proceedings of the IEEE international conference on computer vision. pp. 4799–4807 (2017)
35. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International conference on machine learning. pp. 10347–10357. PMLR (2021)
36. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
37. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 568–578 (2021)
38. Wang, X., Chen, X., Ni, B., Tong, Z., Wang, H.: Learning continuous depth representation via geometric spatial aggregator. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 2698–2706 (2023)
39. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., Change Loy, C.: Esrgan: Enhanced super-resolution generative adversarial networks. In: Proceedings of the European conference on computer vision (ECCV) workshops. pp. 0–0 (2018)
40. Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H.: Uformer: A general u-shaped transformer for image restoration. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 17683–17693 (2022)
41. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. Advances in neural information processing systems **34**, 12077–12090 (2021)
42. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Curves and Surfaces: 7th International Conference, Avignon, France, June 24-30, 2010, Revised Selected Papers 7. pp. 711–730. Springer (2012)
43. Zhang, K., Zuo, W., Gu, S., Zhang, L.: Learning deep cnn denoiser prior for image restoration. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3929–3938 (2017)
44. Zhang, X., Zeng, H., Guo, S., Zhang, L.: Efficient long-range attention network for image super-resolution. In: European conference on computer vision. pp. 649–667. Springer (2022)
45. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European conference on computer vision (ECCV). pp. 286–301 (2018)
46. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2472–2481 (2018)