CUST : Clustered Unit-level Similarity Transformer for Lightweight Image Super-Resolution

Jeongsoo Kim[1]

Independent Researcher
jeongskim512@gmail.com

Abstract. Recently, Vision Transformer (ViT)-based models have exhibited remarkable performance in image super-resolution. However, the quadratic computational complexity of ViTs with respect to spatial resolution severely constrains their efficiency, leading to high latency and massive memory consumption. To alleviate this, various window-based attention mechanisms have been proposed; yet, they inherently compromise the long-range dependency modeling that is the primary advantage of ViTs. To overcome these limitations, we propose the Clustered Unit-level Similarity Transformer (CUST), a novel architecture that efficiently integrates global and local information. Specifically, CUST enables each patch to aggregate and attend to similar patches within a broadened regional scope outside its local window, thereby capturing extensive contextual understanding. Furthermore, it employs overlapping attention windows to capture local dependencies, while explicitly extracting high-frequency details by computing the residual difference between the original features and their downsampled-upsampled counterparts. Comprehensive experiments demonstrate that our proposed model achieves a practical balance between computational efficiency and restoration performance. It achieves a lower memory footprint and faster inference speed compared to recent global context or lightweight models under realistic constraints. Code is available at `https://github.com/jwgdmkj/CUST`.

Keywords: Single Image Super-Resolution · Vision Transformer · Efficient Image Super-Resolution

1 Introduction

Single Image Super-Resolution (SISR), which aims to reconstruct a HR (High-Resolution) image from a given LR (Low-Resolution) counterpart, is a fundamental and actively researched area in computer vision. SISR has been widely applied in various real-world scenarios, ranging from medical imaging requiring precise diagnosis and digital photography for restoring old photos to high-magnification digital zoom in smartphones [14, 24].

Since the pioneering work of SRCNN [8], which first introduced Convolutional Neural Networks (CNNs) to SISR, numerous CNN-based models have

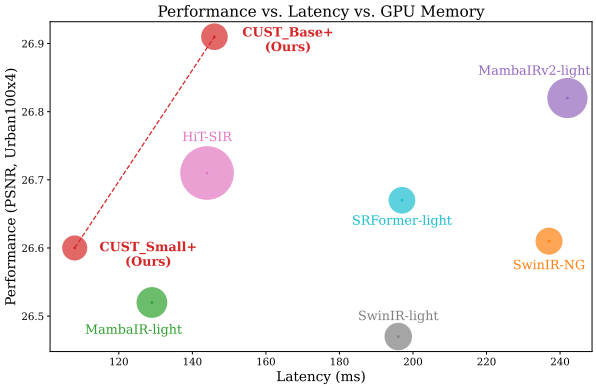

Fig. 1: Efficiency-Performance trade-off analysis on Urban100 (×4).

been proposed [7, 20, 28, 58]. However, CNNs suffer from a fundamental limitation regarding their restricted receptive fields, making it difficult to capture long-range dependencies effectively. Although various approaches have attempted to overcome this issue by stacking deeper layers [26,31,59], this strategy often leads to increased computational costs and model complexity [42].

Following the remarkable success of the attention mechanism in Natural Language Processing (NLP) [46], various models adopting this paradigm have emerged in the computer vision domain [10, 11, 35, 47, 50]. The attention mechanism effectively captures long-range dependencies, addressing the shortcomings of CNNs. However, Vision Transformers (ViTs) suffer from quadratic computational complexity relative to input resolution [5, 6, 21, 55]. To alleviate this, Window-based Self-Attention (WSA) divides images into fixed-size windows [30, 62]; however, this restricts the receptive field and undermines the capture of long-range dependencies [4, 19, 25, 29]. Subsequent attempts to mitigate these limitations—such as generating super-tokens [34, 52] or employing large-window lightweighting strategies [36, 44, 54]—introduce new bottlenecks. Super-token methods are often hindered by slow iterative generation or memory-intensive similarity calculations, while many lightweight models prioritize parameter or FLOPs reduction over practical hardware metrics like inference latency and peak memory usage. Consequently, achieving a practical resource-aware Image Super-Resolution (SR) suitable for resource-constrained deployment remains a significant challenge.

To address these challenges, we propose the Clustered Unit-level Similarity Transformer (CUST). Our CUST is composed of the Cross-window Affinity Neighbor Attention (CANA) module, which aggregates and operates on semantically similar patches beyond window boundaries, and the Multi-frequency Error-driven Dense Attention (MEDA) module, which restores high-frequency details and reinforces local information through multi-scale error signals. Specifically, CANA aggregates semantically similar patches across window boundaries by sorting and clustering them based on their affinity to pooled window tokens.

This mechanism enables the model to effectively capture long-range context beyond fixed constraints.

While CANA provides a long-range context, the subsequent MEDA module restores intricate local fidelity by extracting high-frequency error signals from multi-scale feature differences. These errors, refined via dilated convolutions, explicitly guide overlap window attention toward salient structures like edges while simultaneously expanding the receptive field. By capturing diverse information beyond restricted window areas, CUST achieves superior restoration performance with a lower memory footprint than super-token models and enhanced hardware efficiency compared to conventional WSA-based designs. This dual emphasis on performance and hardware efficiency ensures practical utility for real-world hardware deployment, as illustrated in Fig. 1.

Our extensive experiments demonstrate that the proposed CUST architecture achieves a favorable balance between model complexity and restoration accuracy. Compared to other state-of-the-art window-based attention (WSA) models [30,62], CUST-Base captures a significantly broader spatial context while reducing average inference latency by 26.7%. Furthermore, in comparison to CATANet [34], our model achieves an average of 83% memory reduction across all scales while maintaining comparable inference speeds. Notably, it delivers an average PSNR improvement of 0.094 dB across all benchmark datasets at $\times 4$ scale, demonstrating practical efficiency and restoration fidelity simultaneously.

2 Related Works

Single Image Super-Resolution (SISR) is a fundamental research problem in the field of computer vision. With the advancement of deep learning, diverse architectures have been introduced. Since SRCNN [8] pioneered end-to-end high-resolution reconstruction using a three-layer CNN, subsequent models such as IRCNN [53] and VDSR [26] have emerged, significantly outperforming traditional interpolation methods. However, CNN-based models inherently suffer from restricted receptive fields, limiting their ability to exploit global context. To address this, attention mechanisms were adopted for SISR. IPT [3] demonstrated superiority over CNNs. However, standard self-attention suffers from quadratic complexity relative to input resolution. SwinIR [30] mitigated this through Window-based Self-Attention (WSA), yet restricted the receptive field to local windows, failing to fully exploit global context. To overcome these local constraints, we propose a novel architecture that groups windows into broader search regions and employs a similarity-based cross-window clustering mechanism to capture long-range dependencies efficiently.

Efficient Image Super-Resolution aims to improve the computational efficiency of high-performance SR models. Although deep learning-based SISR has achieved remarkable performance advancements, the accompanying surge in computational cost hinders its deployment on resource-constrained hardware environments. FSRCNN [9] pioneered this effort by replacing the bicubic interpolation in SRCNN [8] with a deconvolution layer, significantly accelerating

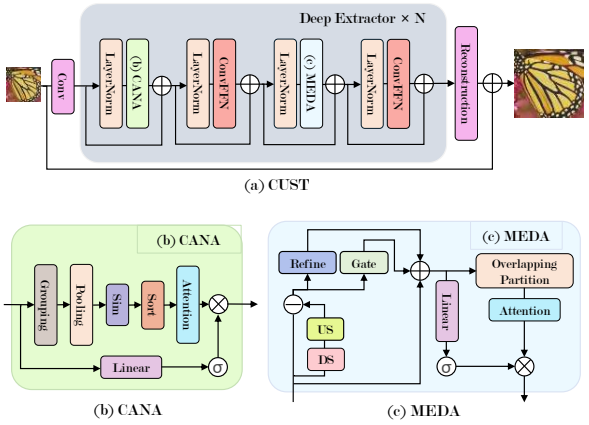

Fig. 2: The architecture of CUST. (a) The overall framework of CUST. (b) The structure of Cross-window Affinity Neighbor Attention (CANA) module. (c) The structure of Multi-frequency Error-driven Dense Attention (MEDA) module.

inference speed. ELAN [57] utilized group-wise multi-scale attention. SPIN [52] adopted superpixel-based interactions. CATANet [34] efficiently captured broad-range attention by generating tokens that represent global image regions. While many state-of-the-art models offer lightweight variants via hyperparameter tuning [30, 44, 62], they often prioritize parameter reduction over practical metrics such as inference latency and peak GPU memory footprint. Moreover, existing token-generation methods frequently encounter efficiency bottlenecks due to iterative computations. In contrast, we demonstrate that refining high-frequency details through multi-scale error signals allows for a reduction in window size and memory footprint while maintaining or even enhancing restoration performance.

3 Method

3.1 Overall Architecture

Figure 2(a) illustrates the overall architecture of our proposed **Clustered Unit-level Similarity Transformer (CUST)**. Given an input low-resolution (LR) image $I_{LR} \in \mathbb{R}^{H\times W\times 3}$, a 3×3 convolutional layer first extracts shallow features $F_0 \in \mathbb{R}^{H\times W\times C}$. Subsequently, these features are passed through a series of deep feature extractor blocks to extract deep-level features. Each deep feature extractor block comprises a Cross-window Affinity Neighbor Attention (CANA) module, a Multi-frequency Error-driven Dense Attention (MEDA) module, two Convolutional Feed-Forward Network (ConvFFN) modules, and four layer normalization [1] layers. Specifically, the CANA module performs a similarity-based clustering operation to interact with semantically related patches beyond window boundaries, capturing long-range dependencies. In contrast, the MEDA module focuses on reinforcing local correlations and restoring high-frequency details using multi-scale error signals. Finally, the deep features are transformed into a high-quality image through a reconstruction module [41], and the final high-resolution output is obtained by adding the global residual of the input

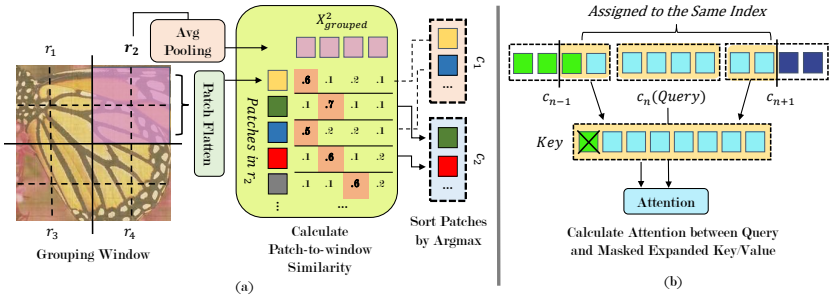

Fig. 3: Illustration of the proposed Cross-window Affinity Neighbor Attention (CANA) mechanism. (a) Patches within a search region are clustered based on their similarity to pooled window tokens. (b) To prevent information disconnection at cluster boundaries, Key and Value sets are expanded by incorporating patches from adjacent clusters that share the same window token affinity.

I_{LR}. Detailed mechanisms of the CANA and MEDA modules are elaborated in Section 3.2 and Section 3.3, respectively.

3.2 CANA : Cross-window Affinity Neighbor Attention

Our proposed Cross-window Affinity Neighbor Attention (CANA) is illustrated in detail in Fig. 2(b) and Fig. 3(a).

Representative Token Generation. First, given a feature map $I \in \mathbb{R}^{H\times W\times C}$, we partition it into multiple search regions $\mathcal{R} = \{r_1, r_2, \ldots, r_M\}$ with a size of $GS \times GS$, where the grid size GS is a multiple of the window size ws. Within each search region $r_m \in \mathbb{R}^{GS\times GS\times C}$, we generate Representative Window Tokens $X^m_{grouped} \in \mathbb{R}^{N\times C}$ to summarize the local context of each window. Here, we apply a non-overlapping average pooling operation with a kernel size and stride of ws, to downsample each $ws \times ws$ window into a 1×1 token. This formulation is expressed as Eq. 1:

$$X^m_{grouped} = \text{AvgPool}_{ws}(r_m) \in \mathbb{R}^{N\times C} \tag{1}$$

where $N = (GS/ws)^2$ denotes the number of partitioned windows (and thus the number of pooled window tokens) within the search region.

Similarity-based Clustering. To establish a robust semantic grouping mechanism, we adapt the similarity-based patch clustering paradigm of CATANet [34] into our window-group-level framework. Specifically, we compute the dot-product similarity between all individual spatial patches within r_m and the representative window tokens $X^m_{grouped}$. Through an Argmax operation along the token dimension, each patch is assigned to the index of the window representative that yields the highest similarity. By gathering patches that share the same index through an Argsort operation, a cluster $\mathcal{C}_n$ sharing the identical index is formed.

Through this, similar patches scattered globally become adjacent to each other. Because the total number of mutually related patches across the search region can often exceed the predefined capacity of a single cluster, some highly relevant features are partitioned into neighboring clusters. To prevent such information disconnection at the cluster boundaries and to expand the receptive field, we extend the Key and Value regions, as illustrated in Fig. 3(b). Specifically, when using the current cluster $\mathcal{C}_n$ as the Query, we fetch the nearest halves of the patches from the adjacent clusters $\mathcal{C}_{n-1}$ and $\mathcal{C}_{n+1}$ to expand the Key and Value sets. Since the entire sequence is sorted by semantic affinity, the index-adjacent clusters $\mathcal{C}_{n-1}$ and $\mathcal{C}_{n+1}$ represent the closest semantic matches rather than mere spatial neighbors. Let $\mathcal{C}^{rear}$ and $\mathcal{C}^{front}$ denote the second and first half of a cluster, respectively. The expanded set is formulated as Eq. 2:

$$\mathbf{K}_{exp}, \mathbf{V}_{exp} = \text{Concat}(\mathcal{C}_{n-1}^{rear}, \mathcal{C}_n, \mathcal{C}_{n+1}^{front}) \tag{2}$$

Attention with Masking. To exclude interactions with semantically unrelated patches among the merged candidates, we apply an attention masking strategy. The attention mask $\mathbf{M}$ is defined such that it takes a value of 0 when a key token j shares the same window ID as the query token i—specifically, when both tokens identify the same window as their most similar neighbor—and $-\infty$ otherwise. This effectively restricts references between patches that do not share the same window ID. Finally, the features are adaptively modulated through a learned gate $\mathbf{G}_n$ [40], and the overall operation is formulated as follows:

$$\mathbf{Y}_n = \text{GELU}\left(\text{Softmax}\left(\frac{\mathbf{Q}_n\mathbf{K}_{exp}^T}{\sqrt{d}} + \mathbf{M}\right)\mathbf{V}_{exp}\right) \odot \mathbf{G}_n \tag{3}$$

This process makes each patch overcomes the constraints of a fixed local window and interacts with semantically similar patches, thereby effectively capturing long-range dependencies.

3.3 MEDA : Multi-frequency Error-driven Dense Attention

While CANA captures global long-range dependencies by overcoming window constraints, the subsequent MEDA module refines intricate local details within the window. The detailed architecture and algorithm of the module are illustrated in Fig. 2(c) and Algorithm 1, respectively. Here, the Overlap Self Attention in Algorithm 1 is adopted from HPI-Net [32].

Motivation. Generally, in window-based attention mechanisms, reducing the window size restricts the receptive field, which leads to performance degradation. To mitigate this, various models have employed overlapping windows to facilitate information exchange across boundaries [32, 34, 52]. However, these methods suffer from a sharp increase in computational overhead as the window size and overlap area expand. To address this, we adopt overlapping window

Algorithm 1 Multi-frequency Error-driven Dense Attention (MEDA)

Require: Input feature $\mathbf{X} \in \mathbb{R}^{H \times W \times C}$
Ensure: Refined feature $\mathbf{Y}$
Step 1: Frequency-aware Error Extraction
$\mathbf{X}_{low} \leftarrow \text{Up}(\text{Down}(\mathbf{X}))$ ▷ Down-sampling and Up-sampling
$\mathbf{X}_{error} \leftarrow \mathbf{X} - \mathbf{X}_{low}$ ▷ Extract High-frequency Residuals
Step 2: Error Refinement & Gating
$\mathbf{F}_{refine} \leftarrow \text{Conv}_{1\times1}(\text{GELU}(\text{DilConv}_{3\times3}(\mathbf{X}_{error})))$ ▷ Dilated Context Aggregation
$\mathbf{G} \leftarrow \sigma(\text{Conv}_{1\times1}(\mathbf{X}_{error}))$ ▷ Generate Spatial Gating Map
Step 3: Feature Compensation
$\mathbf{X}_{refined} \leftarrow \mathbf{X} + \alpha \cdot (\mathbf{F}_{refine} \odot \mathbf{G})$ ▷ $\odot$: Element-wise Product
Step 4: Local Detail Refinement
$\mathbf{Y} \leftarrow \text{OverlapSelfAttention}(\mathbf{X}_{refined})$
return $\mathbf{Y}$

attention while proposing an efficient information correction method that leverages high-frequency refinement based on multi-scale errors, even within limited window regions. The proposed MEDA is formulated as shown in Eq. 4.

$$\mathbf{X_{refined}} = \mathbf{X} + \alpha \cdot (\text{Refiner}(\mathbf{X}_{error}) \odot \text{Gate}(\mathbf{X}_{error})) \tag{4}$$

Algorithm. The MEDA module is computed through the following procedure. First, given the input feature $\mathbf{X}$ from CANA, the error feature $\mathbf{X}_{error}$ is generated by calculating the residual between $\mathbf{X}$ and its downsampled-then-upsampled version: $\mathbf{X}_{error} = \mathbf{X} - \text{Up}(\text{Down}(\mathbf{X}))$. This error feature captures high-frequency information, serving as an explicit guide that highlights essential edge and texture locations for reconstruction. Next, $\mathbf{X}_{error}$ is processed through the **Error Refiner** and the **Spatial Gate**. The **Error Refiner**, composed of dilated convolutions, transforms simple pixel-wise differences into semantically rich contextual features. By enlarging the receptive field with minimal computational overhead, it generates sophisticated error features that account for neighboring texture flows. Simultaneously, the **Spatial Gate**, implemented with 1×1 convolutions, determines the specific locations and intensities where corrections are required. The resulting refined error is multiplied by a learnable scaler α and added to the original feature $\mathbf{X}$, providing guidance on which regions require focused reconstruction. Finally, overlap window-based self-attention is applied to further refine local details and enhance image quality.

4 Experiments

4.1 Settings

Datasets. Following previous works [13,37,43], we utilize the DIV2K [45] dataset for training. The DIV2K dataset consists of 800 training images and 100 validation images, and is widely adopted as a standard benchmark for SISR tasks. For

Table 1: Quantitative comparison of CUST-Base and CUST-Base+ with state-of-the-art lightweight SISR models. The best and second-best performances are highlighted in red and blue, respectively.

Scale	Method	#Params	#FLOPs	Set5	Set14	B100	Urban100	Manga109
×2	SwinIR-light [30]	910K	244G	38.14/0.9611	33.86/0.9206	32.31/0.9012	32.76/0.9340	39.12/0.9783
	ELAN-light [57]	621K	203G	38.17/0.9611	33.94/0.9207	32.30/0.9012	32.76/0.9340	39.11/0.9782
	HPI-Net [32]	783K	429G	38.12/0.9605	33.94/0.9209	32.31/0.9013	32.85/0.9346	39.08/0.9771
	SwinIR-NG [6]	1,181K	274G	38.17/0.9612	33.94/0.9205	32.31/0.9013	32.78/0.9340	39.20/0.9781
	SRformer-Light [62]	853K	236G	38.23/0.9613	33.94/0.9209	32.36/0.9019	32.91/0.9353	39.28/0.9785
	SPIN [52]	497K	114G	38.20/0.9615	33.90/0.9215	32.31/0.9015	32.79/0.9340	39.18/0.9784
	OSFFNet [48]	516K	83G	38.11/0.9610	33.72/0.9190	32.29/0.9012	32.67/0.9331	39.09/0.9780
	HIT-SIR [56]	772K	210G	38.22/0.9613	33.91/0.9213	32.35/0.9019	33.02/0.9365	39.38/0.9782
	SMFANet+ [61]	480K	108G	38.19/0.9611	33.92/0.9207	32.32/0.9015	32.70/0.9331	39.46/0.9787
	MambaIR-light [18]	847K	227G	38.13/0.9610	33.95/0.9208	32.31/0.9013	32.85/0.9349	39.20/0.9782
	MambaIRv2-light [17]	774K	286G	38.26/0.9615	34.09/0.9221	32.36/0.9019	33.26/0.9378	39.35/0.9785
	CATANet [34]	477K	135G	38.28/0.9617	33.99/0.9217	32.37/0.9023	33.09/0.9372	39.37/0.9784
	CUST-Base(Ours)	682K	291G	38.31/0.9617	34.08/0.9221	32.40/0.9026	33.25/0.9375	39.47/0.9785
	CUST-Base+(Ours)	682K	314G	38.32/0.9617	34.10/0.9228	32.40/0.9024	33.21/0.9377	39.49/0.9785
×3	SwinIR-light [30]	918K	111G	34.62/0.9289	30.54/0.8463	29.20/0.8082	28.66/0.8624	33.98/0.9478
	ELAN-light [57]	629K	90G	34.61/0.9288	30.55/0.8463	29.21/0.8081	28.69/0.8624	34.00/0.9478
	HPI-Net [32]	924K	277G	34.70/0.9289	30.63/0.8480	29.26/0.8104	28.93/0.8675	34.31/0.9487
	SwinIR-NG [6]	1,190K	114G	34.64/0.9293	30.58/0.8471	29.24/0.8090	28.75/0.8639	34.22/0.9488
	SRformer-Light [62]	861K	105G	34.67/0.9296	30.57/0.8469	29.26/0.8099	28.81/0.8655	34.19/0.9489
	SPIN [52]	569K	58G	34.65/0.9293	30.57/0.8464	29.23/0.8089	28.71/0.8627	34.24/0.9489
	OSFFNet [48]	524K	38G	34.58/0.9287	30.48/0.8450	29.21/0.8080	28.49/0.8595	34.00/0.9472
	HIT-SIR [56]	780K	94G	34.72/0.9298	30.62/0.8474	29.27/0.8101	28.93/0.8673	34.40/0.9496
	SMFANet+ [61]	487K	48G	34.66/0.9292	30.57/0.8461	29.25/0.8090	28.67/0.8611	34.45/0.9490
	MambaIR-light [18]	913K	149G	34.63/0.9288	30.54/0.8459	29.23/0.8084	28.70/0.8631	34.12/0.9479
	MambaIRv2-light [17]	781K	127G	34.71/0.9298	30.68/0.8483	29.26/0.8098	29.01/0.8689	34.41/0.9497
	CATANet [34]	550K	60G	34.75/0.9300	30.67/0.8481	29.28/0.8101	29.04/0.8689	34.40/0.9499
	CUST-Base(Ours)	754K	138G	34.76/0.9303	30.73/0.8488	29.34/0.8116	29.11/0.8699	34.54/0.9497
	CUST-Base+(Ours)	754K	147G	34.79/0.9303	30.71/0.8485	29.33/0.8114	29.10/0.8698	34.57/0.9500
×4	SwinIR-light [30]	930K	64G	32.44/0.8976	28.77/0.7858	27.69/0.7406	26.47/0.7980	30.92/0.9151
	ELAN-light [57]	640K	54G	32.43/0.8975	28.78/0.7858	27.69/0.7406	26.54/0.7982	30.92/0.9150
	HPINet [32]	896K	137G	32.60/0.8986	28.87/0.7874	27.73/0.7419	26.71/0.8043	31.19/0.9161
	SwinIR-NG [6]	1,201K	63G	32.44/0.8980	28.83/0.7870	27.73/0.7418	26.61/0.8010	31.09/0.9161
	SRformer-Light [62]	873K	63G	32.51/0.8988	28.82/0.7872	27.73/0.7422	26.67/0.8032	31.17/0.9165
	SPIN [52]	555K	42G	32.48/0.8983	28.80/0.7862	27.70/0.7415	26.55/0.7998	30.98/0.9156
	OSFFNet [48]	537K	22G	32.39/0.8976	28.75/0.7852	27.66/0.7393	26.36/0.7950	30.84/0.9125
	HIT-SIR [56]	792K	54G	32.51/0.8991	28.84/0.7873	27.73/0.7424	26.71/0.8045	31.23/0.9176
	SMFANet+ [61]	496K	28G	32.51/0.8985	28.87/0.7872	27.74/0.7412	26.56/0.7976	31.29/0.9163
	MambaIR-light [18]	924K	85G	32.42/0.8977	28.74/0.7847	27.68/0.7400	26.52/0.7983	30.94/0.9135
	MambaIRv2-light [17]	790K	76G	32.51/0.8992	28.84/0.7878	27.75/0.7426	26.82/0.8079	31.24/0.9182
	CATANet [34]	535K	34G	32.58/0.8998	28.90/0.7880	27.75/0.7427	26.87/0.8081	31.31/0.9183
	CUST-Base(Ours)	740K	91G	32.70/0.9008	29.00/0.7899	27.81/0.7441	26.92/0.8085	31.45/0.9190
	CUST-Base+(Ours)	740K	98G	32.59/0.9000	28.94/0.7893	27.81/0.7444	26.91/0.8086	31.48/0.9191

performance evaluation, we employ five standard benchmark datasets: Set5 [2], Set14 [51], BSD100 [38], Urban100 [22], and Manga109 [39].

Model Implementation Details. We design four variants of our model: CUST-Base, CUST-Base+, CUST-Small and CUST-Small+. CUST-Base is configured with 40 channels and 12 blocks, while CUST-Small consists of 30 channels and 8 blocks. For both models, the window sizes in the MEDA modules are cyclically

Table 2: Quantitative comparison of CUST-Small and CUST-Small+ with state-of-the-art efficient SISR models. The best and second-best performances are highlighted in red and blue, respectively.

Scale	Method	#Params	#FLOPs	Set5	Set14	B100	Urban100	Manga109
×2	IMDN [23]	694K	156G	38.00/0.9605	33.63/0.9177	32.19/0.8996	32.17/0.9283	38.88/0.9774
	LatticeNet [37]	756K	170G	38.06/0.9607	33.70/0.9187	32.20/0.8999	32.25/0.9288	38.94/0.9774
	RFDN-L [33]	626K	146G	38.08/0.9606	33.67 /0.9190	32.18/0.8996	32.24/0.9290	38.95/0.9773
	SRPN-Lite [60]	609K	140G	38.10/0.9608	33.70/0.9189	32.25/0.9005	32.26/0.9294	-
	HNCT [13]	357K	82G	38.08/0.9608	33.65/0.9182	32.22/0.9001	32.22/0.9294	38.87/0.9774
	FMEN [12]	748K	172G	38.10/0.9609	33.75/0.9192	32.26/0.9007	32.41/0.9311	38.95/0.9778
	NGswin [6]	990K	140G	38.05/0.9610	33.79/0.9199	32.27/0.9008	32.53/0.9324	38.97/0.9777
	LMLT-Base [25]	652K	158G	38.10/0.9610	33.76/0.9201	32.28/0.9012	32.52/0.9316	39.24/0.9783
	CUST-Small(Ours)	277K	128G	38.19/0.9612	33.88/0.9210	32.31/0.9014	32.73/0.9334	39.15/0.9780
	CUST-Small+(Ours)	277K	140G	38.18/0.9612	33.80/0.9207	32.30/0.9014	32.72/0.9333	39.18/0.9782
×3	IMDN [23]	703K	72G	34.36/0.9270	30.32/0.8417	29.09/0.8046	28.17/0.8519	33.61/0.9445
	LatticeNet [37]	765K	76G	34.40/0.9272	30.32/0.8416	29.10/0.8049	28.19/0.8513	33.63/0.9442
	RFDN-L [33]	633K	66G	34.47/0.9280	30.35/0.8421	29.11/0.8053	28.32/0.8547	33.78/0.9458
	SRPN-Lite [60]	615K	63G	34.47/0.9280	30.38/0.8425	29.16/0.8061	28.22/0.8534	-
	HNCT [13]	363K	38G	34.47/0.9275	30.44/0.8439	29.15/0.8067	28.28/0.8557	33.81/0.9459
	FMEN [12]	757K	77G	34.45/0.9275	30.40/0.8435	29.17/0.8063	28.33/0.8562	33.86/0.9462
	NGswin [6]	1,007K	67G	34.52/0.9282	30.53/0.8456	29.19/0.8078	28.52/0.8603	33.89/0.9470
	LMLT-Base [25]	660K	75G	34.58/0.9285	30.53/0.8458	29.21/0.8084	28.48/0.8581	34.18/0.9477
	CUST-Small(Ours)	317K	62G	34.65/0.9291	30.58/0.8466	29.24/0.8090	28.70/0.8624	34.17/0.9477
	CUST-Small+(Ours)	317K	67G	34.60/0.9289	30.59/0.8464	29.23/0.8089	28.74/0.8630	34.17/0.9479
×4	IMDN [23]	715K	41G	32.21/0.8948	28.58/0.7811	27.56/0.7353	26.04/0.7838	30.45/0.9075
	LatticeNet [37]	777K	44G	32.18/0.8943	28.61/0.7812	27.57/0.7355	26.14/0.7844	30.54/0.9075
	RFDN-L [33]	643K	38G	32.28/0.8957	28.61/0.7818	27.58/0.7363	26.20/0.7883	30.61/0.9096
	SRPN-Lite [60]	623K	36G	32.24/0.8958	28.69/0.7836	27.63/0.7373	26.16/0.7875	-
	HNCT [13]	373K	22G	32.31/0.8957	28.71/0.7834	27.63/0.7381	26.20/0.7896	30.70/0.9112
	FMEN [12]	769K	44G	32.24/0.8955	28.70/0.7839	27.63/0.7379	26.28/0.7908	30.70/0.9107
	NGswin [6]	1,019K	36G	32.33/0.8963	28.78/0.7859	27.66/0.7396	26.45/0.7963	30.80/0.9128
	LMLT-Base [25]	672K	41G	32.38/0.8971	28.79/0.7859	27.70/0.7403	26.44/0.7947	31.09/0.9139
	CUST-Small(Ours)	309K	42G	32.46/0.8982	28.85/0.7862	27.73/0.7411	26.60/0.7995	31.14/0.9145
	CUST-Small+(Ours)	309K	45G	32.44/0.8980	28.85/0.7864	27.73/0.7409	26.60/0.7989	31.12/0.9143

assigned from the set $[12, 14, 16, 18]$ across the consecutive blocks. CUST-Base+ and CUST-Small+ shares the same channel and block configuration as CUST-Base and CUST-Small but utilizes a uniform window size of 18 for all blocks. All models are trained for 5×10^5 iterations with a batch size of 32. We utilize the Adam optimizer [27] with $\beta_1 = 0.9$ and $\beta_2 = 0.99$. For data augmentation, random horizontal flips and 90° rotations are applied. The input patch size for training is fixed at 64×64. The initial learning rate is set to 5×10^{-4} with a warm-up period of 20,000 iterations. Subsequently, the learning rate is managed using the MultiStepLR scheduler, which halves the learning rate at iterations 250,000, 400,000, 450,000, and 475,000. Gradient clipping is applied with a threshold of 0.1.

Evaluation Metrics. To evaluate the quality of the super-resolved images, we employ Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) [49] as the primary metrics. Furthermore, to assess the efficiency of the SR models, we measure GPU memory consumption and inference latency. For inference time, we report the average runtime calculated over 50 randomly selected

Table 3: Efficiency comparison between CUST-Base+ and state-of-the-art models. All evaluations are conducted on an NVIDIA RTX 3090 GPU. SSM-based models are marked with †, while the others are ViT-based models.

Model	Scale ×2		Scale ×3		Scale ×4	
	Mem(M)	Time(ms)	Mem(M)	Time(ms)	Mem(M)	Time(ms)
SwinIR-light [30]	1286.76	920.79	595.74	308.76	350.57	195.89
SwinIR-NG [6]	1236.14	984.24	572.85	365.89	337.68	237.01
SRFormer-light [62]	1184.28	976.84	537.54	330.32	329.08	197.18
SPIN [52]	5799.40	3190.69	1200.02	1296.61	456.64	702.44
HIT-SIR [56]	1803.97	441.74	1464.10	217.60	1331.16	143.74
MambaIR-light [18] †	1695.43	600.75	761.06	235.42	438.25	130.08
MambaIRv2-light [17] †	2823.96	922.30	1250.76	414.67	748.42	241.83
CATANet [34]	6935.17	678.28	3374.99	238.62	1818.34	144.89
CUST-Base+(Ours)	**1211.89**	**613.02**	**540.49**	**243.53**	**328.17**	**146.04**

images with resolutions of 640×360, 427×240, and 320×180 corresponding to $\times 2$, $\times 3$, and $\times 4$ scaling factors, respectively. Peak GPU memory consumption is monitored using `torch.cuda.max_memory_allocated()` in PyTorch.

4.2 Comparisons with State-of-the-Art Methods

Image Reconstruction Comparisons. The comparisons of our proposed CUST with other lightweight or efficient SR models are presented in Table 1 and Table 2. We compare the CUST-Base and CUST-Base+ models against a broad range of competitors, including SwinIR-light [30], ELAN-light [57], HPI-Net [32], SwinIR-NG [6], SRFormer-light [62], SPIN [52], OSFFNet [48], HIT-SIR [56], SMFANet+ [61], MambaIR-light [18], MambaIRv2-light [17], and CATANet [34]. To ensure a fair evaluation under lightweight constraints, all comprehensive benchmark models are constrained to those trained on the DIV2K [45] dataset, while variants utilizing larger joint datasets (*e.g.*, DF2K) or heavy-weight configurations are referenced through their respective lightweight versions (*e.g.*, SwinIR-light [30]). These comparison models are categorized into CNN-based [48, 61], Transformer-based [6, 30, 32, 52, 56, 57, 62], and State-Space Model (SSM)-based [17, 18] architectures. For CUST-Small and CUST-Small+, we evaluate their performance against IMDN [23], LatticeNet [37], RFDN-L [33], SRPN-Lite [60], HNCT [13], FMEN [12], NGSwin [6], and LMLT-Base [25].

The proposed CUST-Base and CUST-Base+ models demonstrate superior performance compared to existing lightweight SR models. In particular, at $\times 3$ and $\times 4$ scales, they achieve the highest performance on all benchmark datasets. Notably, compared to CATANet [34], the second-best performing model, CUST-Base yields an average PSNR improvement of 0.068 dB and 0.094 dB at $\times 3$ and $\times 4$ scales, respectively. Furthermore, it consistently matches or exceeds the performance of state-of-the-art models across various datasets at $\times 2$ scale, demonstrating sustained excellence.

Similarly, the proposed CUST-Small and CUST-Small+ models exhibit outstanding performance compared to other efficient SR models. Especially at $\times 4$

scale, CUST-Small achieves the most superior performance on all datasets, outperforming the next-best model, LMLT-Base [25], by an average of 0.063 dB in PSNR. Beyond this, CUST-Small demonstrates consistent superiority by rivaling or surpassing existing top-performing models at $\times 2$ and $\times 3$ scales. These results confirm that the proposed CUST architecture is not restricted to specific magnification factors but operates robustly across various resolution conditions.

Memory and Running Time Comparisons. To evaluate the efficiency of the proposed CUST, we measure the peak GPU memory footprint and inference latency, comparing it with other ViT-based models [6, 30, 34, 52, 56, 62] and SSM-based models [17, 18]. As presented in Table 3, our model demonstrates a practical trade-off between memory usage and inference speed. Specifically, CUST-Base+ utilizes 82.8% less memory on average across all scales compared to CATANet [34], while maintaining competitive inference latency (146.04 ms vs. 144.89 ms at scale $\times 4$). Furthermore, although our model consumes 0.87% more memory on average than SRFormer-light [62]—the most memory-efficient baseline—it achieves a 29.8% faster inference speed. Compared to other ViT-based models [6,30,52] and the SSM-based MambaIRv2 [17], CUST-Base+ consistently provides a more lightweight and faster solution. These results highlight that our proposed model strikes a favorable balance between performance and practical efficiency. To validate the scalability of our method at larger resolutions, we further analyze the performance on the Test2k [16] dataset in Appendix B, and provide a scaling analysis between FLOPs and inference time in Appendix C.

Qualitative Comparisons. Figure 4 illustrates the superior qualitative performance of the proposed model on the Urban100 [22] dataset compared to other state-of-the-art models [6,18,30,34,62]. As shown in the visual results, our model successfully reconstructs correct geometric orientation and structural integrity, even in dense and repetitive patterns, by accurately capturing the underlying structural context. We further analyze the receptive field of our model using Local Attribution Maps (LAM) [15]. As illustrated in Figure 5, the proposed model references a significantly broader spatial range during inference compared to other ViT-based models [6, 30, 34, 62]. This confirms that our architecture effectively transcends fixed window constraints to capture a wider context.

4.3 Ablation Stuides

Effects of CANA and MEDA. To justify the components of the proposed network, we conduct an ablation study for each module, as presented in Table 4. First, we compare the baseline CUST-Small (**row 1**) with variants where the MEDA and CANA modules are removed (**rows 2 and 3**). Experimental results show that removing MEDA results in a performance drop of 0.29 dB on the Urban100 dataset and 0.34 dB on the Manga109 dataset. The removal of CANA leads to even more significant degradations of 0.39 dB and 0.53 dB, respectively.

Next, to demonstrate the synergy between CANA and MEDA, we compare versions of models using only CANA or MEDA, where the number of blocks

Fig. 4: Qualitative comparison of state-of-the-art SR models at Urban100 ×4 scale. Red boxes indicate the regions selected for detailed comparison.

is increased to 12 to match the parameter count of CUST-Small (**rows 4 and 5**). Our experimental results show that simultaneously utilizing both modules yields a significant synergetic effect compared to using either one in isolation. Specifically, the CANA-only variant exhibits a decrease in PSNR of 0.09 dB and 0.16 dB on Urban100 and Manga109, respectively, compared to the baseline. Similarly, the MEDA-only variant shows a drop of 0.22 dB and 0.33 dB, with SSIM following a consistent trend. The LAM visualization in Figure 6 reveals that CUST-Small utilizes a broader receptive field than the single-module variants. This underscores the importance of capturing long-range dependencies via CANA followed by local detail refinement through MEDA.

Effects of Key Components in CANA and MEDA. We verify the structural soundness and efficacy of CANA and MEDA. Table 5 summarizes the performance variations when replacing or removing key components within each module. First, for CANA, replacing our CANA with standard Swin Attention [35] leads to a performance drop from 26.60 dB to 26.53 dB. Moreover, scaling down search region scale factor $gs = GS/ws$, which represents the number of windows along one dimension of the search region, from the default value of 10 (corre-

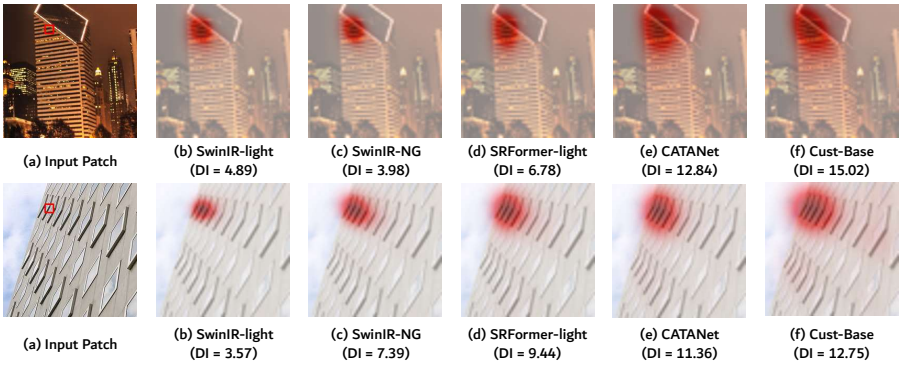

Fig. 5: LAM (Local Attribution Maps) comparison between the proposed model and other ViT-based lightweight SR models. As illustrated, the proposed model utilizes a broader spatial range for image reconstruction compared to other models.

Table 4: Ablation study of CANA and MEDA modules in CUST-Small for scale ×4. The second and third rows show the performance drop when removing each module. The fourth and fifth rows represent networks built exclusively with a single module, scaled to match the parameter count of CUST-Small.

Module	Scaling	#Params	#FLOPs	Urban100	Manga109
CUST-Small (baseline)	-	309K	42G	**26.60/0.7995**	**31.14/0.9145**
Only CANA	✗	227K	29G	26.31/0.7914	30.80/0.9118
Only MEDA	✗	214K	27G	26.21/0.7871	30.61/0.9087
Only CANA	✓	307K	38G	26.51/0.7966	30.98/0.9133
Only MEDA	✓	288K	35G	26.38/0.7926	30.81/0.9113

sponding to a physical grid size of $GS = 80$ under $ws = 8$) to 5 ($GS = 40$), or entirely disabling the Key-Value (KV) Expansion mechanism, degrades the restoration quality, reducing the SSIM by 0.0008 and 0.0010, respectively. As illustrated by the Local Attribution Map (LAM) [15] results in Figure 7, replacing our attention mechanism or altering the group size forces the model to utilize a narrower range of input patches compared to the baseline (CUST-Small). This visual evidence proves that our semantic clustering successfully captures long-range dependencies that are otherwise overlooked by fixed local windows. Next, for MEDA, removing the error-refinement process leads to a decrease of 0.0013 (0.7995 vs. 0.7982) in SSIM, confirming that our explicit error guidance is vital for reconstructing intricate high-frequency details. Additional detailed visualizations of patch clustering in CANA and the high-frequency guidance provided by MEDA are presented in Appendix D.

Effects of Multi-Frequency Error-driven Calculation. To evaluate the efficiency of our Multi-Frequency Error-driven Algorithm (MFA, Algorithm 1: Steps 1-5), we compare CUST-Small against variants that exclude this module across various window sizes leaving only the Overlap Self-Attention (Table 6). Specifically, the

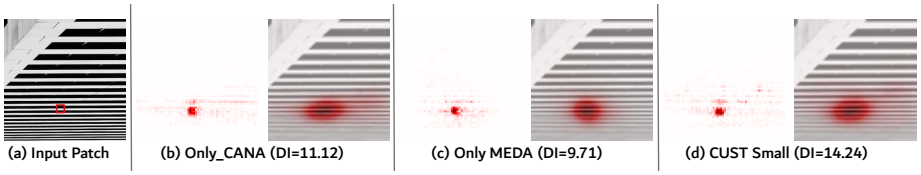

Fig. 6: LAM comparison between CUST-Small and its variants using only the CANA module (b) or only the MEDA module (c). The results demonstrate the synergistic effect achieved when CANA and MEDA are utilized together.

Table 5: Ablation study on the core components of CUST-Small on Urban100 (×4).

Component	Method / Variant	#Params (K)	#FLOPs (G)	Urban100 (PSNR/SSIM)
CANA	SwinAttn instead of CANA	313K	35G	26.53/0.7975
	w/o Expansion	309K	39G	26.58/0.7985
	Reduced group size (10 → 5)	309K	42G	26.59/0.7987
MEDA	w/o Refiner & Gate	296K	41G	26.58/0.7982
Baseline	**CUST-Small (Full)**	**309K**	**42G**	**26.60/0.7995**

odd-indexed rows represent variants equipped with MFA under downscaled window configurations, whereas the even-indexed rows denote counterparts without MFA under upscaled window environments to set comparative upper bounds.

Experimental results confirm that our MFA effectively enhances reconstruction performance beyond restricted window boundaries, consequently contributing to a reduction in GPU memory overhead. As summarized in Table 6, integrating MFA consistently yields superior performance-efficiency trade-offs under varying window configurations. Remarkably, variants with smaller windows but equipped with MFA consistently outperform versions with a window size larger by 2, while maintaining a lower GPU memory footprint. For instance, *WS +2 w/ MFA* (**row 1**) achieves an 8.3% lower GPU memory usage alongside a higher PSNR compared to *WS +4 w/o MFA* (**row 2**) (26.64 dB vs. 26.62 dB). Fur-

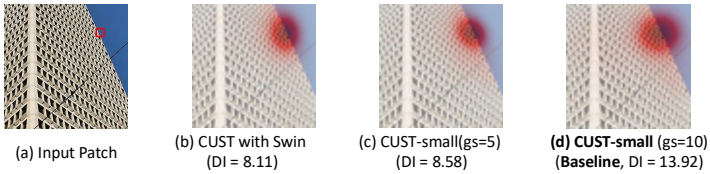

Fig. 7: Visual analysis using Local Attribution Map (LAM). (a) A query patch from Urban100. (b)-(d) LAM results for different configurations. Our baseline (gs=10) shows a significantly higher Diffusion Index (DI = 13.92) compared to the Swin-based counterpart (DI = 8.11), indicating that our model effectively utilizes a broader range of spatial information for high-quality image reconstruction.

Table 6: Performance comparison of CUST-Small with varying window sizes (WS) and the Multi-Frequency Algorithm (MFA, Algorithm 1: Steps 1-5) at scale ×4 on Urban100.

Method	Window size	#GPU Mem [M]	#FLOPs	Urban100 (PSNR/SSIM)
WS +2 w/ MFA	[14,16,18,20]	**314.6M**	**44G**	**26.64/0.8002**
WS +4 w/o MFA	[16,18,20,22]	343.2M	45G	26.62/0.7994
CUST-Small (baseline)	[12,14,16,18]	**291.6M**	**42G**	26.60/**0.7995**
WS +2 w/o MFA	[14,16,18,20]	307.7M	43G	**26.61**/0.7989
WS -2 w/ MFA	[10,12,14,16]	**248.5M**	**40G**	**26.60/0.7989**
No size change w/o MFA	[12,14,16,18]	284.9M	41G	26.58/0.7983

thermore, a more aggressive reduction layout, such as *WS -2 w/ MFA* (**row 5**), achieves a highly comparable PSNR of 26.60 dB and SSIM of 0.7989 against the *WS +2 w/o MFA* (**row 4**) variant (26.61 dB / 0.7989), while drastically reducing the peak GPU memory footprint by 19.2% (248.5M vs. 307.7M). These findings explicitly demonstrate that the multi-frequency algorithm not only compensates for the inherent spatial constraints of window-based architectures but also drastically improves practical hardware memory efficiency. A more extensive comparative analysis regarding the trade-offs between window size reduction and the integration of our proposed algorithm is provided in Appendix E.

5 Conclusion and Limitation

In this paper, we propose the CUST (Clustered Unit-level Similarity Transformer) for lightweight image super-resolution. By grouping windows into search regions and clustering patches based on their affinity to window tokens, CUST effectively captures long-range dependencies while maintaining high computational efficiency. Furthermore, our MEDA module explicitly guides the reconstruction of high-frequency details through multi-scale error signals, expanding the receptive field without an excessive memory burden. Experimental results confirm that CUST achieves a superior balance between restoration quality, inference latency, and memory footprint compared to existing WSA and super-token-based models.

Despite these contributions, our work has certain limitations. First, while CANA effectively captures global context, its clustering efficacy may diminish in regions with highly stochastic or irregular textures where patch-to-window affinity becomes ambiguous. Second, although CUST is optimized for SISR, its generalizability to other low-level vision tasks—such as deblurring and denoising—remains to be fully explored. Future work will focus on enhancing the model's robustness across a wider range of degradation types and diverse vision tasks to further validate its practical utility for real-world hardware deployment.

References

1. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)
2. Bevilacqua, M., Roumy, A., Guillemot, C., Morel, M.L.A.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In: British machine vision conference (BMVC) (2012)
3. 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)
4. Chen, X., Wang, X., Zhou, J., Qiao, Y., Dong, C.: Activating more pixels in image super-resolution transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 22367–22377 (June 2023)
5. Chen, Z., Zhang, Y., Gu, J., Kong, L., Yuan, X., et al.: Cross aggregation transformer for image restoration. Advances in Neural Information Processing Systems **35**, 25478–25490 (2022)
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 (CVPR). 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: European conference on computer vision. pp. 184–199. Springer (2014)
9. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: European conference on computer vision. pp. 391–407. Springer (2016)
10. 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 Computer Vision and Pattern Recognition (CVPR) pp. 12114–12124 (2021)
11. 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)
12. Du, Z., Liu, D., Liu, J., Tang, J., Wu, G., Fu, L.: Fast and memory-efficient network towards efficient image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 853–862 (2022)
13. Fang, J., Lin, H., Chen, X., Zeng, K.: A hybrid network of cnn and transformer for lightweight image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1103–1112 (2022)
14. Fang, W., Tang, Y., Guo, H., Yuan, M., Mok, T.C.W., Yan, K., Yao, J., Chen, X., Liu, Z., Lu, L., Zhang, L., Xu, M.: Cycleinr: Cycle implicit neural representation for arbitrary-scale volumetric super-resolution of medical data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11631–11641 (June 2024)
15. Gu, J., Dong, C.: Interpreting super-resolution networks with local attribution maps. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9199–9208 (2021)

16. Gu, S., Lugmayr, A., Danelljan, M., Fritsche, M., Lamour, J., Timofte, R.: Div8k: Diverse 8k resolution image dataset. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW). pp. 3512–3516. IEEE (2019)
17. Guo, H., Guo, Y., Zha, Y., Zhang, Y., Li, W., Dai, T., Xia, S.T., Li, Y.: Mambairv2: Attentive state space restoration. arXiv preprint arXiv:2411.15269 (2024)
18. Guo, H., Li, J., Dai, T., Ouyang, Z., Ren, X., Xia, S.T.: Mambair: A simple baseline for image restoration with state-space model. In: European Conference on Computer Vision. pp. 222–241. Springer (2024)
19. Hatamizadeh, A., Heinrich, G., Yin, H., Tao, A., Alvarez, J.M., Kautz, J., Molchanov, P.: Fastervit: Fast vision transformers with hierarchical attention. In: International Conference on Learning Representations. vol. 2024, pp. 29368–29391 (2024)
20. Hu, Y., Li, J., Huang, Y., Gao, X.: Channel-wise and spatial feature modulation network for single image super-resolution. IEEE Transactions on Circuits and Systems for Video Technology **30**(11), 3911–3927 (2019)
21. Huang, H., Zhou, X., He, R.: Orthogonal transformer: An efficient vision transformer backbone with token orthogonalization. Advances in Neural Information Processing Systems **35**, 14596–14607 (2022)
22. 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)
23. Hui, Z., Gao, X., Yang, Y., Wang, X.: Lightweight image super-resolution with information multi-distillation network. In: ACM MM (2019)
24. Ignatov, A., Perevozchikov, G., Timofte, R., Zhang, Z., Gao, T., Yang, Y., Zhu, S., Wang, S., Yoon, K., Gankhuyag, G., et al.: Quantized image super-resolution on mobile npus, mobile ai 2025 challenge: Report. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 1908–1921 (2025)
25. Kim, J., Nang, J., Choe, J.: Lmlt : Low-to-high multi-level vision transformer for lightweight image super-resolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops. pp. 5509–5519 (October 2025)
26. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
27. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
28. 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)
29. Li, Y., Fan, Y., Xiang, X., Demandolx, D., Ranjan, R., Timofte, R., Gool, L.V.: Efficient and explicit modelling of image hierarchies for image restoration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2023)
30. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. Proceedings Eighth IEEE International Conference on Computer Vision (ICCV) (2021)
31. 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 (CVPR) Workshops (July 2017)

32. Liu, J., Chen, C., Tang, J., Wu, G.: From coarse to fine: Hierarchical pixel integration for lightweight image super-resolution. In: Proceedings of the AAAI Conference on Artificial Intelligence (2023)
33. Liu, J., Tang, J., Wu, G.: Residual feature distillation network for lightweight image super-resolution. In: European Conference on Computer Vision. pp. 41–55. Springer (2020)
34. Liu, X., Liu, J., Tang, J., Wu, G.: Catanet: Efficient content-aware token aggregation for lightweight image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 17902–17912 (June 2025)
35. 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 (ICCV). pp. 10012–10022 (2021)
36. Long, W., Zhou, X., Zhang, L., Gu, S.: Progressive focused transformer for single image super-resolution. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 2279–2288 (2025)
37. Luo, X., Xie, Y., Zhang, Y., Qu, Y., Li, C., Fu, Y.: Latticenet: Towards lightweight image super-resolution with lattice block. In: European Conference on Computer Vision. pp. 272–289. Springer (2020)
38. 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)
39. 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**(20), 21811–21838 (2017)
40. Qiu, Z., Wang, Z., Zheng, B., Huang, Z., Wen, K., Yang, S., Men, R., Yu, L., Huang, F., Huang, S., et al.: Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. Advances in Neural Information Processing Systems **38**, 100092–100118 (2026)
41. Shi, W., Caballero, J., Huszar, 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 (CVPR) (2016)
42. Sun, L., Dong, J., Tang, J., Pan, J.: Spatially-adaptive feature modulation for efficient image super-resolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 13190–13199 (October 2023)
43. Sun, L., Pan, J., Tang, J.: Shufflemixer: An efficient convnet for image super-resolution. Advances in Neural Information Processing Systems **35**, 17314–17326 (2022)
44. Tian, Y., Chen, H., Xu, C., Wang, Y.: Image processing gnn: Breaking rigidity in super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 24108–24117 (June 2024)
45. 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)
46. 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)

47. 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 (ICCV). pp. 568–578 (October 2021)
48. Wang, Y., Zhang, T.: OSFFNet: Omni-Stage Feature Fusion Network for Lightweight Image Super-Resolution. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 5660–5668 (2024)
49. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
50. Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., Yan, S.: Metaformer is actually what you need for vision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10819–10829 (2022)
51. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Curves and Surfaces. pp. 711–730. Springer Berlin Heidelberg, Berlin, Heidelberg (2012)
52. Zhang, A., Ren, W., Liu, Y., Cao, X.: Lightweight image super-resolution with superpixel token interaction. In: International Conference on Computer Vision (2023)
53. Zhang, K., Zuo, W., Gu, S., Zhang, L.: Learning deep cnn denoiser prior for image restoration. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3929–3938 (2017)
54. Zhang, L., Li, Y., Zhou, X., Zhao, X., Gu, S.: Transcending the limit of local window: Advanced super-resolution transformer with adaptive token dictionary. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2856–2865 (June 2024)
55. Zhang, Q., Zhang, J., Xu, Y., Tao, D.: Vision transformer with quadrangle attention. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(5), 3608–3624 (2024)
56. Zhang, X., Zhang, Y., Yu, F.: Hit-sr: Hierarchical transformer for efficient image super-resolution. In: ECCV (2024)
57. Zhang, X., Zeng, H., Guo, S., Zhang, L.: Efficient long-range attention network for image super-resolution. Proceedings of the IEEE Conference on European Conference on Computer Vision (ECCV) (2022)
58. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: ECCV (2018)
59. 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 (CVPR) (2018)
60. Zhang, Y., Wang, H., Qin, C., Fu, Y.: Learning efficient image super-resolution networks via structure-regularized pruning. In: International Conference on Learning Representations (2021)
61. Zheng, M., Sun, L., Dong, J., Pan, J.: Smfanet: A lightweight self-modulation feature aggregation network for efficient image super-resolution. In: ECCV (2024)
62. Zhou, Y., Li, Z., Guo, C.L., Bai, S., Cheng, M.M., Hou, Q.: Srformer: Permuted self-attention for single image super-resolution. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 12780–12791 (2023)