HiT-SR: Hierarchical Transformer for Efficient Image Super-Resolution

Xiang Zhang[1], Yulun Zhang[2*], and Fisher Yu[1]

[1] ETH Zürich, Switzerland
[2] MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University, China
{xiangz.ethz,yulun100}@gmail.com, i@yf.io
https://github.com/XiangZ-0/HiT-SR

Abstract. Transformers have exhibited promising performance in computer vision tasks including image super-resolution (SR). However, popular transformer-based SR methods often employ window self-attention with quadratic computational complexity to window sizes, resulting in fixed small windows with limited receptive fields. In this paper, we present a general strategy to convert transformer-based SR networks to hierarchical transformers (HiT-SR), boosting SR performance with multi-scale features while maintaining an efficient design. Specifically, we first replace the commonly used fixed small windows with expanding hierarchical windows to aggregate features at different scales and establish long-range dependencies. Considering the intensive computation required for large windows, we further design a spatial-channel correlation method with linear complexity to window sizes, efficiently gathering spatial and channel information from hierarchical windows. Extensive experiments verify the effectiveness and efficiency of our HiT-SR, and our improved versions of SwinIR-Light, SwinIR-NG, and SRFormer-Light yield state-of-the-art SR results with fewer parameters, FLOPs, and faster speeds ($\sim 7\times$).

Keywords: Super-Resolution · Transformer · Hierarchical Windows

1 Introduction

Image super-resolution (SR) is a classical low-level vision task that aims to convert a low-resolution (LR) image to its high-resolution (HR) counterpart with better visual details. How to tackle the ill-posed SR problem has attracted considerable interest for decades [13, 17, 18, 20, 46, 47]. Many popular approaches employ convolutional neural networks (CNNs) to learn the projection between LR inputs and HR images [11, 13, 34, 35, 51, 52]. Despite significant progress being achieved, CNN-based methods usually focus on utilizing local features via convolution and often fall short in aggregating long-range information across the image, limiting the performance of CNN-based SR.

The recent development of vision transformers provides a promising solution for establishing long-range dependencies [8, 12, 14, 30, 42], benefiting many

* Corresponding author: Yulun Zhang.

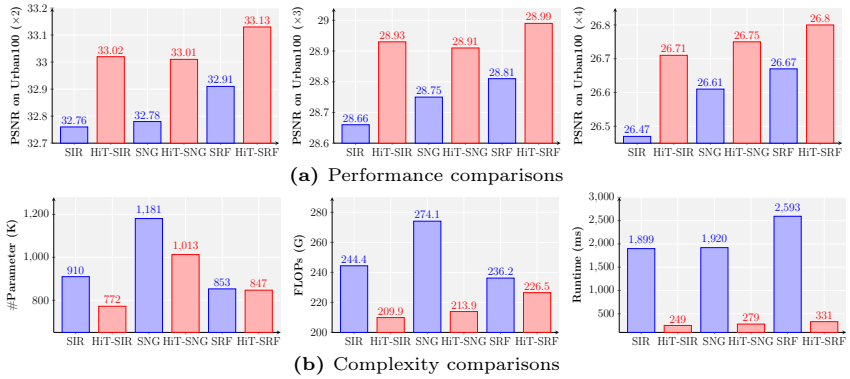

Fig. 1: Comparisons of the popular efficient SR transformers, *i.e.*, SwinIR-Light (SIR) [26], SwinIR-NG (SNG) [10], and SRFormer-Light (SRF) [55], and the corresponding HiT-SR versions, *i.e.*, HiT-SIR, HiT-SNG, and HiT-SRF. The complexity metrics are calculated under $\times 2$ upscaling on an A100 GPU, with the output size set to 720×1280.

computer vision tasks including image SR [9, 10, 16, 26, 50]. An essential component in popular transformer-based SR methods is the window self-attention (W-SA) [10,26,29,30]. By bringing locality into self-attention, the W-SA mechanism not only better utilizes spatial information from input images but also mitigates the computational burden when processing high-resolution images [26,30]. However, current transformer-based SR methods often employ W-SA with fixed small window sizes, *e.g.*, 8×8 in SwinIR [26], limiting the receptive field to a single scale and preventing the network from gathering multi-scale information such as local textures and repetitive patterns [22, 23, 25, 27]. In addition, the quadratic computational complexity of W-SA to the window size also makes the expansion of receptive fields unaffordable in practice. To mitigate the computational overhead, previous attempts often reduce channels to support large windows, *e.g.*, channel splitting of group-wise multi-scale self-attention (GMSA) in ELAN [50] and channel compression of permuted self-attention block (PSA) in SRFormer [55]. However, these methods not only suffer from the trade-off between spatial and channel information but also remain quadratic complexity to window sizes, limiting the window scaling (max 16×16 in ELAN [50] and 24×24 in SRFormer [55] *vs*. 64×64 and larger in ours). Therefore, how to effectively aggregate multi-scale features while maintaining computational efficiency remains a critical problem for transformer-based SR approaches.

To this end, we develop a general strategy to convert popular transformer-based SR networks to hierarchical transformers for efficient image SR (HiT-SR). Motivated by the success of multi-scale feature aggregation in SR [22,23,25,50], we first propose to replace the fixed small windows with expanding hierarchical windows in transformer layers, enabling HiT-SR to leverage informative multi-scale features with gradually enlarging receptive fields. To cope with the increasing computational burdens of W-SA in handling large windows, we further

design a spatial-channel correlation (SCC) method for efficient aggregation of hierarchical features. Specifically, our SCC consists of a dual feature extraction (DFE) layer to improve feature projection by combining spatial and channel information, a spatial and channel self-correlation (S-SC and C-SC) approach to efficiently exploit hierarchical features with **linear computational complexity** to window sizes, better supporting window scaling. In addition, unlike the conventional W-SA that employs hardware inefficient softmax layers [5] and time-consuming window shifting operations, our SCC directly uses feature correlation matrices for transformation and hierarchical windows for receptive field expansion, boosting computational efficiency while preserving functionality.

Overall, our main contributions are three-fold:

- We propose a simple yet effective strategy, *i.e.*, HiT-SR, to convert popular transformer-based SR methods to our hierarchical transformers, boosting SR performance by exploiting multi-scale features and long-range dependencies.
- We design a spatial-channel correlation method to efficiently leverage spatial and channel features with linear computational complexity to window sizes, enabling utilization of large hierarchical windows, *e.g.*, 64×64 windows.
- We convert SwinIR-Light [26], SwinIR-NG [10] and SRFormer-Light [55] to HiT-SR versions, *i.e.*, HiT-SIR, HiT-SNG, and HiT-SRF, achieving better performance with fewer parameters, FLOPs, and $\sim 7\times$ speed-up (Fig. 1).

2 Related Work

Efficient SR. Several CNN-based methods are proposed to approach SR in an efficient way. Previous works first explore compact building blocks for SR networks [2,15,27]. Meanwhile, information distillation methods are developed to progressively refine image features for better performance [22,28], and LatticeNet designs an economical SR structure based on the lattice filter bank [31]. Recent methods further improve the efficiency of image SR by utilizing network pruning techniques [41,53]. However, most CNN-based SR approaches focus on exploiting local features and struggle to utilize long-range dependencies in image SR.

Transformer-based SR. The self-attention (SA) mechanism has been widely employed to establish long-range dependencies in both high-level [6, 14, 29, 30, 37, 38, 54] and low-level vision tasks [7, 26, 44, 48]. For the SR task, SwinIR [26] designs a general image restoration framework based on the shifted window self-attention (SW-SA) [30], and several techniques have been developed to utilize a wider range of features, including N-Gram [10], omni self-attention (OSA) [40], and permuted self-attention (PSA) [55]. Although many advances have been made, most existing works neglect the importance of hierarchical features in SR due to the intensive computation required by W-SA on large windows.

Hierarchical Feature. Hierarchical features have been proven effective in boosting SR performance [22,23,25,27,50], but it is generally challenging for transformer-based SR methods to utilize hierarchical features due to the quadratic complexity of W-SA to window sizes. Recent method ELAN designs a group-wise multi-scale SA (GMSA) technique to gather features at different scales [50]. However,

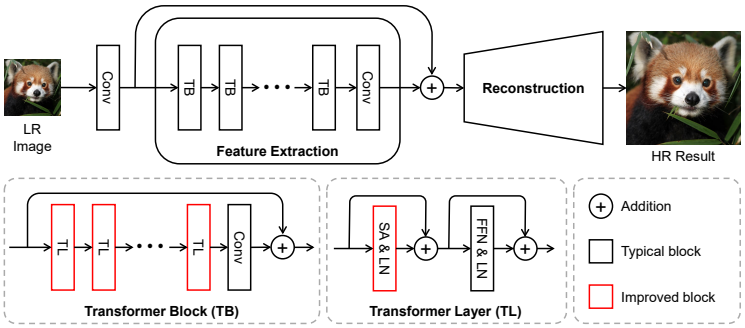

Fig. 2: Typical framework for transformer-based SR methods, where the block-level and layer-level improvements made by our HiT-SR are colored red. SA, FFN, and LN indicate self-attention, feed-forward network, and layer normalization, respectively.

GMSA remains quadratic complexity and suffers from information trade-off due to channel splitting. In our approach, a spatial-channel correlation (SCC) method is designed to efficiently utilize hierarchical features with linear complexity to window sizes, while preserving spatial and channel information.

3 Method

We first introduce the basic methodology of HiT-SR in Sec. 3.1 and then present the block-level and layer-level designs in HiT-SR in Sec. 3.2 and 3.3, respectively.

3.1 Hierarchical Transformer

We first review the commonly adopted pipeline of transformer-based SR methods [10, 26, 55]. As depicted in Fig. 2, popular transformer-based SR framework often consists of convolutional layers to extract shallow features $F_S \in \mathbb{R}^{C\times H\times W}$ from LR input images $I_{LR} \in \mathbb{R}^{3\times H\times W}$, a feature extraction module to aggregate deep image features $F_D \in \mathbb{R}^{C\times H\times W}$ via transformer blocks (TBs), and a reconstruction module to restore HR images $I_{HR} \in \mathbb{R}^{3\times sH\times sW}$ (s denotes upscaling factor) from shallow and deep features. In the feature extraction module, TBs are generally built with cascaded transformer layers (TLs) followed by convolutional layers, where each TL consists of self-attention (SA), feed-forward network (FFN), and layer normalization (LN). Since the computational complexity of SA is quadratic to input sizes [14], window partition is often employed in TL to limit SA on local regions, which is known as window self-attention (W-SA) [26, 30]. Although W-SA alleviates the computational burden, its receptive field is restricted to small local regions, hindering SR networks from utilizing long-range dependencies and multi-scale information.

To efficiently aggregate hierarchical features, we propose a general strategy to convert the above SR framework to hierarchical transformers. As shown in Fig. 2, our approach mainly contains improvements in two aspects: (i) Instead of using

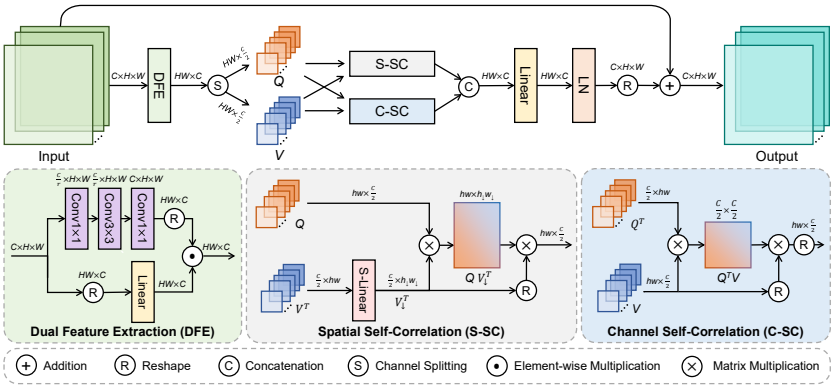

Fig. 4: Layer-Level design in HiT-SR composed of dual feature extraction (DFE), spatial and channel self-correlation (S-SC and C-SC). DFE is designed to extract features from spatial and channel domains. S-SC and C-SC are proposed to efficiently aggregate hierarchical information with linear computational complexity to window sizes.

fixed small window sizes for all TLs, we apply hierarchical windows to different TLs in the block level, enabling HiT-SR to establish long-range dependency and gather multi-scale information; (ii) To overcome the computational burden caused by large windows, we replace the W-SA in TLs with a novel spatial-channel correlation (SCC) method, which better supports window scaling with linear computational complexity to window sizes. Based on the above strategies, HiT-SR not only gains better performance by exploiting hierarchical features but also maintains computational efficiency thanks to SCC.

3.2 Block-Level Design: Hierarchical Windows

At the block level, we assign hierarchical windows to different TLs, collecting multi-scale features. Given a base window size $h_B \times w_B$, we set the window size $h_i \times w_i$ for the i-th TL to

$$h_i = \alpha_i h_B, \quad w_i = \alpha_i w_B, \qquad (1)$$

where $\alpha_i > 0$ is the hierarchical ratio for the i-th TL.

Expanding Windows. To better aggregate hierarchical features, we arrange the windows with an expanding strategy. As illustrated in Fig. 3, we first use small window sizes in the initial layers to gather the most relevant features from local regions, and

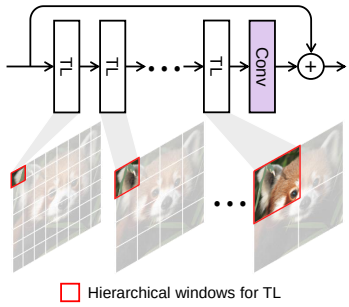

Fig. 3: Block-Level design in HiT-SR. Hierarchical windows are applied to different transformer layers (TLs) to aggregate features with expanding receptive fields.

then gradually expand the window size to utilize the information gained from long-range dependencies. Previous approaches often apply shifting and masking operations on fixed small windows [9, 10, 26, 55] to enlarge receptive fields, but these operations are time-consuming and inefficient in practice. Compared with them, our approach directly utilizes the cascaded TLs to form a hierarchical feature extractor, enabling small to large receptive fields while maintaining overall efficiency. Fig. 1 shows the better performance of our HiT-SR methods with $\sim 7\times$ faster speeds over the original models.

3.3 Layer-Level Design: Spatial-Channel Correlation

At the layer level, we propose spatial-channel correlation (SCC) to efficiently leverage spatial and temporal information from hierarchical inputs. As depicted in Fig. 4, our SCC mainly consists of dual feature extraction (DFE), spatial self-correlation (S-SC), and channel self-correlation (C-SC). Besides, unlike commonly adopted multi-head strategies [9,26,39], different correlation head strategies are applied to S-SC and C-SC to better utilize image features.

Dual Feature Extraction. Linear layers are often employed for feature projection, which only extracts channel information and neglects to model spatial relations. Instead, we propose DFE with a two-branch design to utilize features from two domains. As shown in Fig. 4, DFE consists of a convolutional branch to exploit spatial information and a linear branch to extract channel features. Given an input feature $X \in \mathbb{R}^{C\times H\times W}$, the output of DFE is computed as

$$\begin{aligned} \mathrm{DFE}(X) &= X_{ch} \odot X_{sp}, \quad \text{with} \\ X_{ch} &= \mathrm{Linear}(X),\ X_{sp} = \mathrm{Conv}(X), \end{aligned} \tag{2}$$

where $\odot$ denotes element-wise multiplication. The reshaped channel feature $X_{ch} \in \mathbb{R}^{HW\times C}$ and spatial feature $X_{sp} \in \mathbb{R}^{HW\times C}$ are captured by linear and convolutional layers, respectively. In the spatial branch, we use an hourglass structure to stack three convolutional layers with the hidden dimension reduced by ratio r for efficiency. Finally, the spatial and channel features interact with each other by multiplication, yielding the DFE output.

Unlike standard SA methods that predict queries, keys, and values by linear projection, we equate keys with values as they both reflect the intrinsic properties of input features, and only estimate queries $Q \in \mathbb{R}^{HW\times \frac{C}{2}}$ and values $V \in \mathbb{R}^{HW\times \frac{C}{2}}$ by splitting the DFE output as displayed in Fig. 4,

$$[Q, V] = \mathrm{DFE}(X), \tag{3}$$

which reduces the information redundancy caused by key estimation. Then, we partition queries and keys to non-overlapped windows according to the assigned window size, *e.g.*, $Q_i,\ V_i \in \mathbb{R}^{h_i w_i\times \frac{C}{2}}$ for the i-th TL (the number of windows is omitted for simplicity), and use the partitioned queries and values for the subsequent self-correlation.

Spatial Self-Correlation. Compared with W-SA, our S-SC aggregates spatial information in an efficient manner. Considering the expanding window sizes in

Table 1: Complexity of different layer types. C, N, h, w correspond to channel dimension, window number, height, and width. $m = \max(C, h_\downarrow w_\downarrow)$ is upper bounded.

Layer Type	Complexity per layer	Summary
Global Self-Attention	$O(C \cdot N^2 \cdot h^2 \cdot w^2)$	Quadratic to image size
Window Self-Attention	$O(C \cdot N \cdot h^2 \cdot w^2)$	Quadratic to window size
Spatial-Channel Correlation (Ours)	$O(C \cdot N \cdot m \cdot h \cdot w)$	**Linear** to window size

our hierarchical strategy, we first adaptively summarize the spatial information of values V_i in different TLs by applying linear layers on the spatial dimension (denoted as S-Linear and detailed in supplementary material), *i.e.*,

$$V_{\downarrow,i}^T = \text{S-Linear}_i(V_i^T), \tag{4}$$

where $V_{\downarrow,i} \in \mathbb{R}^{h_\downarrow w_\downarrow \times \frac{C}{2}}$ denotes the projected values with

$$[h_\downarrow, w_\downarrow] = \begin{cases} [h_i, w_i], & \text{if } \alpha_i \leq 1, \\ [h_B, w_B], & \text{if } \alpha_i > 1. \end{cases} \tag{5}$$

Thus, our HiT-SR is able to summarize high-level information from large windows, *i.e.*, $\alpha_i > 1$, and simultaneously preserve fine-grained features with small windows, *i.e.*, $\alpha_i \leq 1$. Afterward, we compute the S-SC based on Q_i and $V_{\downarrow,i}$ as

$$\text{S-SC}(Q_i, V_{\downarrow,i}) = \left(\frac{Q_i V_{\downarrow,i}^T}{D} + B \right) \cdot V_{\downarrow,i}, \tag{6}$$

where B denotes the relative position encoding [43], and the constant denominator $D = \frac{C}{2}$ is used for normalization. Compared with the standard W-SA, our S-SC shows advantages in efficiency and complexity: (i) We utilize correlation maps instead of attention maps to aggregate information, dropping the hardware inefficient operation softmax to improve inference speeds [5]; (ii) Our S-SC supports large windows with linear computational complexity to window sizes. Supposing the input contains N windows with each window in the $\mathbb{R}^{hw \times C}$ space, the numbers of mult-add operations required for W-SA and S-SC are

$$\begin{aligned} \text{Mult-Add(W-SA)} &= 2NC(hw)^2, \\ \text{Mult-Add(S-SC)} &= 2NCh_\downarrow w_\downarrow hw, \end{aligned} \tag{7}$$

where the former is quadratic to window sizes hw. Since $h_\downarrow w_\downarrow$ is upper bounded by the fixed base window size $h_B w_B$, the computational complexity of our S-SC is linear to the window size, benefiting window scaling-up.

Channel Self-Correlation. Apart from spatial information, we further design C-SC to gather features from the channel domain, as depicted in Fig. 4. Given the partitioned queries and values in the i-th TL, the output of C-SC is

$$\text{C-SC}(Q_i, V_i) = \frac{Q_i^T V_i}{D_i} \cdot V_i^T, \tag{8}$$

where the denominator $D_i = h_i w_i$. Compared with the prevalent transposed attention for channel aggregation [3, 9, 48], our C-SC benefits from hierarchical windows and utilizes rich multi-scale information to boost SR performance. For computational complexity, the mult-add operations needed for C-SC is

$$\text{Mult-Add(C-SC)} = 2NC^2hw \tag{9}$$

under inputs in the $\mathbb{R}^{N\times hw\times C}$ space. Combining Eq. (7) and (9), the complexity of our spatial-channel correlation maintains **linear** to window sizes as noted in Tab. 1, enabling scalable windows to make full use of hierarchical information.

Different Correlation Head. Multi-head strategy is commonly employed in SA to gather information from different representation subspaces [39], and it has exhibited promising performance when dealing with spatial information [9, 26]. However, when processing channel information, the multi-head strategy instead restricts the receptive field of channel information aggregation, *i.e.*, each channel can only interact with a limited set of other channels, leading to sub-optimal performance. To address this, we propose to apply the standard multi-head strategy to S-SC but use a single-head strategy in C-SC, enabling full channel interaction. Therefore, the S-SC can utilize information from different channel subspaces by the multi-head strategy, and the C-SC can exploit information from different spatial subspaces via hierarchical windows.

4 Experiments and Analysis

4.1 Experimental Settings

Implementation Details. We apply our HiT-SR strategy to the popular SR method SwinIR-Light [26] and the recent state-of-the-art SR approaches SwinIR-NG [10] and SRFormer-Light [55], corresponding to HiT-SIR, HiT-SNG, and HiT-SRF in this paper. To fairly verify the effectiveness and adaptivity of our method, we convert each method to the HiT-SR version with minimal changes and apply the same hyperparameter settings for all SR transformers. Specifically, we follow the original settings of SwinIR-Light [26] and set the TB number, TL number, channel number, and head number of all HiT-SR improved models to 4, 6, 60, and 6, respectively. The base window size $h_B \times w_B$ are set to the widely adopted value, *i.e.*, 8×8, and we set the hierarchical ratios $[0.5, 1, 2, 4, 6, 8]$ for the 6 TLs in each TB.

We apply the same training strategy to HiT-SIR, HiT-SNG, and HiT-SRF. All the models are implemented based on PyTorch [36] and trained with patch size 64×64 and batch size 64 for 500K iterations. L_1 loss and Adam optimizer [24] ($\beta_1 = 0.9$ and $\beta_2 = 0.99$) are employed for model optimization. We set the initial learning rate as 5×10^{-4}, and half it at [250K,400K,450K,475K] iterations. We also randomly utilize 90°, 180°, and 270° rotations and horizontal flips for data augmentation during model training.

Data and Evaluation. Following previous practice [10, 26, 55], we employ the popular DIV2K [1] dataset for training and the five classical benchmark datasets:

Table 2: Quantitative comparison with state-of-the-art SR methods. The output size is set to 720 × 1280 for all scales to compute FLOPs. Best results are colored red.

Method	Scale	Complexity		Set5		Set14		B100		Urban100		Manga109	
		#Para.	FLOPs	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
EDSR-B [27]	×2	1370K	316.3G	37.99	0.9604	33.57	0.9175	32.16	0.8994	31.98	0.9272	38.54	0.9769
CARN [2]	×2	1592K	222.8G	37.76	0.9590	33.52	0.9166	32.09	0.8978	31.92	0.9256	38.36	0.9765
IMDN [22]	×2	694K	158.8G	38.00	0.9605	33.63	0.9177	32.19	0.8996	32.17	0.9283	38.88	0.9774
LatticeNet [31]	×2	756K	169.5G	38.06	0.9607	33.70	0.9187	32.20	0.8999	32.25	0.9288	38.94	0.9774
RFDN-L [28]	×2	626K	145.8G	38.08	0.9606	33.67	0.9190	32.18	0.8996	32.24	0.9290	38.95	0.9773
SRPN-Lite [53]	×2	609K	139.9G	38.10	0.9608	33.70	0.9189	32.25	0.9005	32.26	0.9294	-	-
FMEN [15]	×2	748K	172.0G	38.10	0.9609	33.75	0.9192	32.26	0.9007	32.41	0.9311	38.95	0.9778
GASSL-B [41]	×2	689K	158.2G	38.08	0.9607	33.75	0.9194	32.24	0.9005	32.29	0.9298	38.92	0.9777
SwinIR-L [26]	×2	910K	244.4G	38.14	0.9611	33.86	0.9206	32.31	0.9012	32.76	0.9340	39.12	0.9783
HNCT [16]	×2	357K	82.4G	38.08	0.9608	33.65	0.9182	32.22	0.9001	32.22	0.9294	38.87	0.9774
ELAN-L [50]	×2	621K	201.3G	38.17	0.9611	33.94	0.9207	32.30	0.9012	32.76	0.9340	39.11	0.9782
Omni-SR [40]	×2	772K	194.5G	38.22	0.9613	33.98	0.9210	32.36	0.9020	33.05	0.9363	39.28	0.9784
SwinIR-NG [10]	×2	1181K	274.1G	38.17	0.9612	33.94	0.9205	32.31	0.9013	32.78	0.9340	39.20	0.9781
SRFormer-L [55]	×2	853K	236.2G	38.23	0.9613	33.94	0.9209	32.36	0.9019	32.91	0.9353	39.28	0.9785
HiT-SIR (Ours)	×2	772K	209.9G	38.22	0.9613	33.91	0.9213	32.35	0.9019	33.02	0.9365	39.38	0.9782
HiT-SNG (Ours)	×2	1013K	213.9G	38.21	0.9612	34.00	0.9217	32.35	0.9020	33.01	0.9360	39.32	0.9782
HiT-SRF (Ours)	×2	847K	226.5G	38.26	0.9615	34.01	0.9214	32.37	0.9023	33.13	0.9372	39.47	0.9787
EDSR-B [27]	×3	1555K	160.2G	34.37	0.9270	30.28	0.8417	29.09	0.8052	28.15	0.8527	33.45	0.9439
CARN [2]	×3	1592K	118.8G	34.29	0.9255	30.29	0.8407	29.06	0.8034	28.06	0.8493	33.50	0.9440
IMDN [22]	×3	703K	71.5G	34.36	0.9270	30.32	0.8417	29.09	0.8046	28.17	0.8519	33.61	0.9445
LatticeNet [31]	×3	765K	76.3G	34.40	0.9272	30.32	0.8416	29.10	0.8049	28.19	0.8513	33.63	0.9442
RFDN-L [28]	×3	633K	65.6G	34.47	0.9280	30.35	0.8421	29.11	0.8053	28.32	0.8547	33.78	0.9458
SRPN-Lite [53]	×3	615K	62.7G	34.47	0.9276	30.38	0.8425	29.16	0.8061	28.22	0.8534	-	-
FMEN [15]	×3	757K	77.2G	34.45	0.9275	30.40	0.8435	29.17	0.8063	28.33	0.8562	33.86	0.9462
GASSL-B [41]	×3	691K	70.4G	34.47	0.9278	30.39	0.8430	29.15	0.8063	28.27	0.8546	33.77	0.9455
SwinIR-L [26]	×3	918K	110.8G	34.62	0.9289	30.54	0.8463	29.20	0.8082	28.66	0.8624	33.98	0.9478
HNCT [16]	×3	363K	37.8G	34.47	0.9275	30.44	0.8439	29.15	0.8067	28.28	0.8557	33.81	0.9459
ELAN-L [50]	×3	629K	89.5G	34.61	0.9288	30.55	0.8463	29.21	0.8081	28.69	0.8624	34.00	0.9478
Omni-SR [40]	×3	780K	88.4G	34.70	0.9294	30.57	0.8469	29.28	0.8094	28.84	0.8656	34.22	0.9487
SwinIR-NG [10]	×3	1190K	114.1G	34.64	0.9293	30.58	0.8471	29.24	0.8090	28.75	0.8639	34.22	0.9488
SRFormer-L [55]	×3	861K	104.8G	34.67	0.9296	30.57	0.8469	29.26	0.8099	28.81	0.8655	34.19	0.9489
HiT-SIR (Ours)	×3	780K	94.2G	34.72	0.9298	30.62	0.8474	29.27	0.8101	28.93	0.8673	34.40	0.9496
HiT-SNG (Ours)	×3	1021K	99.5G	34.74	0.9297	30.62	0.8474	29.26	0.8100	28.91	0.8671	34.38	0.9495
HiT-SRF (Ours)	×3	855K	101.6G	34.75	0.9300	30.61	0.8475	29.29	0.8106	28.99	0.8687	34.53	0.9502
EDSR-B [27]	×4	1518K	114.0G	32.09	0.8938	28.58	0.7813	27.57	0.7357	26.04	0.7849	30.35	0.9067
CARN [2]	×4	1592K	90.9G	32.13	0.8937	28.60	0.7806	27.58	0.7349	26.07	0.7837	30.47	0.9084
IMDN [22]	×4	715K	40.9G	32.21	0.8948	28.58	0.7811	27.56	0.7353	26.04	0.7838	30.45	0.9075
LatticeNet [31]	×4	777K	43.6G	32.18	0.8943	28.61	0.7812	27.57	0.7355	26.14	0.7844	30.54	0.9075
RFDN-L [28]	×4	643K	37.4G	32.28	0.8957	28.61	0.7818	27.58	0.7363	26.20	0.7883	30.61	0.9096
SRPN-Lite [53]	×4	623K	35.8G	32.24	0.8958	28.69	0.7836	27.63	0.7373	26.16	0.7875	-	-
FMEN [15]	×4	769K	44.2G	32.24	0.8955	28.70	0.7839	27.63	0.7379	26.28	0.7908	30.70	0.9107
GASSL-B [41]	×4	694K	39.9G	32.17	0.8950	28.66	0.7835	27.62	0.7377	36.16	0.7888	30.70	0.9100
SwinIR-L [26]	×4	930K	63.6G	32.44	0.8976	28.77	0.7858	27.69	0.7406	26.47	0.7980	30.92	0.9151
HNCT [16]	×4	373K	22.0G	32.31	0.8957	28.71	0.7834	27.63	0.7381	26.20	0.7896	30.70	0.9112
ELAN-L [50]	×4	640K	53.7G	32.43	0.8975	28.78	0.7858	27.69	0.7406	26.54	0.7982	30.92	0.9150
Omni-SR [40]	×4	792K	50.9G	32.49	0.8988	28.78	0.7859	27.71	0.7415	26.64	0.8018	31.02	0.9151
SwinIR-NG [10]	×4	1201K	64.4G	32.44	0.8980	28.83	0.7870	27.73	0.7418	26.61	0.8010	31.09	0.9161
SRFormer-L [55]	×4	873K	62.8G	32.51	0.8988	28.82	0.7872	27.73	0.7422	26.67	0.8032	31.17	0.9165
HiT-SIR (Ours)	×4	792K	53.8G	32.51	0.8991	28.84	0.7873	27.73	0.7424	26.71	0.8045	31.23	0.9176
HiT-SNG (Ours)	×4	1032K	57.7G	32.55	0.8991	28.83	0.7873	27.74	0.7426	26.75	0.8053	31.24	0.9176
HiT-SRF (Ours)	×4	866K	58.0G	32.55	0.8999	28.87	0.7880	27.75	0.7432	26.80	0.8069	31.26	0.9171

Set5 [4], Set14 [49], B100 [32], Urban100 [21], and Manga109 [33], for evaluation. The LR images are obtained from their HR counterparts by bicubic degradation. We conduct comparisons under three upscaling factors: ×2, ×3, and ×4, and assess the SR performance by the two commonly used metrics PSNR and SSIM [45], which are computed on the Y channel in the YCbCr space.

4.2 Benchmarking

We evaluate the proposed HiT-SR by comparing our HiT-SIR, HiT-SNG, and HiT-SRF with existing state-of-the-art efficient SR approaches, including CNN-based algorithms EDSR-B [27], CARN [2], IMDN [22], LatticeNet [31], RFDN-

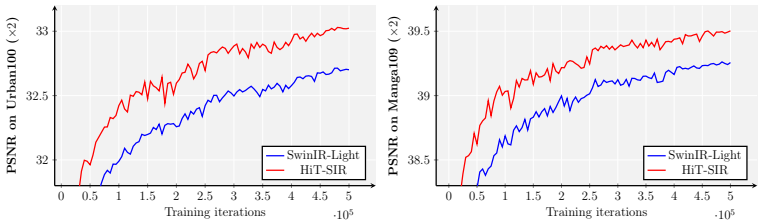

Fig. 5: Convergence comparison of SwinIR-Light [26] and our improved version HiT-SIR on Urban100 (×2) and Manga109 (×2) datasets under the same training settings.

Table 3: Comparisons between state-of-the-art efficient SR transformers and our HiT-SR approaches. Complexity metrics are measured under ×2 upscaling on an A100 GPU with the output image size set to 720 × 1280. Improvements are highlighted in red.

Method	Computational complexity			Urban100 (×2)	
	#Para. (K)	FLOPs (G)	Runtime (ms)	PSNR	SSIM
SwinIR-Light [26]	910	244.4	1899	32.76	0.9340
HiT-SIR (Ours)	772 (138↓)	209.9 (34.5↓)	249 (7.6×)	33.02 (0.26↑)	0.9365 (0.0025↑)
SwinIR-NG [10]	1181	274.1	1920	32.78	0.9340
HiT-SNG (Ours)	1013 (168↓)	213.9 (60.2↓)	279 (6.9×)	33.01 (0.23↑)	0.9360 (0.0020↑)
SRFormer-Light [55]	853	236.2	2593	32.91	0.9353
HiT-SRF (Ours)	847 (6↓)	226.5 (9.7↓)	331 (7.8×)	33.13 (0.22↑)	0.9372 (0.0019↑)

L [28], SRPN-Lite [53], FMEN [15], and GASSL-B [41], as well as transformer-based methods SwinIR-Light [26], HCNT [16], ELAN-Light [50], Omni-SR [40], SwinIR-NG [10], and SRFormer-Light [55].

Quantitative Comparisons. The results in Tab. 2 show the remarkable performance of our HiT-SR methods on all benchmark datasets. Compared with the previous state-of-the-art approach SRFormer-Light, our HiT-SIR achieves comparable results with fewer parameters and FLOPs, and our HiT-SRF sets new state-of-the-art SR results across all upscaling factors. Furthermore, our HiT-SR improved methods show advantages over their original models in three main aspects: performance, efficiency, and convergence. (i) As illustrated in Tab. 3, our HiT-SR method contributes to significant improvements for SR performance, *e.g.*, 0.26/0.23/0.22 dB PSNR gains for HiT-SIR/HiT-SNG/HiT-SRF on Urban100 (×2) dataset. (ii) By replacing the computationally inefficient shifted window self-attention with our SCC, all HiT-SR models require fewer computational resources and significantly boost inference speeds, achieving ∼ 7× speed-up. Although the parameters and FLOPs improvements of HiT-SRF over SRFormer-Light are relatively smaller as SRFormer also improves the efficiency of transformer layers via permuted self-attention [55], our HiT-SRF still shows better SR performance with 7.8× faster inference speed, benefiting practical usage. (iii) We finally compare the convergence curves of SwinIR-Light and our improved HiT-SIR on Urban100 (×2) and Manga109 (×2) datasets in Fig. 5. It

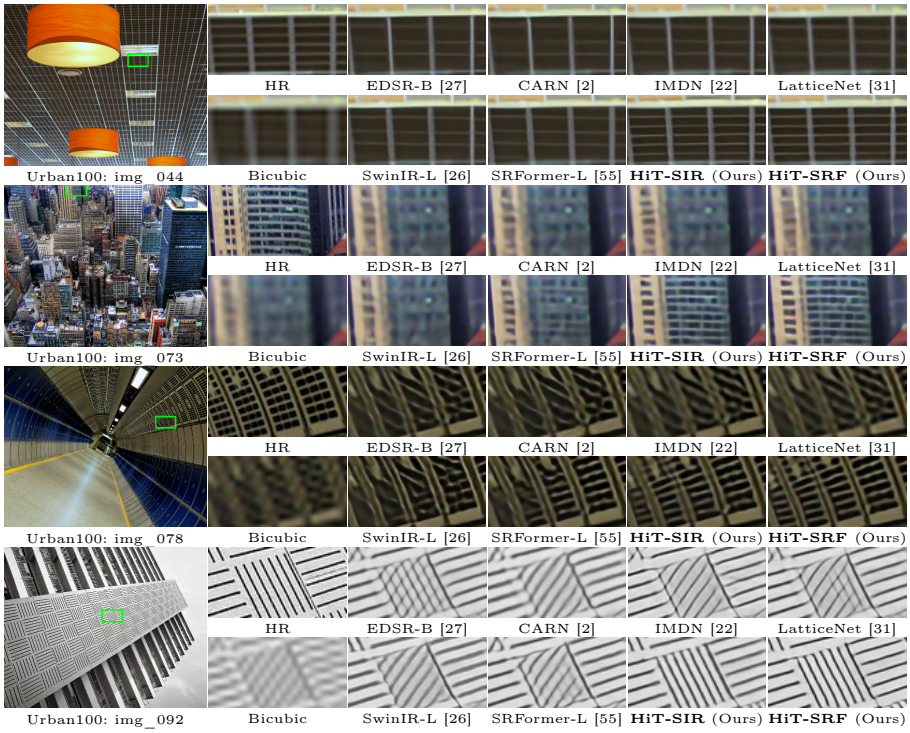

Fig. 6: Qualitative comparisons for image SR (×4) in challenging scenes.

is evident that our HiT-SR improved method converges faster than the original networks and achieves similar SR performance with only around half iterations.

Qualitative Comparisons. Fig. 6 shows qualitative comparisons in some challenging image SR scenes. Previous SR approaches often suffer from blurry details and artifacts, *e.g.*, img_078, as they mainly utilize local features with limited receptive fields. In contrast, our HiT-SIR and HiT-SRF produce finer details and sharper textures by leveraging long-range dependencies with large windows. In addition, structure distortion is another common problem in SR results, especially in challenging scenarios like img_092. We can observe that previous methods fail to recover the image structure and deteriorate SR performance with distorted lines. Instead, our HiT-SR methods utilize multi-scale information *e.g.*, repetitive patterns, and better retain the overall structure. The effectiveness of our HiT-SR is further validated by comparing the visual results of SwinIR-Light/SRFormer-Light and their improved versions HiT-SIR/HiT-SRF, which restore better image structures with finer details.

4.3 Method Analysis

In this section, we further verify the advantages of our HiT-SR method by analyzing its information aggregation ability and window scalability.

Information Aggregation. We employ the local attribution maps (LAM) [19] to analyze the information aggregation performance of HiT-SR. LAM is a visual tool aiming at finding input pixels that strongly influence the SR outputs, *i.e.*, informative areas, and larger informative areas mean better aggregation ability. As shown in Fig. 7, we apply the LAM approach to SwinIR-Light [26] and our improved version HiT-SIR, and compare the informative areas for the same target regions. Due to the fixed small windows used in SwinIR-Light, the informative areas are limited to a local region, hindering the utilization of long-range dependencies. By contrast, our HiT-SIR significantly expands the informative areas by leveraging hierarchical features. Therefore, our proposed method is able to gather a wider range of information, *e.g.*, similar structures and repetitive patterns, to boost SR performance.

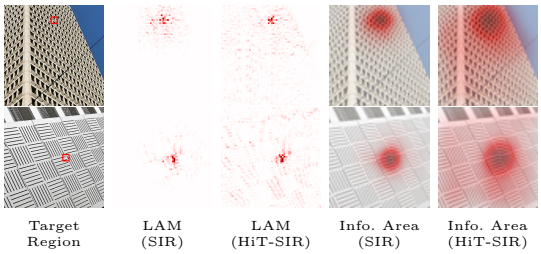

Fig. 7: Comparisons between SwinIR-Light (SIR) and HiT-SIR, where local attribution maps (LAM) and informative areas (Info. Area) are displayed.

Window Scalability. We measure the required FLOPs for a single W-SA [26], PSA [55], and our SCC layer, under the same hyperparameter settings (channel and head numbers are set to 60 and 6, respectively). As depicted in Fig. 8, the FLOPs of W-SA increase drastically with the expansion of window size, making it unaffordable to utilize large windows. Although PSA alleviates the computational burden with permutation techniques, its complexity still remains quadratic to window sizes, limiting window scaling. Benefiting from the linear complexity of SCC, our method equips SR transformers with scalable large windows to efficiently gather long-range information.

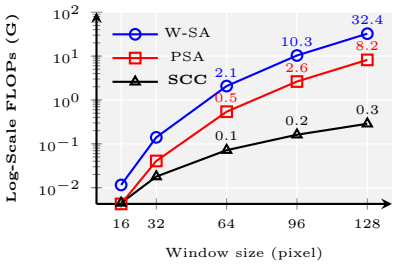

Fig. 8: FLOPs of W-SA in SwinIR [26], PSA in SRFormer [55], and our SCC, with respect to square window sizes.

4.4 Ablation Study

In Tab. 4, we study the effectiveness of each component in our design based on HiT-SIR network. All models are trained on the DIV2K dataset [1] for 300K iterations and tested on the Manga109 ($\times 2$) dataset [33]. We set the output size to $3 \times 720 \times 1280$ to compute FLOPs. Note that we maintain similar complexity across all variants by adjusting channel numbers for fair comparisons.

Table 4: Ablation study. We train all models on DIV2K for 300K iterations, and test on Manga109 (×2). The ablations about hierarchical windows, DFE, SCC, and different head strategies correspond to #1-4, #5-7, #8-12, and #13-15, where our strategies are noted in **bold**. Win. indicates the assigned window sizes for a transformer block, and #head means the number of correlation heads. Best results are colored red.

ID	Strategies	#Para.	FLOPs	PSNR	SSIM
#1	**Win.=[4,8,16,32,48,64]**	772K	209.9G	39.21	0.9780
#2	Win.=[8,8,8,8,8,8]	772K	208.6G	38.88	0.9775
#3	Win.=[64,64,64,64,64,64]	773K	241.4G	39.09	0.9775
#4	Win.=[64,48,32,16,8,4]	772K	209.9G	39.12	0.9779
#5	**DFE**	772K	209.9G	39.21	0.9780
#6	DFE → Linear	792K	214.9G	38.99	0.9777
#7	DFE → Conv.	781K	212.3G	38.95	0.9775
#8	**SCC**	772K	209.9G	39.21	0.9780
#9	QV → QKV	826K	222.6G	39.19	0.9780
#10	Correlation → Attention	772K	209.9G	39.08	0.9779
#11	S-SC only	819K	222.1G	39.04	0.9775
#12	C-SC only	816K	217.2G	39.18	0.9778
#13	**#head of S-SC, C-SC=[6,1]**	772K	209.9G	39.21	0.9780
#14	#head of S-SC, C-SC=[1,1]	772K	209.9G	39.14	0.9778
#15	#head of S-SC, C-SC=[6,6]	772K	201.2G	39.08	0.9778

Hierarchical Windows. To verify the effect of our window strategy, we conduct experiments with different window arrangement methods, including fixed small windows (#2), fixed large windows (#3), and shrinking hierarchical windows (#4). The model with small windows can only utilize local features for SR, resulting in sub-optimal performance. By establishing long-range dependencies, #3 boosts the PSNR results but simultaneously brings more computational burdens due to the fixed large window sizes. Compared with them, the models with hierarchical windows (#1 and #4) make full use of multi-scale features for SR, showing promising improvements in both performance and efficiency. Moreover, the proposed expanding hierarchical windows gain the best performance. This is because our expanding window strategy allows networks to first utilize the most relevant information and improve feature representation with small windows, benefiting the establishment of reliable long-range dependencies with the subsequent large windows.

Dual Feature Extraction. In the layer-level design, we first investigate the impact of DFE by comparing it with linear projection (#6) and convolutional projection (#7). As shown in Tab. 4, the performance of linear projection is limited since only channel information is leveraged. Although spatial relation is exploited in convolution, efficiency-related techniques like dimension reduction or depth-wise operation are often employed to alleviate the computational burdens of convolutional layers, leading to insufficient utilization of channel information. By contrast, our DFE extracts spatial and channel information in a

two-branch structure and fuses features from different domains, gaining significant performance improvements as shown in Tab. 4.

Spatial-Channel Correlation. Our SCC efficiently aggregates spatial and channel information via self-correlation with queries and values. We first design a variant using commonly adopted queries, keys, and values for computation (QKV in #9). As illustrated in Tab. 4, the QKV setting performs similarly to our QV setting but consumes more computational resources. This is because keys and values both reflect the intrinsic properties of input features, and thus using only queries and values mitigates the information redundancy during computation. In addition, involving values in computing the correlation matrix, *i.e.*, Eq. (6) and (8), also contributes to SR performance with more precise transforming relations. Next, we add softmax layers to Eq. (6) and (8), converting our correlation methods to attention approaches (#10) for comparison. The results in Tab. 4 validate the better performance of our proposed methods without limiting the transformation scales by softmax operations. Finally, we conduct experiments by replacing SCC with only S-SC (#11) or C-SC (#12). From the metrics in Tab. 4, we can observe that C-SC performs better than S-SC, and utilizing both spatial and channel information leads to the overall best performance.

Different Correlation Head. To study our different correlation head strategy, we train two models with the same single-head and multi-head strategies for both S-SC and C-SC, corresponding to #14 and #15 in Tab. 4. By comparing #13 and #14, one can see that adopting the single-head method for S-SC is inappropriate as single-head prevents S-SC from aggregating features from different representation spaces. Meanwhile, applying the multi-head approach to C-SC, *i.e.*, #15, also impairs SR performance since splitting the channel into different heads prohibits each channel from gathering information from all the other channels. Compared with these variants, our different correlation head strategy achieves the best results by enabling S-SC to aggregate features from different channel subspaces while allowing full channel interaction in C-SC.

5 Conclusion

In this paper, we propose a general strategy to convert popular transformer-based SR methods to hierarchical transformers for efficient image SR (HiT-SR). Our approach consists of block-level and layer-level designs. In each transformer block, we apply expanding hierarchical windows to establish long-range dependencies and leverage multi-scale features, boosting SR performance. Considering the quadratic complexity of self-attention methods, we devise a spatial-channel correlation (SCC) method with linear complexity to window sizes, benefiting the efficient aggregation of hierarchical features. Extensive evaluations are made to verify the effectiveness of the proposed HiT-SR, and our HiT-SIR, HiT-SNG, and HiT-SRF set new state-of-the-art results for efficient image SR.

Acknowledgements

This work was supported by Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), the Fundamental Research Funds for the Central Universities, and Huawei Technologies Oy (Finland) Project.

References

1. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: Dataset and study. In: CVPRW. pp. 126–135 (2017)
2. Ahn, N., Kang, B., Sohn, K.A.: Fast, accurate, and lightweight super-resolution with cascading residual network. In: ECCV. pp. 252–268 (2018)
3. Ali, A., Touvron, H., Caron, M., Bojanowski, P., Douze, M., Joulin, A., Laptev, I., Neverova, N., Synnaeve, G., Verbeek, J., et al.: Xcit: Cross-covariance image transformers. In: NIPS. vol. 34, pp. 20014–20027 (2021)
4. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In: BMVC (2012)
5. Cai, H., Li, J., Hu, M., Gan, C., Han, S.: Efficientvit: Lightweight multi-scale attention for high-resolution dense prediction. In: ICCV. pp. 17302–17313 (2023)
6. Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., Wang, M.: Swin-unet: Unet-like pure transformer for medical image segmentation. In: ECCV. pp. 205–218 (2022)
7. 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: CVPR. pp. 12299–12310 (2021)
8. Chen, Q., Wu, Q., Wang, J., Hu, Q., Hu, T., Ding, E., Cheng, J., Wang, J.: Mixformer: Mixing features across windows and dimensions. In: CVPR. pp. 5249–5259 (2022)
9. Chen, Z., Zhang, Y., Gu, J., Kong, L., Yang, X., Yu, F.: Dual aggregation transformer for image super-resolution. In: ICCV. pp. 12312–12321 (2023)
10. Choi, H., Lee, J., Yang, J.: N-gram in swin transformers for efficient lightweight image super-resolution. In: CVPR. pp. 2071–2081 (2023)
11. Dai, T., Cai, J., Zhang, Y., Xia, S.T., Zhang, L.: Second-order attention network for single image super-resolution. In: CVPR. pp. 11065–11074 (2019)
12. Ding, M., Xiao, B., Codella, N., Luo, P., Wang, J., Yuan, L.: Davit: Dual attention vision transformers. In: ECCV. pp. 74–92 (2022)
13. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: ECCV. pp. 184–199 (2014)
14. 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. In: ICLR (2021)
15. Du, Z., Liu, D., Liu, J., Tang, J., Wu, G., Fu, L.: Fast and memory-efficient network towards efficient image super-resolution. In: CVPR. pp. 853–862 (2022)
16. Fang, J., Lin, H., Chen, X., Zeng, K.: A hybrid network of cnn and transformer for lightweight image super-resolution. In: CVPRW. pp. 1103–1112 (2022)
17. Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning low-level vision. IJCV **40**, 25–47 (2000)

18. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: ICCV. pp. 349–356 (2009)
19. Gu, J., Dong, C.: Interpreting super-resolution networks with local attribution maps. In: CVPR. pp. 9199–9208 (2021)
20. Hu, Y., Li, J., Huang, Y., Gao, X.: Channel-wise and spatial feature modulation network for single image super-resolution. TCSVT **30**(11), 3911–3927 (2019)
21. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: CVPR. pp. 5197–5206 (2015)
22. Hui, Z., Gao, X., Yang, Y., Wang, X.: Lightweight image super-resolution with information multi-distillation network. In: ACMMM. pp. 2024–2032 (2019)
23. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: CVPR. pp. 1646–1654 (2016)
24. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. ICLR (2015)
25. Lai, W.S., Huang, J.B., Ahuja, N., Yang, M.H.: Deep laplacian pyramid networks for fast and accurate super-resolution. In: CVPR. pp. 624–632 (2017)
26. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In: ICCVW. pp. 1833–1844 (2021)
27. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: CVPRW. pp. 136–144 (2017)
28. Liu, J., Tang, J., Wu, G.: Residual feature distillation network for lightweight image super-resolution. In: ECCV. pp. 41–55 (2020)
29. Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., Ning, J., Cao, Y., Zhang, Z., Dong, L., et al.: Swin transformer v2: Scaling up capacity and resolution. In: CVPR. pp. 12009–12019 (2022)
30. 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: ICCV. pp. 10012–10022 (2021)
31. Luo, X., Xie, Y., Zhang, Y., Qu, Y., Li, C., Fu, Y.: Latticenet: Towards lightweight image super-resolution with lattice block. In: ECCV. pp. 272–289 (2020)
32. 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: ICCV. vol. 2, pp. 416–423 (2001)
33. 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)
34. Mei, Y., Fan, Y., Zhou, Y., Huang, L., Huang, T.S., Shi, H.: Image super-resolution with cross-scale non-local attention and exhaustive self-exemplars mining. In: CVPR. pp. 5690–5699 (2020)
35. Niu, B., Wen, W., Ren, W., Zhang, X., Yang, L., Wang, S., Zhang, K., Cao, X., Shen, H.: Single image super-resolution via a holistic attention network. In: ECCV. pp. 191–207 (2020)
36. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. In: NIPS. vol. 32 (2019)
37. Ramachandran, P., Parmar, N., Vaswani, A., Bello, I., Levskaya, A., Shlens, J.: Stand-alone self-attention in vision models. In: NIPS. vol. 32 (2019)
38. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: ICML. pp. 10347–10357 (2021)
39. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NIPS. vol. 30 (2017)

40. Wang, H., Chen, X., Ni, B., Liu, Y., Liu, J.: Omni aggregation networks for lightweight image super-resolution. In: CVPR. pp. 22378–22387 (2023)
41. Wang, H., Zhang, Y., Qin, C., Van Gool, L., Fu, Y.: Global aligned structured sparsity learning for efficient image super-resolution. PAMI (2023)
42. 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: ICCV. pp. 568–578 (2021)
43. Wang, W., Chen, W., Qiu, Q., Chen, L., Wu, B., Lin, B., He, X., Liu, W.: Crossformer++: A versatile vision transformer hinging on cross-scale attention. In: ICLR (2022)
44. Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H.: Uformer: A general u-shaped transformer for image restoration. In: CVPR. pp. 17683–17693 (2022)
45. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. TIP **13**(4), 600–612 (2004)
46. Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution as sparse representation of raw image patches. In: CVPR. pp. 1–8 (2008)
47. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. TIP **19**(11), 2861–2873 (2010)
48. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H.: Restormer: Efficient transformer for high-resolution image restoration. In: CVPR. pp. 5728–5739 (2022)
49. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Proc. 7th Int. Conf. Curves Surf. pp. 711–730 (2012)
50. Zhang, X., Zeng, H., Guo, S., Zhang, L.: Efficient long-range attention network for image super-resolution. In: ECCV. pp. 649–667 (2022)
51. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: ECCV. pp. 286–301 (2018)
52. Zhang, Y., Li, K., Li, K., Zhong, B., Fu, Y.: Residual non-local attention networks for image restoration. ICLR (2019)
53. Zhang, Y., Wang, H., Qin, C., Fu, Y.: Learning efficient image super-resolution networks via structure-regularized pruning. In: ICLR (2021)
54. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: CVPR. pp. 6881–6890 (2021)
55. Zhou, Y., Li, Z., Guo, C.L., Bai, S., Cheng, M.M., Hou, Q.: Srformer: Permuted self-attention for single image super-resolution. In: ICCV. pp. 12780–12791 (October 2023)