Token-Aligned Hierarchies for Lightweight Super-Resolution

Anonymous CVPR submission

Paper ID

Abstract

Windowed self-attention (WSA) has become a strong backbone for single-image super-resolution (SR), yet its high overhead often leads to latency inefficiency. We revisit Swin-style SR from a hierarchical perspective and introduce a token-aligned encoder–decoder built entirely with grouped and depthwise convolutions, replacing attention windows with efficient spatial mixing. Our architecture preserves the locality bias of WSA while substantially improving speed and stability. It incorporates (i) symmetry-preserving padding for consistent token partitioning, (ii) a token pyramid that expands channels through patch merging to aggregate broader context, and (iii) Token-Aligned Skip Fusion (TASF) for precise multi-scale feature reuse. Built upon the SwinIR hierarchy, our model attains both the relatively high reconstruction quality (PSNR $\approx$ 37.8 dB for $\times 2$) and the lowest latency among all compared methods, including faster inference than SwinIR-light while maintaining strong texture consistency and low memory usage. These results demonstrate that hierarchical, convolution-based modeling can match or surpass transformer performance at a fraction of the cost, making our design highly suitable for real-time and edge SR applications.

1. Introduction

Single-image super-resolution (SISR) aims to reconstruct a high-resolution (HR) image from a low-resolution (LR) input. Early SISR progress was driven by convolutional neural networks (CNNs) [13, 23, 43, 44], which offered strong spatial priors and efficient local feature extraction. However, CNN-based approaches often struggle to capture long-range dependencies without incurring higher computational cost or training instability [11, 39]. The introduction of Vision Transformers (ViTs), originally developed for NLP [35] and later adapted for vision [3, 15, 24, 34, 36], brought effective global modeling to image restoration, including super-resolution [5, 8, 22, 37]. Models such as SwinIR [22] demonstrated that windowed and shifted-window self-attention can balance locality and context. Subsequent works refined this paradigm through hybrid attention, cross-scale interaction, and adaptive aggregation [7, 8, 32, 42].

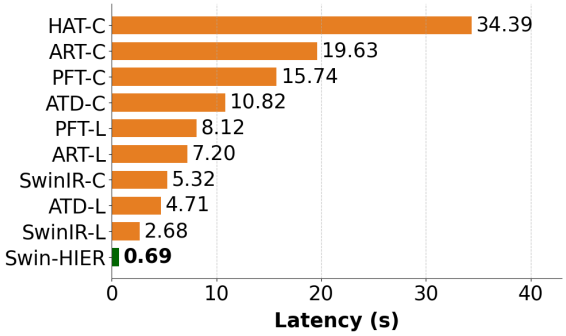


Figure 1. Model latency comparison among transformer-based SR methods. Our proposed Swin-HIER achieves the lowest latency (**0.69 s**), outperforming both SwinIR variants and recent transformer-based competitors while maintaining strong reconstruction quality. Methods with superscripts C and L denote classical and lightweight variants, respectively.

Nonetheless, transformer-based SR models still face two key limitations: (1) dense attention mixes both relevant and irrelevant tokens, increasing redundancy; and (2) local windowed attention restricts evidence sharing between neighboring regions. This can produce inconsistent textures and subtle boundary artifacts across adjacent windows.

To address these inefficiencies, we replace the standard windowed self-attention module with a ConvEncoder Layer, a lightweight depthwise–pointwise convolutional encoder. Each layer performs two spatial convolutional stages with residual connections, followed by a compact MLP projection. This substitution retains local inductive bias and contextual mixing while eliminating the quadratic attention cost. Practically, it enables faster inference without degrading representational richness, especially when combined with our hierarchical token design. As shown in Fig. 1, this design choice contributes significantly to runtime efficiency: our model achieves a latency of only **0.69 s**, outperforming SwinIR (lightweight, 1.63 s) and SwinIR (classical, 3.54 s), while being substantially faster than ATD [42],

PFT [25], HAT [7] and ART [41], which range from 10.82 s to 34.39 s. This result illustrates that careful architectural alignment and convolutional substitution can yield transformer-level performance at a fraction of the cost. The latency was measured and averaged over 10 randomly chosen DIV2k [30] validation dataset.

Building on these insights, we propose a hierarchical, token-aligned Swin architecture (Swin-HIER) designed to balance locality, cross-window consistency, and computational efficiency. Our main design choices are:

1. **Strict token hierarchy**: patch merging halves resolution and doubles channel depth, enhancing contextual aggregation without heavy computation.
2. **Token-Aligned Skip Fusion (TASF)**: encoder features are fused with decoder tokens at matched resolutions before upscaling, improving texture consistency.
3. **Conv-based local reasoning**: replace windowed attention with efficient convolutional encoding layers, preserving structure awareness at a fraction of the cost.
4. **Modular SR head**: a factorized PixelShuffle head performs progressive upscaling with lightweight refinement.

2. Related Works

2.1. Convolution-based Super-Resolution

Early deep SISR methods established the now-standard "feature extraction nonlinear mapping upsampling" pipeline using purely convolutional operators. SRCNN [13] first demonstrated that a shallow CNN can outperform sparse-coding SR, and FSRCNN [14] inverted the order of feature extraction and upsampling to improve speed. To increase reconstruction capacity, very-deep residual networks such as VDSR [18] and DRCN [19] stacked many convolutional layers with global residual learning to stabilize training. Subsequent works exploited recursive and multi-path connections to reuse features more effectively, e.g. DRRN [31] and LapSRN [20], while EDSR [23] simplified residual blocks and removed unnecessary modules to push accuracy on DIV2K. Later CNN-based SR models focused on strengthening long-range and cross-channel interactions within a convolutional framework: RDN [44] introduced densely connected residual blocks to aggregate hierarchical features, and RCAN [43] employed channel attention to selectively amplify informative responses. In parallel, efficiency-oriented designs such as CARN [1], IMDN [17], and LAPAR [21] explored grouped/depthwise convolutions and lightweight feature distillation to achieve better accuracy–latency trade-offs. These CNNs show that, with proper residual/attention routing, convolution alone can still deliver strong SR quality while preserving favorable inductive bias and hardware efficiency, an observation our convolutional replacement layers also leverage.

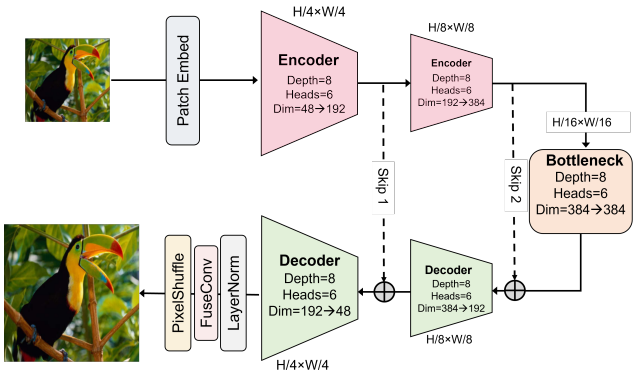


Figure 2. Overall architecture of Swin-HIER. Encoder stage with patch merging and exact window padding. Decoder stage with TASF and patch expand. Modular SR head that factors the total scale into $\times 2$ and $\times 3$ by PixelShuffle.

2.2. Transformer-based Super-Resolution

Transformers, initially introduced for natural language processing [12, 35], were soon extended to visual understanding tasks [3, 15, 24, 34, 36]. Their strong long-range dependency modeling made them competitive for low-level restoration, particularly SISR [5, 8, 22, 37]. Classical CNN-based SR approaches [13, 23, 43, 44] are computationally efficient but limited by local receptive fields and weaker cross-region interactions. Transformer-based SR models, in contrast, have gradually surpassed them in both fidelity and perceptual realism. For instance, IPT [5] introduces large-scale pre-training for SR, while SwinIR [22] enhances local–global balance through shifted window attention and residual Swin blocks. Subsequent methods refine this paradigm via hybrid or cross-window attention, improving cross-region aggregation and hierarchical representation [7, 8, 32, 42].

2.3. Sparse and Selective Attention

Despite their success, dense self-attention often entangles both relevant and irrelevant tokens, increasing redundancy and noise. Sparse or selective mechanisms aim to preserve only the most informative relations. NLSA [28] integrates non-local operations with sparse attention to improve robustness and efficiency, while ART [41] combines dense and sparse branches to expand the effective receptive field. Other approaches explicitly retain only top-k relations [9, 38], thereby filtering distractors and reducing computational cost. Building on this principle, DRSformer [6] learns a dynamic top-k selector to adaptively retain relevant self-attention values for image restoration. Complementary designs that enhance cross-window or channel–window interactions [7] further alleviate locality bias and improve long-range feature correlation.

3. Methodology

3.1. Preliminaries

Problem motivation. Windowed self-attention (WSA) in Swin-style Transformers restricts feature interaction to a finite $M \times M$ window. While this improves locality and efficiency, it limits cross-window communication causing neighboring regions to reconstruct textures with mismatched phase or orientation, a phenomenon refered as regional drift [10]. Visualizing intermediate token activations across Swin stages reveals that early layers primarily capture intra-window detail, while deeper layers partially mitigate drift through patch merging but cannot fully eliminate inter-window discontinuities. These inconsistencies manifest as faint blocking and texture misalignment in reconstructed SR images [10].

Let $Q, K, V \in \mathbb{R}^{M^2 \times D}$ denote the token query, key, and value matrices for one window. The WSA operation is

$$\mathrm{WSA}(Q, K, V) = \mathrm{Softmax}\left(\frac{QK^\top}{\sqrt{D}} + B\right) V, \quad (1)$$

where $B \in \mathbb{R}^{M^2 \times M^2}$ encodes relative positional bias. Since Eq. (1) is applied independently to each window, inter-window dependencies are ignored. As depth increases, Swin's hierarchical merging introduces partial context expansion, but residual inconsistencies persist at window boundaries, appearing as faint blocking or phase shifts in super-resolved images.

3.2. Design Rationale

Our architecture explicitly addresses these limitations through three core principles, implemented directly in our network code:

1. **Exact geometric alignment.** We enforce symmetry-preserving padding so that both the patch stride evenly divide the spatial dimensions. This guarantees consistent token partitioning across encoder and decoder stages and avoids boundary misalignment.
2. **Hierarchical token pyramid.** Each Patch Merging step halves spatial resolution and doubles channel width, expanding receptive fields while maintaining constant compute per stage.
3. **TASF.** During decoding, encoder features at matching resolutions are linearly fused before upsampling, ensuring that cross-scale aggregation preserves texture phase and semantic alignment.

Fig. 2 illustrates our proposed architecture, Swin-HIER, a hierarchical, token-aligned encoder–decoder designed for efficient super-resolution. The model retains the structural intuition of Swin-based backbones but replaces all attention modules with lightweight convolutional layers, combining transformer-style token hierarchies with the efficiency and stability of groupwise spatial convolutions.

3.3. Hierarchical Swin Architectures for SR

We adopt a hierarchical encoder–decoder architecture inspired by Swin backbones but replace the encoder and decoder self-attention stages with lightweight convolutional feature mixers. The input is first tokenized by a patch embedding layer (stride-p convolution + LayerNorm), producing a $(H/p) \times (W/p)$ token grid. Each encoder and decoder stage applies several ConvEncoder Layer blocks, which follow the same pre-norm + residual design as transformer layers but use depthwise–pointwise convolutions for local spatial mixing. Patch Merging layers downsample by $2\times$ while doubling channels, forming a multi-scale hierarchy. The decoder mirrors this with Patch Expand layers and token-aligned skip connections, enabling cross-scale feature reuse and preserving detail. Exact padding keeps patch and window divisions consistent. Finally, the reconstruction head factors the upsampling ratio ($p \times s$) into successive PixelShuffle stages ($\times 2$ or $\times 3$) and a light convolutional refinement.

3.4. Hierarchical Token-Aligned Architecture

The network begins with a convolutional Patch Embed layer (stride-p) that tokenizes the input image into a feature grid of size $(H/p) \times (W/p)$ with embedding dimension C. The encoder then processes these tokens through successive stages, each composed of several convolutional blocks followed by a Patch Merging operation that downsamples by $2\times$ while doubling channel width. This yields a multiscale representation sequence $(H/4, W/4) \rightarrow (H/8, W/8) \rightarrow (H/16, W/16)$, $(H/4, W/4) \rightarrow (H/8, W/8) \rightarrow (H/16, W/16)$, as shown in Fig. 2. Skip connections are recorded before each merge and later reused in the decoder for spatially aligned feature fusion. To ensure consistent token geometry, both the patch stride and internal window size evenly divide the spatial dimensions through symmetric padding so that encoder and decoder tokens remain precisely aligned at every scale.

The proposed hierarchical design leads to efficient token utilization by progressively reducing the spatial resolution and, consequently, the number of tokens processed at deeper stages. If $H \times W$ denotes the number of tokens after the initial patch embedding, subsequent stages operate on $(H/2) \times (W/2)$ and $(H/4) \times (W/4)$ tokens, respectively, resulting in a $4\times$ reduction in token count per level. This hierarchical downsampling structure significantly lowers activation memory and computational complexity, while maintaining feature expressiveness through increased channel dimensionality.

During training, we employ a patch stride of $p = 32$, yielding an initial token grid of $(I_H = 32) \times (I_W = 32)$ for $2\times$ SR. At inference, the model seamlessly generalizes to arbitrary input sizes while preserving token alignment through symmetric padding and window consistency.

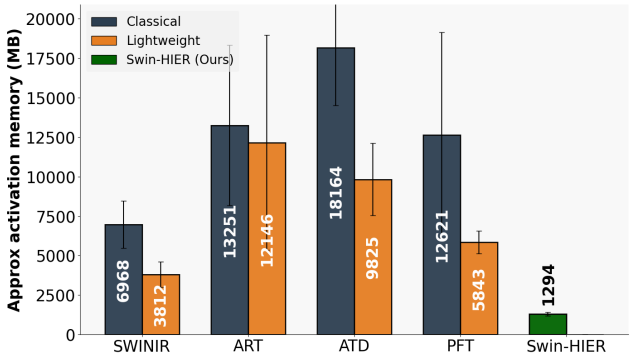


Figure 3. Activation Memory for input size 256 × 256.

The hierarchical token compression substantially decreases activation memory footprint and inference latency, providing a favorable efficiency–performance balance. These properties make the model particularly suitable for deployment on edge and low-power devices, where memory and compute resources are limited. As shown in Fig. 3, the proposed design achieves competitive accuracy while maintaining considerably lower peak activation memory compared to other lightweight transformer and CNN baselines.

3.5. Convolution-based Mixing

To reduce the cost and instability of attention, we replace each MHSA block with a lightweight ConvEncoder Layer, which preserves the transformer interface but performs spatial and channel mixing entirely through convolutions. Specifically, each block applies Layer Normalization, then reshapes tokens back to feature maps and performs:

1. A depthwise 3×3 convolution to capture local spatial context independently per channel, and
2. A pointwise 1×1 convolution for cross-channel communication.

The output is flattened back into tokens and passed through a residual MLP projection. This combination effectively substitutes windowed self-attention with an inductive, convolutional alternative that maintains locality and linear complexity while remaining drop-in compatible with transformer-based architectures.

Fig. 4 illustrates the internal design of the proposed ConvEncoder layer, which serves as a drop-in replacement for MHSA. Unlike standard transformer layers that compute dense token-wise correlations, the ConvEncoder performs spatial mixing directly in the feature domain through depthwise and pointwise convolutions. This design preserves locality and structural awareness, significantly reducing quadratic attention overhead while maintaining compatibility with the hierarchical token architecture. In practice, the ConvEncoder achieves comparable representational capacity with linear complexity and improved numerical stability during training.

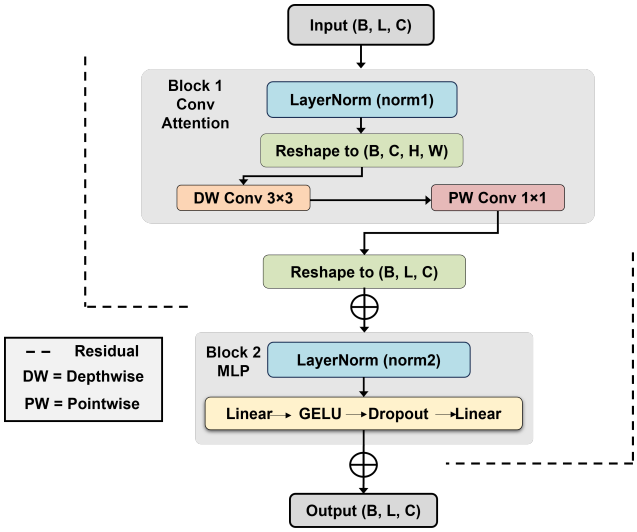


Figure 4. **Detailed structure of the ConvEncoder layer.** Each block replaces the windowed self-attention (WSA) with an efficient depthwise–pointwise convolutional mixer while preserving the same residual and normalization layout as transformer layers. The input tokens are first normalized and reshaped into feature maps, followed by a depthwise 3×3 convolution for local spatial aggregation and a pointwise 1×1 convolution for channel interaction. A lightweight MLP projection and residual connection complete the block, maintaining the expressive capacity of attention at a fraction of the computational cost.

3.6. Reconstruction Head

After the final decoder, tokens are reshaped into spatial features via linear projection and a 3×3 fusion convolution:

$$F = \mathrm{Conv}_{3\times3}(\mathrm{Reshape}(T_0)), \quad F \in \mathbb{R}^{C_0 \times H_0 \times W_0}. \quad (2)$$

The feature map F is then upsampled to the target scale r using a modular PixelShuffle-based head.

3.7. Computational Complexity

For a feature map (H, W) with C channels, our convolutional mixing block consists of a depthwise 3×3 convolution followed by a pointwise 1×1 convolution. The depthwise part costs

$$\mathcal{O}(\mathrm{DW}) = k^2 HWC, \quad k = 3,$$

while the pointwise part costs

$$\mathcal{O}(\mathrm{PW}) = HWC^2.$$

Since typically $C \gg k^2$, the overall complexity is dominated by the pointwise layer:

$$\mathcal{O}(\mathrm{ConvMix}) = HWC^2.$$

In contrast, WSA with window size M splits the feature map into $\frac{HW}{M^2}$ windows and performs attention within each window. Its total complexity consists of two components: the QKV and output projection layers contribute $\mathcal{O}(HWC^2)$, while the attention computation within each $M \times M$ window contributes $\mathcal{O}(HWM^2C)$. The total complexity is thus

$$\mathcal{O}(\text{WSA}) = \mathcal{O}(HWC^2 + HWM^2C).$$

Both approaches scale linearly with the number of spatial tokens HW. However, because each stage in Swin-HIER halves (H, W) and doubles C,

$$H_{i+1}W_{i+1}C_{i+1}^2 = \frac{H_i}{2} \cdot \frac{W_i}{2} \cdot (2C_i)^2 = H_iW_iC_i^2.$$

the dominant HWC^2 term remains constant across stages for both methods. The key difference lies in the window-size dependency: our convolutional replacement eliminates the $\mathcal{O}(HWM^2C)$ term present in WSA. When the window size M is significant (e.g., $M = 8$ as commonly used in Swin-based architectures), this term adds $64C$ operations per spatial location. Thus, the proposed approach achieves a complexity reduction of $\mathcal{O}(M^2C)$ per token compared to WSA, while maintaining the same stage-wise constant complexity property and linear scaling with respect to the number of tokens.

4. Experiments

4.1. Experimental Setup

Datasets and Benchmarks. We adopt the DF2K training set, which merges DIV2K [33] and Flickr2K [23], following common practice in recent SR works [22, 23, 43]. Our pipeline supports any scale factors, but we are only showing the results of $\times 2$, $\times 3$, and $\times 4$. LR–HR pairs are provided by the dataset loader, and on-the-fly patch sampling/augmentations follow the dataset defaults. Unless otherwise specified, we report both PSNR and SSIM scores on the Y-channel (luminance) of the BT.601 color space.

We use AdamW with learning rate 2×10^{-4}, $\beta = (0.9, 0.999)$, and zero weight decay. The loss is the Charbonnier loss [4]. Training runs for 300 epochs with a cosine learning-rate schedule and linear warmup of 5 epochs; the LR decays to a minimum of 1×10^{-6}. The batch size is 8 and gradient norms are clipped to 1.0. We enable mixed precision (FP16 when available, BF16 on supported GPUs) and TF32 for CUDA matmul/cuDNN to improve throughput on Ampere/Lovelace devices.

4.2. Comparison with State-of-the-Art Methods

4.2.1. Quantitative Evaluation on Benchmark Datasets

We evaluate our proposed Swin-HIER against recent state-of-the-art SISR models, including EDSR [23], RCAN [44], HAN [29], IPT [5], SwinIR [22], CAT-A [8], ART [7], HAT [41], IPG [32], ATD [42], and PFT [25]. Table 1 summarizes the quantitative results on the five standard SR benchmarks: Set5 [2], Set14 [40], BSD100 [26], Urban100 [16], and Manga109 [27], across upscaling factors $\times 2$, $\times 3$, and $\times 4$.

Despite having only 18.8M parameters and a much lower latency of 0.69 s, Swin-HIER consistently matches or closely approaches much larger transformer-based models across all scales. As shown in Table 1, it achieves comparable PSNR and SSIM on all benchmarks, with performance gaps typically below 0.5 dB and 0.005 SSIM relative to heavy-weight models such as HAT, PFT, and IPT (115 M params). This demonstrates that Swin-HIER attains a favorable balance between accuracy and efficiency, offering transformer-level performance with significantly reduced computational cost, making it well-suited for practical deployment.

4.2.2. Efficiency and Memory Behavior

Activation Memory. As shown in Table 2, transformer-based architectures such as ART and ATD incur heavy activation footprints ranging from 13 GB to 18 GB due to dense cross-window fusion. In contrast, Swin-HIER operates at merely 1.3 GB, achieving a $10\times$ reduction in memory consumption. Even compared with lightweight SwinIR (3.8 GB), our model saves roughly threefold memory. This improvement is driven by two key design choices: (i) a strict token pyramid that halves resolution and doubles channels per stage, keeping per-stage compute balanced; and (ii) the TASF module, which fuses features only at matched token geometries, eliminating redundant buffering and ensuring stable inter-stage alignment. Together, they form a highly memory-coherent transformer that scales efficiently to high-resolution inference.

Latency. Table 2 reports the inference latency of the $2\times$ super-resolution models on 10 randomly chosen DIV2K validation LR images, evaluated on a single NVIDIA A100 GPU. Swin-HIER achieves **0.69 s**, significantly faster than SwinIR (1.63 s), ATD (10.8 s), and ART (19.1 s). .

4.2.3. Deployment on NVIDIA Jetson AGX Orin

To evaluate real-world edge deployment feasibility, we benchmarked representative $\times 2$ SR models on the NVIDIA Jetson AGX Orin platform. We report end-to-end inference latency (milliseconds) for increasing input resolutions (256, 320, 512, and 1024). For ART, both the standard and small (S) variants are tested using the official PyTorch checkpoints. OOM indicates out-of-memory during execution or engine conversion (TensorRT / ONNX).

As shown in Table 3, Swin-HIER achieves dramatically lower latency across all tested resolutions. At 256 resolution, our model runs in only 86.7 ms, compared to 1.33 s for

Table 1. Quantitative comparison with state-of-the-art SR methods on standard benchmarks. Our Swin-HIER achieves comparable or superior performance with significantly lower computational cost.

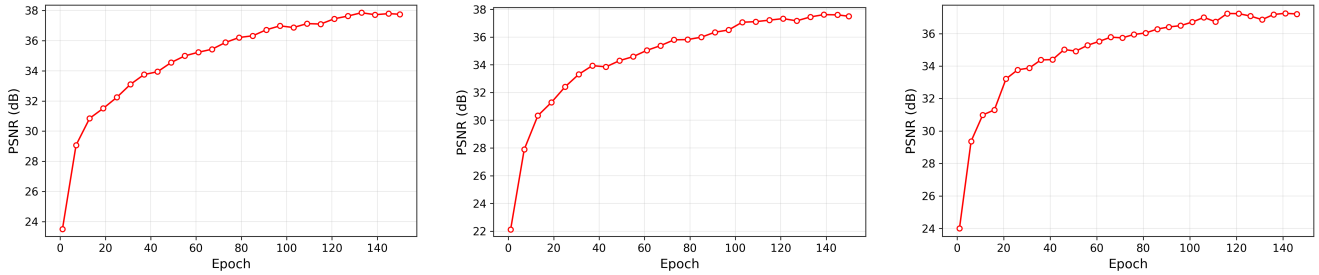
Method	Scale	Params	Set5		Set14		BSD100		Urban100		Manga109	
			PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Scale $\times 2$												
EDSR [23]	$\times 2$	42.6M	38.11	0.9602	33.92	0.9195	32.32	0.9013	32.93	0.9351	39.10	0.9773
RCAN [43]	$\times 2$	15.4M	38.27	0.9614	34.12	0.9216	32.41	0.9027	33.44	0.9384	39.44	0.9786
HAN [29]	$\times 2$	63.6M	38.27	0.9614	34.16	0.9217	32.41	0.9027	33.35	0.9385	39.46	0.9785
IPT [5]	$\times 2$	115M	38.37	–	34.43	–	32.48	–	33.76	–	–	–
SwinIR [22]	$\times 2$	11.8M	38.42	0.9623	34.46	0.9250	32.53	0.9041	33.81	0.9433	39.92	0.9797
CAT-A [8]	$\times 2$	16.5M	38.51	0.9626	34.78	0.9265	32.59	0.9047	34.26	0.9440	40.10	0.9805
ART [41]	$\times 2$	16.4M	38.56	0.9629	34.59	0.9267	32.58	0.9048	34.30	0.9452	40.24	0.9808
HAT [7]	$\times 2$	20.6M	38.63	0.9630	34.86	0.9274	32.62	0.9053	34.45	0.9466	40.26	0.9809
IPG [32]	$\times 2$	18.1M	38.61	0.9622	34.73	0.9270	32.60	0.9052	34.48	0.9464	40.24	0.9810
ATD [42]	$\times 2$	20.1M	38.61	0.9629	34.95	0.9276	32.65	0.9056	34.70	0.9476	40.37	0.9810
PFT [25]	$\times 2$	19.6M	38.68	0.9635	35.00	0.9280	32.67	0.9058	34.90	0.9490	40.49	0.9815
Swin-HIER (Ours)	$\times 2$	18.8M	37.85	0.9619	33.51	0.9195	32.11	0.9026	31.73	0.9259	38.38	0.9770
Scale $\times 3$												
EDSR [23]	$\times 3$	43.0M	34.65	0.9280	30.52	0.8462	29.25	0.8093	28.80	0.8653	34.17	0.9476
RCAN [43]	$\times 3$	41.6M	34.74	0.9299	30.65	0.8482	29.32	0.8111	29.09	0.8702	34.44	0.9449
HAN [29]	$\times 3$	63.4M	34.75	0.9299	30.67	0.8483	29.32	0.8110	29.10	0.8705	34.48	0.9500
IPT [5]	$\times 3$	116M	34.81	–	30.85	–	29.38	–	29.49	–	–	–
SwinIR [22]	$\times 3$	11.9M	34.97	0.9318	30.93	0.8534	29.46	0.8145	29.75	0.8826	35.12	0.9537
CAT-A [8]	$\times 3$	16.6M	35.06	0.9326	31.04	0.8538	29.52	0.8160	30.12	0.8862	35.38	0.9546
ART [41]	$\times 3$	16.6M	35.07	0.9325	31.02	0.8541	29.51	0.8159	30.10	0.8871	35.39	0.9548
HAT [7]	$\times 3$	20.8M	35.07	0.9329	31.08	0.8555	29.54	0.8159	30.23	0.8896	35.53	0.9552
IPG [32]	$\times 3$	18.3M	35.10	0.9332	31.10	0.8554	29.53	0.8168	30.36	0.8901	35.53	0.9554
ATD [42]	$\times 3$	20.3M	35.11	0.9330	31.13	0.8556	29.57	0.8176	30.46	0.8917	35.63	0.9558
PFT [25]	$\times 3$	19.8M	35.15	0.9333	31.16	0.8561	29.58	0.8178	30.56	0.8931	35.67	0.9560
Swin-HIER (Ours)	$\times 3$	19.9M	33.90	0.9273	30.13	0.8453	29.00	0.8100	27.79	0.8470	33.08	0.9425
Scale $\times 4$												
EDSR [23]	$\times 4$	43.0M	32.46	0.8968	28.80	0.7876	27.71	0.7420	26.64	0.8033	31.02	0.9148
RCAN [43]	$\times 4$	15.6M	32.63	0.9002	28.87	0.7889	27.77	0.7436	27.77	0.7436	31.22	0.9177
HAN [29]	$\times 4$	64.2M	32.64	0.9002	28.90	0.7890	27.80	0.7442	26.65	0.8094	31.42	0.9177
IPT [5]	$\times 4$	116M	32.64	–	29.01	–	27.82	–	27.26	–	–	–
SwinIR [22]	$\times 4$	11.9M	32.92	0.9044	29.09	0.7950	27.92	0.7489	27.45	0.8254	32.03	0.9260
CAT-A [8]	$\times 4$	16.6M	33.08	0.9052	29.18	0.7960	27.98	0.7510	27.87	0.8339	32.39	0.9285
ART [41]	$\times 4$	16.6M	33.04	0.9051	29.16	0.7958	27.97	0.7510	27.77	0.8321	31.93	0.9283
HAT [7]	$\times 4$	20.8M	33.04	0.9056	29.23	0.7973	28.00	0.7571	27.97	0.8368	32.48	0.9292
IPG [32]	$\times 4$	17.0M	33.15	0.9062	29.24	0.7973	27.99	0.7519	28.13	0.8392	32.53	0.9300
ATD [42]	$\times 4$	20.3M	33.10	0.9058	29.24	0.7974	28.01	0.7526	28.17	0.8404	32.62	0.9306
PFT [25]	$\times 4$	19.8M	33.15	0.9065	29.29	0.7978	28.02	0.7527	28.20	0.8412	32.63	0.9306
Swin-HIER (Ours)	$\times 4$	20.2M	31.54	0.8902	28.30	0.7819	27.39	0.7387	25.49	0.7683	29.60	0.8973

ART-S and 4.46 s for full ART. The gap widens at 320 resolution, where ART exceeds 16 seconds per image, while Swin-HIER remains below 140 ms.

Moreover, ART and ART-S fail to scale to higher resolutions due to memory constraints during TensorRT/ONNX conversion, whereas Swin-HIER remains deployable up to 512 resolution. These results demonstrate that dense cross-window transformer architectures suffer from severe computational scaling on embedded hardware, while our hierarchical and geometry-aligned design maintains practical edge-level latency.

4.3. Ablation Study on Encoder Design

To understand the impact of our Convolution based encoder decoder design on reconstruction quality, we perform an ablation study with three variants of our backbone: (i) ConvEncoderLayer described in section 3.5,), (ii) a generic multi-head self-attention encoder (MHA), and (iii)



(a) Conv-based encoder (b) MHA-based encoder (c) Swin-based encoder

Figure 5. Validation PSNR (dB) vs. training epoch for different encoder architectures on DF2K ×2 SR (Y-channel).

Table 2. Activation memory and inference latency comparison across methods (lower is better). Swin-HIER achieves the lowest memory usage and fastest runtime among all listed methods.

Method	**Activation Memory (GB)** ↓	**Latency (s)** ↓
SwinIR [22] - C	6.96	5.32
SwinIR [22] - L	3.80	2.68
ART [41] - C	13.25	19.63
ART [41] - L	12.14	7.20
HAT [7] - C	24.99	34.39
ATD [42] - C	18.16	10.82
ATD [42] - L	9.82	4.71
PFT [25] - C	12.62	15.74
PFT [25] - L	5.84	8.12
Swin-HIER	**1.29**	**0.69**

Table 3. Inference latency (ms) on Jetson AGX Orin for ×2 SR at different input resolutions. OOM indicates out-of-memory during execution or engine conversion (TensorRT/ONNX).

Model	**256 × 256**	**320×320**	**512×512**	**1024×1024**
Swin-HIER (Ours)	86.73	139.73	362.41	OOM
ART (×2)	4464.81	16716.10	OOM (TRT)	OOM (ONNX)
ART-S (×2)	1325.28	2922.05	OOM (TRT)	OOM (ONNX)

Table 4. Encoder ablation on DF2K ×2 SR (Y-channel). We report the best validation PSNR (dB) over 150 epochs.

Block Type	Best PSNR (dB)	ΔPSNR vs. Conv (dB)
Conv	**38.10**	–
MHA	37.66	-0.44
Swin	37.43	-0.67

a Swin-style windowed-attention encoder (Swin). All three models share the same loss, optimization hyperparameters, and training protocol described in Sec. 4.1; only the encoder/decoder block type is changed with these given type of layers. We train each variant on DF2K for ×2 SR for 150 epochs and report validation PSNR/SSIM on the Y-channel. Fig. 5 shows the evolution of PSNR as a function of training epoch for all encoder decoder block choices shown in Fig. 2, while Table 4 summarizes the corresponding best-validation scores.

All three variants exhibit a similar behavior in Fig. 5: a rapid PSNR increase in the first 20–30 epochs followed by a slower, near-logarithmic improvement as training progresses. However, their final performance differs noticeably. As reported in Table 4, the Conv encoder achieves the highest best-validation PSNR of **38.10 dB** (reached at epoch 141), while the MHA and Swin encoders saturate at 37.66 dB and 37.43 dB, respectively. Thus, the convolutional encoder outperforms the MHA and Swin counterparts by about +0.44 dB and +0.67 dB, even though all three models are trained under identical conditions. The Conv variant also exhibits slightly smoother convergence in the later stages of training, with less epoch-to-epoch fluctuation in PSNR compared to the attention-based encoders.

These results suggest that, in our hierarchical setting, a convolutional encoder provides a more effective inductive bias for SR than directly applying dense self-attention. We hypothesize that local convolutional processing offers stronger priors for edge and texture reconstruction, and that these locally structured features are better aligned with our token hierarchy and TASF fusion than features produced by globally mixed attention. In contrast, the MHA and Swin encoders tend to reach a plateau earlier and yield slightly noisier validation curves as shown in Fig. 5, indicating a weaker optimization landscape for high-PSNR refinement. Based on this, we selected the convolutional encoder as the default configuration in Swin-HIER. As discussed earlier, it consistently provides the best balance between reconstruction accuracy and training stability across all experiments.

4.4. Qualitative Comparison

Although Table 1 reports competitive PSNR/SSIM scores for Swin-HIER, it is known that higher PSNR does not necessarily translate into higher perceptual quality, especially in the high-PSNR regime. We therefore complement the quantitative evaluation with a qualitative comparison and an analysis of reconstruction variability to better understand the visual behavior and stability of our model.

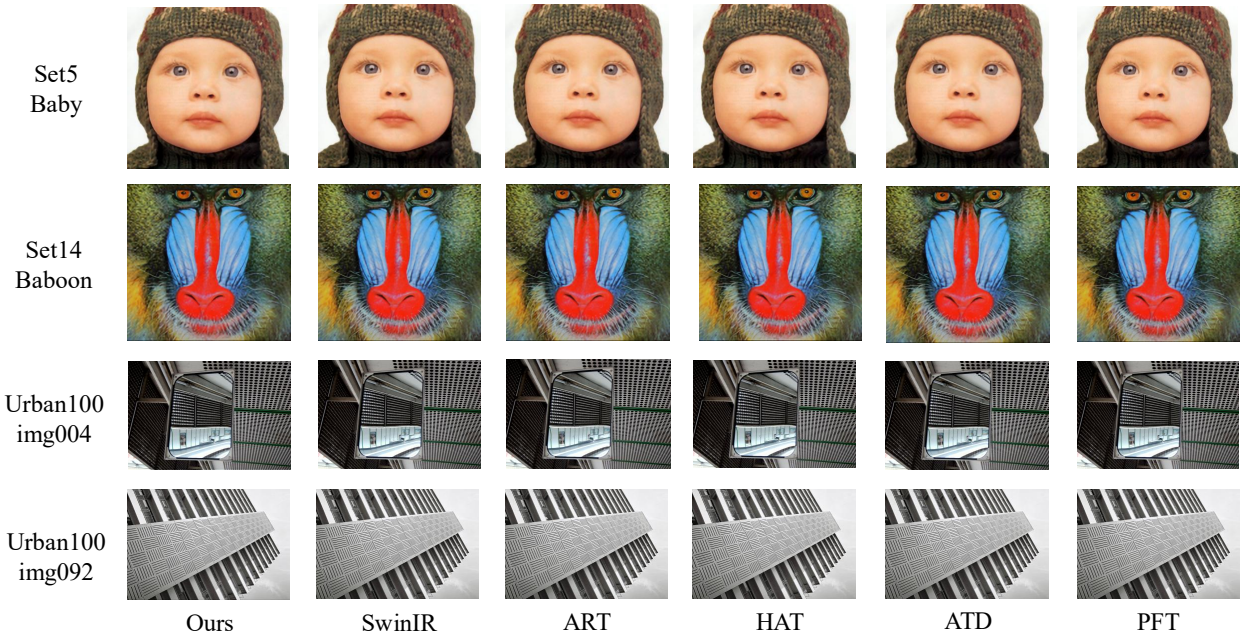


Figure 6. Qualitative comparison of different methods on Set5, Set14, and Urban100 for $\times 4$ scale. Our results are comparable to state-of-the-art approaches, demonstrating consistent reconstruction quality.

To better understand the perceptual characteristics of our model, we show a qualitative comparison of $\times 4$ scale against recent transformer-based SR approaches, including SwinIR, ART, HAT, ATD, and PFT. As shown in Fig. 6, our approach demonstrates remarkable structural coherence and stability across diverse scenes. Specifically, in challenging textures such as the Baboon hair region or the fine grids of Urban100 images, Swin-HIER tends to generate smoother and more consistent patterns, avoiding the over-sharpening and artificial ringing artifacts often observed in HAT or ART. This behavior stems from our Conv-based local reasoning and TASF, which emphasize structure-aware representation and feature consistency across scales rather than extreme local contrast. Overall, the qualitative results indicate that our hierarchical and token-aligned design achieves a balanced trade-off between locality preservation, cross-window consistency, and perceptual naturalness. Consistent with these observations, our variability analysis (see supplementary materials), shows that Swin-HIER produces more stable reconstructions, with lower variance across images and random seeds, than recent transformer-based baselines, which aligns with the visually smoother and less artifact-prone textures observed in Fig. 6. We acknowledge that further enhancing high-frequency detail recovery remains an open challenge, and we plan to extend our approach with frequency-aware attention and adaptive sharpening priors to better capture ultra-fine textures in future work.

5. Conclusion

In this work, we introduced **Swin-HIER**, a geometry-aligned hierarchical transformer for efficient single-image super-resolution. A central design choice is the use of a convolutional encoder–decoder built from depthwise and pointwise convolution layers instead of dense self-attention blocks. These lightweight convolutions provide a strong inductive bias for edges and textures and align naturally with the multi-scale token hierarchy and TASF, enabling stable optimization and high-fidelity reconstruction without the heavy computational burden of global attention. Extensive experiments on standard SR benchmarks show that Swin-HIER achieves competitive PSNR/SSIM while significantly reducing activation memory and computational cost compared with recent transformer-based models. In particular, Swin-HIER achieves an inference latency of 0.69 s, corresponding to roughly $4\times$ and $8\times$ speedup over SwinIR variants and up to about $50\times$ faster than heavy transformer baselines such as HAT. These results highlight that carefully structured convolutional hierarchies, combined with geometry-aware token alignment, can deliver transformer-level performance with substantially improved efficiency.

References

[1] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 252–268, 2018. 2

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

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 1, 2

[4] Pierre Charbonnier, Laure Blanc-Féraud, Gilles Aubert, and Michel Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Transactions on image processing*, 6(2):298–311, 1997. 5

[5] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12299–12310, 2021. 1, 2, 5, 6

[6] Xiang Chen, Hao Li, Mingqiang Li, and Jinshan Pan. Learning a sparse transformer network for effective image deraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5896–5905, 2023. 2

[7] Xiangyu Chen, Xintao Wang, Jiantao Zhou, Yu Qiao, and Chao Dong. Activating more pixels in image super-resolution transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22367–22377, 2023. 1, 2, 5, 6, 7

[8] Zheng Chen, Yulun Zhang, Jinjin Gu, Linghe Kong, Xin Yuan, et al. Cross aggregation transformer for image restoration. *Advances in Neural Information Processing Systems*, 35:25478–25490, 2022. 1, 2, 5, 6

[9] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019. 2

[10] Haram Choi, Jeongmin Lee, and Jihoon Yang. N-gram in swin transformers for efficient lightweight image super-resolution, 2023. 3

[11] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11065–11074, 2019. 1

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019. 2

[13] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015. 1, 2

[14] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network, 2016. 2

[15] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 2

[16] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5197–5206, 2015. 5

[17] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the 27th acm international conference on multimedia*, pages 2024–2032, 2019. 2

[18] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016. 2

[19] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution, 2016. 2

[20] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution, 2017. 2

[21] Wenbo Li, Kun Zhou, Lu Qi, Nianjuan Jiang, Jiangbo Lu, and Jiaya Jia. Lapar: Linearly-assembled pixel-adaptive regression network for single image super-resolution and beyond. *Advances in Neural Information Processing Systems*, 33:20343–20355, 2020. 2

[22] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1833–1844, 2021. 1, 2, 5, 6, 7

[23] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017. 1, 2, 5, 6

[24] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 1, 2

[25] Wei Long, Xingyu Zhou, Leheng Zhang, and Shuhang Gu. Progressive focused transformer for single image super-resolution. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2279–2288, 2025. 2, 5, 6, 7

[26] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and

measuring ecological statistics. In *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, pages 416–423. IEEE, 2001. 5

[27] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia tools and applications*, 76(20):21811–21838, 2017. 5

[28] Yiqun Mei, Yuchen Fan, and Yuqian Zhou. Image super-resolution with non-local sparse attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3517–3526, 2021. 2

[29] Ben Niu, Weilei Wen, Wenqi Ren, Xiangde Zhang, Lianping Yang, Shuzhen Wang, Kaihao Zhang, Xiaochun Cao, and Haifeng Shen. Single image super-resolution via a holistic attention network. In *European conference on computer vision*, pages 191–207. Springer, 2020. 5, 6

[30] Go Ohtani, Ryu Tadokoro, Ryosuke Yamada, Yuki M. Asano, Iro Laina, Christian Rupprecht, Nakamasa Inoue, Rio Yokota, Hirokatsu Kataoka, and Yoshimitsu Aoki. Rethinking image super-resolution from training data perspectives, 2024. 2

[31] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2790–2798, 2017. 2

[32] Yuchuan Tian, Hanting Chen, Chao Xu, and Yunhe Wang. Image processing gnn: Breaking rigidity in super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24108–24117, 2024. 1, 2, 5, 6

[33] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 114–125, 2017. 5

[34] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021. 1, 2

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. 1, 2

[36] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 568–578, 2021. 1, 2

[37] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17683–17693, 2022. 1, 2

[38] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297, 2020. 2

[39] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5728–5739, 2022. 1

[40] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010. 5

[41] Jiale Zhang, Yulun Zhang, Jinjin Gu, Yongbing Zhang, Linghe Kong, and Xin Yuan. Accurate image restoration with attention retractable transformer. *arXiv preprint arXiv:2210.01427*, 2022. 2, 5, 6, 7

[42] Leheng Zhang, Yawei Li, Xingyu Zhou, Xiaorui Zhao, and Shuhang Gu. 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*, pages 2856–2865, 2024. 1, 2, 5, 6, 7

[43] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 286–301, 2018. 1, 2, 5, 6

[44] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2472–2481, 2018. 1, 2, 5