Enriched CNN-Transformer Feature Aggregation Networks for Super-Resolution

Jinsu Yoo[1*] Taehoon Kim[2] Sihaeng Lee[2] Seung Hwan Kim[2] Honglak Lee[2] Tae Hyun Kim[1]

[1]Hanyang University [2]LG AI Research

Abstract

Recent transformer-based super-resolution (SR) methods have achieved promising results against conventional CNN-based methods. However, these approaches suffer from essential shortsightedness created by only utilizing the standard self-attention-based reasoning. In this paper, we introduce an effective hybrid SR network to aggregate enriched features, including local features from CNNs and long-range multi-scale dependencies captured by transformers. Specifically, our network comprises transformer and convolutional branches, which synergetically complement each representation during the restoration procedure. Furthermore, we propose a cross-scale token attention module, allowing the transformer branch to exploit the informative relationships among tokens across different scales efficiently. Our proposed method achieves state-of-the-art SR results on numerous benchmark datasets.

1. Introduction

Super-resolution (SR) is a longstanding problem that aims to restore a high-resolution (HR) image from the given low-resolution (LR) image. With the advancements in deep learning, various CNN-based SR methods are introduced [10, 23, 25, 29, 34, 36, 48]. The emergence of CNN-based SR networks have verified the efficiency in processing 2D images with the inductive bias (*i.e.*, local connectivity and translation invariance). However, such architectures have certain limitations in exploiting global information [11] or restoring weak texture details [6].

To solve the problem, SR researchers [3, 6, 27, 28] have recently applied Vision Transfomer (ViT) [11] to SR architectures. A self-attention mechanism, the core component of ViT, enables the network to capture long-range spatial dependencies within an image. In particular, ViT inherently contains superiority over CNN in exploiting self-similar patches within the input image by calculating similarity among tokens (patches) over the entire image region. Built upon standard [11] or sliding window-based [31] self-attention, such transformer-based SR networks have remarkably improved the restoration performance.

However, existing approaches [3, 6, 27, 28] suffer from features limitedly extracted through a certain type of self-attention mechanism. Accordingly, the networks cannot utilize various sets of features, such as local or multi-scale features, which are proven effective for SR [17, 35]. This problem raises three concerns in building transformer-based SR architecture. First, although ViTs extract non-local dependencies better, CNN is still a preferable way to efficiently leverage repeated local information within an image [7, 12, 45]. Next, restoring images solely with tokenized image patches can cause undesired artifacts at the token boundaries. While tokenization with large overlapping alleviates such a problem, this approach will considerably increase the computational cost of self-attention. Finally, tokenization with an identical token size limits the exploitation of multi-scale relationships among tokens. Notably, reasoning across different scaled patches (tokens) is exceptionally beneficial in SR [35,39] as it utilizes the internal self-similar patches [51] within an image.

In this paper, we propose to **A**ggregate enriched features extracted from both **C**NN and **T**ransformer (**ACT**) mechanisms and introduce an effective hybrid architecture that takes advantage of multi-scale local and non-local information. Specifically, we construct two different branches (*i.e.*, CNN and transformer branches) and fuse the intermediate representations during the SR procedure. Consequently, local features extracted from the CNN branch and long-range dependencies captured in the transformer branch are progressively fused to complement each other and extract robust features. Furthermore, we propose a **C**ross-**S**cale **T**oken **A**ttention module (**CSTA**) inside the transformer branch, which overcomes the limitation of prior

*Work done while interning at LG AI Research. Code is available at: https://github.com/jinsuyoo/act.

transformer-based SR methods [3,6,28] in exploiting multi-scale features. Inspired by re-tokenization [46], CSTA efficiently generates multi-scale tokens and enables the network to learn multi-scale relationships. Lastly, we investigate the necessity of commonly used techniques with transformers, such as positional embeddings, for SR task.

The proposed network, ACT, achieves state-of-the-art SR performances on several benchmark datasets and even outperforms recent transformer-based SR methods [6, 28]. Notably, our ACT extraordinarily improves SR quality when test image contains numerous repeating patches. To sum up, our contributions are presented as follows:

- We introduce a novel hybridized SR method, combining CNN and ViT, to effectively aggregate an enriched set of local and non-local features.
- We propose a cross-scale token attention module to leverage multi-scale token representations efficiently.
- Extensive experiments on numerous benchmark SR datasets demonstrate the superiority of our method.

2. Related Work

CNN-based SR: Several SR architectures have been designed upon convolutional layer to extract beneficial features from a given LR image [8, 10, 17, 23, 25, 29, 36, 48]. In particular, researchers have focused on exploiting a non-local internal information (*i.e*., patch-recurrence [51]) within an image [30, 34, 35, 49, 50]. This is achieved by adding various modules such as recurrent-based module [30, 35] or graph neural networks [50]. Constructed on top of baseline architectures (*e.g*., EDSR [29] or RCAN [48]), these methods improve the SR performance further. Our work is inspired by the recent success of ViT [11] in exploiting global representations within an image. Unlike previous studies, we separate CNN and Transformer branches to leverage local and non-local features individually and fuse the intermediate representations to compensate for each other.

ViT-based SR: Recent ViT-based networks have escalated the performance for various computer vision tasks [4, 12, 16, 26, 31, 31, 37, 43, 45]. Moreover, researchers have explored building the architecture for image restoration [6, 28, 44]. For image SR, Chen *et al*. [6] proposed a pure ViT-based network [11] to handle various image restoration tasks, including denoising, deraining, and SR. First, the network is pre-trained with a multi-task learning scheme, including the entire tasks. Then, the pre-trained network is fine-tuned to the desired task (*e.g*., $\times 2$ SR). Instead of the standard self-attention, Liang *et al*. [28] adapted the Swin Transformer block [31] and included convolutional layers inside the block to impose the local connectivity. Motivated to overcome the limitations within standard self-attention-based networks (*i.e*., IPT [6]), we model an effective hybridized architecture to aggregate enriched features across different scales.

Multi-scale ViTs: Many prior arts have studied multi-scale token representations for transformer [5, 42, 44]. Among them, Chen *et al*. [5] explicitly tokenized an image with different token sizes, while Wang *et al*. [42] constructed the pyramid architecture. Differently, we basically maintain a single token representation. Then, we utilize channel-wise splitting and re-tokenization [46] to efficiently generate multi-scale tokens and reason across them. Furthermore, our cross-scale attention aims to exploit self-similar patches across scales [51] within an input image.

Multi-branch architectures: Numerous works have constructed multi-branch networks [7,13–15,40] to handle several input data containing different information effectively. For SR, Jin *et al*. [22] and Isobe *et al*. [21] decomposed the input image/video into structural and texture information to advantageously restore the missing high-frequency details. In this study, we borrow the recent ViT to enhance global connectivity and model capacity.

3. Proposed Method

In this section, we present ACT, which leverages both CNN and transformer, in detail. ACT is composed of a shallow feature extraction module (head), transformer/CNN modules (body), and a high-quality image reconstruction module (tail). Figure 1 illustrates the overall structure.

3.1. Head

First, head module H_{Head} extracts shallow feature $\mathbf{F}_0$ from a given low-resolution input image $\mathbf{I}_{LR}$ as:

$$\mathbf{F}_0 = H_{Head}(\mathbf{I}_{LR}), \tag{1}$$

where H_{Head} is composed of two residual convolution blocks as suggested in previous works [6, 29, 48].

3.2. Body

Next, the extracted feature $\mathbf{F}_0$ is passed to the body module to acquire robust features as:

$$\mathbf{F}_{DF} = H_{Body}(\mathbf{F}_0) + \mathbf{F}_0, \tag{2}$$

where $\mathbf{F}_{DF}$ indicates deep feature, and H_{Body} contains proposed two-stream branches to extract residual features correspondingly.

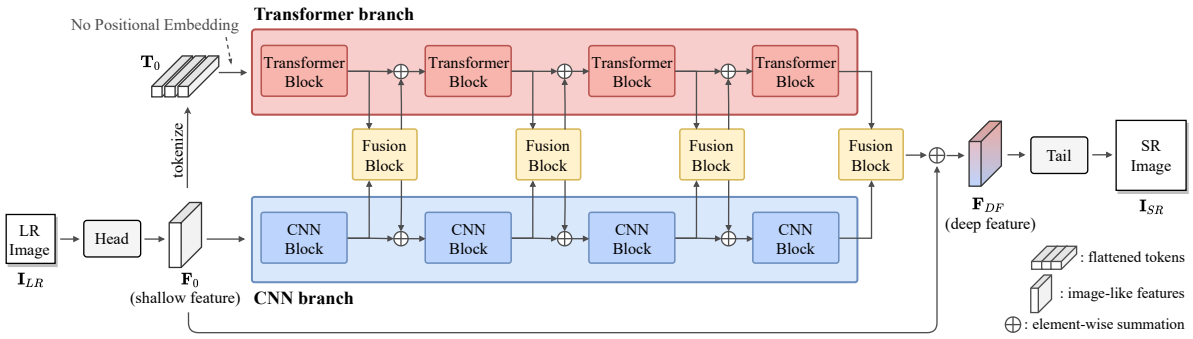


Figure 1: The overall flow of ACT. The input image goes through two separate branches constructed with a CNN and ViT. Each branch extracts local features and global representations and actively exchanges beneficial information during the intermediate fusion. The final SR result is acquired by aggregating enriched representations.

3.2.1 CNN branch

In the body module, convolutional kernels in the CNN branch slide over the image-like features with a stride of 1. Such a procedure compensates for the transformer branch's lack of intrinsic inductive bias. In building our CNN branch, we adopt the residual channel attention module (RCAB) [48], which has been widely used in the recent SR approaches [17, 36]. Specifically, we stack N CNN Blocks, which yields the following:

$$\mathbf{F}_i = H^i_{CB}(\mathbf{F}_{i-1}), \quad 1 \leq i \leq N, \tag{3}$$

where H^i_{CB} denotes CNN Block including RCAB modules, and $\mathbf{F}_i$, represents extracted CNN feature at ith CNN Block.

3.2.2 Transformer branch

We construct transformer branch in body module based on standard multi-head self-attention (MHSA) [6, 11]. Moreover, we propose to add cross-scale token attention (CSTA) modules in the transformer branch to exploit repeating structures within an input image across different scales.

First, we tokenize image-like shallow feature $\mathbf{F}_0 \in \mathbb{R}^{c \times h \times w}$ into non-overlapping n tokens $\mathbf{T}_0 \in \mathbb{R}^{n \times d}$, where d is dimension of each token vector. Notably, $n = {}^{h}/_{t} \times {}^{w}/_{t}$ and $d = c \cdot t^2$, where t is the token size. Moreover, unlike previous ViTs [6, 11], we observe that positional information becomes insignificant in SR. Thus, we do not add positional embeddings to tokens.

Then, acquired tokens are fed into the transformer branch, including N Transformer Blocks, which are symmetric to the CNN branch as:

$$\mathbf{T}_i = H^i_{TB}(\mathbf{T}_{i-1}), \quad 1 \leq i \leq N, \tag{4}$$

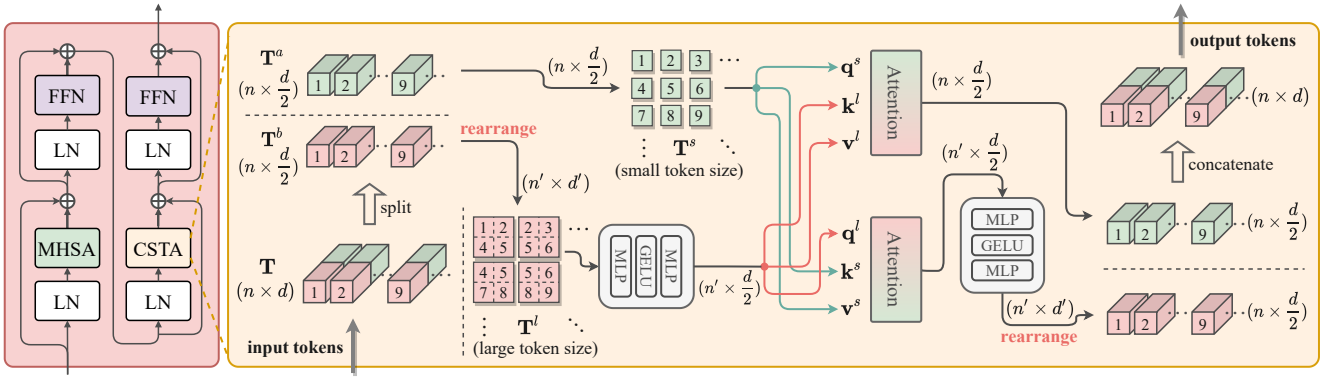
where $\mathbf{T}_i$ indicates extracted token representations at ith Transformer Block H^i_{TB}. As depicted in Figure 2a, each Transformer Block includes two sequential attention operations: multi-head self-attention (MHSA) [6, 11] and cross-scale token attention (CSTA), which yield:

$$\mathbf{T}_i = \text{FFN}(\text{CSTA}(\mathbf{T}'_{i-1})), \mathbf{T}'_{i-1} = \text{FFN}(\text{MHSA}(\mathbf{T}_{i-1})), \tag{5}$$

where FFN (feed forward network) includes two MLP layers with expansion ratio r with GELU activation function [19] in middle of the layers. Here, we omit layer normalization (LN) [1] and skip connections for brevity, and details of our CSTA are as follows.

Cross-scale token attention (CSTA): Standard MHSA [11, 41] performs attention operation by projecting queries, keys, and values from the same source of single-scale tokens. In addition to self-attention, we propose to exploit information from tokens across different scales, and we illustrate operational flow of proposed CSTA module in Figure 2b. Concretely, we split input token embeddings $\mathbf{T} \in \mathbb{R}^{n \times d}$ of CSTA along the last (*i.e.*, channel) axis into two, and we represent them as $\mathbf{T}^a \in \mathbb{R}^{n \times d/2}$ and $\mathbf{T}^b \in \mathbb{R}^{n \times d/2}$. Then, we generate $\mathbf{T}^s \in \mathbb{R}^{n \times d/2}$ and $\mathbf{T}^l \in \mathbb{R}^{n' \times d'}$, which include n tokens from $\mathbf{T}^a$ and n' tokens by rearranging $\mathbf{T}^b$, respectively. In practice, we use $\mathbf{T}^a$ for $\mathbf{T}^s$ as is, and re-tokenize [46] $\mathbf{T}^b$ to generate $\mathbf{T}^l$ with larger token size and overlapping. Here, we can control number of tokens in $\mathbf{T}^l$ by setting as $n' = \left\lfloor \frac{h-t'}{s'} + 1 \right\rfloor \times \left\lfloor \frac{w-t'}{s'} + 1 \right\rfloor$, where s' is stride and t' denotes token size, and token dimension is $d' = ({}^{c}/_{2}) \cdot t'^2 = {}^{(d \cdot t'^2)}/_{2t^2}$. One can acquire numerous tokens of large size by overlapping, which enables the network to enjoy patch-recurrence across scales actively. Notably, large tokens are essential for reasoning repeating patches across scales during the CSTA procedure.

In particular, to effectively exploit self-similar patches across different scales and pass a larger patch's information to small but self-similar ones, we produce a smaller number of tokens (*i.e.*, $n' < n$) with a relatively larger token size (*i.e.*, $t' > t$), and then compute cross-scale attention scores between tokens in both $\mathbf{T}^s$ and $\mathbf{T}^l$.



(a) Transformer Block (b) Cross-Scale Token Attention (CSTA)

Figure 2: (a) Our Transformer Block includes multi-head self-attention (MHSA) and cross-scale token attention (CSTA). (b) CSTA effectively exploits information across different scaled tokens by channel-wise splitting and token rearrangement. Two different token embeddings exchange keys and values for the attention operation.

Specifically, we generate queries, keys and values in $\mathbf{T}^s$ and $\mathbf{T}^l$: ($\mathbf{q}^s \in \mathbb{R}^{n\times d/2}$, $\mathbf{k}^s \in \mathbb{R}^{n\times d/2}$, $\mathbf{v}^s \in \mathbb{R}^{n\times d/2}$) from $\mathbf{T}^s$, and ($\mathbf{q}^l \in \mathbb{R}^{n'\times d/2}$, $\mathbf{k}^l \in \mathbb{R}^{n'\times d/2}$, $\mathbf{v}^l \in \mathbb{R}^{n'\times d/2}$) from $\mathbf{T}^l$. Next, we carry out attention operation [41] using triplets $(\mathbf{q}^l, \mathbf{k}^s, \mathbf{v}^s)$ and $(\mathbf{q}^s, \mathbf{k}^l, \mathbf{v}^l)$ as inputs by exchanging key-value pairs from each other. Notably, last dimension of queries, keys, and values from $\mathbf{T}^l$ is lessened from d' to $d/2$ by projections for the attention operation, and attention result for $\mathbf{T}^l$ is re-projected to dimension of $n' \times d'$, then rearranged to dimension of $n \times \frac{d}{2}$. Finally, we generate an output token representation of CSTA by concatenating attention results.

Our CSTA can exploit cross-scale information without additional high overhead. More specifically, computation costs of MHSA and CSTA are as follows:

$$\begin{aligned} \mathcal{O}(MHSA) &= n^2 \cdot d + n \cdot d^2 \\ \mathcal{O}(CSTA) &= (n \cdot n') \cdot d + (n + n') \cdot d^2, \end{aligned} \tag{6}$$

and computational cost of CSTA is competitive with MHSA because $n > n'$.

The notion of our CSTA module is computing attention scores across scales in a high-dimensional feature space and explicit reasoning across multi-scale tokens. Thus, our network can utilize recurring patch information across different scales within the input image [35, 39], while conventional MHSA limitedly extract informative cross-scale cues.

3.2.3 Multi-branch feature aggregation

We bidirectionally connect intermediate features extracted from independent branches. Figure 3 depicts our Fusion Block. Concretely, given intermediate features $\mathbf{T}_i$ and $\mathbf{F}_i$ from ith CNN Block and Transformer Block, we aggregate feature maps by using Fusion Block H_{fuse} as:

$$\mathbf{M}_i = H^i_{fuse}(\texttt{rearrange}(\mathbf{T}_i) \,\|\, \mathbf{F}_i), \quad 1 \le i \le N, \tag{7}$$

where $\mathbf{M}_i \in \mathbb{R}^{2c\times h\times w}$ denotes fused features, and `rearrange` and $\|$ represent image-like rearrangement and concatenation, respectively. We build our Fusion Block H_{fuse} with 1×1 convolutional blocks for a channel-wise fusion. Except for last Fusion Block (*i.e.*, $i = N$), fused features $\mathbf{M}_i$ are split into two features along channel dimension, *i.e.*, $\mathbf{M}^T_i \in \mathbb{R}^{c\times h\times w}$ and $\mathbf{M}^F_i \in \mathbb{R}^{c\times h\times w}$, followed by MLP blocks and convolutional blocks, respectively. Then, each fused feature flows back to each branch and is individually added to the original input feature $\mathbf{T}_i$ and $\mathbf{F}_i$. Fused feature $\mathbf{M}_N$ at last Fusion Block takes a single 3×3 convolution layer to resize channel dimension from $2c$ to c. The extracted deep residual feature is added to $\mathbf{F}_0$ and produces a deep feature $\mathbf{F}_{DF}$. Subsequently, $\mathbf{F}_{DF}$ is transferred to the final tail module.

3.3. Tail

For the last step, aggregated feature $\mathbf{F}_{DF}$ is upscaled and reconstructed through tail module H_{Tail} and produces the final SR result as:

$$\mathbf{I}_{SR} = H_{Tail}(\mathbf{F}_{DF}). \tag{8}$$

H_{Tail} includes PixelShuffle [38] operation, which upscales feature maps by rearranging channel-wise features to the spatial dimension, followed by a single convolution layer to predict the final SR result.

4. Experiments

In this section, we quantitatively and qualitatively demonstrate the superiority of ACT.

4.1. Implementation details

Datasets and evaluation metrics: Following previous works [6, 10], we train our network with the ImageNet dataset [9]. Therefore, the transformer can fully utilize its

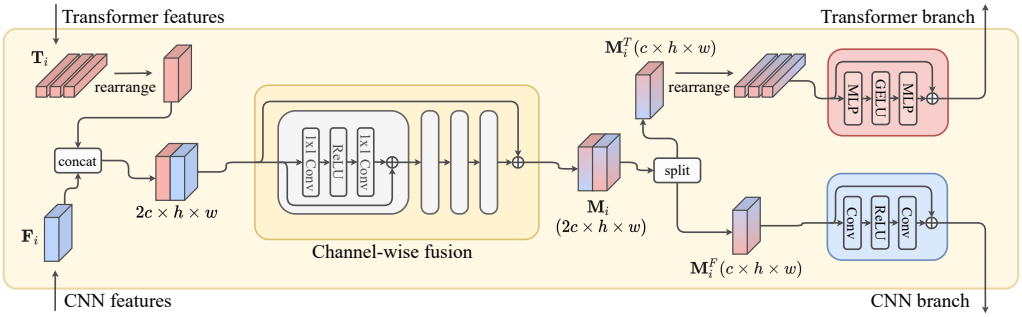


Figure 3: Our Fusion Block. Concatenated features from two branches are fused with 1×1 convolutions. Then, complemented information is bidirectionally transferred to the original branches.

representation capability [6]. Specifically, we use approximately 1.3M images such that the length of the shortest axis exceeds 400 pixels as ground-truth HR images. Input LR images are generated by downscaling HR images using bicubic interpolation. Moreover, we evaluate performance of SR networks on conventional SR benchmark datasets: Set5 [2], Set14 [47], B100 [32], Urban100 [20], and Manga109 [33]. The experimental results are evaluated with two metrics, namely peak signal-to-noise ratio (PSNR) and structural similarity (SSIM), on Y channel in YCbCr color space following baselines [6, 29, 48].

Hyperparameters: We have four CNN Blocks in the CNN branch and four Transformer Blocks in the transformer branch (*i.e.*, $N = 4$). Each CNN Block includes 12 RCAB modules with channel size $c = 64$. We use $d = 576$, and expansion ratio r inside FFN module is set to 4. For Fusion Block, we stack four 1×1 residual blocks [18]. During training, input LR patches are fixed to 48×48, and token size is 3×3 (*i.e.*, $t = 3$). We also use conventional data augmentation techniques, such as rotation ($90°$, $180°$, and $270°$) and horizontal flipping. Large token size t', stride s', and d' for CSTA module are set to 6, 3, and 1152, respectively. Moreover, we use Adam optimizer [24] to train our network and minimize L_1 loss following previous studies [6, 36, 48]. We train our network for 150 epochs with a batch size of 512. The initial learning rate is 10^{-4} ($\beta_1 = 0.9$ and $\beta_2 = 0.999$) and we reduce it by half every 50 epochs. We implement our model using the PyTorch framework with eight NVIDIA A100 GPUs.

4.2. Ablation studies

4.2.1 Impact of positional embeddings

Unlike high-level vision tasks (*e.g.*, classification), the transformer in SR utilizes a relatively small patch size. To investigate the necessity of positional embeddings for SR, we train two types of SR networks by using 1) MHSA instead of CSTA and removing the CNN branch (*i.e.*, standard ViT [11]) and 2) our ACT. Networks are trained with and without learnable positional embeddings [4, 6, 11], and results are provided in Table 1. We observe that transformer without positional embeddings does not degrade SR performance. According to our observation, we do not use positional embeddings in the remainder of our experiments.

	Standard ViT [11]		ACT (Ours)	
Learnable PE [6]	w/	w/o	w/	w/o
Set5 [2]	38.31	**38.33**	38.43	**38.46**
Set14 [47]	34.29	**34.33**	34.57	**34.60**

Table 1: Ablation on the impact of positional embeddings for SR, reported in PSNR value. PE indicates positional embeddings.

	Single-stream		**Two-stream**			
Transformer	w/	w/o	w/	w/	w/	w/
CNN	w/o	w/	w/	w/	w/	w/
Fusion	w/o	w/o	w/o	T $\rightarrow$ C	C $\rightarrow$ T	T $\leftrightarrow$ C
# Params.	32.8M	4.3M	36.6M	37.4M	45.1M	45.3M
Set14 [47]	34.33	34.21	34.32	34.38	34.44	**34.45**
Manga109 [33]	39.48	39.46	39.56	39.71	39.77	**39.85**

Table 2: Ablation experiments on various architectural choices w.r.t. PSNR metric. T and C means Transformer and CNN branches respectively. $\rightarrow$ indicates unidirectional flow from left to right, and $\leftrightarrow$ indicates bidirectional flow.

4.2.2 Impact of fusing strategies

In Table 2, we ablate various architectural choices related to multi-stream network. Specifically, we conduct experiments on each branch and fusion strategies. First, due to the large model capacity, a single-stream network with only a transformer branch performs better than a relatively lightweight single-stream CNN branch. Next, we observe that the performance of a two-stream network without Fusion Block consistently drops due to significantly separated pathways. However, performance is largely improved when intermediate features are unidirectionally fused. Finally, the proposed bidirectional fusion between CNN and trans-

Method	# Params.	Set14/Urban100
MHSA only	45.3M	34.45/33.93
MHSA + CSTA	46.0M	**34.60/34.07**
CSTA only	46.7M	34.51/33.92

(a) MHSA vs. CSTA. Utilizing both attentions performs better than MHSA or CSTA alone.

CSTA	w/o	w/		
Stride (s')	-	5	4	3
# Large tokens (n')	-	81	121	225
FLOPs	22.2G	**21.4G**	21.6G	22.2G
PSNR on Set14	34.45	34.51	34.55	**34.60**

(b) Effect of the number of large tokens $n^{'}$. CSTA efficiently boost performance.

# Scales	Token sizes	Set14/Urban100
1	(3)	34.45/33.93
2	(3, 6)	**34.60/34.07**
3	(3, 6, 12)	34.52/33.95

(c) Effect of various scaled tokens. CSTA with two scales performs best.

Table 3: Ablation experiments on CSTA module. We demonstrate the effectiveness of our proposed CSTA on various aspects w.r.t. PSNR metric.

former features (T $\leftrightarrow$ C) shows the best performance over the entire fusion strategy. The experimental results show that transformer and CNN branches contain complementary information, and intermediate bidirectional fusion is necessary for satisfactory restoration results.

4.2.3 Impact of CSTA

MHSA vs. CSTA: We compare our CSTA against standard MHSA in Table 3a. Specifically, we train ACT by replacing all CSTA/MHSA with MHSA/CSTA (*i.e.*, *MHSA only* and *CSTA only*). The comparison between *MHSA only* and *MHSA + CSTA* shows that CSTA largely boosts performance with a small number of additional parameters (+ 0.7M). Moreover, the result of *CSTA only* indicates that CSTA alone cannot cover the role of MHSA capturing self-similarity within the same scale (performs better than *MHSA only* but lower than *MHSA + CSTA*).

Impact of the number of large tokens: We conduct experiments to observe large tokens' impact and efficiency in Table 3b. Specifically, we vary sequence length of large token (*i.e.*, n') of $\mathbf{T}^l$ by controlling stride s'. The results show that our CSTA module, even with a small number of large tokens ($n' = 81$), efficiently outperforms conventional self-attention without cross-attention (*i.e.*, MHSA) by 0.06dB. Furthermore, performance improvement is remarkable when we increase the number of large tokens with a small overhead in terms of FLOPs. This experimental result demonstrates that CSTA can efficiently exploit informative cross-scale features with larger tokens.

Impact of more token scales: We investigate whether performing CSTA with more token sizes is beneficial or not in Table 3c. In doing so, we embed three token sizes (3, 6, and 12) with similar overall computational costs to the CSTA module. Comparing two token scales and three token scales shows that cross attention with an additional larger scale drops the performance. Since the number of recurring patches decreases as scale increases [51], we observe that exploiting self-similar patches across proper scales is more effective than solely performing certain types of attention or adding various scales.

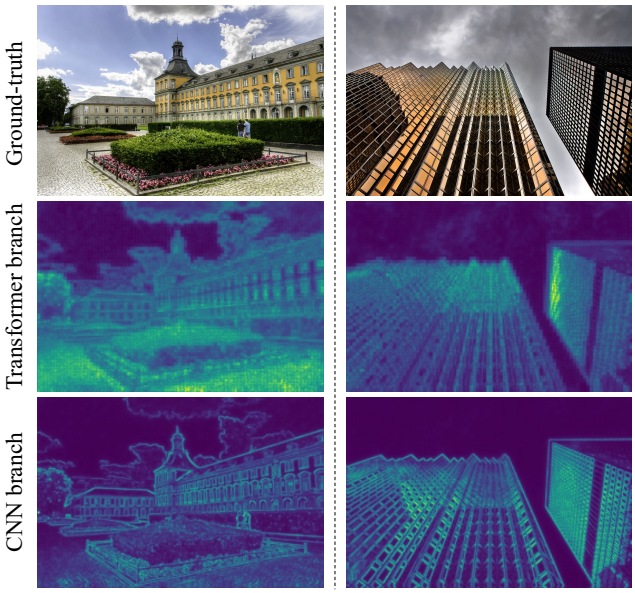


Figure 4: Feature map visualizations of transformer branch and CNN branch. The transformer branch focuses on restoring texture detail and repeated small patterns, while the CNN branch emphasizes the reconstruction of sharp and strong edges.

4.2.4 Feature visualization

In Figure 4, we visualize features to analyze the role of each branch. Specifically, we compare the output features from the last blocks for each branch (*i.e.*, $\mathbf{T}_4$ and $\mathbf{F}_4$). According to the visualization, both branches provide minimal attention to flat and low-frequency areas (*e.g.*, sky). However, the output feature from the transformer branch focuses on recovering tiny and high-frequency texture details, while producing blurry and checkerboard artifacts due to tokenization. Moreover, we observe that the transformer branch attends to a small version of recurring patches within the image (*e.g.*, upper side window in the right example), leveraging multi-scale representations with CSTA module.

Method	Scale	**Set5** [2]		**Set14** [47]		**B100** [32]		**Urban100** [20]		**Manga109** [33]	
		PSNR	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
EDSR [29]	$\times 2$	38.11	0.9602	33.92	0.9195	32.32	0.9013	32.93	0.9351	39.10	0.9773
RCAN [48]	$\times 2$	38.27	0.9614	34.12	0.9216	32.41	0.9027	33.34	0.9384	39.44	0.9786
RNAN [49]	$\times 2$	38.17	0.9611	33.87	0.9207	32.32	0.9014	32.73	0.9340	39.23	0.9785
SAN [8]	$\times 2$	38.31	0.9620	34.07	0.9213	32.42	0.9028	33.10	0.9370	39.32	0.9792
HAN [36]	$\times 2$	38.27	0.9614	34.16	0.9217	32.41	0.9027	33.35	0.9385	39.46	0.9785
NLSA [34]	$\times 2$	38.34	0.9618	34.08	0.9231	32.43	0.9027	33.42	0.9394	39.59	0.9789
IPT [6]	$\times 2$	38.37	-	34.43	-	32.48	-	33.76	-	-	-
SwinIR [28]	$\times 2$	38.42	0.9623	34.46	0.9250	32.53	0.9041	33.81	0.9427	39.92	0.9797
ACT (Ours)	$\times 2$	<u>38.46</u>	<u>0.9626</u>	<u>34.60</u>	<u>0.9256</u>	<u>32.56</u>	<u>0.9048</u>	<u>34.07</u>	<u>0.9443</u>	<u>39.95</u>	<u>0.9804</u>
ACT+ (Ours)	$\times 2$	**38.53**	**0.9629**	**34.68**	**0.9260**	**32.60**	**0.9052**	**34.25**	**0.9453**	**40.11**	**0.9807**
EDSR [29]	$\times 3$	34.65	0.9280	30.52	0.8462	29.25	0.8093	28.80	0.8653	34.17	0.9476
RCAN [48]	$\times 3$	34.74	0.9299	30.65	0.8482	29.32	0.8111	29.09	0.8702	34.44	0.9499
SAN [8]	$\times 3$	34.75	0.9300	30.59	0.8476	29.33	0.8112	28.93	0.8671	34.30	0.9494
HAN [36]	$\times 3$	34.75	0.9299	30.67	0.8483	29.32	0.8110	29.10	0.8705	34.48	0.9500
NLSA [34]	$\times 3$	34.85	0.9306	30.70	0.8485	29.34	0.8117	29.25	0.8726	34.57	0.9508
IPT [6]	$\times 3$	34.81	-	30.85	-	29.38	-	29.38	-	-	-
SwinIR [28]	$\times 3$	34.97	0.9318	30.93	0.8534	29.46	0.8145	29.75	0.8826	35.12	0.9537
ACT (Ours)	$\times 3$	<u>35.03</u>	<u>0.9321</u>	<u>31.08</u>	<u>0.8541</u>	<u>29.51</u>	<u>0.8164</u>	<u>30.08</u>	<u>0.8858</u>	<u>35.27</u>	<u>0.9540</u>
ACT+ (Ours)	$\times 3$	**35.09**	**0.9325**	**31.17**	**0.8549**	**29.55**	**0.8171**	**30.26**	**0.8876**	**35.47**	**0.9548**
EDSR [29]	$\times 4$	32.46	0.8968	28.80	0.7876	27.71	0.7420	26.64	0.8033	31.02	0.9148
RCAN [48]	$\times 4$	32.63	0.9002	28.87	0.7889	27.77	0.7436	26.82	0.8087	31.22	0.9173
RNAN [49]	$\times 4$	32.49	0.8982	28.83	0.7878	27.72	0.7421	26.61	0.8023	31.09	0.9149
SAN [8]	$\times 4$	32.64	0.9003	28.92	0.7888	27.78	0.7436	26.79	0.8068	31.18	0.9169
HAN [36]	$\times 4$	32.64	0.9002	28.90	0.7890	27.80	0.7442	26.85	0.8094	31.42	0.9177
NLSA [34]	$\times 4$	32.59	0.9000	28.87	0.7891	27.78	0.7444	26.96	0.8109	31.27	0.9184
IPT [6]	$\times 4$	32.64	-	29.01	-	27.82	-	27.26	-	-	-
SwinIR [28]	$\times 4$	32.92	**0.9044**	29.09	0.7950	27.92	0.7489	27.45	0.8254	32.03	0.9260
ACT (Ours)	$\times 4$	<u>32.97</u>	0.9031	<u>29.18</u>	<u>0.7954</u>	<u>27.95</u>	<u>0.7507</u>	<u>27.74</u>	<u>0.8305</u>	<u>32.20</u>	<u>0.9267</u>
ACT+ (Ours)	$\times 4$	**33.04**	<u>0.9041</u>	**29.27**	**0.7968**	**28.00**	**0.7516**	**27.92**	**0.8332**	**32.44**	**0.9282**

Table 4: Quantitative comparison of the proposed method with numerous state-of-the-art SR methods. The best and the second-best values are highlighted with **bold** and <u>underline</u>, respectively.

In contrast, the CNN branch recovers sharp and strong edges, which the transformer branch fails to capture. This observation indicates that ACT endowed independent roles for each pathway such that the two branches complement each other.

4.3. SR results

Quantitative evaluation: In Table 4, we quantitatively compare our ACT for $\times 2$, $\times 3$, and $\times 4$ SR tasks with eight state-of-the-art SR networks: EDSR [29], RCAN [48], RNAN [49], SAN [8], HAN [36], NLSA [34], IPT [6], and SwinIR [28]. We also report test-time self-ensembling-based results following baselines [28, 29, 36, 48] to improve performance further, and ACT+ indicates approach with self-ensemble [29]. Compared with all previous works, ACT and ACT+ achieve the best or second-best performance in terms of PSNR and SSIM for all scale factors. In particular, our method substantially outperforms IPT [6], which is recognized as the first transformer-based restoration approach, through proposed multi-scale feature extraction and effective hybridized architecture of CNN and transformer. Moreover, performance improvement over SwinIR [28] is considerable for Urban100 dataset [20] (more than 0.3dB PSNR gain for all scale factors) with high patch-recurrence in the dataset, indicating that our CSTA module successfully exploits multi-scale features.

Qualitative evaluation: We provide a qualitative comparison with existing SR methods. Figure 5 shows that our method obtains more accurately recovered details than conventional methods. Specifically, the restoration result of image "MisutenaideDaisy" demonstrates that our method can generate more human-readable characters than other existing methods. Moreover, by taking "barbara" as an example, baseline methods have generated sharp edges/patterns despite being far from the ground-truth structure. By contrast, our method correctly reconstructs the main structure without losing high-frequency details. The result of "img092", which contains an urban scene, shows that most conventional methods fail to recover the structure and produce

Figure 5: Visual comparison of the proposed method with various methods for $\times 4$ SR task. Our method restores sharp and complicated structures more accurately.

Measure	EDSR	RCAN	NLSA	IPT	SwinIR	ACT (Ours)
# Params.	43M	16M	44M	114M	12M	46M
FLOPs	116G	37G	125G	35G	29G	22G

Table 5: Comparison of the proposed method's resources with state-of-the-art SR methods.

blurry results. Meanwhile, our method alleviates blurring artifacts and accurately reconstructs correct contents. The above observation indicates the general superiority of the proposed method in recovering sharp and accurate details.

4.4. Model size analysis

Finally, we compare the number of network parameters and floating-point operations (FLOPs) of various SR methods in Table 5. Our ACT shows the best SR results as in Table 4 with competitive hardware resources in comparison with existing approaches, including IPT [6] and SwinIR [28]. Notably, although SwinIR [28] has few parameters, its computational cost is relatively high due to small window and token sizes, 8×8 and 1×1, respectively. The comparison demonstrates an effective trade-off between ACT's performance and model complexity.

5. Conclusion

In this study, we proposed to aggregate various beneficial features for SR and introduced a novel architecture combining transformer and convolutional branches, advantageously fusing both representations. Moreover, we presented an efficient cross-scale attention module to exploit multi-scale feature maps within a transformer branch. The effectiveness of the proposed method has been extensively demonstrated under numerous benchmark SR datasets, and our method records the state-of-the-art SR performance in terms of quantitative and qualitative comparisons.

Acknowledgments

This work was partially supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2022-0-00156, Fundamental research on continual meta-learning for quality enhancement of casual videos and their 3D metaverse transformation.)

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[2] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2012.

[3] Jiezhang Cao, Yawei Li, Kai Zhang, and Luc Van Gool. Video super-resolution transformer. *arXiv preprint arXiv:2106.06847*, 2021.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[5] Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[6] 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 (CVPR)*, 2021.

[7] Yinpeng Chen, Xiyang Dai, Dongdong Chen, Mengchen Liu, Xiaoyi Dong, Lu Yuan, and Zicheng Liu. Mobile-former: Bridging mobilenet and transformer. *arXiv preprint arXiv:2108.05895*, 2021.

[8] 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 (CVPR)*, 2019.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[10] 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 (PAMI)*, 2015.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[12] Stéphane d'Ascoli, Hugo Touvron, Matthew L Leavitt, Ari S Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. In *International Conference on Machine Learning (ICML)*, 2021.

[13] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[14] Christoph Feichtenhofer, Axel Pinz, and Richard Wildes. Spatiotemporal residual networks for video action recognition. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[15] Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Spatiotemporal multiplier networks for video action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[16] Jianyuan Guo, Kai Han, Han Wu, Chang Xu, Yehui Tang, Chunjing Xu, and Yunhe Wang. Cmt: Convolutional neural networks meet vision transformers. *arXiv preprint arXiv:2107.06263*, 2021.

[17] Yong Guo, Jian Chen, Jingdong Wang, Qi Chen, Jiezhang Cao, Zeshuai Deng, Yanwu Xu, and Mingkui Tan. Closed-loop matters: Dual regression networks for single image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[19] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[20] 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 (CVPR)*, 2015.

[21] Takashi Isobe, Xu Jia, Shuhang Gu, Songjiang Li, Shengjin Wang, and Qi Tian. Video super-resolution with recurrent structure-detail network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[22] Zhi Jin, Muhammad Zafar Iqbal, Dmytro Bobkov, Wenbin Zou, Xia Li, and Eckehard Steinbach. A flexible deep cnn framework for image restoration. *IEEE Transactions on Multimedia (TMM)*, 2019.

[23] 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 (CVPR)*, 2016.

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[25] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[26] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021.

[27] Jingyun Liang, Jiezhang Cao, Yuchen Fan, Kai Zhang, Rakesh Ranjan, Yawei Li, Radu Timofte, and Luc Van Gool. Vrt: A video restoration transformer. *arXiv preprint arXiv:2201.12288*, 2022.

[28] 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 Workshops (ICCVW)*, 2021.

[29] 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/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017.

[30] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[31] 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 (ICCV)*, 2021.

[32] 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 of the IEEE International Conference on Computer Vision (ICCV)*, 2001.

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

[34] 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 (CVPR)*, 2021.

[35] Yiqun Mei, Yuchen Fan, Yuqian Zhou, Lichao Huang, Thomas S Huang, and Honghui Shi. Image super-resolution with cross-scale non-local attention and exhaustive self-exemplars mining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[36] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[37] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[38] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[39] Assaf Shocher, Nadav Cohen, and Michal Irani. "zero-shot" super-resolution using deep internal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[40] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[42] 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 (ICCV)*, 2021.

[43] Zhendong Wang, Xiaodong Cun, Jianmin Bao, and Jianzhuang Liu. Uformer: A general u-shaped transformer for image restoration. *arXiv preprint arXiv:2106.03106*, 2021.

[44] 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 (CVPR)*, 2022.

[45] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. *arXiv preprint arXiv:2103.15808*, 2021.

[46] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[47] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, 2010.

[48] 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)*, 2018.

[49] Yulun Zhang, Kunpeng Li, Kai Li, Bineng Zhong, and Yun Fu. Residual non-local attention networks for image restoration. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

[50] Shangchen Zhou, Jiawei Zhang, Wangmeng Zuo, and Chen Change Loy. Cross-scale internal graph neural network for image super-resolution. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[51] Maria Zontak and Michal Irani. Internal statistics of a single natural image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.