A Hybrid Network of CNN and Transformer for Lightweight Image Super-Resolution

Jinsheng Fang[1], Hanjiang Lin[1], Xinyu Chen[1], Kun Zeng[2*]
[1]Minnan Normal University, China [2]Minjiang University, China

Abstract

Recently, a number of CNN based methods have made great progress in single image super-resolution. However, these existing architectures commonly build massive number of network layers, bringing high computational complexity and heavy memory consumption, which is inappropriate to be applied on embedded terminals such as mobile platforms. In order to solve this problem, we propose a hybrid network of CNN and Transformer (HNCT) for lightweight image super-resolution. In general, HNCT consists of four parts, which are shallow feature extraction module, Hybrid Blocks of CNN and Transformer (HBCTs), dense feature fusion module and up-sampling module, respectively. By combining CNN and Transformer, HBCT extracts deep features beneficial for super-resolution reconstruction in consideration of both local and non-local priors, while being lightweight and flexible enough. Enhanced spatial attention is introduced in HBCT to further improve performance. Extensive experimental results show our HNCT is superior to the state-of-the-art methods in terms of super-resolution performance and model complexity. Moreover, we won the second best PSNR and the least activation operations in NTIRE 2022 Efficient SR Challenge. Code is available at https://github.com/lhjthp/HNCT.

1. Introduction

Single image super-resolution (SR) is a low-level computer vision task to reconstruct a high-resolution (HR) image from a low-resolution (LR) image. SR is an ill-posed problem due to the fact that an LR image can be degraded by infinite number of HR images. Since the SR technique is capable of recovering image texture details, it can be applied in many applications such as surveillance system, smart camera and so on.

Recently, a variety of convolutional neural network (CNN) based methods [5, 10, 16, 22, 31, 32, 34, 37] have been proposed and achieved prominent SR performance. Dong et al. first proposed a three-layer network SRCNN [5] to learn the end-to-end mapping from LR image to HR image. Then Kim et al. developed deeper network VDSR [13] with 20 layers and obtained better results than SRCNN, indicating that deeper networks can obtain better SR performance. EDSR [22] made further demonstration by deepening and widen the network architecture, and won the champion of NTIRE 2017 [36]. RDN [45] and RCAN [43] surpassed EDSR [22] by increasing the network depth to over 100 and 400 layers, respectively.

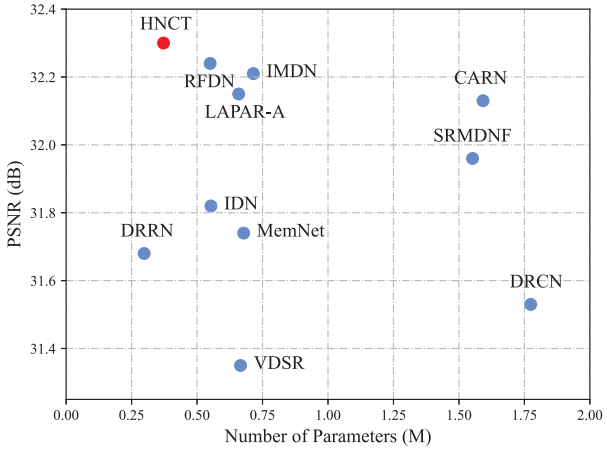

Figure 1. PSNR vs. Parameters on Set5 (×4).

Although SR has made considerable improvements, the existing CNN-based models still face some limitations. With the increasing of network depth, these methods require exploding computational cost and memory consumption, so that they cannot be deployed on embedded terminals like mobile devices. Moreover, CNN can only deal with the local region of the image, subjecting to the limited kernel sizes of convolutional operations, and is not able to achieve satisfied efficiency on long-range dependency modeling. So, it is important to take account of both local and non-local information for the enhancement of network performance.

*Corresponding author

To this end, new architecture different from CNN like Transformer [38] provides a self-attention mechanism to capture global information and exploit the self-similarity properties of image. LocalViT [19] introduces convolutional neural network (CNN) to bring locality mechanism into Transformers. In this way, LocalViT is capable of combining local and non-local information to increase model capacity. Lately, Liang et al. proposed a strong baseline model SwinIR [21] based on Swin Transformer [25]. In the main components of SwinIR, several Swin Transformer layers are utilized for local attention and cross-window interaction, while a convolutional layer is also added for feature enhancement. Through mutual cooperation of Transformer and CNN, SwinIR outperformed other state-of-the-art SR methods.

Inspired by SwinIR [21] and LocalViT [19] , we propose a lightweight SR model namely hybrid network of CNN and Transformer (HNCT), integrating CNN and transformer to model local and non-local priors, simultaneously. Specifically, HNCT consists of four parts: shallow feature extraction (SFE) module, Hybrid Blocks of CNN and Transformer (HBCTs), dense feature fusion (DFF) module and up-sampling module. Firstly, shallow features containing low-frequency information are extracted by only one convolution layer in the shallow feature extraction module. Then, four HBCTs are used to extract hierarchical features. Each HBCT contains a Swin Transformer block (STB) with two Swin Transformer layers inside, a convolutional layer and two enhanced spatial attention (ESA) modules [24]. Afterwards, these hierarchical features produced by HBCTs are concatenated and fused to obtain residual features in SFE. Finally, SR results are generated in the up-sampling module. Integrating CNN and transformer, our HNCT is able to extract more effective features for SR. As shown in Figure 1, HNCT achieves better SR results compared with state-of-the-art lightweight methods with fewer parameters.

The main contributions of this work can be summarized as follows:

1. We propose a lightweight hybrid network of CNN and Transformer (HNCT) for image super-resolution, which achieves better SR performance with fewer parameters than other methods.

2. We propose a hybrid block of CNN and Transformer (HBCT) that exploits local and non-local priors simultaneously to extract features beneficial for SR.

2. Related work

Recently, deep learning based methods, especially CNN-based methods [16, 37], have achieved dramatic improvements in image SR problem. Meanwhile, attention mechanism [8, 46], including self-attention mechanism [38], which is widely used in high-level vision tasks, has been introduced to further improve the SR performance. In this section, we briefly review on works related to CNN-based networks and attention-based networks.

CNN-based networks. Dong et al. first proposed SRCNN [5], which learns a end-to-end mapping from LR image to its HR counterpart via a CNN containing only three convolutional layers. Then, VDSR [13] and DRCN [14] further improved SR performance by learning larger networks with residual learning and recursive learning, respectively. By employing both residual learning and recursive learning strategies, DRRN [30] achieved better performance with fewer parameters. MemNet [35] was proposed to tackle the long-term dependency problem by mining persistent memory. In these methods, the original LR image is up-scaled to desired size before fed to the network. In order to increase SR speed, majority of new SR models took the original LR image as input and increased the spatial resolution by de-convolution or sub-pixel convolution [33] at the end of the networks. Different from other SR methods, LapSRN [15] reconstructed SR image by progressively increasing image resolution and predicting sub-band residuals of HR images. Inspired by ResNet [7], SRResNet [16] and EDSR [22] proposed SR models by stacking a flurry of residual blocks to improve SR performance. Specially, EDSR modified the residual block by removing batch normalization (BN) layer to achieve performance improvement. Based on EDSR, RDN [45] introduced dense connection [9] to make full use of hierarchical features from all the preceding layers.

Despite the great performances, most of CNN-based methods are not practical in real world due to heavy computation complexity. To solve this problem, Ahn et al. proposed an efficient model CARN-M [1] using a cascading network structure and group convolution operation, which achieved comparable results to state-of-the-art methods with fewer computations and parameters. Hui et al. proposed IDN [12] to gradually extract both long and short-path features and distill more useful information for SR reconstruction. Based on IDN, IMDN [11] proposed multi-distillation and contrast-aware channel attention mechanism and won the AIM 2019 constrained image super-resolution challenge [41]. Liu et al. proposed RFDN [24], which introduced feature distillation connection and shallow residual block for fast SR with fewer parameters than IMDN.

Attention-based networks. Inspired by human visual system which can focus on significant regions automatically, attention mechanism is designed to concentrate the most informative components of an input signal. Recently, several works introduced the attention mechanism to SR task. Zhang et al. presented RCAN [43] to focus on the most important channels by introducing channel attention mechanism into simplified residual block. Magid et al. proposed DFSA [26] to predict attention map of features in frequency domain using a matrix multi-spectral channel atten-

tion mechanism. Liu et al. proposed an enhanced spatial attention (ESA) module [24] to efficiently exploit local spatial information with fewer parameters. Besides, non-local attention mechanism aiming at capturing long-distance spatial information is studied. Methods, such as NLRN [23], RNAN [44], CSNLN [29], ENLCN [39], introduced non-local attention to achieve performance improvement. Recently, models like [3,17,21] introduced Transformer based on self-attention to further improve SR performance. Self-attention mechanism designed to encode distant dependencies and capture global interactions can be treated as a special case of non-local attention mechanism. Specially, Liang et al. presented SwinIR [21] based on Swin Transformer [25] to achieve excellent performance.

Moreover, multiple attention mechanisms are employed collaboratively to improve the SR results. Dai et al. proposed SAN [4] to refine features using both non-local attention and second-order channel attention. Niu et al. presented HAN [30] to not only learn the channel and spatial interdependencies of features in each layer by using channel attention and spatial attention, but also introduce a layer attention to explore correlations among hierarchical layers.

3. METHOD

3.1. Network Structure

As shown in Figure 2, the proposed HNCT consists four parts: shallow feature extraction (SFE), hybrid blocks of CNN and Transformer (HBCTs), dense feature fusion (DFF) and up-sampling module.

Given an input LR image I_{LR}, we first extract shallow features

$$F_0 = H_{SF}(I_{LR}) = W_0 * I_{LR}, \tag{1}$$

where H_{SF} denotes the one-convolution-layer SFE with weight W_0, and symbol $*$ denotes convolution operation. For simplicity, the bias term of convolutional layer is omitted. F_0 is then used for deep feature extraction with several HBCTs. Supposing the number of HBCTs is D, the output of the d-th HBCT F_d ($1 \leq d \leq D$) can be formulated as

$$F_d = f^d_{HBCT}(f^{d-1}_{HBCT} \cdots ((f^1_{HBCT}(F_0)))), \tag{2}$$

where f^d_{HBCT} denotes the function of d-th HBCT and F_d represents the output of d-th HBCT. HBCT is proposed to extract higher-level features from input features. More details of HBCT will be given in Section 3.2.

All the outputs of these HBCTs are concatenated and sent to DFF which includes two stacked convolutional layers to fuse all hierarchical features, and global residual learning strategy is added to ease learning difficulty. DFF uses features from all preceding HBCT layers and the output can be expressed as

$$F_{DFF} = W_1 * (W_2 * [F_1, F_2, \cdots, F_D]) + F_0, \tag{3}$$

where $[F_1, F_2, \cdots, F_D]$ is the concatenation of features generated by all HBCTs. W_1 and W_2 are the weights of 3×3 convolutional layer and 1×1 convolutional one, respectively. The 1×1 convolutional layer is introduced for feature fusion and the following 3×3 convolutional layer is used for further feature extraction.

Finally, in up-sampling module comprised of a 3×3 convolutional layer and a pixel shuffle layer, the SR image I_{SR} is reconstructed as follows

$$I_{SR} = F_{UP}(W_3 * F_{DFF}), \tag{4}$$

where W_3 is the weight of the convolution layer and F_{UP} denotes the pixel shuffling operation.

The loss function of our HNCT can be formulated as

$$L(\theta) = \frac{1}{N} \sum_{n=1}^{N} \|I^i_{SR} - I^i_{HR}\|_1, \tag{5}$$

where θ denotes the parameters of HNCT, $\|\|_1$ is the l_1 norm, N is the number of image patch for training, I^i_{SR} and I^i_{HR} are the i-th reconstructed SR images and the corresponding ground-truth HR image, repectively.

3.2. Hybrid Block of CNN and Transformer (HBCT)

In this Section, we introduce our proposed Hybrid Block of CNN and Transformer (HBCT). HBCT is composed of a Swin Transformer Block(STB), one 3×3 convolutional layer and two Enhanced Spatial Attention (ESA) modules. STB is proposed because it can greatly improve the representation ability of the model. ESA is characterized by light weight and high efficiency. STB and ESA will be discussed in details in Section 3.3 and Section 3.4. The structure of HBCT is shown in Figure 2. According to Equation (2), the feature maps of $(d-1)$-th HBCT F_{d-1} are directly fed to the d-th HBCT. Given the input feature F_{d-1}, the d-th HBCT first selects important features from input with an ESA module and then extracts intermediate features by a Swin Transformer Block (STB). Afterwards, a 3×3 convolutional layer is added to ensure the translational equivariance of our network. Finally, another ESA module is also introduced to obtain features that are more focused on the regions of interest. The function of the d-th HBCT can be described as

$$\begin{aligned} F_d &= f^d_{HBCT}(F_{d-1}) \\ &= H_{ESA}(W_4 * H_{STB}(H_{ESA}(F_{d-1}))), \end{aligned} \tag{6}$$

where H_{STB} denotes the function of STB, W_4 is the weight of the convolution layer, and H_{ESA} denotes the function of ESA.

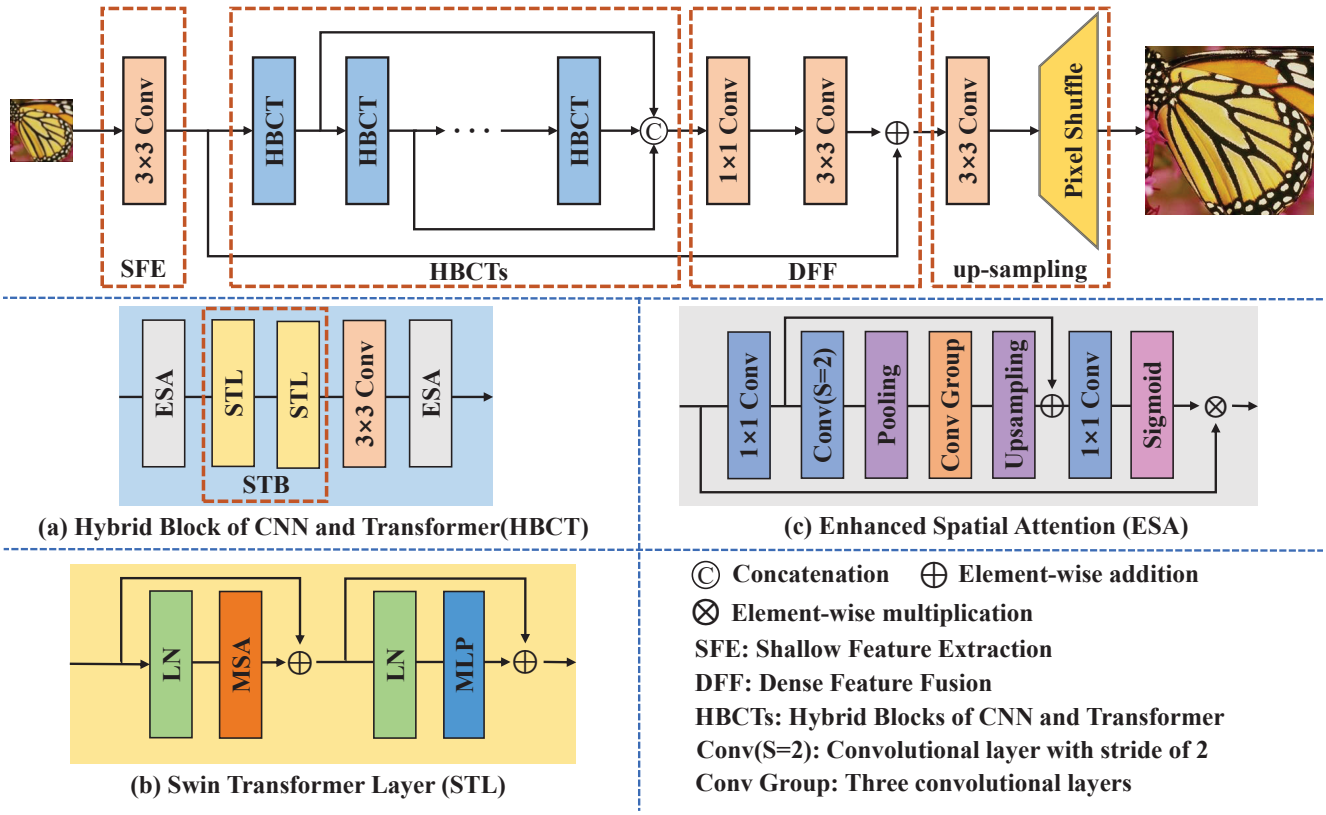

Figure 2. The architecture of the proposed HNCT for lightweight image super-resolution. (a) The module of Hybrid Blocks of CNN and Transformer (HBCTs), (b) structure of Swin Transformer Layer (STL), (c) enhanced spatial attention module (ESA), first proposed in RFANet [24].

3.3. Swin Transformer Block (STB)

Swin Transformer layer (STL) adopts the architecture of the original Transformer layer based on standard multi-head self-attention [38]. Moreover, Swin Transformer introduces local attention and shifted window mechanism. As shown in Figure 2(b), given an input with the size of $h \times w \times c$, Swin Transformer first reshapes the input into a $\frac{hw}{M^2} \times M^2 \times c$ feature by window partitioning, where $\frac{hw}{M^2}$ is the total number of windows with the size of $M \times M$. Then, for each window, Swin Transformer calculates self-attention for h times in parallel, where h is the number of self-attention head. Given a local window feature $F_{in}^{swt} \in R^{M^2 \times c}$, the query, key and value matrices Q, K and $V \in R^{M^2 \times d}$ are computed as

$$Q = F_{in}^{swt} W_Q, K = F_{in}^{swt} W_K, V = F_{in}^{swt} W_V, \quad (7)$$

where $d = \frac{c}{h}$, W_Q, W_K and W_V are shared learnable projection matrices across different windows. The attention matrix $Attn(Q, K, V)$ is calculated through the self-attention mechanism in the local window.

$$Attn(Q, K, V) = SoftMax(\frac{QK^T}{\sqrt{d}} + b)V, \quad (8)$$

where b is the learnable relative positional encoding. The results of multi-head self-attention (MSA) are concatenated to keep embedding dimension unchanged. After the attention function, there is a two-layer MLP with GELU activation in between. Layer Norm (LN) layer is added before MSA and MLP, and residual connection is used. The whole function of Transformer can be described as

$$\begin{cases} F_{inter}^{swt} = H_{MSA}(H_{LN}(F_{in}^{swt})) + F_{in}^{swt}, \\ F_{out}^{swt} = H_{MLP}(H_{LN}(F_{inter})) + F_{inter}^{swt}, \end{cases} \quad (9)$$

where H_{LN} denotes LN function, F_{MSA} represents multi-head self-attention operation and F_{MLP} denotes MLP function. However, there is no information interaction between windows with fixed window partition. Swin Transformer [25] uses regular and shift window partition alternately to realize the efficient information transmission and interaction of different windows.

By exploiting cross-window information, Swin Transformer has shown great promising performance in computer vision tasks. Because the length of shift step is half of the window size, even number of successive Swin Transformer layers are usually used to keep the positions of obtained features consistent with that of the corresponding LR im-

age patches in image space. In our HNCT, a STB contains two STLs to balance between SR performance and network complexity.

3.4. Enhanced Spatial Attention (ESA)

We used an enhanced spatial attention (ESA) model proposed in [24], which is more powerful than ordinary SA module [46]. The structure of ESA module is depicted in Figure 2(c). Given an input F_{in}^{esa}, ESA firstly extracts compact features F_1^{esa} as follows,

$$F_1^{esa} = W_1^{esa} * F_{in}^{esa}, \tag{10}$$

where W_1^{esa} is the weight of 1×1 convolutional layer used to reduce embedding dimension, Then ESA further extracts features F_2^{esa} as follows,

$$F_2^{esa} = H_{up}(H_g(H_{pool}(W_2^{esa} * F_1^{esa}))), \tag{11}$$

where W_2^{esa} is the weight of 3×3 convolution with stride of 2, H_{pool} is a max-pooling operation, H_g is the function of the group composed of three 3×3 convolution layers, and H_{up} is up-sampling function realized by bilinear interpolation. The spatial dimensions are reduced by both strided convolutional layer and max pooling layer, and then recovered by the up-sampling layer. Finally, the output of ESA module F_{out}^{esa} can be computed as

$$F_{out}^{esa} = H_{sigmoid}(W_3^{esa} * (F_1^{esa} + F_2^{esa})) \times F_{in}^{esa}, \tag{12}$$

where W_3^{esa} is the weight of 1×1 convolutional layer used to recover the embedding dimension, $H_{sigmoid}$ is the sigmoid function, and symbol $\times$ denotes point-wise multiplication operation.

The ESA mechanism works at the beginning and the end of HBCT, making the features more focused on the regions of interest. When these highlighted features are aggregated together, we can get more representative features, which are more beneficial for image SR reconstruction.

3.5. Discussions

Difference to RFDN. RFDN [24] proposes feature distillation connection (FDC), which is functionally equivalent to channel splitting operation. Based on FDC, RFDN uses multiple feature connections to learn more distinctive feature representations. A shallow residual block is also proposed as the main building block of RFDN, so that RFDN benefits from residual learning while maintaining lightweight. Unlike RFDN, HNCT assembles Transformer and CNN. Thanks to Transformer's ability of modeling long-distance dependence and CNN's ability of local feature extraction, our HNCT can improve SR performance greatly.

Difference to SwinIR. SwinIR [21] proposes a strong baseline model for image restoration based on Swin Transformer. The main component of SwinIR is constructed by stacking several residual Swin Transformer blocks. Different from SwinIR, HNCT adopts dense connection to fully integrate hierarchical features generated by preceding HBCTs. Moreover, ESA module is deployed to highlight more representative features, boosting the SR performance further.

Table 1. Investigations of HBCT on the Manga109 benchmark datasets with $\times 4$ super-resolution.

Model	Block Structure	PSNR/SSIM
HNCT	ESA+STB+Conv+ESA	30.70/0.9112
Model 1	ESA+Conv+ReLU+Conv +ReLU+Conv+ESA	30.13/0.9034
Model 2	ESA+STB+Conv	30.65/0.9104
Model 3	STB+Conv+ESA	30.60/0.9104
Model 4	STB+Conv	30.56/0.9093
Model 5	ESA+STB+Conv+SA	30.64/0.9106
Model 6	SA+STB+Conv+ESA	30.65/0.9107
Model 7	SA+STB+Conv+SA	30.56/0.9099

4. Experiments

4.1. Experimental Setup

We train our HNCT using 800 training images from DIV2K [36] dataset. Data augmentation is performed by rotating 90°, 180°, 270° and flipping horizontally. For testing, we use five benchmark datasets: Set5 [2], Set14 [40], BSD100 [27], Urban100 [10] and Manga109 [28]. Peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) are employed to measure the quality of SR images.

For each training mini-batch, 16 patches of size 64×64 are cropped randomly from LR images as input. Adam optimizer is used to trained our HNCT by setting β1=0.9, β2=0.999, and ϵ=1e-8. There are 1200 training epochs in total. The learning rate is initialized to 5e-4, reduced by half every 200 epochs, and fixed after 1000 epochs. Window size, embedding dimension and attention head number in STL are set to 8, 50 and 5, respectively. To trade-off the size and performance of the model, our HNCT contains four HBCTs, each of which includes two STLs.

4.2. Ablation study

We conduct several ablation experiments to evaluate the effectiveness of our proposed HBCT, on Manga109 benchmark dataset. The results are listed in Table 1, where Conv denotes a 3×3 convolution layer and SA is ordinary spatial attention module introduced in [46]. First, Model 1 is a CNN based network constructed by replacing STB in HBCT with two convolutional layers, and adding a ReLU layer between every two successive convolutional layers. The SR results of model 1 show that HNCT is superior to

Table 2. Average PSNR/SSIM for scale factor 2, 3 and 4 on datasets Set5, Set14, BSD100, Urban100, and Manga109. The best and second best results are highlighted in red and blue respectively.

Method	Scale	Params	Set5	Set14	BSD100	Urban100	Manga109
			PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
Bicubic	×2	-	33.66/0.9299	30.24/0.8688	29.56/0.8431	26.88/0.8403	30.80/0.9339
SRCNN [5]		8K	36.66/0.9542	32.45/0.9067	31.36/0.8879	29.50/0.8946	35.60/0.9663
FSRCNN [6]		13K	37.00/0.9558	32.63/0.9088	31.53/0.8920	29.88/0.9020	36.67/0.9710
VDSR[19] [13]		666K	37.53/0.9587	33.03/0.9124	31.90/0.8960	30.76/0.9140	37.22/0.9750
DRCN[32] [14]		1774K	37.63/0.9588	33.04/0.9118	31.85/0.8942	30.75/0.9133	37.55/0.9732
DRRN [30]		298K	37.74/0.9591	33.23/0.9136	32.05/0.8973	31.23/0.9188	37.88/0.9749
MemNet [35]		678K	37.78/0.9597	33.28/0.9142	32.08/0.8978	31.31/0.9195	37.72/0.9740
IDN [12]		553K	37.83/0.9600	33.30/0.9148	32.08/0.8985	31.27/0.9196	38.01/0.9749
SRMDNF [42]		1511K	37.79/0.9601	33.32/0.9159	32.05/0.8985	31.33/0.9204	38.07/0.9761
CARN [1]		1592K	37.76/0.9590	33.52/0.9166	32.09/0.8978	31.92/0.9256	38.36/0.9765
LAPAR-A [18]		548k	38.01/0.9605	33.62/0.9183	32.19/0.8999	32.10/0.9283	38.67/0.9772
IMDN [11]		694K	38.00/0.9605	33.63/0.9177	32.19/0.8996	32.17/0.9283	38.88/0.9774
RFDN [24]		534K	38.05/0.9606	33.68/0.9184	32.16/0.8994	32.12/0.9278	38.88/0.9773
HNCT (Ours)		356K	38.08/0.9608	33.65/0.9182	32.22/0.9001	32.22/0.9294	38.87/0.9774
Bicubic	×3	-	30.39/0.8682	27.55/0.7742	27.21/0.7385	24.46/0.7349	26.95/0.8556
SRCNN [5]		8K	32.75/0.9090	29.30/0.8215	28.41/0.7863	26.24/0.7989	30.48/0.9117
FSRCNN [6]		13K	33.18/0.9140	29.37/0.8240	28.53/0.7910	26.43/0.8080	31.10/0.9210
VDSR[19] [13]		666K	33.66/0.9213	29.77/0.8314	28.82/0.7976	27.14/0.8279	32.01/0.9340
DRCN[32] [14]		1774K	33.82/0.9226	29.76/0.8311	28.80/0.7963	27.15/0.8276	32.24/0.9343
DRRN [30]		298K	34.03/0.9244	29.96/0.8349	28.95/0.8004	27.53/0.8378	32.71/0.9379
MemNet [35]		678K	34.09/0.9248	30.00/0.8350	28.96/0.8001	27.56/0.8376	32.51/0.9369
IDN [12]		553K	34.11/0.9253	29.99/0.8354	28.95/0.8013	27.42/0.8359	32.71/0.9381
SRMDNF [42]		1528K	34.12/0.9254	30.04/0.8382	28.97/0.8025	27.57/0.8398	33.00/0.9403
CARN [1]		1592K	34.29/0.9255	30.29/0.8407	29.06/0.8034	28.06/0.8493	33.50/0.9440
LAPAR-A [18]		544k	34.36/0.9267	30.34/0.8421	29.11/0.8054	28.15/0.8523	33.51/.09441
IMDN [11]		703K	34.36/0.9270	30.32/0.8417	29.09/0.8046	28.17/0.8519	33.61/0.9445
RFDN [24]		541K	34.41/0.9273	30.34/0.8420	29.09/0.8050	28.21/0.8525	33.67/0.9449
HNCT (Ours)		363K	34.47/0.9275	30.44/0.8439	29.15/0.8067	28.28/0.8557	33.81/0.9459
Bicubic	×4	-	28.42/0.8104	26.00/0.7027	25.96/0.6675	23.14/0.6577	24.89/0.7866
SRCNN [5]		8K	30.48/0.8626	27.50/0.7513	26.90/0.7101	24.52/0.7221	27.58/0.8555
FSRCNN [6]		13K	30.72/0.8660	27.61/0.7550	26.98/0.7150	24.62/0.7280	27.90/0.8610
VDSR[19] [13]		666K	31.35/0.8838	28.01/0.7674	27.29/0.7251	25.18/0.7524	28.83/0.8870
DRCN[32] [14]		1774K	31.53/0.8854	28.02/0.7670	27.23/0.7233	25.14/0.7510	28.93/0.8854
DRRN [30]		298K	31.68/0.8888	28.21/0.7720	27.38/0.7284	25.44/0.7638	29.45/0.8946
MemNet [35]		678K	31.74/0.8893	28.26/0.7723	27.40/0.7281	25.50/0.7630	29.42/0.8942
IDN [12]		553K	31.82/0.8903	28.25/0.7730	27.41/0.7297	25.41/0.7632	29.41/0.8942
SRMDNF [42]		1552K	31.96/0.8925	28.35/0.7787	27.49/0.7337	25.68/0.7731	30.09/0.9024
CARN [1]		1592K	32.13/0.8937	28.60/0.7806	27.58/0.7349	26.07/0.7837	30.47/0.9084
LAPAR-A [18]		659K	32.15/0.8944	28.61/0.7818	27.61/0.7366	26.14/0.7871	30.42/0.9074
IMDN [11]		715K	32.21/0.8948	28.58/0.7811	27.56/0.7353	26.04/0.7838	30.45/0.9075
RFDN [24]		550K	32.24/0.8952	28.61/0.7819	27.57/0.7360	26.11/0.7858	30.58/0.9089
HNCT (Ours)		372K	32.31/0.8957	28.71/0.7834	27.63/0.7381	26.20/0.7896	30.70/0.9112

CNN based network due to combination of CNN and Transformer. Then, models 2-4 are constructed by removing one or both ESA modules in HNCT, respectively. Their performances drop slightly, demonstrating that spatial attention can improve SR performance of HNCT. Finally, built by replacing one or both ESA modules with SA module, models 5-7 perform worse than HNCT, indicating that ESA is more powerful to highlight significant features than SA.

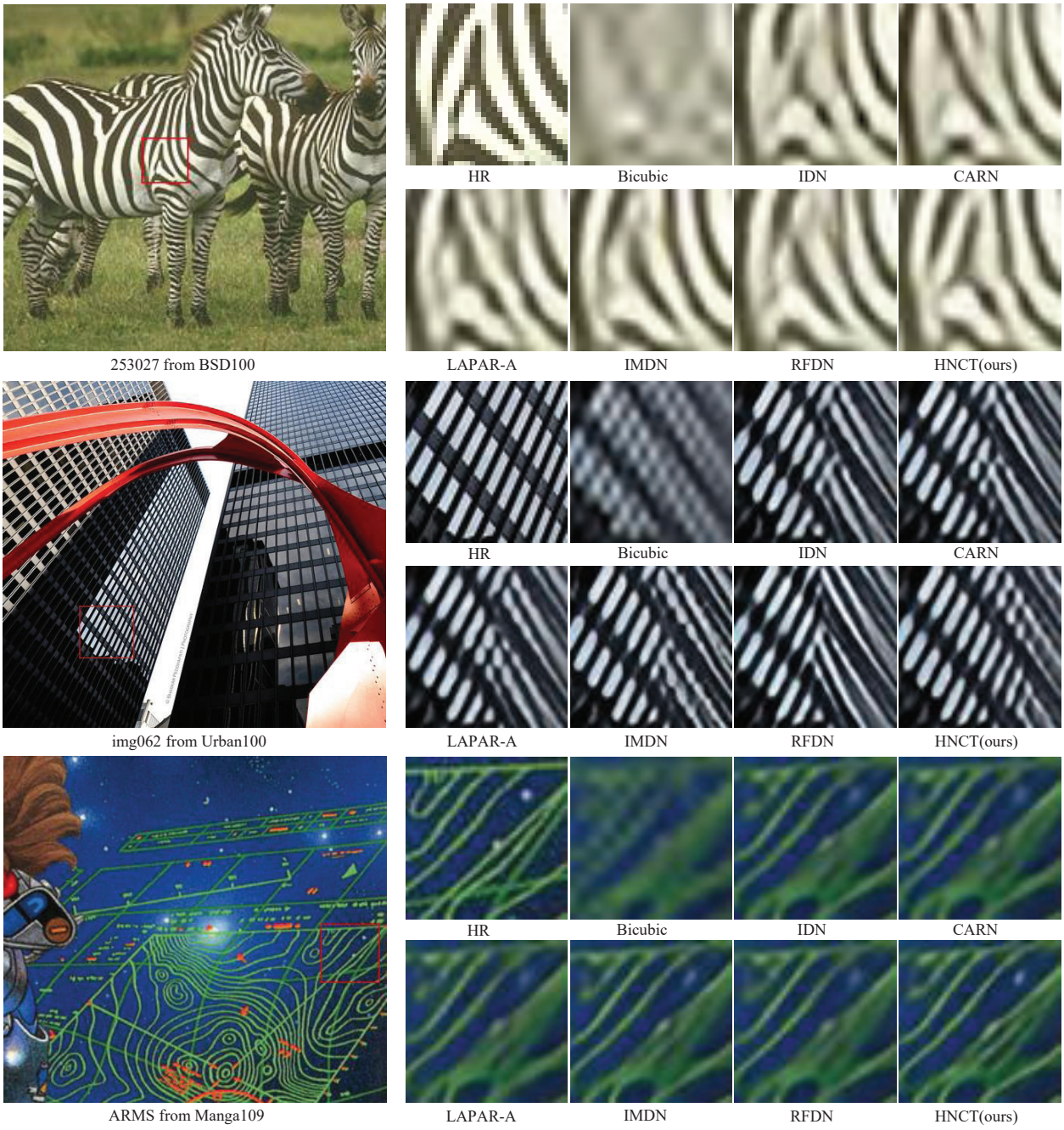

Figure 3. Visual comparisons of HNCT with other SR methods on BSD100, Urban100 and Manga109 $\times 4$ datasets.

Thanks to ESA and combination of CNN and Transformer, our HNCT outperforms other models listed in Table 1.

4.3. Complexity Analysis

The compared of PSNRs of $\times 4$ SR on Set5 and parameter numbers of different models is described in Figure1. The compared models include VDSR [13], DRCN [14], LAPAR-A [18], DRRN [30], MemNet [35], IDN [12], SRMDNF [42], CARN [1], IMDN [11], RFDN [24] and our HNCT. As we know, the parameter number is one of the significant factors in a lightweight model. As shown in Figure 1, our HNCT achieves the best performance with fewer parameter number compared with other methods except DRRN. HNCT obtains much better performance than

Table 3. NTIRE 2022 Efficient SR Challenge results. Noting that only six methods are included.

Team Name	Network	PSNR (dB)	Runtime (ms)	Params (K)	FLOPS (G)	Acts (M)	Mem (M)
ByteESR	RLFN	28.72	26.76	317	19.7	80.05	377.91
NJU_Jet	FMEN	28.69	27.67	341	22.28	72.09	204.6
NEESR	PlainRFDN	28.71	29.58	272	16.86	79.59	575.99
Just Try	LWFANet	28.81	251.45	832	135.3	392.43	2387.93
ncepu_explorers	MDAN	28.79	324.5	390	23.73	994.25	771.54
mju_mnu	HNCT[1]	28.79	339.61	345	78.81	46.76	1310.72

DRRN with slightly larger parameter number. It is proved that HNCT is an efficient lightweight SR method in Figure 1.

4.4. Comparison with State-Of-The-Arts

We compare our HNCT with other lightweight SR methods, including SRCNN [5], FSRCNN [6], VDSR [13], DRCN [14], DRRN [30], MemNet [35], IDN [12], SRMDNF [42], CARN [1], LAPAR-A [18], IMDN [11] and RFDN [24]. Table 2 shows quantitative results of five benchmark datasets. We can find that the proposed HNCT achieve the best performance under both $\times 3$ and $\times 4$ on all datasets, except on Set14 and Manga109 under $\times 2$, due to the fact that these competitors are efficient enough to reconstruct the images with only 2 scales of down-sampling. Parameter comparison is also listed in Table 2. It is clearly shown that although RFDN has closer results with the proposed HNCT method, it has approximate 50% (180K) the parameters larger than that of our model under all cases. SRCNN and FSRCNN have the least parameters, but their performance are far behind that of the proposed HNCT. Hence, profit from ESA, CNN and Transformer, our proposed HNCT substantially obtains the best results with least parameters.

Figure 3 shows three visual comparisons between HNCT and the other lightweight competitors on $\times 4$. The original images "253027", "img062" and "ARMS" are selected from BSD100, Urban100 and Manga109, respectively. From the enlarged views, we can observe that the stripes and lines reconstructed by HNCT are more closer to the ground truth than the competitors. Especially in the reconstruction of img062, more accurate rectangles are reconstructed. This visual comparison can further demonstrate the effectiveness of our proposed HNCT.

4.5. NTIRE 2022 Efficient SR Challenge

This work is proposed initially for participating in the NTIRE 2022 Efficient SR Challenge [20]. The challenge aims to devise a network that reduces one or several aspects such as runtime, parameters, FLOPS, activations and depth, while at least maintaining PSNR of 29.00dB on DIV2K validation dataset. According to the competing rules, RLFN, FMEN and PlainRFDN are the top three winner methods because they have the least runtime. For simplicity, Table 3 lists these three methods and the other top three methods of PSNR. The proposed HNCT has the least activation number and achieves better PSNR than the top winners, RLFN, FMEN and PlainRFDN with comparable parameter number. Compared with LWFANet and MDAN that obtain the top two PSNR, HNCT only has 345K parameters, while other two methods have 390K and 832K parameters, respectively.

[1]The original name of the proposed HNCT model in NTIRE 2022 Efficient SR Challenge is CCSTN.

5. Conclusion

In this paper, we propose a hybrid network of CNN and Transformer (HNCT) for lightweight image SR. By integrating CNN and Transformer, HNCT can exploit both local and non-local priors and extract deep features more beneficial for image SR. Furthermore, enhanced spatial attention (ESA) is employed to further improve SR results. Extensive experiments demonstrate that our HNCT is superior to the compared lightweight SR methods, achieving the best performances with the least parameters. However, HNCT runs much slower than CNN-based methods due to heavy computation complexity of Transformer. In future, we will focus on improving the inference speed of HNCT.

Acknowledgement. This work was partly supported by Fujian Provincial Natural Science Foundation Projects (No.2020J01824, 2021J011005), the National Natural Science Foundation of China (No.61601389) and Open Project of the Key Laboratory of Plasma and Magnetic Resonance in Fujian Province, Xiamen University(No.20191201).

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*, pages 252–268, 2018. 2, 6, 7, 8

[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, 2012. 5

[3] 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 Conference on Computer Vision and Pattern Recognition*, pages 12299–12310, 2021. 3

[4] 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 Conference on Computer Vision and Pattern Recognition*, pages 11065–11074, 2019. 3

[5] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Proceedings of the European Conference on Computer Vision*, pages 184–199, 2014. 1, 2, 6, 8

[6] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *Proceedings of the European Conference on Computer Vision*, pages 391–407, 2016. 6, 8

[7] 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*, pages 770–778, 2016. 2

[8] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018. 2

[9] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017. 2

[10] 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. 1, 5

[11] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the ACM International Conference on Multimedia*, pages 2024–2032, 2019. 2, 6, 7, 8

[12] Zheng Hui, Xiumei Wang, and Xinbo Gao. Fast and accurate single image super-resolution via information distillation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 723–731, 2018. 2, 6, 7, 8

[13] 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. 1, 2, 6, 7, 8

[14] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1637–1645, 2016. 2, 6, 7, 8

[15] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 624–632, 2017. 2

[16] 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*, pages 4681–4690, 2017. 1, 2

[17] Wenbo Li, Xin Lu, Jiangbo Lu, Xiangyu Zhang, and Jiaya Jia. On efficient transformer and image pre-training for low-level vision. *arXiv preprint arXiv:2112.10175*, 2021. 3

[18] 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. 6, 7, 8

[19] Yawei Li, Kai Zhang, Jiezhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021. 2

[20] Yawei Li, Kai Zhang, Luc Van Gool, Radu Timofte, et al. Ntire 2022 challenge on efficient super-resolution: Methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2022. 8

[21] 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 Conference on Computer Vision and Pattern Recognition*, pages 1833–1844, 2021. 2, 3, 5

[22] 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

[23] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. *Advances in Neural Information Processing Systems*, 31, 2018. 3

[24] Jie Liu, Jie Tang, and Gangshan Wu. Residual feature distillation network for lightweight image super-resolution. In *Proceedings of the European Conference on Computer Vision*, pages 41–55, 2020. 2, 3, 4, 5, 6, 7, 8

[25] 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 Conference on Computer Vision and Pattern Recognition*, pages 10012–10022, 2021. 2, 3, 4

[26] Salma Abdel Magid, Yulun Zhang, Donglai Wei, Won-Dong Jang, Zudi Lin, Yun Fu, and Hanspeter Pfister. Dynamic high-pass filtering and multi-spectral attention for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4288–4297, 2021. 2

[27] 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 International Conference on Computer Vision*, pages 416–423, 2001. 5

[28] 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

[29] 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 Conference on Computer Vision and Pattern Recognition*, pages 5690–5699, 2020. 3

[30] 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*, pages 191–207, 2020. 2, 3, 6, 7, 8

[31] Yajun Qiu, Ruxin Wang, Dapeng Tao, and Jun Cheng. Embedded block residual network: A recursive restoration model for single-image super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4180–4189, 2019. 1

[32] Mohammad Saeed Rad, Behzad Bozorgtabar, Urs-Viktor Marti, Max Basler, Hazim Kemal Ekenel, and Jean-Philippe Thiran. Srobb: Targeted perceptual loss for single image super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2710–2719, 2019. 1

[33] 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*, pages 1874–1883, 2016. 2

[34] Dehua Song, Yunhe Wang, Hanting Chen, Chang Xu, Chunjing Xu, and DaCheng Tao. Addersr: Towards energy efficient image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 15648–15657, 2021. 1

[35] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4539–4547, 2017. 2, 6, 7, 8

[36] 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. 1, 5

[37] Radu Timofte, Rasmus Rothe, and Luc Van Gool. Seven ways to improve example-based single image super resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1865–1873, 2016. 1, 2

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017. 2, 4

[39] Bin Xia, Yucheng Hang, Yapeng Tian, Wenming Yang, Qingmin Liao, and Jie Zhou. Efficient non-local contrastive attention for image super-resolution. *arXiv preprint arXiv:2201.03794*, 2022. 3

[40] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *Proceedings of the International Conference on Curves and Surfaces*, pages 711–730, 2010. 5

[41] Kai Zhang, Shuhang Gu, Radu Timofte, Zheng Hui, Xiumei Wang, Xinbo Gao, Dongliang Xiong, Shuai Liu, Ruipeng Gang, Nan Nan, et al. Aim 2019 challenge on constrained super-resolution: Methods and results. In *Proceedings of the 2019 IEEE International Conference on Computer Vision Workshop*, pages 3565–3574, 2019. 2

[42] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3262–3271, 2018. 6, 7, 8

[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*, pages 286–301, 2018. 1, 2

[44] Yulun Zhang, Kunpeng Li, Kai Li, Bineng Zhong, and Yun Fu. Residual non-local attention networks for image restoration. *arXiv preprint arXiv:1903.10082*, 2019. 3

[45] 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

[46] Xizhou Zhu, Dazhi Cheng, Zheng Zhang, Stephen Lin, and Jifeng Dai. An empirical study of spatial attention mechanisms in deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6688–6697, 2019. 2, 5