Fast and Accurate Single Image Super-Resolution via Enhanced U-Net

Le Chang [1], Fan Zhang [2*], and Biao Li[3]
[1] Institute of Electronic Information Engineering, Shanxi Polytechnic College
Taiyuan, 03006 China
[e-mail: 285694237@qq.com]
[2] Key Laboratory of the Ministry of Education for Optoelectronic Measurement Technology and Instrument
Beijing Information Science and Technology University
Beijing, 100101 China
[e-mail: zhangfan2015@bupt.edu.cn]
[3] School of Business Administration, Southwestern University of Finance and Economics
Chengdu, 611130 China
[e-mail: biaoli@swufe.edu.cn]
[*]Corresponding author: Fan Zhang

Received Septebmer 10, 2020; revised December 28, 2020; revised February 3, 2021; March 28, 2021; published April 30, 2021

Abstract

Recent studies have demonstrated the strong ability of deep convolutional neural networks (CNNs) to significantly boost the performance in single image super-resolution (SISR). The key concern is how to efficiently recover and utilize diverse information frequencies across multiple network layers, which is crucial to satisfying super-resolution image reconstructions. Hence, previous work made great efforts to potently incorporate hierarchical frequencies through various sophisticated architectures. Nevertheless, economical SISR also requires a capable structure design to balance between restoration accuracy and computational complexity, which is still a challenge for existing techniques. In this paper, we tackle this problem by proposing a competent architecture called Enhanced U-Net Network (EUN), which can yield ready-to-use features in miscellaneous frequencies and combine them comprehensively. In particular, the proposed building block for EUN is enhanced from U-Net, which can extract abundant information via multiple skip concatenations. The network configuration allows the pipeline to propagate information from lower layers to higher ones. Meanwhile, the block itself is committed to growing quite deep in layers, which empowers different types of information to spring from a single block. Furthermore, due to its strong advantage in distilling effective information, promising results are guaranteed with comparatively fewer filters. Comprehensive experiments manifest our model can achieve favorable performance over that of state-of-the-art methods, especially in terms of computational efficiency.

Keywords: Single Image Super-resolution, Convolutional Neural Networks, Information Propagation, U-Net Block

This work was supported in part by the establishment of the Dynamic Adjustment Mechanism for the cultivation of core professional skills in higher vocational colleges (No. GH-19238).

http://doi.org/10.3837/tiis.2021.04.004 **ISSN : 1976-7277**

1. Introduction

Single image super-resolution (SISR), which aims to reconstruct a high-resolution (HR) image from its low-resolution (LR) counterpart, has attracted wide attention in computer vision community. Thanks to constant efforts, a bunch of real-world applications can be found owing to this method ranging from medical imaging [1], face recognition [2] to surveillance [3]. Besides, many electronic products with built-in camera are equipped with this technology to dramatically decrease the hardware cost. Back to SISR research regime, in addition to signal processing techniques [4], various learning-based image-resolution algorithms have been recently proposed to tackle this ill-posed problem (an underdetermined inverse problem), including sparse coding [5], random forest [6] and deep convolutional neural networks (CNNs) [7-13].

Among the aforementioned methods, deep CNNs based algorithms have demonstrated its advantage in boosting the performance, where peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [14] are two generic measurements. More specifically, Dong et al. [7] firstly employed three convolutional layers to build a neural network, which achieved significant improvement compared with conventional algorithms. Later, Kim et al. [9] increased the depth of network via introducing a global residual directly from the input to the output, and state-of-the-art results were obtained by this revision. Then, Lim et al. built an extremely deep network EDSR [11] via stacking the residual block, which made slight changes to the original version [15] by removing batch normalization. In addition, SRFBN was proposed to refine low-level representations with high-level information, and a feedback block was designed to generate powerful high-level representations [16]. Residual non-local attention network for image super-resolution was proposed in [17], where trunk branch and non-local mask branch in each non-local attention block was designed. The trunk branch was used to extract hierarchical features. Non-local mask branches aimed to adaptively rescale hierarchical features with mixed attentions. Scale-wise convolutional network (SCN) proposed in [18] learnt to dynamically activate and aggregate features from different input scales in each residual block to exploit contextual information on multiple scales. Pan et al. proposed to impose the image formation constraint on the deep neural networks via pixel substitution, and the output of pixel substitution was further refined by the deep neural network in a cascaded manner [19].

Generally speaking, feature maps from different layers contain different information frequencies. In other words, lower layers generate information in low-frequency and higher layers bring about relatively high-frequency one. The key concern is how to utilize diverse information frequencies to achieve efficient and accurate recovery, which is crucial to satisfying super-resolution image reconstructions. From this perspective, previous methods only attain information with limited frequency diversity at the last layer after repeated convolution operations. As a result, they failed to leverage information in different frequencies when reconstructing the final SR image, which intensively undermines SISR performance.

To cope with this puzzle, especially after the emergence of dense connection network [20], several SISR architectures [13, 21] have adopted dense block as underlying bread and butter. Most of them achieved better performances compared with their former counterparts. Thereby, it is plausible that dense connection sheds lights on generating and manipulating multiple types of information once and for all, which is essential to accurate SISR. Indeed, in practice, the deeper each dense block is and the more blocks the structure possess, the more types of information it may extract and deliver. Additionally, Zhang et al. [21] has proven higher growth rate in dense blocks can improve the network performance in SISR. But these dense-

connection based structures conceal their innate limitations when generating different types of information in network constructions. On one hand, it is consistently difficult to collect the information of excessive layers within one dense block because it would characterize computational complexity and memory consumption as unacceptable, which results in adequate network representation capacity intractable. On the other hand, the growth rate of dense blocks is extremely constrained by the GPU memory, which may also impact the quality of information in feature maps.

Accordingly, instead of dense block, in this paper, we propose the enhanced U-Net block (EUB) to tackle all the drawbacks mentioned above. The architecture is shown in **Fig. 2**, which is inspired by U-Net [22] with some revisions. More specifically, skip concatenation with local residual learning constitutes the main part of the block, which means the convolutional layers on the right (deep) have access to the corresponding layers on the left (shallow). It is unrealistic to directly extract feature maps by constructing a neural network merely by this block, so EUB is just as building blocks to establish our final network.

Two advantages can be immediately delivered by this specific design. Firstly, the concatenation of feature maps directly from the lower layers to the higher ones, i.e., information propagation between different layers, has been profoundly proved to be effective and essential to SISR task, as mentioned above. Secondly, the proposed block enables the network to develop fairly deep in depth, which always brings out permission on generating more types of information. Our network can successfully attain more fidelity compared with most of the advanced models, as shown in **Fig. 1**.

Fig. 1. Visualized results on "img061" image from "Urban100" with scaling factor ×2. Compared with the recently proposed state-of-the-art network RDN, EUN can recover more realistic visual results, which is more faithful to the ground truth.

In summary, the main contributions of this work can be unpacked in four folds:

(1) We propose a neural network called the enhanced U-Net network (EUN) for SISR, which can combine multiple types of information to construct high quality super-resolution image.

(2) We provide a new block EUB for SISR task, which can produce multi-type information via skip concatenation mechanism. Furthermore, the proposed block can also be employed to other image task, such as image denoising and image enhancement.

(3) Due to its strong ability in distilling effective information, our SISR model can achieve a promising performance with comparatively fewer filters. Then, lightweight model is feasible.

(4) To the best of our knowledge, it is the first time that using U-Net-style structure as fundamental block to develop a deep neural network for such specific task.

2. Related Work

SRCNN [7] was the first end-to-end CNNs based SISR algorithm, and achieved performance superior to previous traditional algorithms. In this work, an interpolated image was the input and three convolution layers were applied for feature extraction. However, it suffered from computation complexity introduced by the large interpolated input image. Thus, an accelerated version FSRCNN was proposed [8] by performing a transposed convolution layer to upscale the image before the output layer instead of the bicubic interpolation in the beginning.

Nevertheless, there were only three convolution layers in both of the above works, and a straightforward improvement is to merely increase the network depth. Provided that deeper network retains larger receptive field to exploit more contextual information from LR images, VDSR [9] grew the network into twenty layers via skip connection, where residual could speed up the converging speed in training process and also improve the performance [15]. Again, Ledig et al. [10] proposed SRResNet to directly stack multiple residual blocks for their discriminator in SRGAN. Thanks to the powerful GAN mechanism, it achieved significantly improvement. Later, Lim et al. [11] optimized the residual network structure by removing unnecessary modules and performed well in both SSIM and PSNR. However, the contribution is limited to the final performance when only increasing the depth of network, not to mention the unacceptable parameter scale.

Besides, Shi et al. proposed ESPCN [23], where an efficient sub-pixel convolution was proposed to upscale feature maps to HR output. Until now, sub-pixel convolution and transposed convolution have become two mainstreams for LR image upscaling in SISR, mainly because of their high efficiency in reconstructing images. Thanks to these upscaling techniques, we can abandon the pre-processing of LR image interpolation, which always ends up with details losing and computation complexity increasing. These two up-sampling methods can be found in almost all the up-to-date methods, such as EDSR [11], RDN [21], SRGAN [10].

On the other hand, in order to better tailor models to benefit from multi-layer features in CNNs, Huang et al. [20] proposed the dense connection to synthesize former information within the dense block. In addition, Tong et al. [13] developed SRDenseNet via stacking dense blocks, which were employed to extract high level features. Zhang et al. [21] improved the dense block via a residual shortcut to construct a new block, which was adopted to extract abundant local features. However, as mentioned above, the size and the number of dense blocks are extremely obsessed by the budget on GPU memory.

To facilitate this problem, we design a new U-Net based block to extract more information and directly connect more blocks in the overall structure, which is highly crucial to SISR. The details of our architecture will be shown in Section 3.

3. Enhanced U-Net Network

3.1 Enhanced U-Net Block

The designed block is very similar to U-Net [22]. As shown in **Fig. 2**, the proposed block also consists of two parts except without bottleneck. Instead of the encoder-decoder mechanism, we repeatedly apply 3×3 convolution operation with padding rather than down-sampling, in order to preserve the image spatial size. More importantly, the frequency augment requirement of restoration turns out no pooling in SISR. Meanwhile, it inherits the skip concatenation from U-Net to combine information in different frequencies. In addition, the amounts of channels in each layer is identical, while it is inversely proportional to the feature map dimension in the original U-Net. A detailed comparison will be discussed in Section 4. Accordingly, we call our block the enhanced U-Net block (EUB).

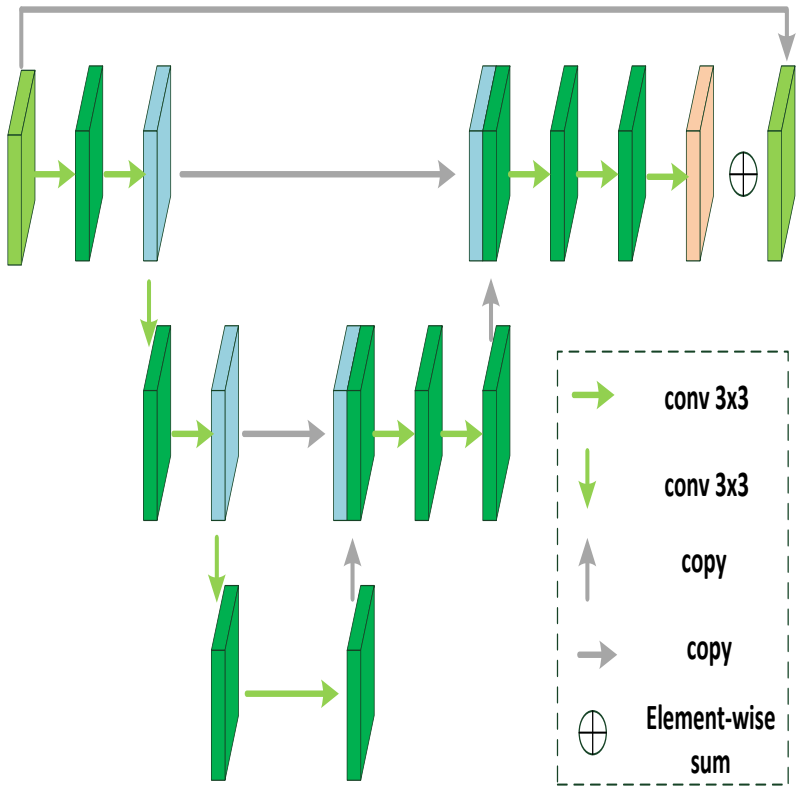


Fig. 2. The architecture of enhanced U-Net block (EUB). The gray arrows denote the copy of feature maps, which are used for connection to other layers or local residual learning.

As a matter of fact, an EUB can be described by four components: left side layer, right side layer, the last layer and local residual shortcut. With slight alteration on terminology, we group every two cascaded convolution operations as one layer, so there are five layers in **Fig. 2**. Providing that the left side has N layers, the total number of layers will be $2N-1$, where the right side contains $N-1$ layers. The total number of convolution operations will be $4N-1$, because there is one more convolution at the end of the whole block.

We choose LReLU [24] as the activation function, and the convolution operation with same padding is adopted to keep the final feature dimension unchanged. In practice, the number of layers within an EUB is a hyper-parameter. We will explain the details in Section 5.1.

Left side layer It follows typical CNNs architecture and each layer consists of two repeated convolution operations with filter size 3×3. In particular, for the i^{th} layer of left side in the d^{th} block, the output can be formulated as:

$$H_{i,d} = \sigma(W_{i2,d} * \sigma(W_{i1,d} * H_{i-1,d}) \tag{1}$$

where $H_{i-1,d}$ denotes the output of the $(i-1)^{th}$ layer in the d^{th} block. Similarly, the input of the first layer in the d^{th} block is the output of the $(d-1)^{th}$ block. In addition, $W_{\cdot,\cdot}$ is the parameters in the 3×3 filter.

Right side layer Similar to the left side, each layer of the right side is also composed by two repeated convolution operations. Yet the input of each layer in the right side contains a concatenation with the correspondingly feature maps from the left side. For the i^{th} layer of right side in the d^{th} block, the output can be formulated as:

$$H_{i,d} = \sigma(W_{i2,d} * \sigma(W_{i1,d} * [H_{i-1,d}, H_{2N-i,d}]) \tag{2}$$

where $[.,.]$ refers to the concatenation of feature maps.

Last layer We only apply one convolution operation for the last layer (pink in **Fig. 2**) with the activation function removed, which has been verified to be important to the improvement of representation ability in super-resolution task. Consequently, the formulation becomes:

$$H_{2N,d} = W_{2N,d} * H_{2N-1,d} \tag{3}$$

Local residual shortcut To further improve the representation ability, we introduce the residual connection to our design, which turns out to be the following formulation:

$$H_d = H_{d-1} \oplus H_{2N,d} \tag{4}$$

Here, H_{d-1} represents the output of the $(d-1)^{th}$ block and operation $\oplus$ is element-wise summation.

3.2 Network Structure

Our proposed architecture is shown in **Fig. 3**, which is an end-to-end mapping from LR image I_{LR} to HR image I_{HR}. In detail, it consists of three parts: shallow feature extraction sub-network (FENet), feature transformation sub-network (FTNet) and up-sampling sub-network (UPNet).

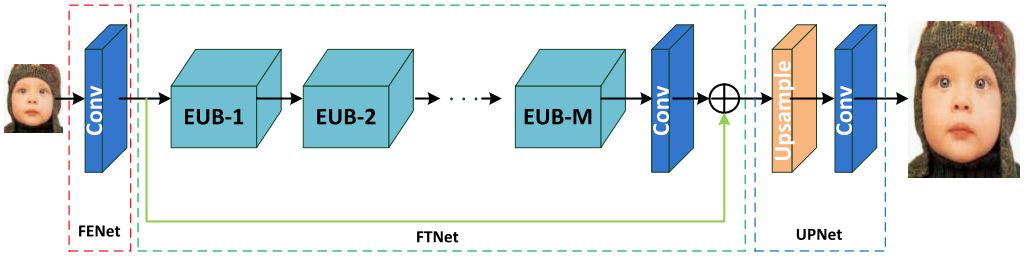


Fig. 3. The architecture of the proposed network, which consists of three parts: shallow feature extraction sub-network (FENet), feature transformation sub-network (FTNet) and up-sampling sub-network (UPNet). The EUB-*i* in the network denotes the proposed block.

Firstly, following the method in [10, 11], a convolution layer is employed for the shallow feature extraction sub-network. Specifically, the process can be expressed as:

$$H_0 = F_{EF}(I_{LR}) \tag{5}$$

Later on, the output H_0 is used as the input of the following two sub-networks. On one hand, it is input to the feature transformation sub-network to obtain fine features further. On the other hand, it is served for the global residual learning to perform element-wise summation with the input of the up-sampling sub-network.

Basically, in the feature transformation sub-network, multiple EUBs are cascaded by recurrent implementation. For the k^{th} EUB, the formulation can be expressed as:

$$H_k = F_{EUB,k}(H_{k-1}) = F_{EUB,k}\left(F_{EUB,k-1}(\cdots(F_{EUB,1}(H_0)))\right) \tag{6}$$

where $F_{EUB,k}$ denotes the k^{th} EUB operation with the output H_k. Provided that there are M EUBs in the cascaded structure, the final output will be H_M.

Next, a convolution operation is employed for further feature extraction:

$$H_{FT} = F_{CONV}(H_M) \tag{7}$$

where H_{FT} can be regarded as the output from feature transformation sub-network (FTNet).

Specifically, a global residual operation is conducted before the up-sampling sub-network, where the residual comes from the output of shallow feature extraction sub-network (FENet). Then, the output of global residual operation is upscaled by an upscale module:

$$H_{UP} = F_{UP}(H_{FT} \oplus H_0) \tag{8}$$

where $F_{UP}(.)$ is an upscale function. Several methods have been proposed for image upscaling, such as transposed convolution [25] and sub-pixel convolution [23]. In this paper, we choose the sub-pixel convolution as our upscale module, which has been proven to be efficient with promising performance in SISR.

The final reconstructed process is a convolution operation with upscaled result H_{UP} as its input:

$$I_{HR} = F_{CONV}(H_{UP}) = F_{EUN}(I_{LR}) \tag{9}$$

where F_{EUN} denotes our total SISR model and I_{HR} is the output of final HR image.

3.3 Loss Function

In fact, several loss functions are available for SISR, such as L_1loss [11, 20], L_2 loss [7, 8, 12] and adversarial loss [10].

In this paper, we regard SISR as a regression in pixel-level and construct the loss function through L_1 loss, which has been proven to be more powerful in performance [11].

Suppose that the i^{th} sample is $\{I_{LR}^i, I_{HR}^i\}$, $i = 1,2,\dots,n$, where n is the total number of training patches and I_{HR}^i is the ground truth corresponding to I_{LR}^i.

Hence, the final loss function can be straightforward formulated as:

$$L(\theta) = \frac{1}{n}\Sigma_{i=1}^{n} \parallel F_{EUN}(I_{LR}^i) - I_{HR}^i \parallel_1 \tag{10}$$

where θ is the parameters of the proposed model. The details of optimization process are shown in Section 5.1.

4. Structure Comparison

Besides U-Net, another analogous structure to our work is RDN [21]. In this section, we will discuss the network characteristics and investigate the differences.

4.1 Difference to U-Net

Our basic block is inspired by the structure of U-Net [22], which is originally designed for image segmentation task. In order to leverage traditional U-Net to SISR task, we improve it in four aspects. First of all, we introduce the input-output residual path to U-Net to boost the circulation both the information flow in forward propagation and the gradient flow in back propagation. Moreover, the total training process can be more stable by this residual shortcut. The second aspect is that the up-sampling and down-sampling are removed compared with the original U-Net, which is specifically designed for SISR to avoid the details impairment incurred by down-sampling. Instead, we implement same padding to maintain the spatial dimension, so there is no cropping either. The third revision will be the replacement of ReLU activation function [26] with LReLU [24] and the removal of activation function at the last layer. We discover that the training process will be more stable with LReLU and a better performance can be obtained. Lastly, the number of channels in each layer is fixed in our architecture, which is inversely proportional to feature dimension in the original U-Net. This is mainly due to the removal of pooling and up-sampling, which are the principle reason for altering channel numbers, in order to balance the information loss.

4.2 Difference to RDN

Although the overall configuration between RDN and our network is similar, they are still distinguishable in three folds. Firstly, we only employ one convolution layer to extract shallow feature, while two repeated convolution layers are used in RDN. The experimental results demonstrate our setup helps to reduce parameters without weakening the performance. Secondly, the basic block employed to extract information is disparate. RDN is constructed based on DenseNet [20]. Our design enables our block to grow much deeper, which is in favor of generating diverse information. Hence, we only need one-tenth blocks of that in RDN to reach a comparable result. Thirdly, RDN employs global feature fusion (GFF) among different blocks, while only one convolution operation is hardly fusing all information from large amounts of previous features. Differently, we replace GFF with cascaded connection in our

model. As a result, our proposed method has a better performance than that of RDN.

5. Experiments

5.1 Implementation Details

Datasets and metrics For training stage, we employ a newly released high-quality image dataset "DIV2K", which is originally proposed for image restoration tasks and consists of 800 training images, 100 validation images and 100 test images. We randomly select 5 images from 100 validation images to constitute the validation dataset. Meanwhile, we select five standard benchmark datasets: "Set5" [27], "Set14" [28], "Urban100" [29], "B100" [30] and "Manga109" [31] to evaluate final results, where PSNR and SSIM are two measurements computing values based on Y channel. Higher value in PSNR and SSIM means better performance.

Block setups. In our proposed EUN, the input and output are color images with channel of 3. We fix the kernel size of all the convolutional layers as 3×3, and employ same padding to keep the dimension unchanged after convolution operation. Furthermore, sub-pixel convolution is used to upscale the feature maps, which are extracted after the FTNet. In addition, following the definition in Section 3.1, we set $N = 5$, then the total number of convolution layers within one EUB is $4N - 1 = 19$.

Training settings. In accordance with the algorithm in [11], we subtract the mean RGB values of all training images. In each training epoch, 16 RGB LR images are chosen as input with size of 48×48. In practice, the size of input image can be arbitrary as EBN only contains convolution layers. Furthermore, data augmentation is performed for the input with flipping horizontally or vertically and rotating 90°, 180° and 270°, which is similar to the method in [11]. We set the initial learning rate to 10^{-4}, which is halved after every 200 epochs. The total network is training on the PyTorch framework [32] with ADAM optimizer [33], where the parameters setting is $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. In this paper, β_1, β_2 and ϵ are experienced values.

5.2 Study on Principle Hyper-parameters

Now, we discuss the effects of different hyper-parameter selections in our model. In detail, two principle hyper-parameters are considered when construing the network: the number of EUBs (denoted as M) and the number of channels in every convolution layer in EUB (denoted as C).

We evaluate the effect of each hyper-parameter in three settings: 1, 5 and 8 for the block number (M); 32, 64 and 128 for the channel number (C). We fix one hyper-parameter and evaluate the effect of the other one. The final results are shown in **Fig. 4**. Here, we randomly select 5 validation images from "DIV2K" to check the performance in 200 epochs, where the y-axis represents the average PSNR of the 5 images.

From the result of **Fig. 4(b)**, we can observe that larger C will result in a better performance, which is mainly because more filter parameters can increase the representation capacity of the network. Meanwhile, the effect of M to final performance is relatively small compared with that of C when decreasing the hyper-parameters. This can be observed from **Fig. 4(a)**, where 5 blocks can achieve comparable performance as that of 8 blocks. A main reason is that the parameter reduction is not in same scale. In this circumstance, the parameters will reduce in proportions of 1/2 when halving M and 3/4 when halving C respectively.

More importantly, **Fig. 4(a)** indicates that our model can achieve a relatively good

performance with only one EBU, which justifies the high efficiency of the proposed block in extracting and utilizing information. Consequently, we can construct lightweight networks with comparable performances (see Section 5.4).

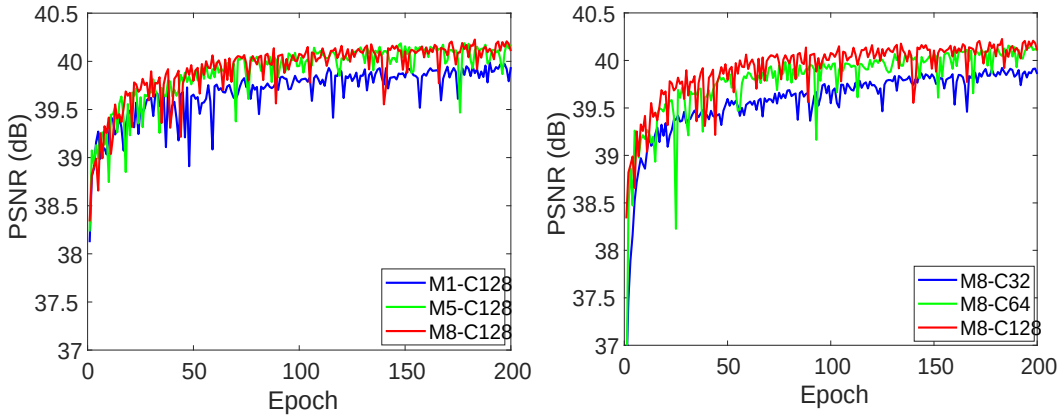


Fig. 4. Convergence results with different values in *M* and *C*, where (a) is the investigation for *M* and (b) is for *C*. The curves are obtained at × 2 scaling factor with 5 validation images from "DIV2K"

5.3 Comparison with the Advanced Methods

We compare the proposed EUN with 9 concurrent state-of-the-art methods: SRCNN [7], LapSRN [34], EDSR [11], D-DBPN [35], RDN [21], SRFBN [16], IFM [19], SCN [18], and RNAN [17]. Similar to the methods from [21, 34], the self-ensemble strategy is adopted to further improve the performance, which is denoted as EUN+. From the investigation above, large M and C would be likely to end up with a satisfying performance. However, considering fair comparison, we report the final results with a relatively moderate hyper-parameters setting, i.e. $M = 8$, $C = 128$, which demonstrates high effectiveness of the proposed network. Noting that RCAN [36] is excluded in the comparison, because of its excessive depth.

Table 1 shows quantitative comparisons of different algorithms at × 2, × 3 and × 4 scaling factors, where we mark the best and the second best results in red and blue respectively. Compared with the state-of-the-art methods, our EUN+ outperforms all other networks on all datasets under all the scaling factors. In practice, our EUN exceeds other methods on most of the datasets without the self-ensemble strategy as well. Exceptionally, when scaling factor is ×2, IFM achieves a second best on "Set5", and RDN achieves a second best SSIM on "Set14", while our EUN has beat IFM and RDN when scaling factor is ×3 and ×4.

Table 1. Test results on benchmark datasets under various scaling factors.
The best results are marked in red, and the second best ones are marked in blue.

Dataset	Scale	Set5	Set14	B100	Urban 100	Manga109
Bicubic	x2 x3 x4	33.66/0.9299 30.39/0.8682 28.42/0.8104	30.24/0.8688 27.55/0.7742 26.00/0.7027	29.56/0.8431 27.21/0.7385 25.96/0.6675	26.88/0.7388 24.46/0.7349 23.14/0.6577	30.80/0.9339 26.95/0.85562 4.89/0.7866

LapSRN	x2 x3 x4	37.52/0.9591 33.82/0.9227 31.54/0.8855	33.08/0.9130 29.96/0.8349 28.19/0.7720	31.80/0.8950 28.82/0.7973 27.32/0.7280	30.41/0.9101 27.07/0.8272 25.21/0.7553	37.27/0.9740 32.19/0.9334 29.09/0.8893
SRCNN	x2 x3 x4	36.66/0.9542 32.75/0.9090 30.48/0.8628	32.42/0.9063 29.28/0.8209 27.49/0.7503	32.42/0.9063 29.28/0.8209 27.49/0.7503	32.42/0.9063 29.28/0.8209 27.49/0.7503	32.42/0.9063 29.28/0.82092 7.49/0.7503
EDSR	x2 x3 x4	38.11/0.9601 34.65/0.9282 32.46/0.8968	33.92/0.9195 30.52/0.8462 28.80/0.7876	33.92/0.9195 30.52/0.8462 28.80/0.7876	33.92/0.9195 30.52/0.8462 28.80/0.7876	33.92/0.9195 30.52/0.84622 8.80/0.7876
D-DBPN	x2 x3 x4	38.09/0.9600 –/– 32.46/0.8969	33.85/0.9190 –/– 28.82/0.7860	33.85/0.9190 –/– 28.82/0.7860	33.85/0.9190 –/– 28.82/0.7860	33.85/0.9190 – /– 28.82/0.7860
RDN	x2 x3 x4	38.24/0.9614 34.71/0.9296 32.47/0.8990	34.01/0.9212 30.57/0.8468 28.81/0.7871	32.34/0.9017 29.26/0.8093 27.72/0.7419	32.89/0.9353 28.80/0.8653 26.61/0.8028	39.18/0.9780 34.13/0.9484 31.00/0.9151
SRFBN	x2 x3 x4	38.11/0.9609 34.70/0.9292 32.47/0.8983	33.82/0.9196 30.51/0.8461 28.81/0.7868	32.29/0.9010 29.24/0.8084 27.72/0.7409	32.62/0.9328 28.73/0.8641 26.60/0.8015	39.08/0.9779 34.18/0.9481 31.15/0.9160
IFM	x2 x3 x4	38.26/0.9614 34.75/0.9298 32.56/0.8995	33.99/0.9200 30.61/0.8466 28.80/0.7882	32.37/0.9020 29.29/0.8102 27.73/0.7422	33.09/0.9365 28.97/0.8683 27.73/0.7422	39.26/0.9784 34.14/0.9490 27.73/0.7422
SCN	x2 x3 x4	38.18 / 0.9614 34.60 / 0.9295 32.39 / 0.8981	33.99 / 0.9208 30.50 / 0.8467 28.74 / 0.7869	32.39 / 0.9024 29.26 / 0.8104 27.69 / 0.7415	33.13 / 0.9374 28.79 / 0.8667 26.50 / 0.8000	35.10 / 0.9411 31.28 / 0.8800 29.18 / 0.8253
RNAN	x2 x3 x4	38.17/0.9611 –/– 32.49/0.8982	33.87/0.9207 –/– 28.83/0.7878	32.32/0.9014 –/– 27.72/0.7421	32.73/0.9340 –/– 26.61/0.8023	39.23/0.9785 –/– 31.09/0.9149
EUN(ours)	x2 x3 x4	38.23/0.9614 34.72/0.9298 32.57/0.8998	34.05/0.9208 30.62/0.8475 28.85/0.7885	32.38/0.9022 29.30/0.8106 27.76/0.7432	33.08/0.9370 28.91/0.8679 26.73/0.8063	39.32/0.9786 34.26/0.94923 1.18/0.9175
EUN+(ours)	x2 x3 x4	38.29/0.9616 34.82/0.9306 32.70/0.9012	34.10/0.9213 30.73/0.8494 28.96/0.7905	32.43/0.9027 29.37/0.8119 27.84/0.7448	33.26/0.9384 29.14/0.8716 26.96/0.8112	39.49/0.9789 34.61/0.95083 1.55/0.9205

In **Fig. 5**, we illustrate the visualized comparisons of different methods, where *zebra* from "Set14" and *img-93* from "Urban100" are the test images. For the image *zebra*, we observe that all the other methods fail to recover its stripe texture in the bottom of the image, while both EUN and EUN+ can generate a continuous stripe. For the image *img-93*, our EUN can deliver a more realistic result to the ground truth, that is, the recovered lines by EUN and EUN+ are straighter and finer to the naked eyes. These results mainly benefit from the high efficiency in extracting and utilizing information.

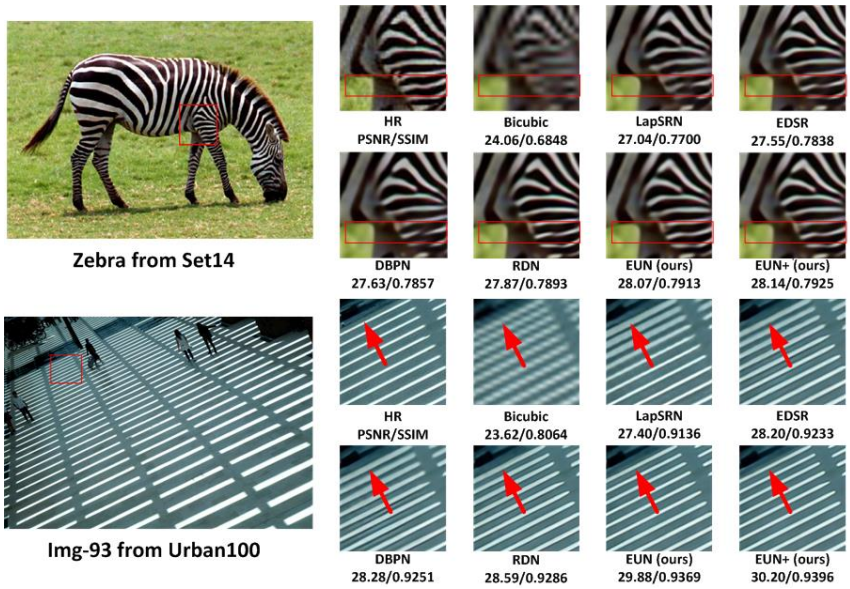


Fig. 5. Visualized results on × 4 scaling factor with the image *zebra* from "Set14" and *img-93* from "Urban100" respectively. The red rectangle boxes and the red arrows denote the regions with relatively clear distinctions.

5.4 Results with Lightweight Network

In line with the method in [13, 16, 37, 39], we also construct two lightweight networks EUNM and EUNS with the hyper-parameters setting as $M = 1$, $C = 64$ and $M = 1$, $C = 32$ respectively. Because of the space limitation, we only choose several recently published state-of-the-art methods as the baseline to illustrate our networks, including VDSR [9], IDN [38], CARN, CARN-M [37], SRFBN-S [16], and AWSRN-S [39], where all those methods do have mechanism as a tradeoff between model performance and computing complexity.

Table 2. Test results on benchmark datasets under various scaling factors.
The best results are marked in red, and the second best ones are marked in blue.

Dataset	Scale	Parameters	Multi-Adds	Set5	Set14	B100	Urban100
VDSR	x2	665K	612.6G	37.53/0.9587	33.03/0.9124	31.90/0.8960	30.76/0.9140
	x3	665K	612.6G	33.66/0.9213	29.77/0.8314	28.82/0.7976	27.14/0.8279
	x4	665K	612.6G	31.35/0.8838	28.01/0.7674	27.29/0.7251	25.18/0.7524
IDN	x2	579K	133.4G	37.83/0.9600	33.30/0.9148	32.08/0.8985	31.27/0.9196
	x3	587K	60.1G	34.11/0.9253	29.99/0.8354	28.95/0.8031	27.42/0.8359
	x4	600K	34.5G	31.82/0.8903	28.25/0.7730	27.41/0.7297	25.41/0.7632
CARN	x2	1592K	222.8G	37.76/0.9590	33.52/0.9166	32.09/0.8978	31.92/0.9256
	x3	1592K	118.8G	34.29/0.9255	30.29/0.8407	29.06/0.8034	28.06/0.8493
	x4	1592K	90.9G	32.13/0.8937	28.60/0.7806	27.58/0.7349	26.07/0.7837
CARN-M	x2	412K	91.2G	37.53/0.9583	33.26/0.9141	31.92/0.8960	31.23/0.9193
	x3	412K	46.1G	33.99/0.9236	30.08/0.8367	28.91/0.8000	27.55/0.8385
	x4	412K	32.5G	31.92/0.8903	28.42/0.7762	27.44/0.7304	25.62/0.7694

SRFBN-S	x2 x3 x4	483K 483K 483K	– – 132.5G	37.78/0.9597 34.20/0.9255 31.98/0.8923	33.35/0.9156 30.10/0.8372 28.45/0.7779	32.00/0.8970 28.96/0.8010 27.44/0.7313	31.41/0.9207 27.66/0.8415 25.71/0.7719
AWSRN-S	x2 x3 x4	397K 477K 588K	91.2G 48.6G 37.7G	37.75/0.9596 34.02/0.9240 31.77/0.8893	33.31/0.9151 30.09/0.8376 28.35/0.7761	32.00/0.8974 28.92/0.8009 27.41/0.7304	31.39/0.9207 25.57/0.8391 25.56/0.7678
EUNM (ours)	x2 x3 x4	1032K 1216K 1179K	237.8G 124.5G 67.9G	37.89/0.9601 34.31/0.9268 32.06/0.8940	33.53/0.9168 30.27/0.8409 28.52/0.7803	32.15/0.8994 29.07/0.8044 27.54/0.7349	31.91/0.9263 28.07/0.8505 26.00/0.7837
EUNS (ours)	x2 x3 x4	258K 304K 294K	59.5G 31.1G 16.9G	37.72/0.9597 34.10/0.9249 31.87/0.8902	33.27/0.9150 30.07/0.8370 28.38/0.7761	32.03/0.8979 28.95/0.8015 27.42/0.7307	31.39/0.9211 27.65/0.8413 25.62/0.7703

Table 2 gives PSNR and SSIM values of different networks on four benchmark datasets. Furthermore, we also present the amounts of total parameters and the numbers of Multi-Adds operations, similar to the methods in CARN and CARN-M. More specifically, the Multi-Adds is the total multiplications and additions, and computed based on the HR image with size *720p* (1280×720).

When comparing with other lightweight models, our EUNM can achieve the best average results on most datasets with scaling factors at ×2 and ×3. Moreover, EUNS achieves similar performance with comparatively fewer parameters and memory compared with CARN-M, SRFBN-S, and AWSRN-S.

However, EUNM fails to maintain the similar advantage to CARN when the scaling factor is ×4, which is primarily due to the following two reasons. On one hand, CARN has more parameters than EUNM, thus results in better representation capacity. On the other hand, CARN is trained through multi-scale strategy, which not only contains the information from scaling factors at ×2 and ×3, but also on average 1/3 of the parameters are account for ×4 super-resolution. Nevertheless, it is noticeable that not all the situations require a multi-scale super-resolution task. In those single scale scenarios, the parameters of CARN would increase to three times compared to that in **Table 2**.

Besides, we also show the visualized comparison of different methods on *img-96* from "Urban100" under scaling factor × 4 in **Fig. 6**. The edges of windows produced by most competitive methods are blurred, while our EUNM can recover clearer and sharper edges, the most faithful to the ground truth.

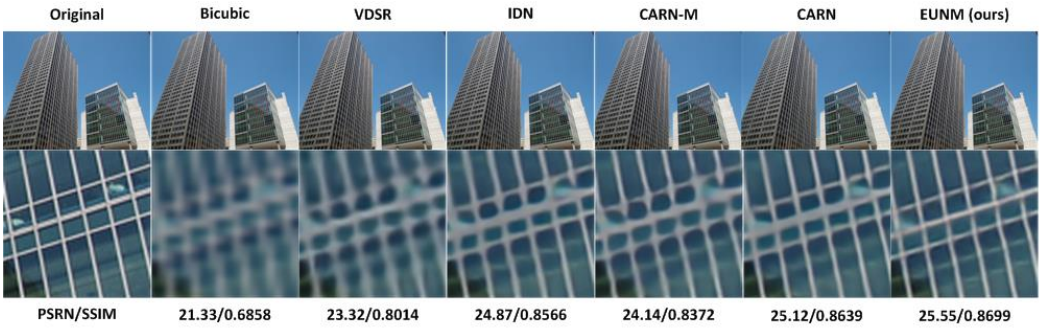


Fig. 6. Visualized results on the *img-96* from "Urban100" with scaling factor at × 4. EUNM can recover a clearer and sharper edges of the windows compared to other networks, the most faithful to the ground truth.

5.5 Tradeoff Between Performance and Parameters

Now we analyze the tradeoff between the performance and parameters of different methods. **Fig. 7** illustrates the comparison of different methods with our EUNM and EUNS in terms of Multi-Adds and PSNR, where the *x*-axis represents the total Multi-Adds operations, and *y*-axis is the average PSNR on "Set5" dataset with scaling factor at × 2. The number of parameters is not displayed, because it will be proportional to the number of operations when LR image size fixing, except for VDSR. In particular, we only choose the methods whose total Multi-Adds are less than 700G.

Obviously, we can tell that our EUNM outperforms other methods list in the figures. Note that EUNM can exceed the state-of-the-art method CARN, whose Multi-Adds is almost equivalent with our EUNM. Furthermore, our EUNS exceeds the methods VDSR, SRCNN, LapSRN, CARN-M in performance with comparable Multi-Adds. For example, the Multi-Adds of our EUNS is equivalent to SRCNN, while a visible increase in performance can be reached by EUNS. Besides, the performance of EUNS is equivalent to the recently proposed method AWSRN-S, whose Multi-Adds is larger than our EUNS.

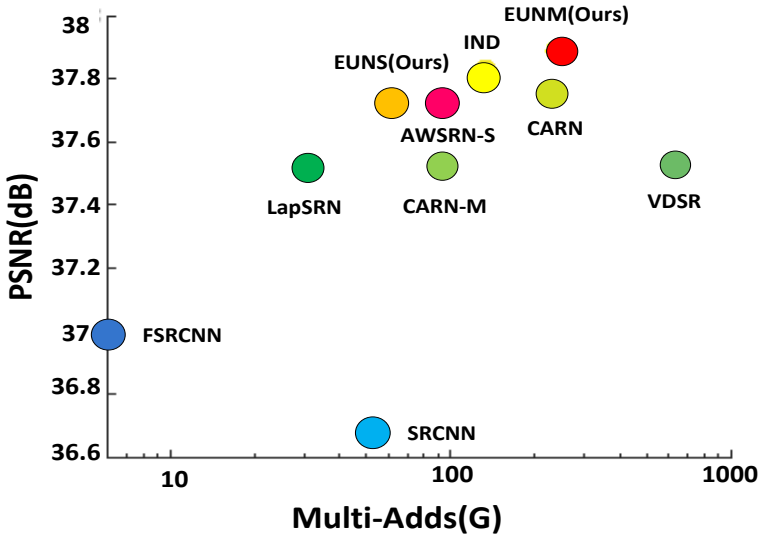


Fig. 7. Tradeoffs between performance and the Multi-Adds of different methods on the "Set5" dataset with × 2 scaling factor. The parameter scale is proportional to Multi-Adds except for VDSR.

6. Conclusion

In this paper, we design an enhanced U-Net block (EUB) to efficiently extract abundant information from an image and propose a novel single image super-resolution model EUN based on our EUB. Taking advantage of multiple skip concatenations, EUB can successfully propagate information from lower layers to higher ones, which is also useful when stacking EUBs to form EUN. The proposed method achieves competitive results compared with advanced SISR neural networks in terms of PSNR and SSIM. Several result comparisons confirm this improvement, even in visualized sense. Meanwhile, in order to deploy a lightweight network, we carefully set the hyper-parameters to extremely reduce the inference time, as well as strikingly maintain comparable performance. Hence, the proposed EUN could greatly facilitate real-world applications.

References

[1] S. Peled and Y. Yeshurun, "Super resolution in mri: application to human white matter fiber tract visualization by diffusion tensor imaging," *Magnetic Resonance in Medicine Official Journal of the Society of Magnetic Resonance in Medicine*, vol. 45, no. 1, pp. 29-35, Apr. 2015. Article (CrossRef Link)

[2] B. K. Gunturk, A. U. Batur, A. Yucel, M. H. Hayes, and R. M. Mersereau, "Eigenface-domain super-resolution for face recognition," *IEEE Transactions on Image Processing*, vol. 12, no. 5, pp. 597-606, Dec. 2003. Article (CrossRef Link)

[3] H. Zhang, L. Zhang, and H. Shen, "A super-resolution reconstruction algorithm for hyperspectral images," *Signal Processing*, vol. 92, no. 9, pp. 2082-2096, Sep. 2012. Article (CrossRef Link)

[4] C. Y. Yang, C. Ma, and M. H. Yang, "Single-image super-resolution: A benchmark," in *Proc. of European Conference on Computer Vision*, pp. 372-386, Sep. 2014. Article (CrossRef Link)

[5] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861-2873, Sep. 2010. Article (CrossRef Link)

[6] S. Schulter, C. Leistner, and H. Bischof, "Fast and accurate image upscaling with super-resolution forests," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3791-3799, June 2015. Article (CrossRef Link)

[7] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. of European Conference on Computer Vision*, pp. 184-199, Sep. 2014. Article (CrossRef Link)

[8] C. Dong, C. C. Loy, K. He, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Proc. of European Conference on Computer Vision*, pp. 391-407, Oct. 2016. Article (CrossRef Link)

[9] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1646-1654, Oct. 2016. Article (CrossRef Link)

[10] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 105-114, 2017. Article (CrossRef Link)

[11] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1132-1140, July 2017. Article (CrossRef Link)

[12] Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: A persistent memory network for image restoration," in *Proc. of International Conference on Computer Vision (ICCV)*, pp. 4549-4557, Oct. 2017. Article (CrossRef Link)

[13] T. Tong, G. Li, X. Liu, and Q. Gao, "Image super-resolution using dense skip connections," in *Proc. of International Conference on Computer Vision (ICCV)*, pp. 4809-4817, Oct. 2017. Article (CrossRef Link)

[14] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, 2004. Article (CrossRef Link)

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, Dec. 2016. Article (CrossRef Link)

[16] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, and W. Wu, "Feedback network for image super-resolution," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3862-3871, 2019. Article (CrossRef Link)

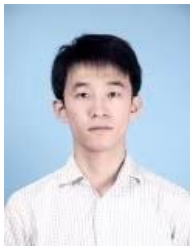
[17] Y. Zhang, K. Li, K. Li, B. Zhong, and Y. Fu, "Residual non-local attention networks for image restoration," *arXiv preprint arXiv:1903.10082*, 2019. Article (CrossRef Link)

[18] Y. Fan, J. Yu, D. Liu, and T. S. Huang, "Scale-wise convolution for image restoration," in *Proc. of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, pp. 10770-10777, Feb. 2020. Article (CrossRef Link)
[19] J. Pan, Y. Liu, D. Sun, J. S. Ren, M. Cheng, J. Yang, and J. Tang, "Image Formation Model Guided Deep Image Super-Resolution," in *Proc. of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, pp. 11807-11814, Feb. 2020. Article (CrossRef Link)
[20] G. Huang, Z. Liu, L. D. Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4700-4708, July 2017. Article (CrossRef Link)
[21] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2472-2481, June 2018. Article (CrossRef Link)
[22] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. of International Conference on Medical Image Computing and Computer-assisted Intervention*, pp. 234-241, Oct. 2015. Article (CrossRef Link)
[23] W. Shi, J. Caballero, F. Huszar, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1874-1883, Oct. 2016. Article (CrossRef Link)
[24] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectfier nonlinearities improve neural network acoustic models," in *Proc. of ICML Workshop on Deep Learning for Audio Speech and Language Processing*, pp. 818-833, June 2013. Article (CrossRef Link)
[25] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. of European Conference on Computer Vision*, pp. 818-833, Sep. 2014. Article (CrossRef Link)
[26] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. of Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 315-323, June 2011. Article (CrossRef Link)
[27] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proc. of British Machine Vision Conference*, pp. 135-145, Sep. 2012. Article (CrossRef Link)
[28] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Proc. of International Conference on Curves and Surfaces*, pp. 711-730, Dec. 2012. Article (CrossRef Link)
[29] J. B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5197-5206, June 2015. Article (CrossRef Link)
[30] D. Martin, C. Fowlkes, and D. Tal, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. of International Conference on Computer Vision (ICCV)*, pp. 416-423, July 2001. Article (CrossRef Link)
[31] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, and K. Aizawa, "Sketch-based manga retrieval using manga dataset," *Multimedia Tools and Applications*, vol. 76, no. 20, pp. 21811-21838, July 2015. Article (CrossRef Link)
[32] A. Paszke, S. Gross, and S. Chintala, "Automatic differentiation in pytorch," in *Proc. of NIPS Workshop*, pp. 1-4, Dec. 2017. Article (CrossRef Link)
[33] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. Article (CrossRef Link)
[34] W. S. Lai, J. B. Huang, N. Ahuja, and M. H. Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5835-5843, July 2017. Article (CrossRef Link)
[35] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1664-1673, July 2017. Article (CrossRef Link)

[36] Y. Zhang, K. Li, L. Kai, L. Wang, B. Zhong, and F. Yun, "Image super-resolution using very deep residual channel attention networks," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 294-310, June 2018. Article (CrossRef Link)
[37] N. Ahn, B. Kang, and K. A. Sohn, "Fast, accurate, and lightweight super-resolution with cascading residual network," in *Proc. of European Conference on Computer Vision*, pp. 256-272, Sep. 2018. Article (CrossRef Link)
[38] Z. Hui, X. Wang, and X. Gao, "Fast and accurate single image super-resolution via information distillation network," in *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 723-731, June 2018. Article (CrossRef Link)
[39] C. Wang, Z. Li, and J. Shi, "Lightweight image super-resolution with adaptive weighted learning network," *arXiv preprint arXiv:1904.02358*, 2019. Article (CrossRef Link)

Le Chang received the master's degree from Taiyuan University of Technology in 2012, and now is a lecturer at Shanxi Polytechnic College. Her research interests include machine learning, computer vision, and artificial intelligence.

Fan Zhang received the PhD degree from Beijing University of Posts and Telecommunications in 2019. Currently, she is the associate professor at Beijing Information Science and Technology University. Her research interests include machine learning and computer vision.



Biao Li received the PhD degree from University of Chinese Academy of Sciences in 2020. Currently, he is the associate professor at Southwestern University of Finance and Economics. Her research interests include machine learning and computer vision.