



Lightweight feature separation, fusion and optimization networks for accurate image super-resolution

Lin Tian¹ · Shaoshuai Gao¹ · Guofang Tu¹

Received: 26 March 2021 / Accepted: 12 October 2021 / Published online: 29 October 2021
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2021

Abstract

Recently, single-image super-resolution (SISR) methods based on deep learning have demonstrated great superiority by deepening or widening the network. However, excessive network layers will not only weaken the information flow during training process, but also increase the storage load and computation cost in practical application. To achieve a better trade-off between model efficiency and accuracy, we propose a lightweight feature separation, fusion and optimization network (SFON) for SISR. For the architecture, we design an efficient feature separation, fusion and optimization block (SFOB) to effectively capture the local cross-level features through successive channel splitting and concatenation first, and then refine them with an improved channel attention mechanism. We also adopt a MAE pooling-based feature optimization and fusion block (MAE-FOFB) to enhance the distinction and utilization of global multi-level features extracted from every SFOB. For the loss function, except for L1 loss, the structural similarity (SSIM) loss is additionally introduced to fine-tune the network, which helps to bring a slight improvement in accuracy. Moreover, we develop a variant of SFON (SFON-P) by applying progressive reconstruction strategy to further boost performance. Extensive experiments show that both SFON and SFON-P achieve favorable reconstruction accuracy against other state-of-the-art lightweight models with relatively low model complexity.

Keywords Single-image super-resolution · Lightweight network · Channel attention · Progressive reconstruction

1 Introduction

Single-image super-resolution (SISR) is a challenging ill-posed problem aiming to reconstruct a high-resolution (HR) image from the corresponding low-resolution (LR) one. Since it plays an increasingly important role in many fields such as remote sensing imaging, medical imaging, and video monitoring, plenty of SISR methods have been developed, which can be classified into three categories: interpolation-based, reconstruction-based, and learning-based. Recently,

deep-learning algorithms have been applied to SISR task and have made significant progress.

In [6] first applied convolutional neural network (CNN) to the SISR field, which has shown vast superiority to the traditional methods. Since [16] proposed a 20 layers network by applying the global residual learning and gave the proof that deepening the network can achieve promising performance, a series of deeper CNN-based models have emerged in SR tasks. Some [22, 26] are based on Resnet [8] structure and some [34, 36, 43] are based on Densenet [12] structure. In [42] even built a very deep SR model with more than 400 layers for superior accuracy. Instead of increasing the network depth, a few works [11, 24] focused on extending the network width to capture multi-scale features. Although deepening or widening the network can significantly boost performance, this usually results in high storage load and computation cost, and further restricts the application of deep learning-based SR methods in real world. Therefore, research on building a lightweight model has become increasingly important.

Communicated by Y. Zhang.

✉ Shaoshuai Gao
ssgao@ucas.ac.cn

Lin Tian
tianlin18@mails.ucas.edu.cn

Guofang Tu
gft@ucas.ac.cn

¹ University of Chinese Academy of Sciences, Beijing, China

One simple way is to adopt shallow network structures [7, 31], but too small model size will limit the feature extraction ability. Some models such as DRCN [17], DRRN [33] and MemNet [34] employ recursive learning scheme to reduce the number of parameters. However, they have to compensate for performance degradation by constructing very deep networks and still require extensive calculation. More works focus on the architecture design to reduce the number of parameters and operations while maintaining satisfying performance. In [20] proposed a Laplacian pyramid framework for fast and accurate SR. In [2] utilized a cascading mechanism upon a residual network to achieve efficient reconstruction. Some [4, 5] are based on neural architecture search mechanism and some [14, 15, 28] are based on information distillation structure. In addition, there exist many lightweight SR models based on multi-scale or multi-level feature fusion structure, in which the way to reduce model complexity is also considered, such as adopting group or depthwise separable convolution operations [21, 23, 27, 32] and utilizing channel grouping schemes [21, 37]. What's more, some of these models introduce attention mechanism [25, 27] to further improve the representational power. It can be observed that all these models try to appropriately balance between model complexity and reconstruction accuracy, but there is still much room for improvement. In addition, this motivates us to build SR models which can achieve favorable performance against other lightweight ones whereas the model complexity remains similar or lower.

To achieve the goal, we first propose a lightweight feature separation, fusion, and optimization network (SFON). Specifically, there are two key components of SFON. One is feature separation, fusion and optimization block (SFOB) which consists of two successive feature separation and fusion units (SFUs) and an improved channel attention unit (ICAU), the other is feature optimization and fusion block based on MAE pooling (MAE-FOFB). Compared with other models, our improvement mainly lies in three aspects.

In terms of architecture, SFU combines the advantage of both dense connected structure [34, 39, 43] in extracting multi-level features and channel grouping scheme [21, 37] in reducing model complexity. However, the skip connections are not as dense as that in [34, 39, 43] and our network is not widened like [21, 37] after channel splitting, which makes our SFU more lightweight. In terms of channel attention mechanism, by introducing a softmax layer, the feature maps obtained after standard deviation pooling and average pooling can be combined using adaptive weighted addition. Since the weight is learned in a self-adaptive manner, the feature discriminability of our ICAU is better than other channel attention modules in [15, 18, 25, 38, 42]. What's more, we also employ a channel attention unit (MAE-CAU) based on a novel pooling mode, named MAE pooling in MAE-FOFB. Note that ICAU will first refine the feature

maps within SFOB in a local level and MAE-CAU will further optimize the output of every SFOB in a global level, contributing to more powerful representational ability. In terms of loss function, unlike most models [2, 7, 16, 34, 37] only using MAE or MSE loss as the loss function, we add SSIM loss on the basis of MAE loss to train the network together which can constrain the smoothness of the reconstructed image to a certain extent and help to bring a higher SSIM value.

In summary, our main contributions are as follows:

- We propose SFON and its variant model SFON-P for fast and accurate SISR. Experiments show that both SFON and SFON-P can achieve competitive or superior performance with a relatively smaller model size compared with other leading lightweight SR models.
- We propose the core block SFOB to be parameter-efficient, in which the local cross-level features can be extracted via successive channel splitting and concatenation, as well as be refined via an improved channel attention unit.
- We propose an effective block MAE-FOFB to boost feature discriminability, in which the global multi-level features will be first optimized with a new MAE pooling-based channel attention mechanism before they are fused together.
- We propose SSIM loss to fine tune the network during training process for better reconstruction accuracy. The experimental results demonstrate that the introduction of SSIM loss is feasible.

The remainder of this paper is organized as follows. Section 2 reviews the related deep learning-based SISR works. In Sect. 3, we introduce our proposed methods in detail. Model analysis and experimental comparison against other methods are demonstrated in Sect. 4. Finally, we conclude our work in Sect. 5.

2 Related works

Single-image super-resolution has been extensively studied recently. In this section, we first present an overview about those advanced SR models based on deep learning. Then, we will give a brief introduction about studies concentrating on lightweight architectures for efficient SISR.

2.1 Advanced SR models based on deep learning

Since the first deep learning-based model SRCNN [6] was applied to SR tasks, many innovative architectures have been designed. Early in the development, because of the difficulty to train a deep network, there exist some models [7,

31] with shallow network layers and their training ability is constrained. Inspired by the great success of very deep networks in other computer vision (CV) tasks [8, 12], some deep models based on residual learning such as VDSR [16], SRResnet [22], and EDSR [26] have sprung up to reduce learning difficulty and improve representational ability while some deep models based on hierarchical feature fusion such as SRDenseNet [36] and RDN [43] utilize dense skip connections to obtain as much feature information as possible. To achieve better performance, the networks become deeper and deeper. However, more parameters and calculations also come. Recently, attention mechanism has been successfully applied to different CV fields [38, 40, 42]. As one kind of it, channel attention mechanism [10] is proposed to concentrate on more useful information with an extra small number of parameters and are widely used in SR tasks [15, 18, 42]. Due to its effectiveness, we also adopt this technique in our models with some modification for performance improvement.

2.2 Lightweight and efficient SR models

Recently, there has been rising much interest in designing lightweight and efficient SR models. DRCN [17], DRRN [33] and MemNet [34] adopt recursive learning to share parameters. In this way, the number of parameters can be controlled with the increasing of network depth. LapSRN [20] achieves fast and accurate SR by progressively reconstructing the high-frequency residuals at multiple pyramid levels. CARN-M [2] employs cascading residual-E blocks by combining recursive learning scheme and group convolutions to slim the network. MoreMNA-S [4] and FALSr [5] introduce neural architecture search [44] to SR field, which can automatically design efficient networks. Inspired by the information distillation mechanism in IDN [14] and IMDN [15], RFDN [28] adopts feature distillation connections (FDC) to be much more lightweight and flexible. Moreover, many SR models [9, 21, 23, 25, 27, 32, 35, 37, 39] dedicate to extract multi-scale or multi-level features with an efficient structure. For example, the multi-scale modules in LFFN [37] and MADnet [21] are based on several convolutional branches and rely on channel splitting strategy to reduce

the number of parameters. Based on the symmetric architecture, S-LWSR [23] utilizes an information pool to mix multi-level information and a compression module borrowed from MobileNet V2 [30] to decrease the number of parameters. WMRN [32] employs the modified residual structure and depthwise separable convolutions to improve operation efficiency. In AMSRN [27], a spatial and channel-wise attention residual block is constructed and group convolution is introduced to further reduce the parameters. In this paper, we also focus on designing more efficient architectures to achieve better performance.

3 Method

In this section, we first introduce the overall architecture of SFON, and then describe SFOB and MAE-FOFB in detail, which are the core of the proposed method. After that, we illustrate the improved loss function used to improve SR accuracy. Besides, we present SFON-P to further boost performance by introducing the progressive reconstruction scheme into SFON. Finally, we make comparison between our proposed method and other related works.

3.1 Framework

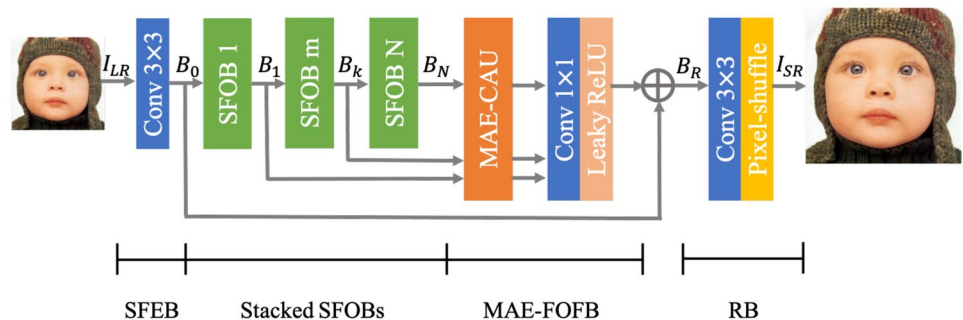
As shown in Fig. 1, the proposed SFON can be roughly divided into four parts: a shallow feature extraction block (SFEB), multiple stacked SFOBs, a MAE pooling-based feature optimization and fusion block (MAE-FOFB) and a reconstruction block (RB). Here, we denote I_{LR} and I_{SR} as the input and output of SFON, respectively.

We first use a 3×3 convolutional layer to simply extract low-level features from the original LR image:

$$B_0 = F_0(I_{LR}), \tag{1}$$

where F_0 represents the shallow feature extraction function and B_0 denotes the extracted feature maps that will pass to the following stage.

Fig. 1 The architecture of our proposed SFON. It can be roughly divided into four parts: SFEB, multiple stacked SFOBs, MAE-FOFB and RB



The next part is composed of multiple stacked SFOBs to exploit and recalibrate the local hierarchical features. This procedure can be expressed as

$$B_k = F_k(B_{k-1}), \quad k = 1, \dots, N, \quad (2)$$

where F_k denotes the k th SFOB function, B_{k-1} and B_k indicate the input and output of the k th SFOB, respectively.

Then, MAE-FOFB is required to first globally optimize the hierarchical features extracted from every SFOB with a MAE pooling based channel attention unit (MAE-CAU) and then assemble them with a 1×1 convolutional layer followed by a Leaky ReLU activation function. We also introduce global residual learning scheme to make deep network training easier. This process can be formulated as

$$B_R = F_M(B_1, \dots, B_k, \dots, B_N) + B_0, \quad (3)$$

where B_0 , $B_k (k = 1, \dots, N)$ and B_R represent the output feature maps of the SFEB, the k th SFOB, and the MAE-FOFB, respectively. F_M denotes the MAE-based feature optimization and fusion function.

Finally, we utilize a 3×3 convolutional layer and a pixel-shuffle layer to generate the HR image:

$$I_{SR} = F_{up}(B_R), \quad (4)$$

where F_{up} denotes the up-sampling operation.

3.2 Feature separation, fusion and optimization block (SFOB)

To make better use of cross-level features with fewer parameters, we propose an efficient block SFOB which is constructed by two successive feature separation and fusion units (SFUs), an improved channel attention unit (ICAU) and the local residual structure. Here, we will provide a detailed description.

3.2.1 SFUs

As shown in Fig. 2, the SFUs consists of two identical successive units. In the j th ($j = 1, 2$) convolutional layer of the i th ($i = 1, 2$) SFU (including Leaky ReLU), let us denote the output as M_i^j and the number of filters as N_i^j . Take the first SFU as illustration, we first adopt a 3×3 convolutional layer with $N_1^1 = 48$ filters followed by a Leaky ReLU activation function to extract input features as well as reduce the number of output feature maps for smaller model size. Then, given the separation rate $s = 3$, we separate M_1^1 into two parts with different number of feature maps by channel splitting, which are $N_1^1/s = 16$ and $N_1^1 - N_1^1/s = 32$, respectively. The part with 32 feature maps will be fed into subsequent 3×3 convolutional layer with $N_1^2 = 48$ filters. The other part with 16 feature maps will fuse with M_1^2 by channel concatenation for better utilization of richer cross-level features. The same process is repeated for the second SFU, which can be described as

$$M_i^1 = F_{s_i}^1(B_i^{\text{in}}), \quad (5a)$$

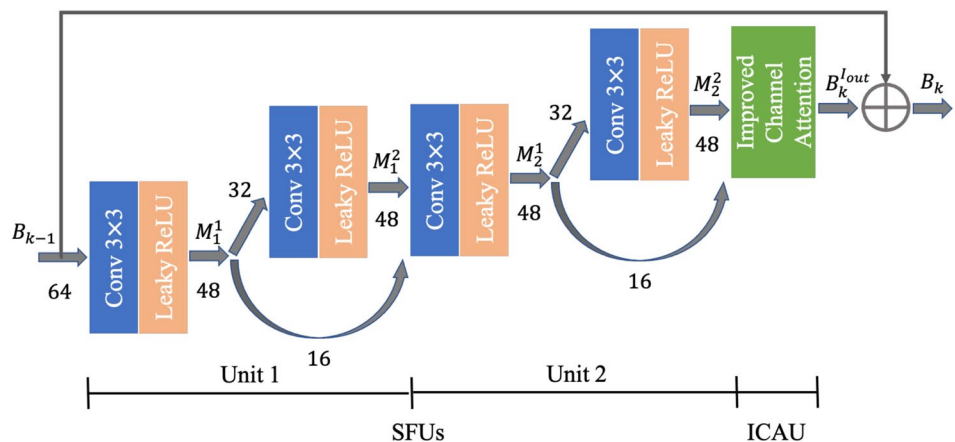
$$M_i^2 = F_{s_i}^2(S(M_i^1, 1 - 1/s)), \quad (5b)$$

$$B_i^{\text{out}} = F(M_i^2, S(M_i^1, 1/s)), \quad (5c)$$

where $F_{s_i}^j (i, j = 1, 2)$ denotes the operation of the j th convolutional layer (including Leaky ReLU) in the i th SFU, S and F indicate feature separation and feature fusion operation, respectively, and B_i^{in} and B_i^{out} represent the input and output of i th SFU, respectively.

In each SFU, on the one hand, output feature separation can reduce the number of feature maps that will be sent to the next layer, leading to fewer parameters and computations. On the other hand, input feature fusion can introduce previous information into the current one, helping to integrate and exploit hierarchical features.

Fig. 2 The architecture of SFOB



3.2.2 ICAU

To focus on more important feature maps, we utilize an improved channel attention mechanism for feature recalibration among different channels and levels in a local manner. As shown in Fig. 3, the channel attention module in RCAB [42] and CBAM [38] is based on global average or maximum pooling to achieve high PSNR value while in MAMB [18] is based on variance pooling to restore high-frequency details. By combining their advantages, IMDB [15] utilizes both the standard deviation pooling and average pooling to achieve better SR performance. However, it ignores that different pooling modes have different importance to the feature maps by simple summation. To solve this problem, we introduce a softmax layer so that the feature maps obtained from standard deviation pooling and average pooling can be weighted adaptively.

Let us denote $B_k^{in} \in R^{H \times W \times C}$ as the input of ICAU in the k th SFOB, where C and $H \times W$ represent the number and the spatial shape of feature maps, respectively. As shown in Fig. 3e, we first use average pooling and standard deviation pooling to squeeze B_k^{in} into two types of feature maps $M_1^i, M_2^i \in R^{1 \times 1 \times C}$. Then, we utilize concatenation operation, softmax activation and split operation to adaptively balance the importance between the two pooling modes and produce the weight maps $W_i^i \in R^{1 \times 1 \times C}, (i = 1, 2)$. This process can be expressed as

$$W_1^i, W_2^i = \text{Split}(\text{Softmax}(\text{Concat}[M_1^i, M_2^i])), \tag{6a}$$

$$W_i^i = \sum_{j=1}^C w_i^{ij}, \tag{6b}$$

where $W_i^i = [w_i^{i1}, \dots, w_i^{ij}, \dots, w_i^{iC}]$. It should be noted that the softmax function will make the formula $\sum_{i=1}^2 w_i^{ij} = 1$ workable. Further, we can obtain the weighted feature map $M_I \in R^{1 \times 1 \times C}$:

$$M_I = \sum_{i=1}^2 M_i^i \cdot W_i^i. \tag{7}$$

Afterwards, based on previous work [10], we exploit dimension reduction, ReLU activation, dimension increasing and sigmoid activation to obtain the target scalar $s_I \in R^{1 \times 1 \times C}$ for reweighting the input B_k^{in} :

$$B_k^{lo} = B_k^{in} \otimes s_I, \tag{8}$$

where \otimes refers to the channel-wise multiplication between B_k^{in} and s_I . B_k^{lo} represents the recalibrated feature maps.

In the end, we utilize a 1×1 convolutional layer to further refine image features and obtain B_k^{out} , which denotes the output of ICAU.

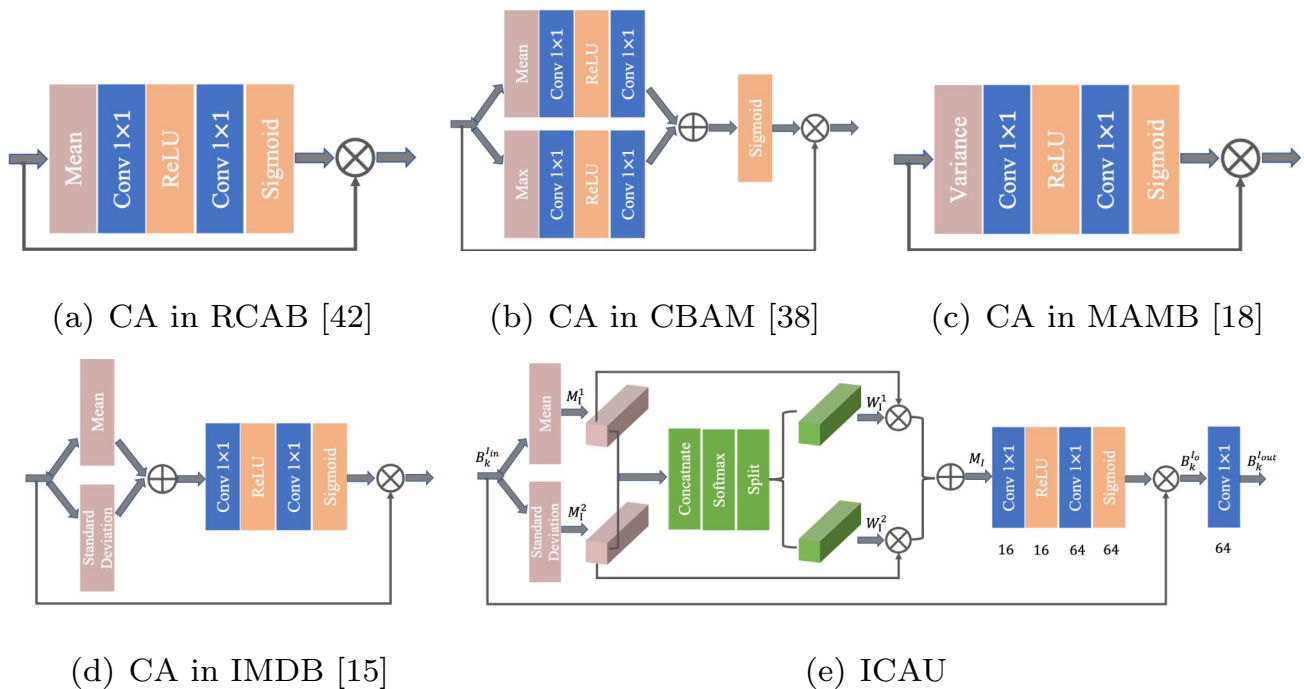


Fig. 3 Different channel attention mechanisms. CA channel attention

3.2.3 Local residual structure

To reduce learning difficulty and speed up convergence during the training process, we adopt residual learning scheme in each SFOB. Therefore, the final output of the k th SFOB can be formulated as

$$B_k = B_k^{\text{out}} + B_{k-1}. \quad (9)$$

3.3 MAE pooling-based feature optimization and fusion block (MAE-FOFB)

To make full use of the global multi-level features, we will apply a bottleneck layer (1×1 convolution) to concatenate the output of every SFOB as well as reduce the number of output feature maps as most previous works do. However, it is not enough to enhance discriminability of features from different levels only by feature fusion. Therefore, before the bottleneck layer, we add a MAE pooling based channel attention unit (MAE-CAU) for optimization. The structure is shown in Fig. 4.

Let us denote the output of the k th SFOB as $[B_{k1}, \dots, B_{kc}, \dots, B_{kC}]$, which has C feature maps with spatial size $H \times W$. The c th map M_{kc} after MAE pooling can be calculated by

$$\mu_{kc} = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W B_{kc}(i, j), \quad (10a)$$

$$M_{kc} = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W |B_{kc}(i, j) - \mu_{kc}(i, j)|. \quad (10b)$$

Noting that the loss function of SFON is mainly based on MAE (see Sect. 3.4), MAE-CAU will help to minimize the reconstruction error and favor a high PSNR value. What's more, the output feature maps of every SFOB have been refined by ICAU using adaptive weighted average and standard deviation pooling in a local manner. Here, they will be further optimized by another simple but effective squeeze



Fig. 4 The structure of MAE-CAU

strategy, MAE pooling, in a global manner, contributing to more powerful representational ability.

3.4 Loss function

Mean square error (MSE) and mean absolute error (MAE) are two types of loss functions that have been widely used in SR tasks. However, the paper [26] shows that MSE loss, also called L2 loss, performs poorly to generate clear images. Therefore, we choose MAE, also called L1 loss, as our first loss function. By denoting the SR image and its corresponding ground truth as I_{SR} and I_{HR} , respectively, we can give the formulation as follows:

$$L_1 = \|I_{\text{SR}} - I_{\text{HR}}\|_1. \quad (11)$$

Since the model using L1 or L2 loss tends to generate smooth images, we introduce the SSIM loss L_{SSIM} to improve the image quality. Our goal is to minimize L_{SSIM} between I_{SR} and I_{HR} :

$$L_{\text{SSIM}} = \lambda [1 - \text{SSIM}(I_{\text{SR}}, I_{\text{HR}})], \quad (12)$$

where $\text{SSIM}(\ast)$ defines the calculation of structural similarity and $\lambda = 0.01$ works well.

In the training process, we first train the network with L1 loss and then fine-tune it with SSIM loss, which can achieve better performance.

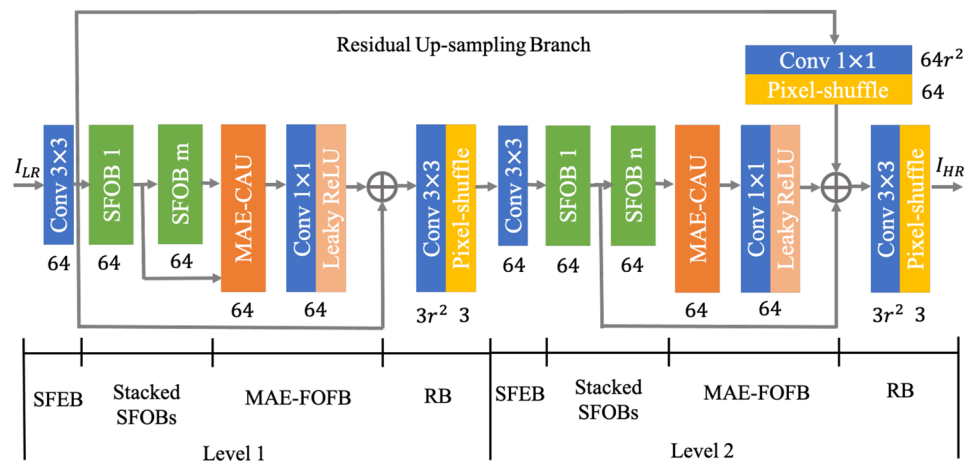
3.5 The progressive version of SFON: SFON-P

To obtain a higher quality image, we slightly modify the original SFON by adopting progressive image reconstruction strategy, which can predict HR image from coarse to fine with the scale factor $r = 2$ as the base. As shown in Fig. 5, SFON-P contains two-level sub-networks for $\times 4$ SR and each sub-network has the same architecture with SFON. The motivation for progressive reconstruction is similar to LapSRN [20]. However, there are two main differences.

First, in LapSRN [20], each sub-network structure is identical, which means the number of convolutional layers is also exactly the same. But in our SFON-P, we stack different number of SFOB at each level, denoted as m, n , respectively, *s.t.* $m > n$. After up-sampling at level one, both the height and width of feature maps will be doubled, leading to the increase of calculations in the subsequent layers. Thus, we reduce the number of SFOB in the level two by setting $n < m$ to control the computational complexity to a certain extent.

Second, LapSRN has two branches at each level that both require up-sampling, while our SFON-P just up-sample the feature maps once at each level to ease the information propagation. What's more, for full use of shallow and deep features, we introduce a residual up-sampling branch from the output of SFEB at level one to the input of RB at level two.

Fig. 5 The architecture of our proposed SFON-P



It should be noted that the kernel size of the convolutional layer in the residual up-sampling branch is 1×1 instead of 3×3 , which helps to control the model parameters.

4 Experiments

4.1 Datasets and metrics

For training, we use DIV2K [1] dataset, which contains 800 high-quality images and are widely used in SR tasks. For testing, we use four standard benchmark datasets: Set5 [3], Set14 [41], B100 [29] and Urban100 [13]. The LR image is obtained by down-sampling corresponding HR one using bicubic interpolation.

For fair comparison with existing methods, we evaluate the performance of generated images using two metrics, peak signal-to-noise-ratio (PSNR) and structure similarity (SSIM), which are calculated on the Y channel of the transformed YCbCr color space. In this paper, all experiments are only performed for the $\times 4$ SR task.

4.2 Implementation details

During the training process, we first randomly crop the RGB color patches with a size of 48×48 from LR images as the input and set the minibatch size to 16. Then, we augment

the training images with random horizontal flips and 90° rotation. Considering the good balance between model complexity and SR accuracy, we set the number of SFOBs to $N = 6$ in SFON, and $m = 4, n = 2$ in each level of SFON-P. Both SFON and SFON-P are trained with the ADAM [19] optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$. The learning rate is initialized as $5e - 4$, and then reduced by half every 10^5 iterations. We implement our models on Pytorch with a GTX 1080 Ti GPU.

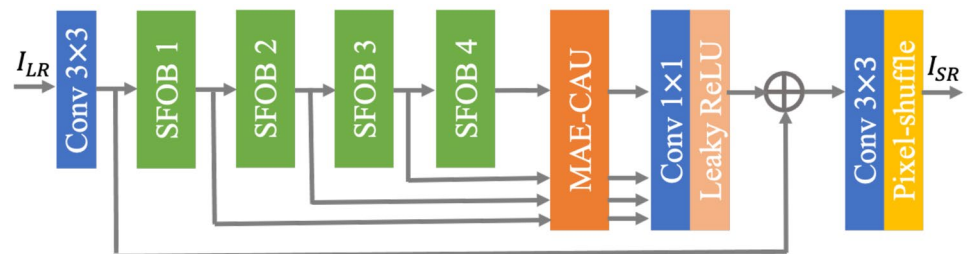
4.3 Analysis of SFON

To have a clear understanding of how SFON achieves better performance with an efficient architecture, we design the ablation experiments. To shorten the training time, we construct a small model (SFON-s, as shown in Fig. 6) by adopting 4 SFOBs to conduct ablation studies from the following three aspects: SFOB, MAE-CAU and loss function.

(1) *Study of SFOB:* As illustrated in Fig. 7, we first choose the plain structure with four cascaded 3×3 convolutional layers (64 channels) as the baseline, and then replace it with two successive SFUs. Afterwards, ICAU and residual structure will be gradually added, which finally constitute the whole SFOB.

Results are shown in Table 1. We can see that from the baseline to SFUs, the number of parameters has a significant drop from 637 to 379 K, which is approximately reduced

Fig. 6 The architecture of SFON-s



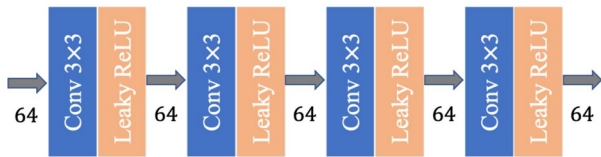


Fig. 7 The plain structure with four cascaded convolutional layers

Table 1 The ablation study of SFOB in SFON-s. RL means residual learning

| Scale | SFUs | ICAU | RL | Params (K) | Set5 PSNR (dB) | |
|-------|------|------|----|------------|----------------|------|
| | | | | | Set5 | B100 |
| ×4 | × | × | × | 637 | 31.91 | |
| | ✓ | × | × | 379 | 31.87 | |
| | ✓ | ✓ | × | 398 | 31.95 | |
| | ✓ | ✓ | ✓ | 398 | 31.98 | |

Table 2 The ablation study of MAE-CAU in SFON-s

| Scale | MAE-CAU | Params (K) | Set5 PSNR (dB) |
|-------|---------|------------|----------------|
| ×4 | × | 397 | 31.95 |
| | ✓ | 398 | 31.98 |

Table 3 The ablation study of loss function when training SFON-s

| Scale | L_{SSIM} | Params (K) | PSNR (dB)/SSIM | |
|-------|------------|------------|----------------|--------------|
| | | | Set5 | B100 |
| ×4 | × | 398 | 31.97/0.8908 | 27.45/0.7311 |
| | ✓ | 398 | 31.98/0.8912 | 27.45/0.7324 |

by 40% at a slight cost of performance. After adding ICAU, the PSNR will improve a lot on Set5, while the number of parameters has merely increased by 5%, which is still far fewer than that of the baseline. Afterwards, The addition of residual structure can lead to performance improvement without increasing parameters. Finally, comparing with the baseline, our SFOB can achieve higher PSNR with much smaller model size.

(2) *Study of MAE-CAU*: To validate the effectiveness of MAE-CAU, we remove it from SFON-s and keep other parts unchanged. Observing the results shown in Table 2, the MAE-CAU can make PSNR value improve by 0.03 dB on Set5 with extra 1 K parameters, which is only an increase of 0.25%. This demonstrates that the introduction of MAE-CAU can further optimize multi-level features, contributing to performance improvement.

(3) *Study of loss function*: To examine the effect of loss functions, we first train SFON-s with L1 loss only, and then introduce SSIM loss to it. As shown in Table 3, using SSIM

Table 4 Results of the allocation of SFOB in SFON-P

| Scale | $m + n$ | Params (K) | PSNR (dB) | |
|-------|---------|------------|-----------|-------|
| | | | Set5 | B100 |
| ×4 | 5 + 1 | 587 | 32.11 | 27.51 |
| | 4 + 2 | 587 | 32.16 | 27.53 |

Table 5 The effect of the residual up-sampling branch (RUB) in SFON-P

| Scale | RUB | Params (K) | PSNR (dB) | |
|-------|-----|------------|-----------|-------|
| | | | Set5 | B100 |
| ×4 | × | 571 | 32.06 | 27.51 |
| | ✓ | 587 | 32.16 | 27.53 |

loss can bring a slight SSIM increase on both datasets. What's more, the PSNR value improves by 0.01 dB on Set5.

Noting that the above all studies are conducted based on the smaller model (SFON-s), it can be inferred that SFON which has more SFOBs will show more obvious performance superiority.

4.4 Analysis of SFON-P

To further analyze the performance behavior of SFON-P, we develop investigations from the following two aspects: the allocation of SFOB and the effect of residual up-sampling branch.

(1) *The allocation of SFOB*: We will train two models with different allocation schemes of SFOB by keeping the number of SFOB at level one larger than that at level two in the case of a fixed total number, which means the relations $m > n$ and $m + n = 6$ should hold. We set $m = 5, n = 1$ in the first model while $m = 4, n = 2$ in the second model, namely our SFON-P. It can be observed from Table 4 that our SFON-P leads to an increase of PSNR by 0.05 dB and 0.02 dB on Set5 and B100, respectively. Thus, we can conclude that based on the above restrictive conditions, the smaller the difference between m and n , the better the performance can be achieved.

(2) *The effect of residual up-sampling branch*: To verify the necessity of adding the residual up-sampling branch, we construct an extra model by removing it from SFON-P and conduct the comparing experiment. From Table 5, we can see that the introduction of residual up-sampling branch can bring absolute performance improvement. By increasing 17K parameters, which is a little increase of 2.98%, the PSNR value improves by 0.1 dB and 0.02 dB on Set5 and B100, respectively, indicating that this structure can effectively fuse the shallow and deep feature information from both levels, thus improving the performance.

4.5 Comparisons with state-of-the-arts

We compare the proposed SFON and SFON-P with several state-of-the-art lightweight SR methods, including FSRCNN [7], VDSR [16], LapSRN [20], MemNet [34], IDN [14], CARN-M [2], MADNet- L_F [21] and s -LWSR₃₂ [23] on aspect of reconstruction accuracy. We also compare the storage and computation efficiency of each method which are measured by parameter consumption and computation cost (multi-adds), respectively. The multi-adds is the number of multiply-accumulate operations and is calculated by assuming that the desired SR image size is 1280×720 .

The quantitative comparisons are listed in Table 6. In terms of image accuracy, our SFON-P significantly outperforms other methods on all datasets both in PSNR and SSIM. For scale factor of $\times 4$, the PSNR value of SFON-P is 0.12 dB and 0.15 dB higher than that of s -LWSR₃₂ [23] on Set5 and Urban100, respectively, and 0.11 dB higher than that of MADNet- L_F [21] on Set14. The SSIM value of SFON-P is also a bit higher than all the other methods. What's more, our SFON achieves the second-best performance result on almost all datasets. Even compared with s -LWSR₃₂ [23], it obtains notable PSNR gain of 0.36 dB and 0.06 dB on Set14 and Urban100, respectively, and achieves similar performance on Set5 and B100. In terms of storage efficiency, both SFON and SFON-P have a relatively small number of parameters in the range of 550K to 600K, which is less than the median number and similar to that of IDN [14] and s -LWSR₃₂ [23]. In terms of computation efficiency, our SFON-P is at the medium level with multi-adds less than 70G and our SFON further reduce it nearly by half. Especially, the multi-adds of SFON is about 1.3% of MemNet [34] and far less than most other methods.

Figure 8 presents visual results on different testing datasets for $\times 4$ SR task. It can be easily observed that the

proposed method SFON-P can reconstruct SR images with much better visualization. For example, for the image “comic” from Set14, SFON-P can preserve more details and produce fewer artifacts. For the image “160068” from B100 and “img_076” from Urban100, SFON-P can generate much clearer stripes and grid structures. What's more, our SFON can produce almost the same results with a smaller model size.

To summarize, SFON shows comparable or even better performance against other methods with relatively fewer parameters and multi-adds. SFON-P further boosts performance and achieves the best result among all methods at a slight cost of computation. Both of our proposed models can make a better balance between SR accuracy and model efficiency.

5 Conclusion

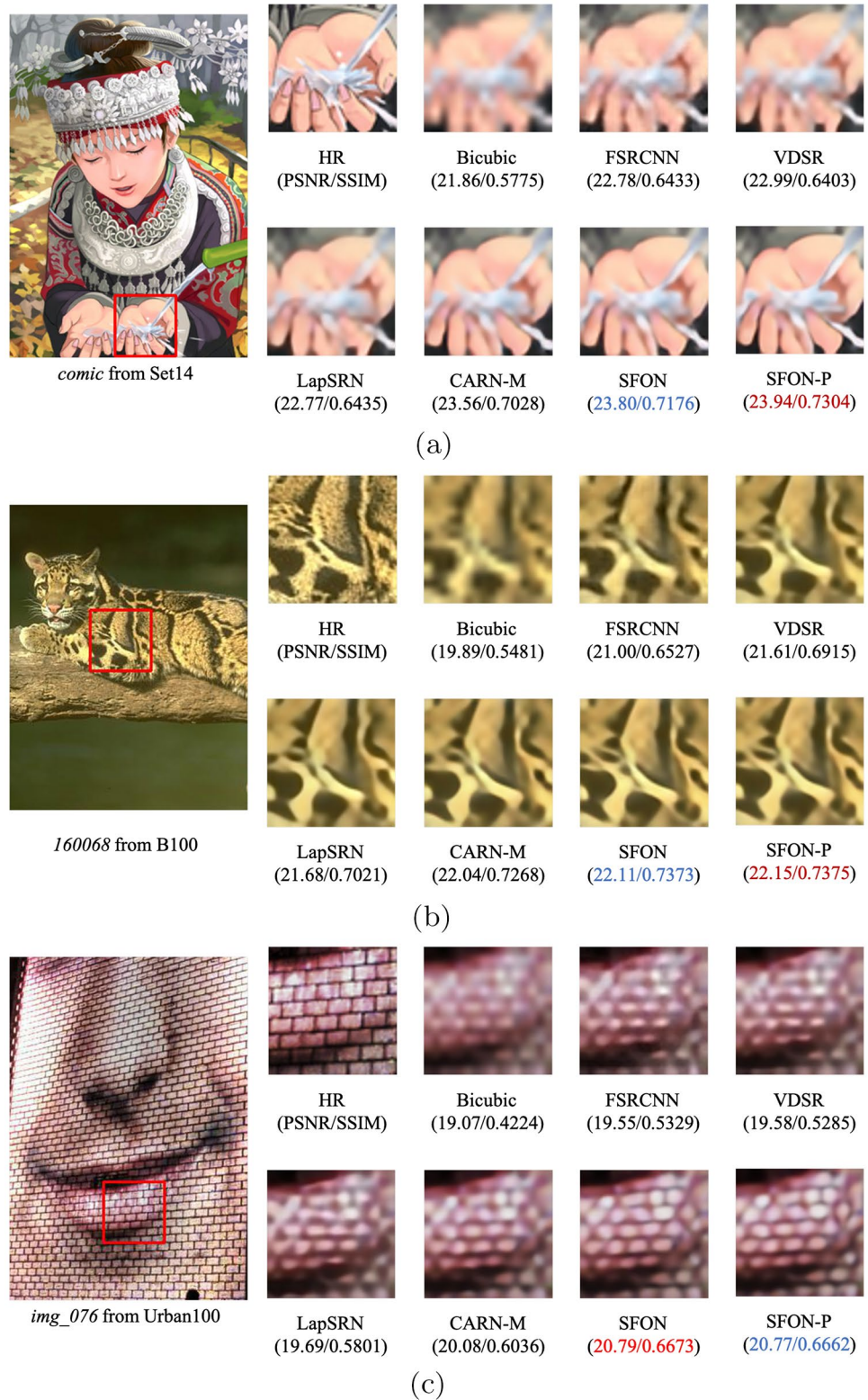
In this paper, we propose a lightweight network SFON and its variant SFON-P for accurate single-image super-resolution. Specifically, we apply multiple SFOBs to capture and refine hierarchical features in a local manner with a simple but effective structure. Following that, we utilize MAE-FOFB to optimize and fuse multi-level features in a global manner. A novel SSIM loss is also adopted to fine-tune the network in the training process. Moreover, we introduce progressive reconstruction strategy to SFON for further performance improvement, allowing the designer to tune trade-off between model performance and complexity. Comparing with other lightweight models, our SFON-P and SFON won the first and second places on all the datasets, respectively, in terms of PSNR and SSIM with a moderate number of parameters (550K–600K). Besides, the number of operations in SFON is only about 35G, which is far less than most other methods. All the experimental results demonstrate that

Table 6 Quantitative comparisons with the state-of-the-art lightweight SR models for scale factor $\times 4$ on different benchmarks

| Model | Scale | Params (K) | Multi-adds (G) | PSNR (dB)/SSIM | | | |
|------------------------------|------------|------------|----------------|---------------------|---------------------|---------------------|---------------------|
| | | | | Set5 | Set14 | B100 | Urban100 |
| FSRCNN [7] | $\times 4$ | 12 | 4.6 | 30.71/0.8657 | 27.59/0.7535 | 26.98/0.7150 | 24.62/0.7280 |
| VDSR [16] | | 665 | 612.6 | 31.35/0.8838 | 28.01/0.7674 | 27.29/0.7251 | 25.18/0.7524 |
| LapSRN [20] | | 813 | 149.4 | 31.54/0.8850 | 28.19/0.7720 | 27.32/0.7280 | 25.21/0.7560 |
| MemNet [34] | | 677 | 2662.4 | 31.74/0.8893 | 28.26/0.7723 | 27.40/0.7281 | 25.50/0.7630 |
| IDN [14] | | 590 | 81.87 | 31.82/0.8903 | 28.25/0.7730 | 27.41/0.7297 | 25.41/0.7632 |
| CARN-M [2] | | 412 | 32.5 | 31.92/0.8903 | 28.42/0.7762 | 27.44/0.7304 | 25.63/0.7688 |
| MADNet- L_F [21] | | 1002 | 54.1 | 32.01/0.8925 | 28.45/0.7781 | 27.47/0.7327 | 25.77/0.7751 |
| s -LWSR ₃₂ [23] | | 571 | 32.9 | 32.04/0.893 | 28.15/0.776 | 27.52/0.734 | 25.87/0.779 |
| SFON (ours) | | 582 | 35.5 | 32.04/0.8920 | 28.51/0.7799 | 27.51/0.7345 | 25.93/0.7808 |
| SFON-P (ours) | | 587 | 66.9 | 32.16/0.8935 | 28.56/0.7804 | 27.53/0.7349 | 26.02/0.7828 |

Bold/italics text means the best/second-best performance

Fig. 8 Visual comparisons of our models with other SR methods for $\times 4$ SR on different benchmarks



both of our models can make a better balance between SR accuracy and model complexity. In the future, we wish our models can be applied to the resource-limited devices and real-time scenarios.

Acknowledgements This work was supported in part by the Fundamental Research Funds for the Central Universities (no. 292021000242), in part by National Key R&D Program of China (2017YFB0403604), in

part by the National Natural Science Foundation of China (Grant nos. 61571416, 61072045, 61032006).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

- Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: dataset and study. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp 1122–1131, <https://doi.org/10.1109/CVPRW.2017.150> (2017)
- Ahn, N., Kang, B., Sohn, K.A.: Fast, accurate, and lightweight super-resolution with cascading residual network. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 252–268 (2018)
- Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.: Low complexity single-image super-resolution based on nonnegative neighbor embedding. In: Proceedings of British Machine Vision Conference (BMVC), p 1–10 (2012)
- Chu, X., Zhang, B., Xu, R., Ma, H.: Multi-objective reinforced evolution in mobile neural architecture search. 1901.01074 (2019)
- Chu, X., Zhang, B., Ma, H., Xu, R., Li, Q.: Fast, accurate and lightweight super-resolution with neural architecture search. 1901.07261 (2020)
- Dong, C., Loy, C. C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: European Conference on Computer Vision. Springer, New York, pp 184–199 (2014)
- Dong, C., Loy, C. C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 391–407 (2016)
- He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 770–778. <https://doi.org/10.1109/CVPR.2016.90> (2016)
- He, Z., Cao, Y., Du, L., Xu, B., Zhuang, Y.: Mrfn: multi-receptive-field network for fast and accurate single image super-resolution. IEEE Trans. Multimed. **PP**(99), 1–1 (2019)
- Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E.: Squeeze-and-excitation networks. IEEE Trans. Pattern Anal. Mach. Intell. **42**(8), 2011–2023 (2020)
- Hu, Y., Gao, X., Li, J., Huang, Y., Wang, H.: Single image super-resolution via cascaded multi-scale cross network. 1802.08808 (2021)
- Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K. Q.: Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 2261–2269, (2017) <https://doi.org/10.1109/CVPR.2017.243>
- Huang, J., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 5197–5206, (2015) <https://doi.org/10.1109/CVPR.2015.7299156>
- Hui, Z., Wang, X., Gao, X.: Fast and accurate single image super-resolution via information distillation network. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 723–731 (2018)
- Hui, Z., Gao, X., Yang, Y., Wang, X.: Lightweight image super-resolution with information multi-distillation network. In: Proceedings of the 27th ACM International Conference on Multimedia (MM'19), pp 2024–2032 (2019)
- Kim, J., Lee, J. K., Lee, K. M.: Accurate image super-resolution using very deep convolutional networks. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1646–1654 (2016)
- Kim, J., Lee, J.K., Lee, K.M.: Deeply-recursive convolutional network for image super-resolution. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1637–1645 (2016)
- Kim, J.H., Choi, J.H., Cheon, M., Lee, J.S.: Mamnet: multi-path adaptive modulation network for image super-resolution. Neurocomputing **402**, 38–49 (2020). <https://doi.org/10.1016/j.neucom.2020.03.069>
- Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: Proceedings of the 3rd International Conference on Learning Representations (ICLR), p 1–15 (2015)
- Lai, W., Huang, J., Ahuja, N., Yang, M.: Deep laplacian pyramid networks for fast and accurate super-resolution. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 5835–5843 (2017)
- Lan, R., Sun, L., Liu, Z., Lu, H., Pang, C., Luo, X.: Madnet: a fast and lightweight network for single-image super resolution. IEEE Trans. Cybern. **51**(3), 1443–1453 (2021). <https://doi.org/10.1109/TCYB.2020.2970104>
- Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., Shi, W.: Photo-realistic single image super-resolution using a generative adversarial network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 105–114 (2017)
- Li, B., Wang, B., Liu, J., Qi, Z., Shi, Y.: s-lwsr: Super lightweight super-resolution network. IEEE Trans. Image Process. **29**, 8368–8380 (2020). <https://doi.org/10.1109/TIP.2020.3014953>
- Li, J., Fang, F., Mei, K., Zhang, G.: Multi-scale residual network for image super-resolution. In: European Conference on Computer Vision, pp 527–542 (2018)
- Li, W., Li, S., Liu, A.: Lightweight image super-resolution reconstruction with hierarchical feature-driven network. In: 2020 IEEE International Conference on Image Processing (ICIP), pp 573–577, (2020) <https://doi.org/10.1109/ICIP40778.2020.9191110>
- Lim, B., Son, S., Kim, H., Nah, S., Lee, K. M.: Enhanced deep residual networks for single image super-resolution. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp 1132–1140 (2017)
- Liu, H., Cao, F., Wen, C., Zhang, Q.: Lightweight multi-scale residual networks with attention for image super-resolution. Knowl.-Based Syst. **203**(4), 106103 (2020)
- Liu, J., Tang, J., Wu, G.: Residual feature distillation network for lightweight image super-resolution. In: European Conference on Computer Vision, Springer, pp 41–55 (2020)
- Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, vol 2, pp 416–423 (2001)
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 4510–4520, (2018) <https://doi.org/10.1109/CVPR.2018.00474>
- Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 1874–1883 (2016)
- Sun, L., Liu, Z., Sun, X., Liu, L., Luo, X.: Lightweight image super-resolution via weighted multi-scale residual network. IEEE/CAA J. Autom. Sin. **PP**(99), 1–10 (2021)

33. Tai, Y., Yang, J., Liu, X.: Image super-resolution via deep recursive residual network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 2790–2798 (2017)
34. Tai, Y., Yang, J., Liu, X., Xu, C.: Memnet: A persistent memory network for image restoration. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp 4549–4557, (2017) <https://doi.org/10.1109/ICCV.2017.486>
35. Tian, C., Zhuge, R., Wu, Z., Xu, Y., Zuo, W., Chen, C., Lin, C.W.: Lightweight image super-resolution with enhanced cnn. *Knowl.-Based Syst.* (2020)
36. Tong, T., Li, G., Liu, X., Gao, Q.: Image super-resolution using dense skip connections. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp 4809–4817 (2017)
37. Wenming, Y., Wei, W., Xuechen, Z., Shuifa, S., Qingmin, L.: Lightweight feature fusion network for single image super-resolution. *IEEE Signal Process. Lett.* (2019)
38. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: convolutional block attention module. In: *Computer Vision—ECCV 2018*, pp. 3–19. Springer International Publishing, Cham (2018)
39. Xu, W., Song, H., Zhang, K., Liu, Q., Liu, J.: Learning lightweight multi-scale feedback residual network for single image super-resolution. *Comput. Vis. Image Underst.* **197–198**, 103005 (2020)
40. Yan, C., Hao, Y., Li, L., Yin, J., Liu, A., Mao, Z., Chen, Z., Gao, X.: Task-adaptive attention for image captioning. *IEEE Trans. Circ. Syst. Video Technol.* (2021)
41. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: *Proceedings of the International Conference on Curves and Surfaces*, Springer, pp 711–730 (2010)
42. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: *Proceedings of the European Conference on Computer Vision*, pp 286–301 (2018)
43. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 2472–2481 (2018)
44. Zoph, B., Vasudevan, V., Shlens, J., QV, L.: Learning transferable architectures for scalable image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 8697–8710 (2018)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.