Lightweight Feature De-redundancy and Self-calibration Network for Efficient Image Super-resolution

ZHENGXUE WANG, College of Automation and College of Artificial Intelligence, Nanjing University of Posts and Telecommunications, China
GUANGWEI GAO, Institute of Advanced Technology for Carbon Neutrality, Nanjing University of Posts and Telecommunications, China
JUNCHENG LI, School of Communication and Information Engineering, Shanghai University, China
HUI YAN, School of Computer Science and Engineering, Nanjing University of Science and Technology, China
HAO ZHENG, Key Laboratory of Intelligent Information Processing, Nanjing Xiaozhuang University, China
HUIMIN LU, Department of Mechanical and Control Engineering, Kyushu Institute of Technology, Japan

In recent years, thanks to the inherent powerful feature representation and learning abilities of the convolutional neural network (CNN), deep CNN-steered single image super-resolution approaches have achieved remarkable performance improvements. However, these methods are often accompanied by large consumption of computing and memory resources, which is difficult to be adopted in real-world application scenes. To handle this issue, we design an efficient Feature De-redundancy and Self-calibration Super-resolution network (FDSCSR). In particular, a Feature De-redundancy and Self-calibration Block (FDSCB) is proposed to reduce the repetitive feature information extracted by the model and further enhance the efficiency of the model. Then, based on FDSCB, a Local Feature Fusion Module is presented to elaborately utilize and fuse the feature information extracted by each FDSCB. Abundant experiments on benchmarks have demonstrated that our FDSCSR achieves superior performance with relatively less computational consumption and storage resource than other state-of-the-art approaches. The code is available at https://github.com/IVIPLab/FDSCSR.

This work was supported in part by the National Key Research and Development Program of China under Project Nos. 2018AAA0100102 and 2018AAA0100100; the National Natural Science Foundation of China under Grants No. 61972212, 61772568, 62076139, and 61833011; the Natural Science Foundation of Jiangsu Province under Grant No. BK20190089; the Six Talent Peaks Project in Jiangsu Province under Grant No. RJFW-011; and the Open Fund Project of Key Laboratory of Intelligent Information Processing (Nanjing Xiaozhuang University).

Authors' addresses: Z. Wang, College of Automation and College of Artificial Intelligence, Nanjing University of Posts and Telecommunications, Qixai District, Nanjing City, 210023, China; email: wzx_0826@163.com; G. Gao (corresponding author), Institute of Advanced Technology for Carbon Neutrality, Nanjing University of Posts and Telecommunications, Qixai District, Nanjing City, 210023, China; email: csggao@gmail.com; J. Li, School of Communication and Information Engineering, Shanghai University, Baoshan District, Shanghai City, 999077, China; email: cvjunchengli@gmail.com; H. Yan, School of Computer Science and Engineering, Nanjing University of Science and Technology, Xuanwu District, Nanjing City, 210094, China; email: yanhui@njust.edu.cn; H. Zheng (corresponding author), Key Laboratory of Intelligent Information Processing, Nanjing Xiaozhuang University, Jiangning District, Nanjing City, 211171, China; email: zhh710@163.com; H. Lu, Department of Mechanical and Control Engineering, Kyushu Institute of Technology, Tobata, Kitakyushu City, 804-8550, Japan; email: dr.huimin.lu@ieee.org.

1551-6857/2023/02-ART110 $15.00
https://doi.org/10.1145/3569900

CCS Concepts: • **Computing methodologies** → **Image representations**; **Vision for robotics**; • **Information systems** → **Multimedia content creation**; • **Security and privacy** → *Social aspects of security and privacy;*

Additional Key Words and Phrases: Lightweight image super-resolution, Feature De-redundancy, feature self-calibration

ACM Reference format:
Zhengxue Wang, Guangwei Gao, Juncheng Li, Hui Yan, Hao Zheng, and Huimin Lu. 2023. Lightweight Feature De-redundancy and Self-calibration Network for Efficient Image Super-resolution. *ACM Trans. Multimedia Comput. Commun. Appl.* 19, 3, Article 110 (February 2023), 15 pages.
https://doi.org/10.1145/3569900

1 INTRODUCTION

The main task of **single-image super-resolution (SISR)** aims to restore faithful **high-resolution (HR)** images with rich texture details from degraded **low-resolution (LR)** images [10, 11]. Nevertheless, SISR is still a challenging ill-posed issue, since a particular LR image may be degraded from a batch of HR ones. To solve this issue, many SISR solutions have been presented.

Recently, as deep learning has played an important role in the fields of computer vision and image processing, many super-resolution solutions on account of **convolutional neural networks (CNN)** have been devised [5, 26, 43]. For example, Dong et al. [7] took the lead in applying CNN to the general image super-resolution tasks and presented a **CNN-based image super-resolution (SRCNN)** method that achieved better performance than traditional methods. After that, with the introduction of residual networks [14] and densely connected networks [16], lots of sophisticated deep SISR solutions have been presented. For instance, Kim et al. [21] first presented a very deep single image super-resolution model, which is a 20-layer deep CNN, and used residual learning to accelerate model convergence and improve super-resolution performance. DRCN [22] and DRRN [38] used a recursive strategy to ensure that the network is deepened while reducing the number of parameters of the model. Tai et al. [39] introduced a persistent memory network containing recurrent units and gating units to improve network feature extraction and representation. Lim et al. [30] presented enhanced deep residual networks by further increasing the network depth better use the relationship between channels. Li et al. [25] suggested to use the multi-scale image features to enhance information and proposed a multi-scale residual network.

Although these image super-resolution models have made great progress, since the performance improvement of these models is often accompanied by the increase of model depth and complexity, their parameters and computational overhead are large and difficult to be applied in realistic scenes. Therefore, how to better deal with the balance between model complexity and performance has become a hot research problem [8, 31, 33, 45, 54, 55]. Ahn et al. [2] presented a fast, accurate, and efficient **cascaded residual network (CARN)**, which used a cascade scheme at the global and local levels to aggregate features from different layers. Hui et al. [18, 19] designed an **information distillation network (IDN)** for lightweight image super-resolution and further designed an **information multi-distillation network (IMDN)**. Zhu et al. [53] presented a **compressed back projection network (CBPN)** by cascading up- and down-sampling layers to simultaneously generate features in both low- and high-resolution spaces. As the width and depth of the network increase, many features containing repeated information will be extracted and used [13, 32], which will increase the computational consumption of the network and may affect the SR performance to some extent. To address this problem and reduce the complexity of the network model, we design an efficient yet effective **Feature De-redundancy and Self-Calibration Block (FDSCB)**. Based on this carefully designed block, we devise a **Local Feature Fusion Module (LFFM)**, which is used to utilize and fuse the features distilled by the proposed FDSCB as much as possible. Finally, based on the LFFM module, a **Feature De-redundancy and Self-Calibration Super-Resolution network (FDSCSR)** is proposed.

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

- We devise a FDSCB, which can reduce the extracted duplicate feature information and calibrate the network, thus reducing the computational resource consumption of the model.
- We propose an LFFM to fully extract and fuse the features extracted by FDSCB. Based on LFFM, we further present a FDSCSR for efficient SISR.
- We explore reducing the repetitive feature information distilled by the network to reduce the amount of computation, re-weight and calibrate the network to improve its feature extraction capability. Our model can achieve an excellent balance between performance and computational consumption.

2 RELATED WORKS

As a pioneer work in SISR tasks, SRCNN [7] achieved competitive performance to traditional SR methods with a three layers CNN network. Since then, deep CNN steered SR models have been continuously designed and obtained excellent performance [4, 20, 25, 44, 46, 48, 49]. Zhang et al. [52] introduced the channel attention mechanism into residual connections to make the network mainly focus on high-frequency information. Hu et al. [15] presented a compression and excitation network, which has achieved great progress in many computer vision tasks. To explore the relationship between different statistical features, Dai et al. [6] propose a second-order channel attention mechanism by using second-order feature statistics to adaptively re-scale channel features. Guo et al. [12] presented a closed-loop dual regression network, which introduced an additional constraint to limit the mapping space between high- and low-resolution images. To exploit the cross-scale feature correlation of images, Mei et al. [36] presented a cross-scale non-local attention network to combine cross-scale non-local priors and intra-scale non-local priors.

Although these solutions mentioned above have achieved remarkable performance, they are difficult to be adopted in real-world applications, since these methods are accompanied by large computational overhead and parameter amount. To handle the above problems, Hui et al. [18, 19] devised IMDN. Li et al. [28] proposed a **linearly assembled pixel adaptive regression scheme (LAPAR)**, which transformed the learning of the mapping from LR space to HR space into a linear coefficient regression problem. CFSRCNN [40] cascades several types of feature extraction modules, which are utilized to learn and fuse features from different paths. Zhu et al. [54] proposed an **expectation-maximizing attention mechanism (EMASRN)** to better equilibrate the number of parameters, computational power, and performance. The above methods have achieved good performance, but there are still many problems, including high model complexity and difficulty in recovering texture details. Wang et al. [41] presented a lightweight **adaptive weight network (AWSRN)** by constructing an adaptive weight residual unit and an adaptive weight multi-scale module. Zhang et al. [50] presented a **global and local adjustment network (GLADSR)** to enhance the network capacity. Li et al. [29] presented an efficient **densely connected distillation network (DCDN)** by combining feature fusion units and densely connected distillation blocks containing selective cascades and dense distillation components. As a complementary solution to CNN, some Transformer-based SISR models have been designed. For example, Gao et al. [9] designed a lightweight bimodal network via symmetric CNN and recursive Transformer. To dynamically adjust the neurons' response to the context, Li et al. [27] proposed a lightweight cross-receptive focused inference network, which is a hybrid outcome of CNNs and Transformers.

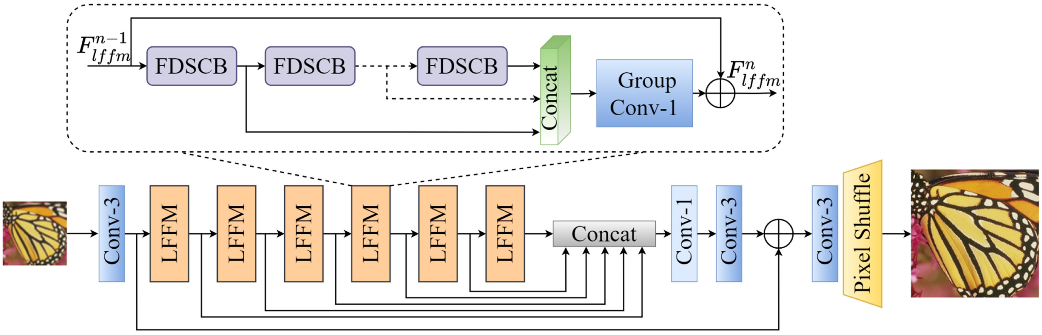

Fig. 1. The pipeline of FDSCSR. Among them, the dashed box is the LFFM. ⊕ denotes the element-wise summation operation.

3 FEATURE DE-REDUNDANCY AND SELF-CALIBRATION NETWORK

This section mainly introduces the proposed network architecture, the FDSCB and the LFFM in detail.

3.1 Network Architecture

In this part, we introduce the backbone of our proposed FDSCSR. As depicted in Figure 1, the entire FDSCSR mainly consists of three parts, including the shallow feature extraction part, the deep feature extraction and fusion part by stacking multiple LFFM, and the final reconstruction part.

In this work, I_{LR} and I_{HR} stand for the LR input and the related HR counterpart, respectively. I_{SR} represents the reconstructed image. In particular, an ordinary 3×3 convolutional layer is utilized to distill superficial features on the primitive observed LR image. This process can be formulated as

$$F_{sf} = H_{sf}(I_{LR}). \tag{1}$$

Among them, $H_{sf}(\cdot)$ is the shallow feature extraction operation, F_{sf} is the extracted rough shallow features, which will be fed to the deep feature extraction portion for further feature extraction.

The deep feature extraction and fusion portion mainly consists of several stacked LFFMs, a 1×1 convolutional layer, and an ordinary 3×3 convolutional layer. The output of each LFFM will be sent to the end of the network for cascading to make full use of the feature information extracted by each LFFM. The 1×1 convolution operation is mainly applied for feature aggregation and channel reduction. After that, an ordinary 3×3 convolutional layer is applied to smooth the aggregated features. This part can be represented as

$$F_1 = H^1_{lffm}(F_{sf}), \tag{2}$$

$$F_k = H^k_{lffm}(F_{k-1}), k = 2, \ldots, m, \tag{3}$$

$$F_{df} = H_{3\times3}(H_{1\times1}(H_{cat}(F_1, \ldots, F_m))), \tag{4}$$

where F_k represents the feature map extracted by the kth LFFM and $H^k_{lffm}(\cdot)$ is the kth LFFM. $H_{3\times3}(\cdot)$ and $H_{1\times1}(\cdot)$ denote 3×3 convolutional layer and 1×1 convolutional layer, respectively; $H_{cat}(\cdot)$ represents the concatenated operation in the channel dimension; and F_{df} is the final feature extracted by the deep feature extraction and fusion part.

The reconstruction section mainly consists of an ordinary 3×3 convolutional layer together with a pixel-shuffle operation [37]. The 3×3 convolutional layer is mainly applied to change the

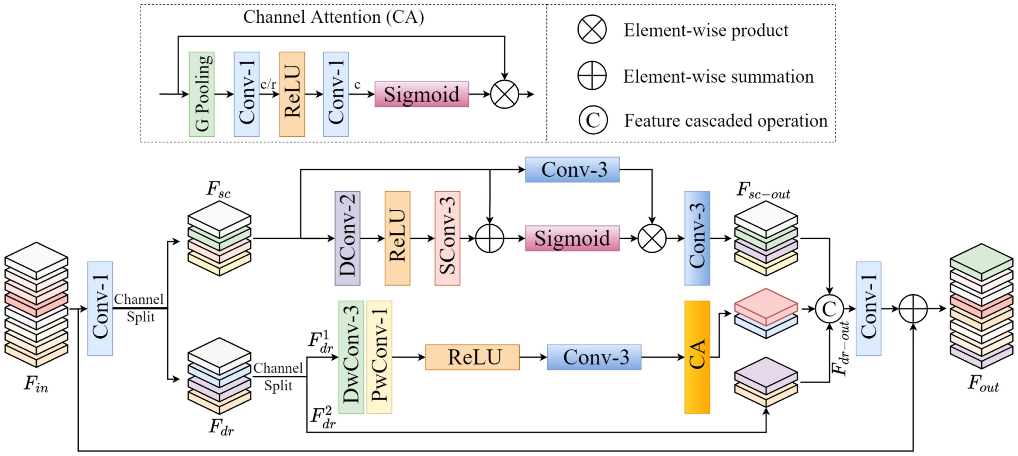

Fig. 2. The pipeline of FDSCB.

number of feature channels into a multiple of s^2 (s is the upsampling factor). Then the pixel-shuffle layer performs pixel-shuffling of the feature maps of size $H \times W \times s^2 \cdot c$ into size $sH \times sW \cdot c$. The episodic feature map of the reconstruction part is the residual sum of shallow features and deep features. This part can be represented as

$$I_{SR} = H_{re}(F_{sf} + F_{df}) = H_{ps}(H_{3\times3}(F_{sf} + F_{df})), \tag{5}$$

where $H_{3\times3}(\cdot)$ denotes 3×3 convolutional layer, $H_{re}(\cdot)$ is the reconstruction function, and $H_{ps}(\cdot)$ stands for the pixel-shuffle layer.

Given a set of training images $\{I_{LR}^j, I_{HR}^j\}_{j=1}^N$, where N represents the number of training image pairs. Therefore, the loss function of FDSCSR can be formulated as

$$L(\Theta) = \frac{1}{N}\sum_{j=1}^{N}\left\|H_{FDSCSR}\left(I_{LR}^j\right) - I_{HR}^j\right\|_1, \tag{6}$$

where $H_{FDSCSR}(\cdot)$ represents the proposed network, Θ denotes the learnable parameter sets in FDSCSR, and $||\cdot||_1$ is the l_1 norm.

3.2 Feature De-Redundancy and Self-Calibration Block

To reduce the redundant information and fully distill useful features, we propose a FDSCB, which is shown in Figure 2. Its input first passes through an ordinary 1×1 convolutional layer for initial feature extraction. Then the channel splitting operation is used to divide the features extracted by the convolutional layer into two parts. The first part serves as the input of the self-calibration module and the second part serves as the input of the de-redundancy module

$$F_{sc}, F_{dr} = H_{cs}(H_{1\times1}(F_{in})), \tag{7}$$

where $H_{1\times1}(\cdot)$ denotes 1×1 convolutional layer; $H_{cs}(\cdot)$ is the channel splitting operation; F_{sc} and F_{dr} represent the input features of the **self-calibration (SC)** module and the **de-redundancy (DR)** module, respectively; and F_{in} represents the input of the FDSCB.

The DR part of FDSCB is mainly composed of a depthwise separable convolutional layer, ordinary 3×3 convolutional layers, and **channel attention (CA)** modules. Compared with ordinary convolution, depthwise separable convolution can significantly reduce the number of parameters and computation cost. The depthwise separable convolution consists of a 3 × 3 channelwise convolution together with a 1 × 1 pointwise convolution. This part further uses the channel splitting operation to split the input feature F_{dr} to two parts: F^1_{dr} and F^2_{dr}. The de-redundant part can be expressed in detail as

$$F^1_{dr}, F^2_{dr} = H_{cs}(F_{dr}), \tag{8}$$

$$F_{dr-out} = H_{cat}\left(F^2_{dr}, H_{ca}\left(H_{3\times3}\left(\sigma\left(H_{pw}\left(H_{dw}\left(F^1_{dr}\right)\right)\right)\right)\right)\right), \tag{9}$$

where $H_{3\times3}(\cdot)$ denotes an ordinary 3 × 3 convolutional layer, $H_{cs}(\cdot)$ represents the channel splitting operation operation, $H_{cat}(\cdot)$ represents the concatenated operation, $H_{dw}(\cdot)$ and $H_{pw}(\cdot)$ represent the channelwise convolution and pointwise convolution operations, σ indicates the ReLU activation operation, H_{ca} is the channel attention module, and F_{dr-out} represents the output of the de-redundant part.

The SC part is shown in the upper in Figure 2, which is mainly composed of a 2 × 2 transposed convolutional layer; a 3 × 3 strided convolutional layer (with stride size 2); ordinary 3 × 3 convolutional layers; and the Sigmoid operation. To enlarge the receptive field of the network, the input feature maps are up-sampled and down-sampled by a general factor of 2 by the transposed convolutional layer and stride convolutional layer, respectively. Then the weights of the extracted features are calculated by Sigmoid. The features extracted by the 3 × 3 convolutional layer are re-weighted and calibrated. Finally, the re-weighted features are fused through a 3 × 3 convolutional layer. The self-calibration part can be represented as

$$F_{sc-out} = H_{3\times3}(H_{3\times3}(F_{sc}) * H_{sig}(F_{sc} + H_{scon}(\sigma(H_{dcon}(F_{sc}))))), \tag{10}$$

where $H_{3\times3}(\cdot)$ denotes 3 × 3 convolutional layer, $H_{sig}(\cdot)$ represents the Sigmoid layer, and $H_{scon}(\cdot)$ and $H_{dcon}(\cdot)$ represent the stride convolution with stride 2 and transposed convolution with kernel size 2, respectively. F_{sc-out} indicates the output of the self-calibration section.

The output features of the de-redundant part and the self-calibration part will be concatenated and fused through a convolutional layer. Finally, we perform residual connection between the input and the fused features to obtain the output feature maps of the FDSCB. This process can be expressed as

$$F_{out} = F_{in} + H_{1\times1}(H_{cat}(F_{dr-out}, F_{sc-out})), \tag{11}$$

where $H_{cat}(\cdot)$ represents the concatenated operation, $H_{1\times1}(\cdot)$ denotes 1 × 1 convolutional layer, and F_{out} represents the output of the FDSCB.

3.3 Local Feature Fusion Module

To fully extract and utilize the feature information extracted by the FDSCB, we design an LFFM that aggregates the output features of each FDSCB efficiently. This module can make full use of the feature information extracted by each FDSCB to distill more useful information for reconstruction. As given in Figure 1, the LFFM mainly consists of k FDSCBs (in this article, the value of k is 6) and a 1 × 1 group convolution. Among them, group convolution is mainly used for feature channel dimension reduction to reduce model computation. The module can be represented as

$$F^1_{fdb} = H^1_{fdb}\left(F^{n-1}_{lffm}\right), \tag{12}$$

$$F^k_{fdb} = H^k_{fdb}\left(F^{k-1}_{fdb}\right), k = 2, 3, \ldots, m, \tag{13}$$

$$F^n_{lffm} = F^{n-1}_{lffm} + H_{gconv}\left(H_{cat}\left(F^1_{fdb}, F^2_{fdb}, \ldots, F^m_{fdb}\right)\right), \tag{14}$$

where F^k_{fdb} indicates the output of the kth FDSCB, F^{n-1}_{lffm} and F^n_{lffm} stand for the import and export features of the nth LFFM, respectively, $H^k_{fdb}(\cdot)$ is the kth FDSCB, $H_{cat}(\cdot)$ represents the concatenated operation, and $H_{gconv}(\cdot)$ means the 1×1 group convolutional layer to perform dimensionality reduction.

4 EXPERIMENTS AND ANALYSES

4.1 Datasets and Metrics

In this experiment, we choose the DIV2K [1] dataset as the training set for our model, which includes 800 training images and 100 validation images. It is widely used for image super-resolution tasks. For testing, we apply five benchmark datasets: Set5 [3], Set14 [47], BSDS100 [34], Urban100 [17], and Manga109 [35]. Meanwhile, to evaluate and validate the effectiveness of SR models, **peak signal-to-noise ratio (PSNR)** and flexible **structural similarity (SSIM)** [35] indexes are chosen as the evaluation criteria. In addition, like many previous methods [2, 18, 19, 30, 52], all the evaluation indicators are calculated on the Y channel embedded in the related YCbCr space.

4.2 Experimental Details

During training, we use bicubic interpolation to downsample the HR images at different scales to obtain corresponding LR ones. Furthermore, we rotate and horizontally flip the samples at random to augment the training dataset. Image patches with 48×48 pixels are randomly acquired from the training dataset as input to the network. We elect Adam as the optimizer to train our SR model, and its parameter is set as $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-8}$. The initial learning rate is set as 2×10^{-4}, and then it is finally decayed to 6.25×10^{-6} in the form of cosine annealing. Two models, FDSCSR and FDSCSR-S, are reported. FDSCSR applies the channel number of 48, while FDSCSR-S adopts a smaller channel number of 36. The presented SR network is conducted using the Pytorch platform with a single NVIDIA 2080Ti GPU.

4.3 Model Complexity Studies

To assess the effectiveness of our presented FDSCSR, this section analyzes the complexity of the FDSCSR model. Figure 3 depicts the comparison of the parameters and PSNR values on Set5 (×4). We can see that our proposed FDSCSR and FDSCSR-S can better balance the parameters and performance of the model. Compared with methods such as AWSRN-M [41], CBPN [53], and CARN [2], which have a huge number of parameters, FDSCSR achieves better performance. Furthermore, our lightweight FDSCSR-S model achieves far fewer parameters than methods such as IMDN [18], LAPAR-A [28], and MADNet [24] but gets better performance than these methods.

In addition, to better assess the effectiveness of our proposed model, Figure 3 also shows the comparison of Multi-Adds and PSNR values on Set5 (×4). We can observe that FDSCSR not only achieves better performance than methods such as GLADSR [24] and MADNet [24] with similar Multi-Adds but also achieves better performance than methods such as SMSR [42], LAPAR-A [28], and CARN [2] with larger Multi-Adds. All the above results illustrate that FDSCSR is an efficient model that can well balance model complexity and performance.

Meanwhile, in Figure 5, we also provide the inference speed comparisons between our FDSCSR with some representative lightweight SISR methods. Fortunately, our FDSCSR achieves competitive results with comparable parameters and acceptable execution time.

4.4 Ablation Study

4.4.1 Verification of the Basic Modules in FDSCB. This part is used to assess the effectiveness of the respective component in our designed FDSCB. Table 1 shows the experimental results of each scheme on the Urban100 dataset (×4). The first row in the table is the experimental result of

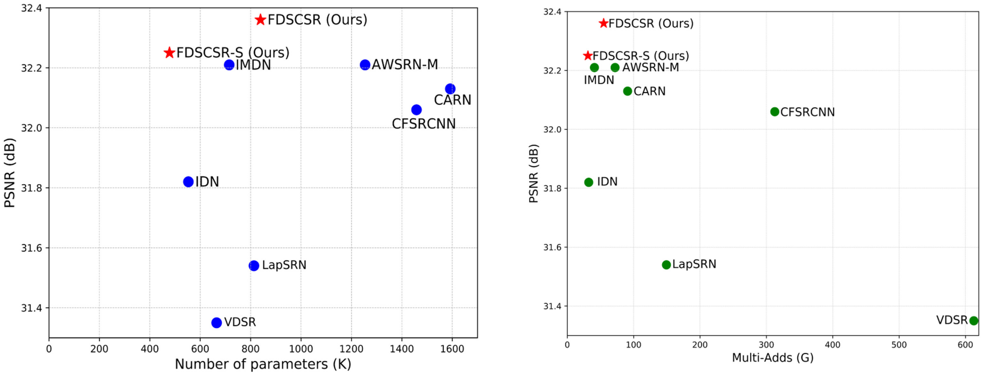

Fig. 3. PSNR vs. Parameters (left) and PSNR vs. Multi-Adds (right) on Set5 dataset (×4).

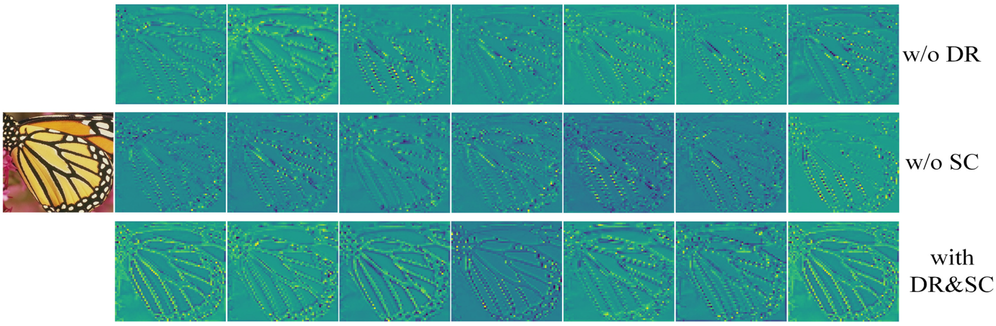

Fig. 4. Visual feature maps of basic modules in FDSCB. It should be noted that these features are derived from the input of the reconstruction part to verify the effectiveness of the DR module and the SC module for reconstructing texture details.

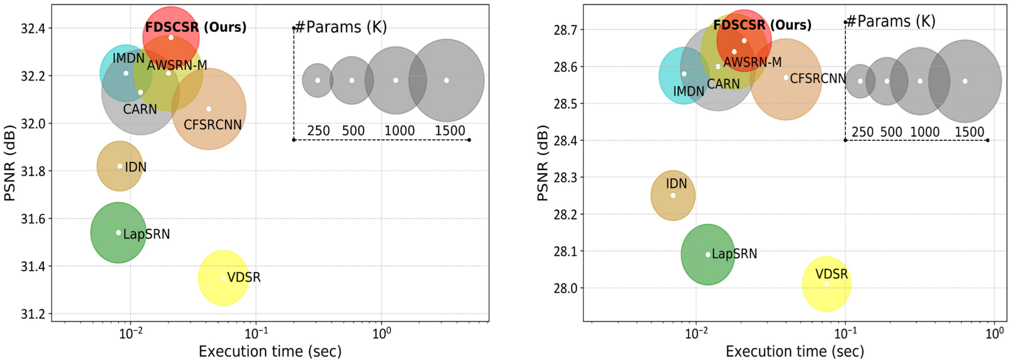

Fig. 5. Inference speed study on Set5 (left) and Set14 (right) (×4).

using ordinary 3 × 3 convolutional layer to replace the feature DR in FDSCB, and the second row is the experiments of using ordinary 3 × 3 convolutional layer to replace the feature SC module in FDSCB, and the last row is the experimental results with both DR and SC modules. It can be seen from the observation between the last row and the first row in the table that the feature DR module has fewer parameters and calculations, and its PSNR value is much higher than that of the ordinary convolutional layer. In addition, the comparison between the last and the second rows

Table 1. Verification of Basic Modules in FDSCB on Urban100 (×4)

Methods	Params	Mult-Adds	PSNR	SSIM
w/o DR	549K	35.3G	25.97	0.7830
w/o SC	401K	22.7G	25.63	0.7708
with DR & SC	478K	31.1G	**26.12**	**0.7866**

The best results are highlighted in bold.

Table 2. Effectiveness of Basic Part in Feature DR Module on Manga109 (×4)

Scale	Conv-3	DPConv	CA	Params	Multi-Adds	PSNR	SSIM
×4	✓	✕	✕	497K	32.304614G	30.37	0.9072
×4	✕	✓	✕	476K	31.128883G	30.37	0.9069
×4	✕	✓	✓	478K	31.128885G	**30.51**	**0.9087**

The best results are highlighted in bold.

shows that although the parameters of an ordinary convolutional layer are much smaller than that of the feature SC module, the PSNR value of the feature SC part is 0.49 dB higher than that of the ordinary convolutional layer. In Figure 4, we further visualize the feature maps of different modules in FDSCB, from which we can see that with both DR and SC modules, the extracted features contain abundant textural details for better reconstruction. These experiments further illustrate the effectiveness of each part in our presented FDSCB.

4.4.2 Effectiveness of Basic Modules in Feature DR Network. This part is used to verify the effectiveness of the **depthwise separable convolution (DPConv)** and CA mechanisms of the feature DR in FDSCB. Table 2 provides the experimental results on the Manga109 (×4) test set. Ordinary 3 × 3 **convolutional (Conv-3)** and DPConv are alternatively verified. It can be observed from the first two rows in the table that the parameters and computation consumption of DPConv is smaller than that of the ordinary Conv-3. Their PSNR values are nearly the same, only the value of SSIM is reduced by 0.0003. We can conclude that the DPConv can reduce the model size without significant performance penalty. In addition, by comparing the last two rows in the table we can observe that the CA mechanism only brings a little increase of parameters, while the PSNR value is increased from 30.37 to 30.51 dB. This fully demonstrates the rationality and effectiveness of using DPConv and CA in DR.

4.5 Comparison with Some States of the Art

To verify the effectiveness of our proposed FDSCSR, we compare it with some representative lightweight SR methods, including DRCN [22], LapSRN [23], DRRN [38], IDN [19], CARN [2], CBPN [53], IMDN [18], AWSRN [41], MADNet [24], RFDN [31], GLADSR [50], DCDN [29], SMSR [42], LAPAR [28], ECBSR [51], and EMASRN [54]. Table 3 reports the quantitative comparisons of our proposed FDSCSR with other competitive approaches on five benchmark test sets. As can be observed from the table that our presented FDSCSR-S solution achieves competitive performance than other methods with similar number of parameters, such as CBPN-S [53], IMDN [18], ECBSR [51], RFDN [31], and LAPAR-A [28]. Furthermore, FDSCSR can achieves better super-resolution performance than those methods with a larger number of parameters and computations, such as CBPN [53], AWSRN-M [41], MADNet [24], and SMSR [42], and so on. These results fully demonstrate the effectiveness of the proposed FDSCSR.

In addition, to more intuitively show the effectiveness of our proposed FDSCSR model, Figures 6, 7, and 8 presents several visual comparisons among FDSCSR, FDSCSR-S, and several

Table 3. Average Quantitative Comparisons on Benchmark Datasets

Methods	Scale	Params	Multi-Adds	Set5	Set14	BSD100	Urban100	Manga109
				PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
DRCN [22]	×2	1,774K	17974.3G	37.63/0.9588	33.04/0.9118	31.85/0.8942	30.75/0.9133	37.55/0.9732
LapSRN [23]		813K	29.9G	37.52/0.9590	33.08/0.9130	31.80/0.8950	30.41/0.9100	37.27/0.9740
DRRN [38]		297K	6796.9G	37.74/0.9591	33.23/0.9136	32.05/0.8973	31.23/0.9188	37.88/0.9749
IDN [19]		553K	124.6G	37.83/0.9600	33.30/0.9148	32.08/0.8985	31.27/0.9196	38.01/0.9749
CARN [2]		1,592K	222.8G	37.76/0.9590	33.52/0.9166	32.09/0.8978	31.92/0.9256	38.36/0.9765
CBPN [53]		1,036K	240.7G	37.90/0.9590	33.60/0.9171	32.17/0.8989	32.14/0.9279	—
CBPN-S [53]		430K	101.5G	37.69/0.9583	33.36/0.9147	32.02/0.8972	31.55/0.9217	—
IMDN [18]		694K	158.8G	38.00/0.9605	33.63/0.9177	32.19/0.8996	32.17/0.9283	38.88/0.9774
AWSRN-M [41]		1,063K	244.1G	38.04/0.9605	33.66/0.9181	32.21/0.9000	32.23/0.9294	38.66/0.9772
MADNet [24]		878K	187.1G	37.85/0.9600	33.38/0.9161	32.04/0.8979	31.62/0.9233	—
RFDN [31]		534K	123.0G	38.05/0.9606	33.68/0.9184	32.16/0.8994	32.12/0.9278	38.88/0.9773
GLADSR [50]		812K	187.2G	37.99/0.9608	33.63/0.9179	32.16/0.8996	32.16/0.9283	—
DCDN [29]		756K	—	38.01/0.9606	33.52/0.9166	32.17/0.8996	32.16/0.9283	38.70/0.9773
SMSR [42]		985K	351.5G	38.00/0.9601	33.64/0.9179	32.17/0.8990	32.19/0.9284	38.76/0.9771
LAPAR-A [28]		548K	171.0G	38.01/0.9605	33.62/0.9183	32.19/0.8999	32.10/0.9283	38.67/0.9772
ECBSR [51]		596K	137.3G	37.90/0.9615	33.34/0.9178	32.10/0.9018	31.71/0.9250	—
FDSCSR-S (Ours)		466K	121.8G	38.02/0.9606	33.51/0.9174	32.18/0.8996	32.24/0.9288	38.67/0.9771
FDSCSR (Ours)		823K	215.8G	38.12/0.9609	33.69/0.9191	32.24/0.9004	32.50/0.9315	38.89/0.9775
DRCN [22]	×3	1,774K	17974.3G	33.82/0.9226	29.76/0.8311	28.80/0.7963	27.15/0.8276	32.31/0.9328
DRRN [38]		297K	6796.9G	34.03/0.9244	29.96/0.8349	28.95/0.8004	27.53/0.8378	32.74/0.9390
IDN [19]		553K	56.3G	34.11/0.9253	29.99/0.8354	28.95/0.8013	27.42/0.8359	32.71/0.9381
CARN [2]		1,592K	118.8G	34.29/0.9255	30.29/0.8407	29.06/0.8034	28.06/0.8493	33.43/0.9427
IMDN [18]		703K	71.5G	34.36/0.9270	30.32/0.8417	29.09/0.8046	28.17/0.8519	33.61/0.9445
AWSRN-M [41]		1,143K	116.6G	34.42/0.9275	30.32/0.8419	29.13/0.8059	28.26/0.8545	33.64/0.9450
MADNet [24]		930K	88.4G	34.16/0.9253	30.21/0.8398	28.98/0.8023	27.77/0.8439	—
RFDN [31]		541K	55.4G	34.41/0.9273	30.34/0.8420	29.09/0.8050	28.21/0.8525	33.67/0.9449
GLADSR [50]		821K	88.2G	34.41/0.9272	30.37/0.8418	29.08/0.8050	28.24/0.8537	—
DCDN [29]		765K	—	34.41/0.9273	30.31/0.8417	29.08/0.8045	28.17/0.8520	33.54/0.9441
SMSR [42]		993K	156.8G	34.40/0.9270	30.33/0.8412	29.10/0.8050	28.25/0.8536	33.68/0.9445
LAPAR-A [28]		594K	114.0G	34.36/0.9267	30.34/0.8421	29.11/0.8054	28.15/0.8523	33.51/0.9441
EMASRN [54]		427K	—	34.36/0.9264	30.30/0.8411	29.05/0.8035	28.04/0.8493	33.43/0.9433
FDSCSR-S (Ours)		471K	54.6G	34.42/0.9274	30.37/0.8429	29.10/0.8052	28.20/0.8532	33.55/0.9443
FDSCSR (Ours)		830K	96.4G	34.50/0.9281	30.43/0.8442	29.15/0.8068	28.40/0.8576	33.78/0.9460
DRCN [22]	×4	1,774K	17974.3G	31.53/0.8854	28.02/0.7670	27.23/0.7233	25.14/0.7510	28.98/0.8816
LapSRN [23]		813K	149.4G	31.54/0.8850	28.19/0.7720	27.32/0.7280	25.21/0.7560	29.09/0.8900
DRRN [38]		297K	6796.9G	31.68/0.8888	28.21/0.7720	27.38/0.7284	25.44/0.7638	29.46/0.8960
IDN [19]		553K	32.3G	31.82/0.8903	28.25/0.7730	27.41/0.7297	25.41/0.7632	29.41/0.8942
CARN [2]		1,592K	90.9G	32.13/0.8937	28.60/0.7806	27.58/0.7349	26.07/0.7837	30.42/0.9070
CBPN [53]		1,197K	97.9G	32.21/0.8944	28.63/0.7813	27.58/0.7356	26.14/0.7869	—
CBPN-S [53]		592K	63.1G	31.93/0.8908	28.50/0.7785	27.50/0.7324	25.85/0.7772	—
IMDN [18]		715K	40.9G	32.21/0.8948	28.58/0.7811	27.56/0.7353	26.04/0.7838	30.45/0.9075
AWSRN-M [41]		1,254K	72.0G	32.21/0.8954	28.65/0.7832	27.60/0.7368	26.15/0.7884	30.56/0.9093
MADNet [24]		1,002K	54.1G	31.95/0.8917	28.44/0.7780	27.47/0.7327	25.76/0.7746	—
RFDN [31]		550K	31.6G	32.24/0.8952	28.61/0.7819	27.57/0.7360	26.11/0.7858	30.58/0.9089
GLADSR [50]		826K	52.6G	32.14/0.8940	28.62/0.7813	27.59/0.7361	26.12/0.7851	—
DCDN [29]		777K	—	32.21/0.8949	28.57/0.7807	27.55/0.7356	26.09/0.7855	30.41/0.9072
SMSR [42]		1,006K	89.1G	32.12/0.8932	28.55/0.7808	27.55/0.7351	26.11/0.7868	30.54/0.9085
LAPAR-A [28]		659K	94.0G	32.15/0.8944	28.61/0.7818	27.61/0.7366	26.14/0.7871	30.42/0.9074
ECBSR [51]		603K	34.7G	31.92/0.8946	28.34/0.7817	27.48/0.7393	25.81/0.7773	—
EMASRN [54]		546K	—	32.17/0.8948	28.57/0.7809	27.55/0.7351	26.01/0.7838	30.41/0.9076
FDSCSR-S (Ours)		478K	31.1G	32.25/0.8959	28.61/0.7821	27.58/0.7367	26.12/0.7866	30.51/0.9087
FDSCSR (Ours)		839K	54.8G	32.36/0.8970	28.67/0.7840	27.63/0.7384	26.33/0.7935	30.69/0.9113

The best and the second best results are highlight in red and blue, respectively.

representative lightweight SR methods. As can be evaluated from the figures, our devised FDSCSR and FDSCSR-S not only have higher PSNR and SSIM values but also recover rich texture details with better visual effects. The recovered images are more close to the ground truth. On the contrary, most of the compared methods cannot recover the textural details (such as edges) well, even

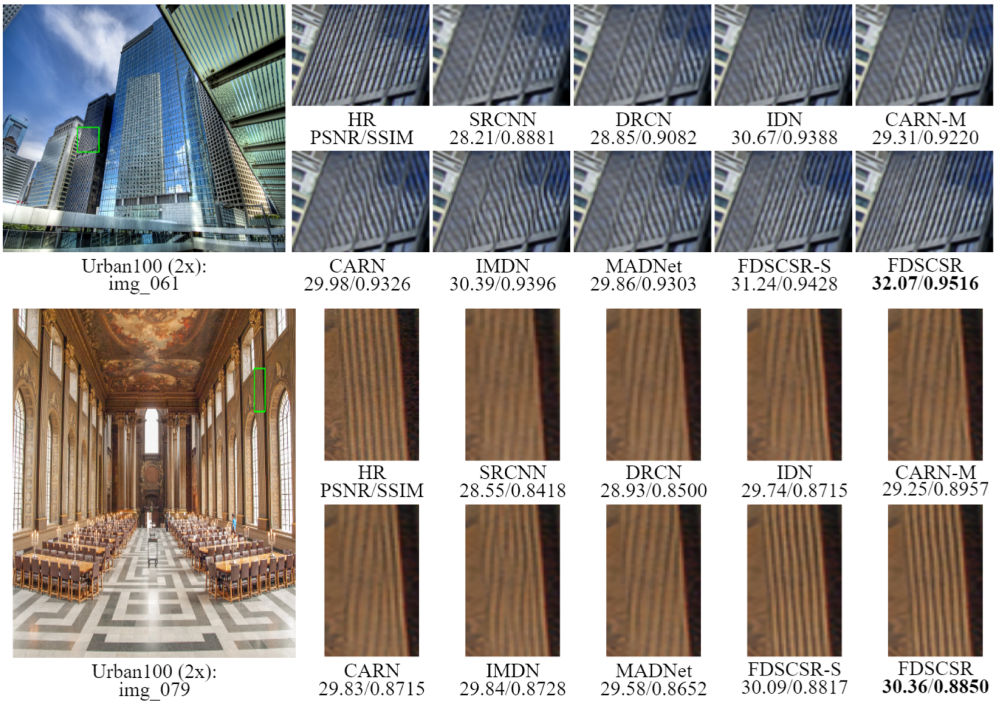

Fig. 6. Visual comparison of FDSCSR, FDSCSR-S and, other lightweight SR methods for scale factor ×2.

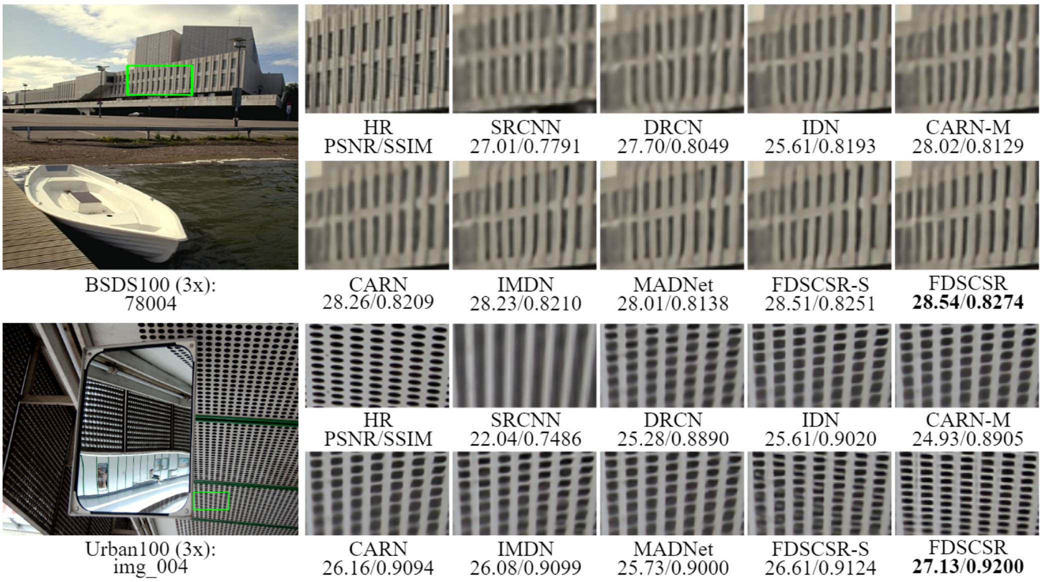

Fig. 7. Visual comparison of FDSCSR, FDSCSR-S, and other lightweight SR methods for scale factor ×3.

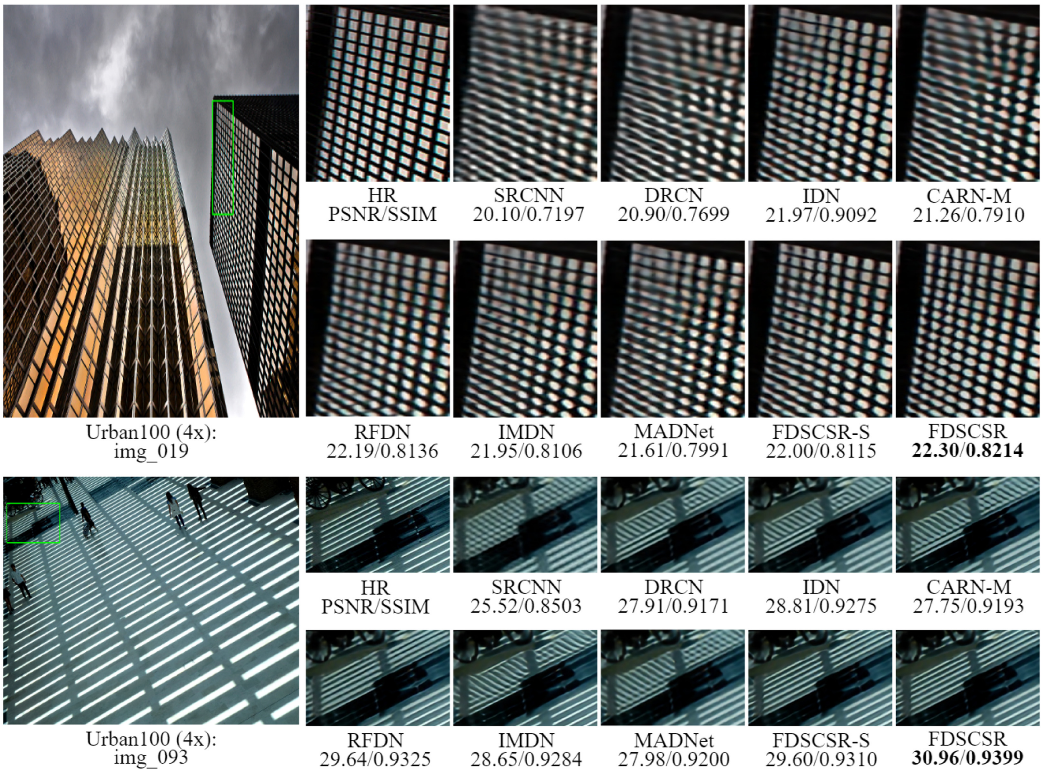

Fig. 8. Visual comparison of FDSCSR, FDSCSR-S, and other lightweight SR methods for scale factor ×4.

though they possess a larger number of parameters and computations. This further validates the effectiveness and excellence of the proposed FDSCSR.

5 CONCLUSIONS

In this article, for the efficient image super-resolution task, we devised a FDSCSR. Specifically, we proposed a lightweight and effective FDSCB as the basic unit of the overall network to reduce the repeated feature information and fully calibrate the features. Furthermore, an LFFM is designed to fully utilize and fuse the feature information extracted by each FDSCB. Extensive experiments have validated that our presented FDSCSR model can achieve comparable performance and can better balance model complexity and performance.

ACKNOWLEDGMENTS

The authors thanks the editor and the anonymous reviewers for their critical and constructive comments and suggestions.

REFERENCES

[1] Eirikur Agustsson and Radu Timofte. 2017. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'17)*. 126–135.

[2] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. 2018. Fast, accurate, and lightweight super-resolution with cascading residual network. In *Proceedings of the European Conference on Computer Vision (ECCV'18)*. 252–268.

[3] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. 2012. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *Proceedings of the British Machine Vision Conference (BMVC'12)*. 135.1–135.10.

[4] Honggang Chen, Xiaohai He, Linbo Qing, Yuanyuan Wu, Chao Ren, Ray E. Sheriff, and Ce Zhu. 2022. Real-world single image super-resolution: A brief review. *Inf. Fusion* 79 (2022), 124–145.

[5] Xiangyu Chen, Xintao Wang, Jiantao Zhou, and Chao Dong. 2022. Activating more pixels in image super-resolution transformer. arXiv:2205.04437. Retrieved from https://arxiv.org/abs/2005.04437.

[6] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. 2019. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'19)*. 11065–11074.

[7] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2014. Learning a deep convolutional network for image super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV'14)*. 184–199.

[8] Guangwei Gao, Wenjie Li, Juncheng Li, Fei Wu, Huimin Lu, and Yi Yu. 2022. Feature distillation interaction weighting network for lightweight image super-resolution. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI'22)*. 661–669.

[9] Guangwei Gao, Zhengxue Wang, Juncheng Li, Wenjie Li, Yi Yu, and Tieyong Zeng. 2022. Lightweight bimodal network for single-image super-resolution via symmetric CNN and recursive transformer. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'22)*. 913–919.

[10] Guangwei Gao, Yi Yu, Huimin Lu, Jian Yang, and Dong Yue. 2022. Context-patch representation learning with adaptive neighbor embedding for robust face image super-resolution. *IEEE Trans Multimedia* (2022).

[11] Guangwei Gao, Yi Yu, Jin Xie, Jian Yang, Meng Yang, and Jian Zhang. 2021. Constructing multilayer locality-constrained matrix regression framework for noise robust face super-resolution. *Pattern Recogn.* 110 (2021), 107539.

[12] Yong Guo, Jian Chen, Jingdong Wang, Qi Chen, Jiezhang Cao, Zeshuai Deng, Yanwu Xu, and Mingkui Tan. 2020. Closed-loop matters: Dual regression networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'20)*. 5407–5416.

[13] Kai Han, Yunhe Wang, Qi Tian, Jianyuan Guo, Chunjing Xu, and Chang Xu. 2020. Ghostnet: More features from cheap operations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'20)*. 1580–1589.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*. 770–778.

[15] Jie Hu, Li Shen, and Gang Sun. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18)*. 7132–7141.

[16] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17)*. 4700–4708.

[17] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. 2015. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15)*. 5197–5206.

[18] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. 2019. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the ACM International Conference on Multimedia (ACMMM'19)*. 2024–2032.

[19] Zheng Hui, Xiumei Wang, and Xinbo Gao. 2018. Fast and accurate single image super-resolution via information distillation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18)*. 723–731.

[20] Kui Jiang, Zhongyuan Wang, Peng Yi, and Junjun Jiang. 2020. Hierarchical dense recursive network for image super-resolution. *Pattern Recogn.* 107 (2020), 107475.

[21] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2016. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*. 1646–1654.

[22] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2016. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*. 1637–1645.

[23] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. 2018. Fast and accurate image super-resolution with deep laplacian pyramid networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 41, 11 (2018), 2599–2613.

[24] Rushi Lan, Long Sun, Zhenbing Liu, Huimin Lu, Cheng Pang, and Xiaonan Luo. 2020. Madnet: A fast and lightweight network for single-image super resolution. *IEEE Trans. Cybernet.* 51, 3 (2020), 1443–1453.

[25] Juncheng Li, Faming Fang, Kangfu Mei, and Guixu Zhang. 2018. Multi-scale residual network for image super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV'18)*. 517–532.

[26] Juncheng Li, Zehua Pei, and Tieyong Zeng. 2021. From beginner to master: A survey for deep learning-based single-image super-resolution. arXiv:2109.14335. Retrieved from https://arixv.org/abs/2109.14335.

[27] Wenjie Li, Juncheng Li, Guangwei Gao, Jiantao Zhou, Jian Yang, and Guo-Jun Qi. 2022. Cross-receptive focused inference network for lightweight image super-resolution. arXiv:2207.02796. Retrieved from https://arxiv.org/abs/2207.02796.

[28] Wenbo Li, Kun Zhou, Lu Qi, Nianjuan Jiang, Jiangbo Lu, and Jiaya Jia. 2021. Lapar: Linearly-assembled pixel-adaptive regression network for single image super-resolution and beyond. arXiv:2105.10422. Retrieved from https://arxiv.org/abs/2105.10422.

[29] Yanchun Li, Jianglian Cao, Zhetao Li, Sangyoon Oh, and Nobuyoshi Komuro. 2021. Lightweight single image super-resolution with dense connection distillation network. *ACM Trans. Multimedia Comput. Commun. Appl.* 17, 1s (2021), 1–17.

[30] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. 2017. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'17)*. 136–144.

[31] Jie Liu, Jie Tang, and Gangshan Wu. 2020. Residual feature distillation network for lightweight image super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV'20)*. Springer, 41–55.

[32] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Changhu Wang, and Jiashi Feng. 2020. Improving convolutional networks with self-calibrated convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'20)*. 10096–10105.

[33] Zhisheng Lu, Hong Liu, Juncheng Li, and Linlin Zhang. 2022. Transformer for single image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'22) Workshops*. 457–466.

[34] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. 2001. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'01)*. 416–423.

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

[36] Yiqun Mei, Yuchen Fan, Yuqian Zhou, Lichao Huang, Thomas S. Huang, and Honghui Shi. 2020. Image super-resolution with cross-scale non-local attention and exhaustive self-exemplars mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'20)*. 5690–5699.

[37] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. 2016. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*. 1874–1883.

[38] Ying Tai, Jian Yang, and Xiaoming Liu. 2017. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17)*. 3147–3155.

[39] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. 2017. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'17)*. 4539–4547.

[40] Chunwei Tian, Yong Xu, Wangmeng Zuo, Bob Zhang, Lunke Fei, and Chia-Wen Lin. 2020. Coarse-to-fine CNN for image super-resolution. *IEEE Trans Multimedia* 23 (2020), 1489–1502.

[41] Chaofeng Wang, Zheng Li, and Jun Shi. 2019. Lightweight image super-resolution with adaptive weighted learning network. arXiv:1904.02358. Retrieved from https://arxiv.org/abs/1904.02358.

[42] Longguang Wang, Xiaoyu Dong, Yingqian Wang, Xinyi Ying, Zaiping Lin, Wei An, and Yulan Guo. 2021. Exploring sparsity in image super-resolution for efficient inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'21)*. 4917–4926.

[43] Yingqian Wang, Zhengyu Liang, Longguang Wang, Jungang Yang, Wei An, and Yulan Guo. 2022. Learning a degradation-adaptive network for light field image super-resolution. arXiv:2206.06214. Retrieved from https://arxiv.org/abs/2206.06214.

[44] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. 2022. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'22)*. 17683–17693.

[45] Zhengxue Wang, Guangwei Gao, Juncheng Li, Yi Yu, and Huimin Lu. 2021. Lightweight image super-resolution with multi-scale feature interaction network. In *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME'21)*. IEEE, 1–6.

[46] Peng Yi, Zhongyuan Wang, Kui Jiang, Junjun Jiang, Tao Lu, and Jiayi Ma. 2022. A progressive fusion generative adversarial network for realistic and consistent video super-resolution. *IEEE Trans. Pattern Anal. Mach. Intell.* 44, 5 (2022), 2264–2280.

[47] Roman Zeyde, Michael Elad, and Matan Protter. 2010. On single image scale-up using sparse-representations. In *Proceedings of the International Conference on Curves and Surfaces (ICCS'10)*. 711–730.

[48] Dongyang Zhang, Jie Shao, Zhenwen Liang, Lianli Gao, and Heng Tao Shen. 2021. Large factor image super-resolution with cascaded convolutional neural networks. *IEEE Trans. Multimedia* 23 (2021), 2172–2184.

[49] Jiqing Zhang, Chengjiang Long, Yuxin Wang, Haiyin Piao, Haiyang Mei, Xin Yang, and Baocai Yin. 2022. A two-stage attentive network for single image super-resolution. *IEEE Trans. Circ. Syst. Vid. Technol.* 32, 3 (2022), 1020–1033.

[50] Xinyan Zhang, Peng Gao, Sunxiangyu Liu, Kongya Zhao, Guitao Li, Liuguo Yin, and Chang Wen Chen. 2021. Accurate and efficient image super-resolution via global-local adjusting dense network. *IEEE Trans. Multimedia* 23 (2021), 1924–1937.

[51] Xindong Zhang, Hui Zeng, and Lei Zhang. 2021. Edge-oriented convolution block for real-time super resolution on mobile devices. In *Proceedings of the ACM International Conference on Multimedia (ACMMM'21)*. 4034–4043.

[52] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. 2018. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV'18)*. 286–301.

[53] Feiyang Zhu and Qijun Zhao. 2019. Efficient single image super-resolution via hybrid residual feature learning with compact back-projection network. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW'19)*. 1–8.

[54] Xiangyuan Zhu, Kehua Guo, Sheng Ren, Bin Hu, Min Hu, and Hui Fang. 2022. Lightweight image super-resolution with expectation-maximization attention mechanism. *IEEE Trans. Circ. Syst. Vid. Technol.* 32, 3 (2022), 1273–1284.

[55] Wenbin Zou, Tian Ye, Weixin Zheng, Yunchen Zhang, Liang Chen, and Yi Wu. 2022. Self-calibrated efficient transformer for lightweight super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'22) Workshops*. 930–939.

Received 7 May 2022; revised 5 September 2022; accepted 20 October 2022