Multi-Scale Residual Fusion Network for Super-Resolution Reconstruction of Single Image

BAITING ZHAO1, RUI HU1, XIAOFEN JIA1,2, AND YONGCUN GUO2
1School of Electrical and Information Engineering, Anhui University of Science and Technology, Huainan 232001, China
2State Key Laboratory of Mining Response and Disaster Prevention and Control in Deep Coal Mines, Anhui University of Science and Technology, Huainan 232001, China
Corresponding author: Xiaofen Jia (jxfzhb2008@163.com)
This work was supported in part by the National Key Research and Development Program under Grant 2016YFC0600908, in part by the Natural Science Research Projects of Colleges and Universities in Anhui Province under Grant KJ2018ZD008, and in part by the University Synergy Innovation Program of Anhui Province under Grant GXXT-2019-048.

ABSTRACT Inferring all missing high-frequency details by using low-resolution image is the key to super-resolution reconstruction of a single image. In order to extract the feature information in low-resolution images fully and maximize the deduction of high-frequency details, we introduce a multi-scale cross merge (MSCM) network based on residual fusion. The MSCM uses feature extraction module with different-sized convolutional kernels to extract multiple features from the low resolution input image and send them into a nonlinear mapping module after concatenate them together. The nonlinear mapping module consists of five cross-merge modules, each of them formed by cascading three residual dual-branch merged structures. This structure can promote information integration of different branches. Dense connection and residual connection are integrated into the nonlinear mapping module to improve the transmission of information and gradient. The nonlinear mapping module is responsible for extracting high-frequency features and sending them to reconstruction module, which combines an improved sub-pixel up-sampling layer with external residual and global residual to generate a high resolution image. Simulation experiments demonstrate that our MSCM network has the ability of achieving single-image super-resolution reconstruction, and offers objective and subjective quality improvement compared to mainstream methods and other state-of-the-art reconstruction methods.

INDEX TERMS Super-resolution reconstruction, multi-scale convolution kernel, cascaded network, feature fusion, residual network.

I. INTRODUCTION Inferring a high-resolution (HR) image from a corresponding low-resolution (LR) image is called single image super-resolution (SISR). It is an important issue in digital image processing, which is widely used in satellite remote sensing, medical image, video surveillance, unmanned driving, hyper-spectral imaging and other fields. In the process of reconstruction, the traditional SISR methods often have some problems, such as loss of high-frequency details, blurred edges and so on. With the rapid progress of computer hardware equipment, deep network model based SISR methods have received widespread attention.

SRCNN [1] is the first convolutional neural network (CNN) model for SISR, which contains three convolutional layers. To improve the gradient dispersion problem, VDSR [2] is proposed based on residual network ResNet [3]. The multi-level fusion network MFFRnet [4] used a recursive residual network to extract features for image reconstruction. FDSR [5] is a symmetric residual convolution structure, which can speed up the training and improve the reconstruction effect without changing the network complexity. DRCN [6] and DRRN [7] introduced recurrent neural network, which greatly reduces network parameters and accelerates the convergence speed. The achievements of Densenet [8] in image classification inspire scholars to use it for super-resolution reconstruction. MemNet [9] used dense connections to stack persistent memory and multiple memory blocks for image restoration. RDN [10] realized the interconnection and fusion of multiple residual dense blocks through the residual dense connections. All these literatures show that residual and dense connections can...
promote information flow, alleviate gradient disappearance and improve reconstruction performance.

In order to enhance the reconstruction effect, scholars try to construct independent sub-networks in the HR reconstruction model. Tang et al. [11] proposed a joint residual network consisting of three parallel sub-networks, two of them learn low-frequency information and the other one learns high-frequency information. CMSC [12] utilized multiple cross sub-networks to enhance the extraction of information. CSN [13] reduced learning burden of deep network by distributing feature information with sub-networks. HDRN [14] reduced memory consumption and promoted feature fusion by constructing multiple hierarchical dense residual blocks. MGE-P-SRCNN [15] proposed to construct an independent pre-classified network, which selects sample sub-database that highly related to the target image before feature extraction, and then performs nonlinear mapping and image reconstruction. The above-mentioned structures of independent sub-network promoted a new idea for the construction of reconstruction models.

The mainstream SISR methods mainly obtain more contextual information by deepening the network layer. Zagoruyko and Komodakis [16] and Zhao et al. [17] found that increasing network width can promote the fully extraction of image features. CNF [18] integrated multiple convolutional layers into network model, and improved network performance by fusing the contextual information of the network. LapSRN [19] combined the traditional Laplace pyramid with deep learning to reconstruct images. MCNN [20] utilized the competition among multi-scale convolution filters to enhance the inference ability of the network. Multi-scale convolution is explored by EEDS [21] to construct deep and shallow networks to extract different features, respectively. HCNN [22] implemented edge extraction, edge enhancement and HR reconstruction by constructing a three-branch network structure. These good reconstruction effects inspired us to extract more information by increasing the network width.

Investigation shows that both the construction of sub-networks and the increase of the network width can promote extraction of image features fully, and the residual dense connections can effectively simplify the difficulty of network training. Therefore, a multi-scale cross merge network (MSCM) for SISR is proposed. The main contributions include three aspects:

1) We design a feature extraction module (FEM), which uses different-sized convolutional kernels to fully extract image features of the input LR image and concatenating them together to form multi-range contextual information. FEM can achieve complementarity of different types of features.

2) We propose a nonlinear mapping module (NMM) by cascading five cross-merge modules. Dense connection and local residual connection are integrated into the nonlinear mapping module to achieve the fusion of multi-level and multi-scale features. NMM can obtain the necessary high-frequency details for reconstructing the textural details of the HR image.

3) We establish an image reconstruction process which combines external residual, global residual and sub-pixel convolution. The global residual connection is used to merge the low-level features extracted from the shallow layer and the high-level features extracted from the deep layer. The low-frequency information in the LR image is combined with the high-frequency information inferred from the network by using the external residual connection. Sub-pixel convolution is used in the last layer of the network to realize up-sampling.

This article is organized as follows. Section 2 introduces the structure of CNN for SISR. Section 3 describes the proposed MSCM hierarchical. Section 4 analyzes the structure of MSCM, presents experimental results and compares them with the leading methods on four public datasets. Section 5 concludes the paper.

II. CNN ARCHITECTURE FOR SISR

A. CONVOLUTION KERNEL

The convolution kernel is an important part of the CNN architecture. Each convolution kernel can extract a feature map. The reconstruction effect of HR image is closely related to the size of convolution kernel when the network depth is determined. Every feature map has its own suitable convolution scale. On this scale, the image feature is the most obvious and can be distinguished accurately from the surrounding environment. The large-scale convolution kernel has the ability to learn complex features, but it will lose the detailed information, while the small-scale convolution kernel is easy to learn and can bring more detailed information, but it has poor ability to learn complex features.

SRCNN [1] is the first CNN model for SR, which uses a single $9 \times 9$ convolution kernel to extract image features. FSRCNN [23], ESPCN [24], VDSR [2], DRCN [6] etc. also extract features through a single-scale convolution layer. GoogLeNet [25], developed by Google, solves the problem of insufficient information extraction caused by a single-scale convolution kernel by increasing the network width, that is, extracting information with multi-scale convolution kernels to enhance the performance of the network. This idea can be introduced into the feature extraction of SR methods.

B. MULTI-BRANCH STRUCTURE

GoogLeNet is a multi-branch structure essentially, which increases the width of the network and promotes information fusion through paralleling convolution kernels of different scales. Zhao et al. [17] merged two residual branches together and then carried out mapping transfer (MR) to realize the rapid transmission of feature information from the front-end module to the back-end module, and realize the rapid backward propagation of gradient from the back-end module to the front-end module. Hu et al. [12] changed the convolutional layer in the MR module to multi-scale convolution, and Zhao et al. [13] introduced residuals to the branch network further, they achieved ideal reconstruction effect. The above methods confirm the feasibility of constructing multi-branch
network module, and provide a new idea for the construction of network structure.

C. HR IMAGE RECONSTRUCTION
SRCNN [1] and FSRCNN [23] enlarged the LR image to the target size firstly by using interpolation, and then used the convolution network to reconstruct the HR image. Enlarging the size of a LR image cannot bring more details information, but will destroy the contextual information of the image. In other words, increasing the image size does not contribute to the inference of high-frequency information. Shi et al. [24] proposed a sub-pixel up-sampling method, which directly extracts features from a LR image and obtains the up-sampled image by periodically arranging multiple feature maps extracted by convolution. Networks such as RDN [10] and MFFRnet [4] also use this method to achieve image up-sampling. The sub-pixel up-sampling avoids the loss of LR image features caused by interpolation and amplification. It can preserve the contextual information and bring abundant image details, thereby improving the effect of super-resolution reconstruction.

III. MULTI-SCALE CROSS MERGE NETWORK

A. THE MSCR MODEL
The purpose of single image super-resolution is to infer all the missing high-frequency details from an input LR image $X$, thereby to obtain a HR image $Y$. Given a training dataset $E = \{X^{(k)}, Y^{(k)}\}, k = 1, 2, \cdots, D$, where $X^{(k)}$ and $Y^{(k)}$ represent the LR image and the corresponding HR image. The SISR reconstruction model is to achieve end-to-end mapping from a LR image to a HR image. The goal of our MSCM model is to learn a deduction model $\Gamma$ to obtain inferring value $\hat{Y}^{(k)}$ of the HR image from the input real LR image $X^{(k)}$,

$$\hat{Y}^{(k)} = \Gamma(X^{(k)}; \Theta)$$  \hspace{1cm} (1)

where $\Theta = [\omega, b]$ are the network parameters of MSCM, $\omega$ is the weight matrix, $b$ is the bias. The model parameters $\Theta$ are determined by minimizing the loss between the reconstructed HR image and the real HR image. We define the loss function as,

$$L(\Theta) = \frac{1}{D} \sum_{k=1}^{D} ||Y^{(k)} - \Gamma(X^{(k)}; \Theta)||$$  \hspace{1cm} (2)

The process of training MSCM with training set $E$ is to minimize the loss and find the optimal parameters $\Theta$ for the model. The structure of the MSCM model is shown in Figure 1. It consists of feature extraction module (FEM), nonlinear mapping module (NMM) and reconstruction module (RM). The FEM is responsible for extracting the shallow features of the input LR image and obtains the up-sampled image by periodically arranging multiple feature maps extracted by convolution. Networks such as RDN [10] and MFFRnet [4] also use this method to achieve image up-sampling. The sub-pixel up-sampling avoids the loss of LR image features caused by interpolation and amplification. It can preserve the contextual information and bring abundant image details, thereby improving the effect of super-resolution reconstruction.

B. FEATURE EXTRACTION MODULE
Using only single-scale convolution kernel to extract low-level features will cause many feature details missing. In this article, the $3\times3$ convolution model composed of $3 \times 3, 5 \times 5$ and $9 \times 9$ is used as the scale of the convolution kernels in the FEM. The multi-scale convolution model can extract more feature details from the LR image, which is conducive to the
reconstruction of HR image details. This structure can make use of the complementary advantages fully between different features. The feature extraction formula is defined as,

\[ F_1 = H_{3 \times 3}(X), F_2 = H_{5 \times 5}(X), F_3 = H_{9 \times 9}(X) \]  

(3)

where \( X \) is the input LR image and \( H \) represents the convolution operation operator, the subscript denotes the size of the convolutional kernel used in the layer, \( F_1 \) is the extracted feature map corresponding to the \( 3 \times 3 \) convolutional kernel. Similarly, the LR image is convoluted with \( 5 \times 5 \) and \( 9 \times 9 \) convolution kernels respectively to obtain the feature maps \( F_2 \) and \( F_3 \). Then, concatenating the three feature maps to obtain \( F \), which is the result of feature extraction by the \( 3, 5, 9 \) convolution model. The formula is defined as,

\[ F = [F_1, F_2, F_3] \]  

(4)

where \([\cdot]\) represents the concatenation operation.

Feature dimensionality reduction of \( F \) is performed through \( 1 \times 1 \) convolution kernel to avoid too many training parameters, which is conducive to improving the robustness of the network. Then, the \( 3 \times 3 \) convolution kernel is used to extract features further to obtain the final feature \( X_0 \) of the FEM. The formula is as follows,

\[ X_0 = H_{3 \times 3}(H_{1 \times 1}(F)) \]  

(5)

C. NONLINEAR MAPPING MODULE

NMM consists of \( n \) cascaded cross-merge modules (CM). We can see the structure of CM in Figure 2. It is cascaded by \( m \) residual dual-branch merged structures (RDM) where integrate dense connections. “Add” the output of the dual-branch of the last RDM, and adjust the dimensions through a \( 1 \times 1 \) convolution kernel. After that, LRC is used to improve the flow of information and gradient.

\[ \begin{align*}
C_{i,j}^{b_1} &= \max(0, H_{3 \times 3}(X_{i,j-1}^{b_1})) \\
C_{i,j}^{b_2} &= \max(0, H_{5 \times 5}(X_{i,j-1}^{b_2})) \\
\begin{bmatrix}
X_{i,j}^{b_1} \\
X_{i,j}^{b_2}
\end{bmatrix}
&= \begin{bmatrix}
C_{i,j}^{b_1} \\
C_{i,j}^{b_2}
\end{bmatrix} + \frac{1}{2} \begin{bmatrix} I & I \\
I & I
\end{bmatrix} \begin{bmatrix}
X_{i,j-1}^{b_1} \\
X_{i,j-1}^{b_2}
\end{bmatrix}
\end{align*} \]  

(6)

(7)

where \( i = 1, 2, \ldots, n, j = 1, 2, \ldots, m \). \( C_{i,j}^{b_1} \) and \( C_{i,j}^{b_2} \) represent the outputs activated by ReLU activation layer after convolving the input \( X_{i,j-1}^{b_1} \) and \( X_{i,j-1}^{b_2} \) of the upper-bran and lower-bran with \( 3 \times 3 \) and \( 5 \times 5 \) convolution kernels, respectively. \( X_{i,j-1}^{b_1} \) and \( X_{i,j-1}^{b_2} \) are the outputs of the upper-bran and lower-bran of the \( j \)-th RDM, satisfying the relationship of \( X_{i,j}^{b_1} = X_{i,j}^{b_2} \).

2) LOCAL RESIDUAL CONNECTION

After cascading \( m \) RDMs, the feature mapping results of the upper-bran and lower-bran are merged by “Add”, thus, we obtain \( M_1 \). The \( M_1 \) passes through a convolutional layer with \( 1 \times 1 \) convolution kernel and obtains \( M_2 \), which is “Add” with the input of the \( i \)-th CM transferring to the upper-layer by LRC, we can get the output of the \( i \)-th CM as follow,

\[ X_i = D_c(M_2, X_{i-1}) = D_c(H_{1 \times 1}(M_1), X_{i-1}) = D_c(H_{1 \times 1}(D_c(H_{1 \times 1}(D_c(H_{1 \times 1}(X_{i,m}^{b_1}, X_{i,m}^{b_2})), X_{i-1})))) \]  

(8)

where \( D_c(\cdot, \cdot) \) represents the mapping function of “Add”, which merges the upper and lower branches.

For the sake of convenience in representation, the mapping relationship between the input \( X_{i-1} \) and the output \( X_i \) in...
Eq. (8) is represented by $T^i_c$, the results $X_n$ of the $n$-th CM can be obtained as,

$$X_n = T^n_c \left( T^{n-1}_c \left( \cdots \left( T^1_c (X_0) \right) \cdots \right) \right)$$  \hspace{1cm} (9)

where $X_0$ is the input of the first CM and the output of FEM.

**D. RECONSTRUCTION MODULE**

The image reconstruction module includes two parts, global feature fusion and image restoration.

1) **GLOBAL FEATURE FUSION**

By concatenating the nonlinear mapping results with dense connections and “concat”, the outputs of $n$ CMs are connected into a tensor $X_M$, its mathematical model is,

$$X_M = [X_0, X_1, \ldots, X_n]$$  \hspace{1cm} (10)

where $X_M$ is the global feature obtained by fusing the local features in all channel blocks of NMM. $X_M$ goes through a $1 \times 1$ convolution and a $3 \times 3$ convolution in turn, we can obtain the first-level feature $\hat{F}_1$. “Add” it with the feature $F_1$ of FEM transferring by GRC, we can get the second-level feature $\hat{F}_2$, which is convolved in turn by three convolution layers with convolution size of $5 \times 5, 3 \times 3$ and $3 \times 3$, respectively, to obtain the third-level feature $\hat{F}_3$. The mathematical models are as follows,

$$\hat{F}_1 = H_{3\times3} (H_{1\times1} (X_M))$$  \hspace{1cm} (11)

$$\hat{F}_2 = D_c \left( F_1, \hat{F}_1 \right)$$  \hspace{1cm} (12)

$$\hat{F}_3 = H_{3\times3} (H_{3\times3} (H_{5\times5} (\hat{F}_2)))$$  \hspace{1cm} (13)

The $1 \times 1$ convolution kernel in Eq. (11) can reduce the number of channels to 64 and reduce the model parameters. Convolutions with larger sizes (e.g., $5 \times 5$ or $9 \times 9$) have more connection parameters, which increases the computational complexity. A good deep learning model often needs to achieve the trade-off between the best performance and the running efficiency. Therefore, only $F_1$ is selected as the object of GRC.

2) **IMAGE RESTORATION**

The sub-pixel up-sampling convolution (SUC) refers to the process of rearranging $r^2$ feature maps of size $h \times w \times c$ extracted from LR space into one $r\times h\times w\times c$ HR image. The effect of the SUC is illustrated in Figure 4.

SUC is easy to destroy the correlation between pixels in the process of periodic arrangement, but LR image and the corresponding HR image have similar topological structure, which can enhance the correlation between pixels. In view of this, we introduce LR image into the last part of HR reconstruction by ERC, then realize up-sampling using sub-pixel convolution. Finally, the reconstruction of HR image is completed by passing the up-sampling result through a convolution layer with convolution size of $3 \times 3$. The mathematical models are as follows,

$$T = D_c \left( \hat{F}_3, \hat{X} \right) = D_c \left( \hat{F}, H_{1\times1} (X) \right)$$  \hspace{1cm} (14)

where $\hat{X}$ is the $1 \times 1$ convolution result of the input LR image $X$, $T$ is the feature map to be reconstructed, $\hat{Y}$ is the up-sampling result, $\text{SUC}(\cdot)$ is the pixel recombination operation for the LR feature image, $r$ is the scale factor of up-sampling amplification, and $c$ is the number of image channels (the corresponding values of color image and gray image are 3 and 1, respectively). $\text{mod}(x, r)$ and $\text{mod}(y, r)$ represent the activation operation, that is, according to the different sub-pixel positions in the $r^2$ LR maps, the pixel regions in the same position are activated in the process of pixel recombination, and extracted to form a region in the HR image $Y$.

In Eqs. (5), (6), (8), (11), (13), (14) and (16), the operator $H$ and the subscript have the same meaning as Eq. (3). The operator $D_c$ in Eqs. (12) and (14) has the same meaning as Eq. (8).

**IV. EXPERIMENTS AND DISCUSSION**

**A. DATASETS AND EXPERIMENTAL SETTING**

The experiments are carried out in the environment TensorFlow. Each convolutional layer of the MSCM network has 64 convolution kernels, and the network is trained using stochastic gradient descent. We trained 30 epochs and used Adam optimizer with variable learning efficiency. Its initial value is set to $10^{-4}$, but it decreases by half every 10 epochs. For each epoch, the batch size is 64.

The training set is constructed using 800 color images ($c = 3$) from the DIV2K Dataset [26]. We extract output HR training samples of size $32 \times 32 (h \times w)$ from the 800 original images with a stride of 16. The corresponding input LR training samples are generated by using Bicubic down-sampling function, the size is set to $32/r \times 32/r$. If $32$ is not an integer multiple of $r$, we perform an integer operation on $32/r$. The MSCM network is trained to obtain the corresponding reconstruction model for the upscale factor $\times 2$, $\times 3$ and $\times 4$, respectively. In order to evaluate the advantages and disadvantages of the MSCM model accurately, we use the current
mainstream datasets, Set5 [27], Set14 [28], BSD100 [29] and Urban100 [30] as the test sets for analysis and research.

Compared with RGB color space, the luminance (Y) channel in YCbCr color space contains more structure information that is more suitable for human eyes. Hence, we convert RGB color space to YCbCr color space, only train and test Y-channel for SISR. Two objective indicators, peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [31] are used to evaluate the image reconstruction effect.

### B. ARCHITECTURE ANALYSIS OF MSCM

In the structure analysis of the MSCM model, we select Set5 as the test set, and set the scale factor \( r = 3 \).
1) DETERMINE THE STRUCTURE OF FEM

In order to determine the number and size of parallel convolution kernels in FEM, an experimental analysis of the 8 FEM structures of different scales in Figure 5 is carried out. The numbers 3, 5, 7, 9 and 11 represent the size of convolution kernel, and each convolution layer contains 64 filters. In the test of 8 FEM structures, the NMM and RM are exactly the same. The experimental results are shown in Figure 6, where the abscissa is the type of FEM structure (list in Figure 5) and the ordinate is the corresponding evaluation results. It can be seen from Figure 6(a), FE 3_9 yields the highest PSNR, and it goes beyond all other FEM structures. FE 3_5_9 is the second-best structure. Figure 6(b) shows the SSIM of 8 FEM structures, we can see that FE 3_5_9 obtained the highest SSIM, followed by FE 3_9. In order to maximize the complementarity of different features, FE 3_5_9 is selected as the structure of our FEM, which is used in subsequent experiments.

2) DETERMINE THE STRUCTURE OF RDM

RDM is an important part of CM. In order to determine the optimal mapping structure, the experimental analysis of 6 RDM models in Figure 7 is carried out. The numbers 3, 5 and 7 represent the size of convolution kernel, and each convolution layer contains 64 filters. The objective indicators of 6 compared RDM models are illustrated in Table 1, where boldface indicates the best indicator. It can be seen that the CM 3_5 model achieved the highest PSNR and SSIM. So, the CM 3_5 model was selected as the RDM structure.

3) DETERMINE THE NUMBER OF CM AND RDM

NMM is formed by cascading $n$ CMs, and each CM is cascaded by $m$ RDMs. We should find the best values of $n$ and $m$, so, two experiments are carried out. Taking $m = 3$, set $n$ as 2, 3, 4, 5, 6, respectively, we compare the PSNR and reconstruction time, the results are shown in Figure 8 (a). Taking $n = 5$, $m$ is set as 2, 3, 4, 5, respectively, the PSNR and reconstruction time are shown in Figure 8 (b). The abscissas of Figure 8 (a) and Figure 8 (b) are the number $n$ of CM and the number $m$ of RDM, respectively. Both ordinates are all PSNR, and the numbers on the polyline are reconstruction time. As shown in Figure 8, with the number of CM increasing, the PSNR and reconstruction time will increase. As the number of RDM increases, the PSNR and reconstruction time will also increase. In order to compromise the operation speed and reconstruction effect, the final determination is $n = 5$, $m = 3$. 
C. EXPERIMENTAL RESULTS

In order to illustrate the effectiveness of MSCM model, two reconstruction experiments of \( \times 2 \), \( \times 3 \) and \( \times 4 \) scale factors are carried out on datasets Set5, Set14, BSD100 and Urban100. Experiment 1: Comparison with mainstream deep learning method. Experiment 2: Comparison with the latest deep learning methods.

1) COMPARISON WITH MAINSTREAM METHODS

We obtain the source codes provided by the authors of methods SRCNN [1], VDSR [3], DRCN [6], DRRN [7], MemNet [9] and LapSRN [19], which are retrained on a PC with an Intel(R) Core (TM) i7-8750 CPU and a GPU model of NVIDIA GeForce GTX 1060 to obtain the results and compared with MSCM. The comparison includes three aspects,
TABLE 2. Average PSNR/SSIM values of mainstream methods for \( \times 2, \times 3, \times 4 \) scales.

| Dataset | Scale | SRCNN[1] | VDSR[3] | DRCN[6] | DRRN[7] | MemNet[9] | LapSRN[19] | MSCM (our) |
|---------|-------|----------|---------|---------|---------|-----------|------------|-----------|
| Set5    | \( \times 2 \) | 36.66/0.9542 | 37.53/0.9587 | 37.63/0.9588 | 37.74/0.9591 | 37.78/0.9597 | 37.52/0.9590 | 38.01/0.9606 |
|         | \( \times 3 \) | 32.75/0.9090 | 33.66/0.9213 | 33.82/0.9226 | 34.03/0.9244 | 34.09/0.9248 | 33.82/0.9227 | 34.44/0.9273 |
|         | \( \times 4 \) | 30.48/0.8628 | 31.35/0.8838 | 31.53/0.8854 | 31.68/0.8888 | 31.74/0.8893 | 31.54/0.8850 | 32.21/0.8915 |
| Set14   | \( \times 2 \) | 32.45/0.9067 | 33.05/0.9124 | 33.04/0.9118 | 33.23/0.9136 | 33.28/0.9142 | 33.08/0.9130 | 33.67/0.9186 |
|         | \( \times 3 \) | 29.30/0.8215 | 29.77/0.8314 | 29.76/0.8311 | 29.96/0.8349 | 30.00/0.8350 | 29.79/0.8320 | 30.34/0.8454 |
|         | \( \times 4 \) | 27.50/0.7513 | 28.01/0.7674 | 28.02/0.7670 | 28.21/0.7720 | 28.26/0.7723 | 28.19/0.7720 | 28.58/0.7813 |
| BSD100  | \( \times 2 \) | 31.36/0.8879 | 31.90/0.8960 | 31.85/0.8942 | 32.05/0.8973 | 32.08/0.8978 | 31.80/0.8950 | 32.19/0.9031 |
|         | \( \times 3 \) | 28.41/0.7863 | 28.82/0.7976 | 28.80/0.7963 | 28.95/0.8004 | 28.96/0.8001 | 28.82/0.7973 | 29.26/0.8093 |
|         | \( \times 4 \) | 26.90/0.7101 | 27.29/0.7251 | 27.23/0.7233 | 27.38/0.7284 | 27.40/0.7284 | 27.32/0.7280 | 27.62/0.7401 |
| Urban100| \( \times 2 \) | 29.50/0.8946 | 30.76/0.9140 | 30.75/0.9133 | 31.23/0.9188 | 31.31/0.9195 | 30.41/0.9101 | 32.05/0.9284 |
|         | \( \times 3 \) | 26.24/0.7989 | 27.14/0.8279 | 27.15/0.8276 | 27.53/0.8378 | 27.56/0.8376 | 27.07/0.8272 | 28.19/0.8562 |
|         | \( \times 4 \) | 24.52/0.7221 | 25.18/0.7540 | 25.14/0.7510 | 25.44/0.7638 | 25.50/0.7630 | 25.21/0.7560 | 26.09/0.7883 |

The average PSNR, average SSIM of MSCM and five compared methods on datasets Set5, Set14, BSD100 and Urban100 are shown in Table 2, in which, the maximal values are boldface, and the second ones are blue. We can find that, for the four datasets with \( \times 2, \times 3 \) and \( \times 4 \) scale factors, the proposed method achieves the highest PSNR and SSIM. On all scale factors (\( \times 2, \times 3 \) and \( \times 4 \)), the MemNet obtains the second best results in terms of PSNR and SSIM for Set5 and Set14, and obtains the second best results in terms of PSNR for BSD100 and Urban100. For Urban100 with scale factors \( \times 2, \times 3 \) and \( \times 4 \), when compared to MemNet, our MSCM shows a notable PSNR increase of 0.74dB, 0.63dB and 0.59dB, respectively.

Figure 8 and Figure 9 show the subjective comparison results of the reconstructed images with scale factors \( \times 3 \) and \( \times 4 \), respectively, for the images from the dataset Urban100. Observe the \( \times 3 \) reconstruction detail image of the high-rise building in Figure 8, and make a local comparison of the wall texture. It can be seen that the wall texture reconstructed by our algorithm is the closest to the HR image, and the contour is obvious, while the image texture reconstructed by other algorithms is disordered and the overall image is blurred.

In Figure 9, a detailed comparison of the station building with a magnification of \( \times 4 \) is performed. Comparing the local details of the window, we can find that MSCM yields the most obvious window outline and restores edge details well, while other methods cannot effectively restore the contours.

The ultimate goal of theoretical research is to achieve practical application, and the background of actual image is more complex than any dataset used in theoretical research, actual image can better measure the advantages and disadvantages of reconstruction methods. Satellite remote sensing images are more disturbed by external factors, making reconstruction more difficult. We use an image obtained by Beijing-2 satellite of China with \( \times 4 \) scale factor to verify the proposed MSCM and compare its result with 6 mainstream methods. Figure 11 shows the reconstruction results, the large image on the left is the actual HR image, (a) ~ (h) are
the partial enlargement results of the 6 compared methods and MSCM. It can be seen that MSCM achieves the best reconstruction result, which has high sharpness and clear edge contour.

From above experimental results, we can see that, our method has better performance when the reconstruction scale factor is larger. It can recover the contour features well and get a better reconstructed image.

2) COMPARISON WITH THE LATEST METHODS
The latest methods of MFFRnet [4], FDSR [5], MGEP-SRCNN [15], EEDS [21] and HCNN [22] achieved good reconstruction results, but we didn’t find the source codes provided by the authors. For objective and fair comparison, we directly compare the results provided in the paper with our MSCM.

The average PSNR and SSIM values of five compared algorithms and MSCM on four datasets are listed in Table 4, where boldface indicates the best results, blue indicates the second results. We can find that, MSCM obtains the highest PSNR and SSIM at $\times 2$, $\times 3$ and $\times 4$ scales on the four datasets. MGEP-SRCNN is the second-best method on datasets Set5, Set14 and BSD100. Under three scales $\times 2$, $\times 3$, $\times 4$, the MSCM illustrates PSNR values increase of 0.23dB, 0.4dB and 0.55dB on dataset Set5 when compared to MGEP-SRCNN, respectively. MSCM obtain the highest PSNR increase on dataset Urban100 when compared to datasets Set5, Set14 and BSD100. For example, at the $\times 2$ scale, the PSNR value of MSCM outperforms MFFRnet, FDSR, and MGEP-SRCNN by [1.16, 1.03, 0.89] dB, respectively.

### TABLE 4. Average PSNR/SSIM values of latest methods for $\times 2$, $\times 3$, $\times 4$ scales.

| Dataset | Scale | MFFRnet[4] | FDSR[5] | MGEP-SRCNN[15] | EEDS[21] | HCNN[22] | MSCM(our) |
|---------|-------|------------|---------|----------------|----------|---------|----------|
| Set5    | $\times 2$ | 37.68/0.9590 | 37.34/0.9514 | **37.78/0.9594** | 37.29/0.9579 | 37.62/0.9594 | **38.01/0.9606** |
|          | $\times 3$ | 33.91/0.9226 | 33.76/0.9098 | **34.04/0.9250** | 33.47/0.9191 | 33.77/0.9230 | **34.44/0.9273** |
|          | $\times 4$ | 31.39/0.8843 | 31.42/0.8876 | **31.66/0.8899** | 31.14/0.8783 | 31.39/0.8849 | **32.21/0.8915** |
| Set14   | $\times 2$ | 32.99/0.9134 | 32.97/0.9044 | **32.29/0.9135** | 32.81/0.9105 | 33.03/0.9127 | **33.67/0.9186** |
|          | $\times 3$ | 29.60/0.8327 | 29.66/0.8190 | **29.97/0.8333** | 29.60/0.8284 | 29.79/0.8318 | **30.34/0.8454** |
|          | $\times 4$ | 27.85/0.7683 | 27.93/0.7521 | **28.19/0.8359** | 27.82/0.7626 | 28.04/0.7674 | **28.58/0.7813** |
| BSD100  | $\times 2$ | 32.03/0.8975 | 31.87/0.8841 | **32.01/0.8979** | 31.64/0.8928 | 31.91/0.8965 | **32.19/0.9031** |
|          | $\times 3$ | 28.87/0.7984 | 28.85/0.7796 | **28.92/0.7972** | 28.64/0.7925 | 28.84/0.7985 | **29.26/0.8093** |
|          | $\times 4$ | 27.33/0.7260 | 27.34/0.7038 | **27.34/0.7238** | 27.11/0.7200 | 27.29/0.7260 | **27.62/0.7401** |
| Urban100 | $\times 2$ | 30.89/0.9156 | 30.96/0.9099 | **31.19/0.9181** | -/- | -/- | **32.05/0.9284** |
|          | $\times 3$ | 27.16/0.8288 | 27.30/0.8207 | **27.24/0.8330** | -/- | -/- | **28.19/0.8562** |
|          | $\times 4$ | 25.20/0.7541 | **25.34/0.7450** | 25.21/0.7837 | -/- | -/- | **26.09/0.7883** |

D. PARAMETERS AND TIME CONSUMING

TABLE 3 lists the parameters and running time of VDSR [3], DRCN [6] and MSCM. We evaluate the three methods by reconstruction images with the scale factor $\times 4$ for datasets Set5, Set14, BSD100 and Urban100. Compared with VDSR [3] and DRCN [6], it is because MSCM has basically the same number of $3 \times 3$ convolutional layers with them, in other words, they have similar model depths, both VDSR [3] and DRCN [6] have 20 layers, and MSCM has 21 layers. From TABLE 3, we can see that, although MSCM is one level deeper than VDSR and DRCN, the model parameters are 205Kb and 1314Kb less than them respectively. And the MSCM is the fastest, that mainly because our method uses the shallow layer features fully, and does not need to preprocess the input LR image before training. Compared with the algorithms of VDSR and DRCN, our algorithm has more obvious advantages in large-scale datasets, and it is more suitable for large scale factor image reconstruction.

V. CONCLUSION

We propose an image reconstruction method for single LR image, which integrates the ideas of multi-scale, residual, cascade and fusion. The use of multi-scale convolution kernels and residual connections can facilitate the comprehensive extraction and fusion of multiple information, which is beneficial to characterize the feature information of a...
LR image fully, and lay the foundation for the inference of high-frequency information. The cascade structure can re- alize the information extraction supervision, prediction, and establish an effective mapping relationship from LR to HR image for super-resolution reconstruction. The multi-scale information fusion structure can promote the information exchange between different modules, and infer the missing high-frequency components by combining multi-contextual information and the mapping relationship of super-resolution reconstruction. For the four mainstream datasets Set5, Set14, BSD100 and Urban100 with ×2, ×3 and ×4 scale factors, our MSCM achieves the highest PSNR and SSIM, especially in Urban100. For example, for Urban100 with the ×2 scale factor, the PSNR value of MSCM outperforms MFFRnet, FDSR, and MGEP-SRCNN by [1.16, 1.03, 0.89] dB, respectively. This is mainly because the image size of Urban100 is larger, the image edge contour is clear, which is conducive to the implementation of our algorithm. That means our algorithm can fully extract the shallow features of the LR image. Meanwhile, MSCM has outstanding performance in large scale factor image reconstruction, it can recover the contour features well and get better reconstructed image.

ACKNOWLEDGMENT
The authors would like to thank the authors of the literatures compared to MSCM for providing their codes.

REFERENCES
[1] C. Dong, C. C. Loy, K. He, and X. Tang, “Image super-resolution using deep convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2014, pp. 184–199.
[2] J. Kim, J. K. Lee, and K. M. Lee, “Accurate image super-resolution using very deep convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 1646–1654.
[3] K. He, X. Zhang, S. Ren, and J. Sun, “Identity mappings in deep residual networks,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2016, pp. 630–645.
[4] X. Jin, Q. Xiong, C. Xiong, Z. Li, and Z. Gao, “Single image super-resolution with multi-level feature fusion recursive network,” Neurocomputing, vol. 370, pp. 166–173, Dec. 2019.
[5] Z. Lu, Z. Yu, P. Yali, L. Shigang, W. Xiaojun, L. Gang, and R. Yuan, “Fast single image super-resolution via dilated residual networks,” IEEE Access, vol. 7, pp. 109729–109738, 2019, doi: 10.1109/ACCESS.2018.2865613.
[6] J. Kim, J. K. Lee, and K. M. Lee, “Deeply-recursive convolutional network for image super-resolution,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 1637–1645.
[7] Y. Tai, J. Yang, and X. Liu, “Image super-resolution via deep recursive residual network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 3147–3155.
[8] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 4700–4708.
[9] Y. Tai, J. Yang, X. Liu, and C. Xu, “MemNet: A persistent memory network for image restoration,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 4549–4557.
[10] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, “Residual dense network for image super-resolution,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 2472–2481.
[11] Z. Tang, L. Luo, H. Peng, and S. Li, “A joint residual network with paired ReLUs activation for image super-resolution,” Neurocomputing, vol. 273, pp. 37–46, Jan. 2018.
[12] Y. Hu, X. Gao, J. Li, Y. Huang, and H. Wang, “Single image super-resolution via cascaded multi-scale cross network,” Feb. 2018, arXiv:1802.08808. [Online]. Available: http://arxiv.org/abs/1802.08808
[13] X. Zhao, Y. Zhang, T. Zhang, and X. Zou, “Channel splitting network for single MR image super-resolution,” IEEE Trans. Image Process., vol. 28, no. 11, pp. 5649–5662, Nov. 2019.
[14] K. Jiang, Z. Wang, P. Yi, and J. Jiang, “Hierarchical dense recursive network for image super-resolution,” Pattern Recognit., vol. 107, Nov. 2020, Art. no. 107475.
[15] J. L. Tang, C. R. Huang, J. Liu, and H. J. Zhu, “Image super-resolution based on CNN using multilabel gene expression programming,” Appl. Sci., vol. 10, no. 3, p. 854, 2020.
[16] S. Zagoruyko and N. Komodakis, “DiracNets: Training very deep neural networks without skip-connections,” Jun. 2017, arXiv:1706.06358. [Online]. Available: http://arxiv.org/abs/1706.06358
[17] L. Zhao, J. Wang, X. Li, Z. Tu, and W. Zeng, “Deep convolutional neural networks with merge-and-run mappings,” Jul. 2017, arXiv:1611.07718. [Online]. Available: http://arxiv.org/abs/1611.07718
[18] H. Ren, M. El-Khamy, and J. Lee, “Image super resolution based on fusing multiple convolutional neural networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Honolulu, HI, USA, Jul. 2017, pp. 1050–1057, doi: 10.1109/CVPRW.2017.142.
[19] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, “Deep Laplacian pyramid networks for fast and accurate super-resolution,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 624–632.
[20] X. Du, X. Qu, Y. He, and D. Guo, “Single image super-resolution based on multi-scale competitive convolutional neural network,” Sensors, vol. 18, no. 3, p. 789, Mar. 2018.
[21] Y. Wang, L. Wang, H. Wang, and P. Li, “End-to-end image super-resolution via deep and shallow convolutional networks,” IEEE Access, vol. 7, pp. 31595–31670, 2019.
[22] B. L. Liu and D. Ait-Boudaoud, “Effective image super resolution via hierarchical convolutional neural network,” Neurocomputing, vol. 374, pp. 109–116, Jun. 2020.
[23] C. Dong, C. C. Loy, and X. Tang, “Accelerating the super-resolution convolutional neural network,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Oct. 2016, pp. 391–407.
[24] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. A. Morel, “Low-complexity single-image super-resolution based on nonnegative truncated self-exemplars,” in Proc. 8th IEEE Int. Conf. Comput. Vis. Pattern Recognit. Workshops (ICCVW), Jun. 2012, pp. 5197–5206.
[25] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600–612, Apr. 2004.