MADNet: A Fast and Lightweight Network for Single-Image Super Resolution

Rushi Lan, Long Sun, Zhenbing Liu, Huimin Lu, Cheng Pang, and Xiaonan Luo

***Abstract*—Recently, deep convolutional neural networks (CNNs) have been successfully applied to the single-image super-resolution (SISR) task with great improvement in terms of both peak signal-to-noise ratio (PSNR) and structural similarity (SSIM). However, most of the existing CNN-based SR models require high computing power, which considerably limits their real-world applications. In addition, most CNN-based methods rarely explore the intermediate features that are helpful for final image recovery. To address these issues, in this article, we propose a dense lightweight network, called MADNet, for stronger multiscale feature expression and feature correlation learning. Specifically, a residual multiscale module with an attention mechanism (RMAM) is developed to enhance the informative multiscale feature representation ability. Furthermore, we present a dual residual-path block (DRPB) that utilizes the hierarchical features from original low-resolution images. To take advantage of the multilevel features, dense connections are employed among blocks. The comparative results demonstrate the superior performance of our MADNet model while employing considerably fewer multiadds and parameters.**

***Index Terms*—Channel attention, dense connections, image super resolution, lightweight, multiscale mechanism.**

I. Introduction

SINGLE-IMAGE super resolution (SISR) is an essential and classical problem in low-level computer vision that is related to reconstructing a visually high-resolution (HR) image from its low-resolution (LR) input. In practice, SISR is generally difficult to process due to its ill-posed nature, wherein multiple HR images can map to the same LR version. Addressing SISR has proven to be useful in many practical cases, such as video streaming [44], [50]; remote sensing [16], [58]; and medical imaging [31], [45], [48].

To mitigate this problem, numerous SR approaches have been proposed from different perspectives, including interpolation-based [17], reconstruction-based [33], and example-based methods [23], [25], [40], [41], [49]. The former two kinds of methods are simple and efficient but suffer a dramatic drop in restoration performance as the scale factors increase, and the example-based methods that try to analyze relationships between LR and HR pairs achieve satisfactory performance but involve time-consuming operations.

Recently, due to the powerful feature representation capability of the deep convolutional neural network (CNN), CNN-based methods have been proposed to learn a nonlinear mapping from an interpolated or LR version to its corresponding high-quality output. By entirely utilizing the inherent relations among images in training datasets, these models have provided outstanding performance in SR tasks [5], [7], [18], [22], [27], [30], [56], [57]. Ranging from the SRCNN [5], which has only three convolution layers (Conv layers), to the recent RCAN [56], which has over 400 layers, these approaches obviously illustrate that as the model becomes deeper, the performance improves.

Although CNN-based models have achieved state-of-the-art performance, these methods face some limitations.

1) Most CNN-based frameworks gain improvement by substantially increasing the depth or width of the network; this means that they rely heavily on computation to produce the HR images, limiting their real-world applications.
2) Most existing CNN-based SR models seldom utilize the multiscale representation for image super resolution and do not fully use the hierarchical features.

Consequently, it is important to design a lightweight architecture that is practical to solve the mentioned problems. The general way to build a lightweight network is to reduce the number of model parameters and computational operations (multiadds). Based on this concept, we provide a feasible solution for the challenge that combines the multiscale mechanism and the dense connection. Specifically, an efficient feature extraction network (EFEN) is proposed for exploring feature maps, and an upsampling network (UN) is used for enlarging

Manuscript received June 23, 2019; revised November 5, 2019; accepted January 17, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61702129, Grant 61772149, Grant U1701267, and Grant 61866009, in part by the National Key Research and Development Program of China under Grant 2018AAA0100305, in part by the China Postdoctoral Science Foundation under Grant 2018M633047, in part by the Guangxi Science and Technology Project under Grant 2019GXNSFAA245014, Grant AD18281079, Grant AD18216004, Grant 2017GXNFDA198025, and Grant AA18118039, and in part by the Innovation Project of GUET Graduate Education under Grant 2019YCXS048. This article was recommended by Associate Editor H. Lu. *(Corresponding author: Zhenbing Liu.)*

Rushi Lan is with the Guangxi Key Laboratory of Image and Graphic Intelligent Processing, Guilin University of Electronic Technology, Guilin 541004, China, and also with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China.

Long Sun, Zhenbing Liu, and Cheng Pang are with the Guangxi Key Laboratory of Images and Graphics Intelligent Processing, Guilin University of Electronic Technology, Guilin 541004, China (e-mail: zbliu2011@163.com).

Huimin Lu is with the Department of Mechanical and Control of Engineering, Kyushu Institute of Technology, Kitakyushu 8048550, Japan.

Xiaonan Luo is with the National Local Joint Engineering Research Center of Satellite Navigation and Location Service, Guilin University of Electronic Technology, Guilin 541004, China.

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2020.2970104

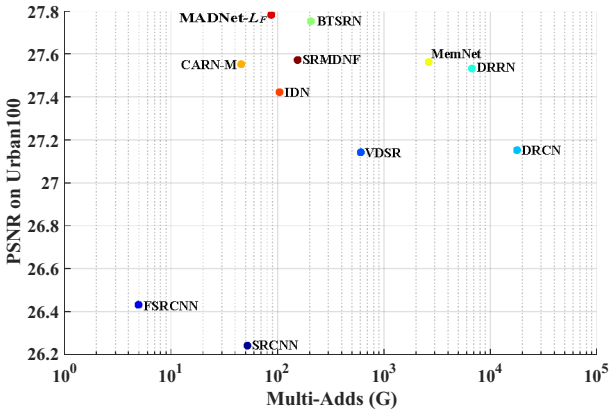


Fig. 1. Multiadds versus PSNR. Comparison between our MADNet model and other advanced lightweight networks on the Urban100 test set (×3). Our MADNet model outperforms other methods. The multiadds are calculated by assuming that the size of the output image is 1280 × 720.

features. The EFEN subnet is the key part of our method. To build this module, we introduce a residual multiscale module with an attention mechanism (RMAM) for better multiscale feature correlation learning. Our RMAM adaptively exploits the discriminative information at different scale spaces. Such a mechanism allows our model to focus on more informative features and enhance the multiscale representation ability. Moreover, for propagating the feature and gradient data, a dual residual-path block (DRPB) is proposed. By stacking the DRPB, we can utilize the hierarchical features from LR images. In addition, we employ a dense connection structure for incorporating features from various layers, which can make full use of multilevel features. As shown in Fig. 1, our network obtains state-of-the-art reconstruction results with fewer multiadd operations.

In summary, our main contributions are listed as follows.

1) We propose an RMAM that can not only effectively extract multiscale features but can also utilize the discriminative information among different channels.
2) We introduce a residual learning-based block, called DRPB, to map the low-level feature to high-level space and gathers more information to the greatest extent possible.
3) We employ a dense connection structure among DRPBs that can integrate multilevel features such as those at local or global levels, and thereby enhance the representational capability.

The remainder of this article is organized as follows. In Section II, we briefly review the relevant works on the proposed method. In Section III, we provide the architecture of our proposed model in detail and discuss the relation between the state-of-the-art models and our proposed one. In Section IV, we show the implementation details and datasets, as well as an ablation study and experimental results. Finally, we conclude the proposed methods in Section V.

II. Related Works

Single-image super resolution has been broadly studied for many years. In this section, we briefly introduce some works that are related to our proposed model.

A. CNN-Based Lightweight Super-Resolution Networks

CNN-based SISR models [4], [5], [7], [8], [18], [24], [27], [56], [57] have shown dramatic improvements in recent years given their powerful nonlinear representation ability. Dong *et al.* [5], [6] first introduced a shallow CNN-based method called SRCNN, which only contains three Conv layers and obtains impressive performance. The input image of SRCNN, however, is a bicubic-interpolated image that reduces high-frequency information and adds a relative amount of computational cost and time. Later, to reduce the computational cost caused by the preprocessed input, FSRCNN [7] and ESPCN [32] explored two different upsampling approaches: 1) the deconvolution layer [52] and 2) the subpixel convolution layer. In their networks, they enlarge images at the end part of the models and thus trim down the number of parameters and operations.

Meanwhile, VDSR [18] employs the global residual learning to train a very deep network, providing proof of the fact that increasing the depth of the network can improve the reconstruction performance. Subsequently, an increasing number of works have been mainly concerned about improvement by designing more complex CNN architectures. For example, by combining residual learning and the channel attention mechanism, Zhang *et al.* [56] proposed the RCAN model, which has more than 400 layers and can achieve great SR performance. However, increasing the reconstruction performance by increasing the model complexity with a deeper network is not free: it comes at the cost of a tremendous increase in computational resources and time. Furthermore, this approach limits real-world applications [1]. Thus, it is still a challenging task to build lightweight SR networks [42].

The Laplacian pyramid super-resolution network (LapSRN) [21] has been introduced to address the speed and accuracy of the SR problem, which takes the LR image as input and progressively reconstructs the subband residuals of HR images. DRRN [37] shares the parameters through a recursive mechanism to not only reduce the number of parameters but also improve the reconstruction accuracy. Ahn *et al.* [1] proposed an architecture that conducts a cascading mechanism upon a residual network to achieve lightweight and efficient reconstruction. Hui *et al.* [15] designed a novel information distillation network (IDN) to maintain the speed of real-time reconstruction.

B. Multiscale Representations

Multiscale feature representations have been widely used in a large number of visual tasks, such as image classification [34]–[36], object detection [28], semantic segmentation [51], and image super resolution [26]. GoogLeNet [35] uses parallel filters with different kernel sizes to enhance the multiscale representation capability in order to find an optimal local sparse structure. After that accomplishment, the improved Inception versions [34], [36] were designed, stacking more filters in each branch of the parallel paths to further expand the receptive field. Moreover, Res2Net [9] currently exhibits a new module to further improve the multiscale feature representation ability of CNNs.

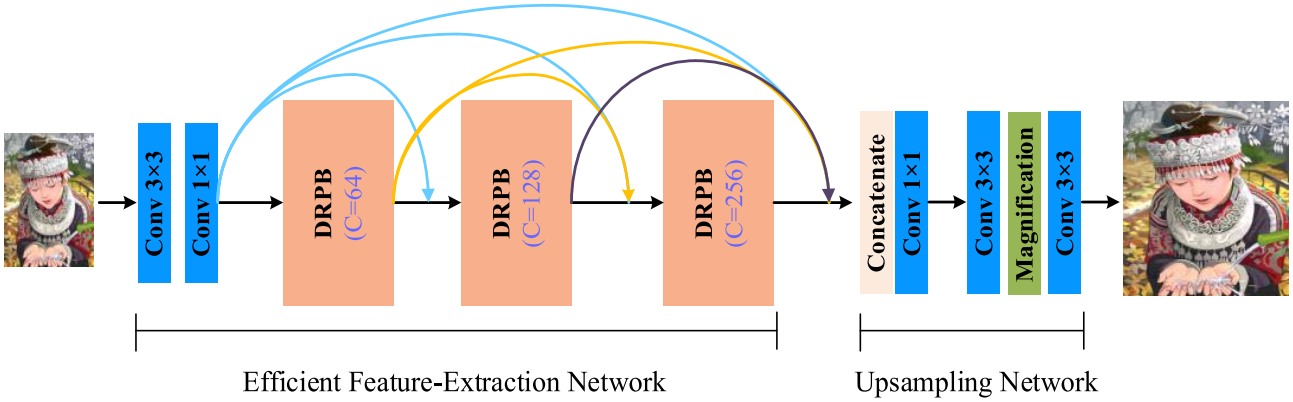


Fig. 2. Architecture of our proposed model (MADNet), which contains two subnetworks: an EFEN and a UN. The former includes three DRPBs; the latter is constructed by three sets of Conv layers and a pixel-shuffle layer.

In the Res2Net module, the input features are divided into several groups, and each group of the parallel groups utilizes a smaller filter to extract features and connects with others via residual shortcuts.

Recently, Li *et al.* [26] introduced a multiscale residual network to exploit the image features to achieve a significant performance gain for image super resolution. However, they simply concatenate the information with two different filter sizes while ignoring the granular-level multiscale feature and thus cannot cover a large range of receptive fields and cause a computational burden. Importantly, for image SR, features with more multiscale information are more accurate for reconstruction, while an SR model with fewer parameters is more feasible for real applications.

C. Attention Mechanism

Attention in human perception refers to how visual systems adaptively exploit a sequence of information items and selectively focus on salient areas [12]. Recently, several attempts have introduced attention processing to improve the performance of CNNs for various computer vision tasks [12], [43], [47], [56].

Hu *et al.* [12] employed an attention module to exploit the interchannel relationship. In their work, the squeeze-and-excitation (SE) module utilizes global average-pooled features to calculate channelwise attention and achieves considerable improvement for image classification. Woo *et al.* [47] further exploited this schema for both spatial and channelwise attention. In addition, Wang *et al.* [43] proposed a novel attention block for video classification in which nonlocal operations are used to capture spatial attention.

III. Methodology

In this section, we first present the network framework of MADNet in detail, and then suggest the multiscale module, which is the core of the proposed method. After that, the loss functions are illustrated and the discussions among the proposed method and other related algorithms are provided at the end of this section.

A. Network Framework

As shown in Fig. 2, the proposed MADNet consists of two components: 1) an EFEN and 2) a UN.

The EFEN utilizes two successive Conv layers with kernel sizes of 3×3 and 1×1 for simply detecting low-level features from the input image. Then, to extract the global and local image features, the output is fed to the DRPBs, and all the results of the intermediary block are connected to the following block as dense connections. Let I_{LR} represent the original input image and I_{SR} be the output; then, this stage can be formulated as

$$F_{\mathrm{FEA}} = H_{\mathrm{EFEN}}(I_{\mathrm{LR}}) = H_{\mathrm{DRPB}}(H_{\mathrm{LL}}(I_{\mathrm{LR}})) \tag{1}$$

where $H_{\mathrm{EFEN}}(\cdot)$ is the feature extraction function and can be divided into the shallow feature extraction step $H_{\mathrm{LL}}(\cdot)$ and the representation learning step $H_{\mathrm{DRPB}}(\cdot)$. F_{FEA} denotes the output feature map from EFEN.

Finally, we concatenate all of the feature maps for further feature fusion. After fusing, these features are processed by two Conv layers and a pixel-shuffle layer to generate the HR image

$$I_{\mathrm{SR}} = H_{\mathrm{UP}}(F_{\mathrm{FEA}}) = H_{\mathrm{GEN}}(H_{\mathrm{CON}}(F_{\mathrm{FEA}})) \tag{2}$$

where $H_{\mathrm{UP}}(\cdot)$ denotes the upsampling procedure and contains two stages: 1) $H_{\mathrm{CON}}(\cdot)$ means feature concatenation and fusion and 2) $H_{\mathrm{GEN}}(\cdot)$ represents the subsequent processing.

B. Efficient Feature Extraction Network

We now describe our EFEN (see Fig. 2) in detail. It is stacked with two Conv layers and three DRPBs, while a single DRPB gains a sequence of our proposed residual module, that is, it operates with the multiscale module and attention mechanism. The details regarding this structure are presented as follows.

DRPB: The DRPB contains $M = 3$ proposed multiscale modules. To utilize different level features and enhance the representation capability of our model, we adopt a dense connection structure for the EFEN, that is, the dth DPRB relays intermediate features to all of the next blocks. The mth multiscale module [see Fig. 3(c)] in the dth DPRB can be represented as

$$F_{d,m} = H_{d,m}\left(F_{d,m-1}\right) \tag{3}$$

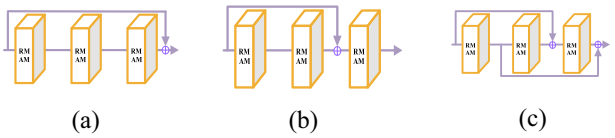


Fig. 3. Exploring different residual forms. We compare the performance of these structures in terms of PSNR, and experimentally show that the dual residual-path schema is more effective to extract features. (a) RPB_1. (b) RPB_2. (c) DRPB.

where $H_{d,m}$ denotes the function of the mth multiscale module in the dth DPRB, and $F_{d,m}$ and $F_{d,m-1}$ are the corresponding output and input. To gain more informative features, the dual residual path is used to generate the block output

$$F_d = F_{d,m-1} + F_{d,m+1}\left(F_{d-1} + F_{d,m}\right) \tag{4}$$

where F_{d-1} and F_d are the outputs of the $(d-1)$th and dth DPRB, respectively. Such a connection schema allows more low-frequency information to be bypassed during training. In fact, to confirm the effectiveness of this combination form, we compare several types of residual blocks and elaborate on the details in Section IV.

RMAM: Multiscale representations are essential for various vision tasks [9], such as semantic segmentation, object detection, and image classification. The multiscale feature extraction ability of CNNs leads to effective representations. In addition, we focus on solving the efficiency limitation that is essentially presented in real-world SR applications. To balance the performance and computational budgets, the channel split strategy is introduced in the residual layer. Meanwhile, the channel attention mechanism [12] is employed to learn discriminative representations. It was empirically found that our multiscale module is not only efficient but also accurate.

Multiscale Structure: Most previous CNN-based SR models do not consider multiscale representations. To exploit such information, MSRN [26] was introduced to detect features at different scales for accurate super-resolution construction. However, the receptive fields within MSRN are limited, and the computational complexity is fairly higher. Inspired by Inception [34] and RFB [28], we propose a multiscale module [see Fig. 4(d)] to learn the multiscale representation ability.

First, we apply a 1×1 Conv to reduce the dimension of the input data for lessening computational burden and then send them to the following four parallel branches. Except for the left (i.e., it includes a 3×3 convolution layer), other branches contain two normal convolutional layers (e.g., 1×1, 3×3) and a depthwise convolution with a dilation rate $r = 2, 3$, and 5, respectively, denoted by $\text{MS}(\cdot)$. These smaller filters first obtain features from the processed input feature maps f_i and then use a large range of receptive fields to describe the information. Specifically, the output of the previous branch is connected to the next branch via an elementwise sum. This procedure is repeated several times until the outputs from all branches are processed. This procedure can be defined as

$$F_i = \begin{cases} \text{Conv}_3(f_i) & i = 1 \\ \text{MS}_i(f_i) & 1 < i \le 4 \end{cases} \tag{5}$$

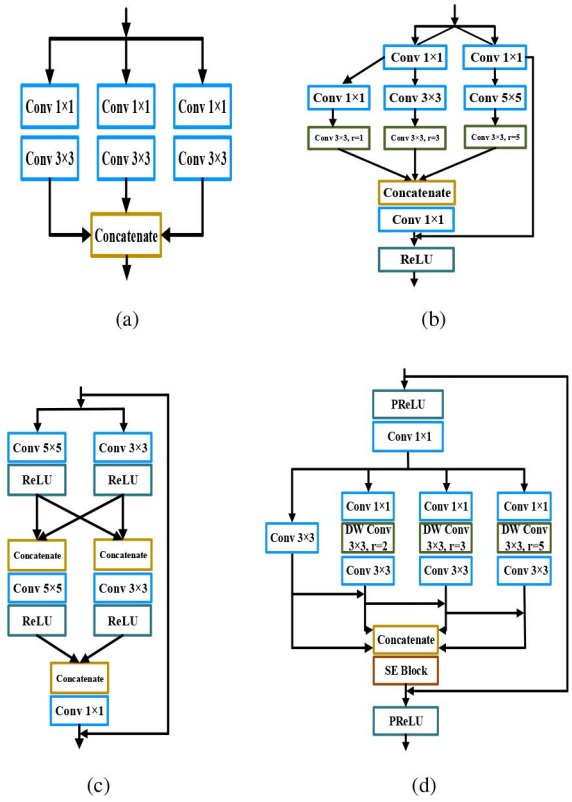


Fig. 4. Comparison of different multiscale modules. From top to bottom are: (a) inception module (simplified form) [34], (b) RFB module [28], (c) MSRB module [26], and (d) RMAM module.

where $\text{Conv}_3(\cdot)$ denotes the process of the left branch, and F_i is the output. Then

$$\text{FU}_i = \begin{cases} F_i & i = 1 \\ F_1 + \cdots + F_{i-1} & 1 < i \le 4 \end{cases} \tag{6}$$

where $\text{FU}_i(\cdot)$ means the mixed features that potentially receive feature information from all preceding feature splits.

After extracting the feature maps, we fuse these features at different scale spaces. The feature maps from all branches are concatenated and sent to the SE block for exploring discriminative representations among channels. For better preserving the inherent information, the output features are then fused with the original input tensors in a residual-like manner. From our observation, this schema is useful for utilizing features at different scale spaces.

Channel Attention Mechanism: The attention mechanism is popular in numerous vision tasks since it adaptively recalibrates the channelwise feature responses by explicitly modeling interdependencies between channels [12]. Recently, this strategy was introduced to further improve CNN-based SR performance [56].

Let $V = [v_1, \ldots, v_n]$ denote the input data that contain n feature maps, and the spatial shape of each feature map is $H \times W$. Then, the statistic S_c of the cth feature map f_c is calculated as

$$S_c = H_{\text{AVGP}}(f_c) = \frac{\sum_{i=1}^{H}\sum_{j=1}^{W} f_c(i,j)}{H \times W} \tag{7}$$

where $H_{\text{AVGP}}(\cdot)$ means the global average pooling operation, and $f_c(i,j)$ represents the corresponding value of f_c.

The attention statistic of the feature f_c is

$$A_c = F(w_1\delta(w_2 S_c)) \tag{8}$$

where $F(\cdot)$ is the ReLU activation function, and $\delta(\cdot)$ represents the sigmoid function and can be treated as a gating mechanism. w_1 is the weight of a dimension-increasing layer (i.e., 1×1 convolution layer for upscaling) and w_2 denotes the weight of a dimension-reduction layer (i.e., 1×1 convolution layer for downscaling). The downscaling layer first reduces the number of input channels by a reduction factor r with w_2, activated by an activation function δ, and then upscaling to the original spatial space with w_1. The attention statistic A_c that is used to rescale the input feature map f_c

$$\hat{f}_c = A_c \cdot f_c. \tag{9}$$

Densely Connected Structure: Due to our DRPB and the multiscale module, the information can be perceived from very different scales. To go a further step to assimilate multilevel features, we densely connect each DRPB. The mth block DPRB_m (see Fig. 2) can be represented as

$$\text{DPRB}_m = \text{Concate}(H_{\text{LL}}, \text{DPRB}_1, \ldots, \text{DPRB}_{m-1}). \tag{10}$$

Concatenating the preceding features as the input of DPRB_m, the output is also connected to the subsequent block employing the same process. Such a dense connection structure [13] allows more abundant low-frequency information to be bypassed during training.

C. Upsampling Network

As stated in Section II, our proposed model directly processes original input images so that it can extract features efficiently. The final high-quality image I_{SR} is reconstructed in the UN, and all of the features from EFEN are concatenated at the input layer of the UN; thus, the dimension of the input data is rather large. Therefore, we use 1×1 to reduce the input dimension before generating the HR pixels.

Then, the magnification layer reshapes the feature maps to a high-level space and outputs nine channels where each channel represents each real-valued tensor of the upsampled pixel.

D. Loss Function

We consider two types of loss functions that measure the difference between the HR output I_{SR} and its corresponding ground truth I_{GT}. The first one is the mean absolute error (MAE), also called the l_1-norm, which is formulated as follows:

$$L_1 = \|I_{\text{SR}} - I_{\text{GT}}\|_1. \tag{11}$$

Alternatively, the mean-square error (MSE) can be used; however, in previous work [27], it was experimentally found to be a poor choice to recover clear images.

Given the perception that MAE or MSE tends to lead a smooth result, we additionally introduce a total variation (TV) regularizer [10], [29] to constrain the smoothness of I_{SR}

$$\begin{aligned} L_{\text{TV}} &= \|\nabla_h(I_{\text{SR}})\|_2 + \|\nabla_v(I_{\text{SR}})\|_2 \\ &= \sum_{i,j} \sqrt{\left(I_{\text{SR}_{\text{i,j+1}}} - I_{\text{SR}_{\text{i,j}}}\right)^2 + \left(I_{\text{SR}_{\text{i+1,j}}} - I_{\text{SR}_{\text{i,j}}}\right)^2} \end{aligned} \tag{12}$$

where $\nabla_h(\cdot)$ and $\nabla_v(\cdot)$ denote the gradient operator among the horizontal and vertical direction, respectively.

Thus, the second loss function is defined as follows:

$$L_F = L_1 + \lambda L_{\text{TV}}. \tag{13}$$

We train our model with these losses, empirically finding that the L_F loss can obtain a better performance than the L_1 loss and $\lambda = 1e^{-5}$ works well. As shown in Fig. 7, the L_F loss enables our model to produce sharper SR results.

E. Relation to Other CNN Methods

Relation to Res2Net: The motivation for exploiting the multiscale potential is similar between the Res2Net [9] module and our RMAM. However, there are three main differences in our mechanism.

1) In general, Res2Net is used in high-level computer vision tasks (e.g., semantic segmentation and object recognition), and some inherent operations of this model are not suitable for image SR such as batch normalization (BN) layers, which increase the computational complexity and hinder the reconstructed performance of the network. Thus, we remove these layers.
2) The procedure of extracting features is different. In Res2Net, the input features are evenly split into several groups, and each group is processed by a corresponding 3×3 convolution except for the first part, where the convolutional output is added to the preceding feature and then fed into the next. However, in our model, we stack three convolutional layers with different kernel sizes and dilation rates for effectively extracting information. All of the previous outputs are added to the following group for integrating multiscale features.
3) For learning the discriminative representation, the SE block [12] is embedded to recalibrate the channelwise feature.

Relation to MSRN: We summarize the main differences between MSRN [26] and our MADNet. The first one is the design of the basic module. In MSRN, the multiscale residual block (MSRB) mainly combines parallel convolutions with multiscale feature fusion by residual learning [11], operating on all feature channels. Such an approach leads to heavy computations. However, our multiscale module is based on several convolutional branches and introduces a split and concatenation strategy to effectively process features and reduce the number of parameters. The second one is the activation function. MSRN uses the ReLU function, whereas we utilize the PReLU activation function. From the comparisons in Fig. 5, in the negative part, PReLU introduces a learnable parameter that can counterweigh the positive mean of the ReLU, making it slightly symmetric; moreover, previous experiments have proven that PReLU converges faster than ReLU and obtains better performance [55]. Thus, our proposed multiscale module possesses more powerful representational ability.

Relation to MemNet: MemNet [38] uses a dense block and various shortcuts. The differences in our method are listed as follows. First, Lim *et al.* trained the network with the L2 loss, while it was empirically found that training with the L1 loss

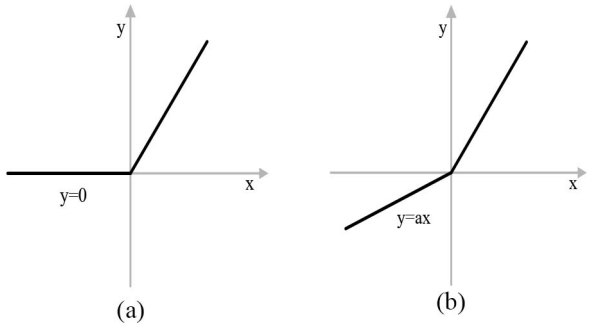


Fig. 5. (a) ReLU versus (b) PReLU. PReLU introduces a learnable parameter that can counterweigh the positive mean of the ReLU, making it slightly symmetric.

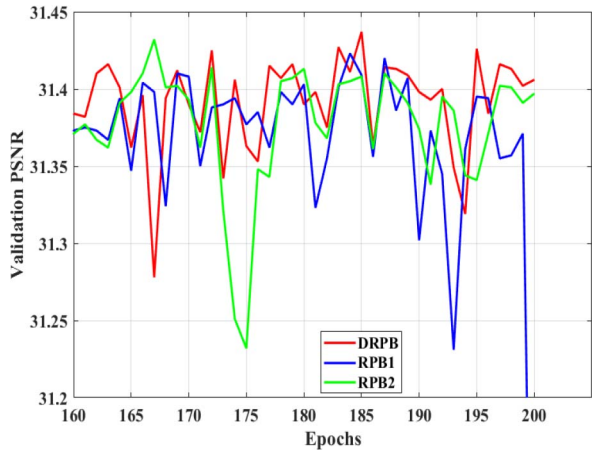


Fig. 6. Effect of MADNet with different residual structures. The curves are based on the PSNR (dB) on DIV2K (val) with an upsampling factor of 3 in 200 epochs.

provides better convergence and results than L2 [27]. In this article, we further improve the L1 loss, and the experimental results demonstrate that the modification is feasible. Second, MemNet takes the bicubic-upsampled images as input. Such input images dramatically increase the number of multiadds. However, our MADNet directly extracts hierarchical features from the original LR image and upsamples it at the end of the network in order to achieve computational efficiency and improve SR performance. Third, the components are totally different. Inside of the memory blocks of MemNet, the output features of each recursive unit are concatenated at the gate unit for fusing multilevel representations with 1×1 convolution. The analysis in [1] and [57] shows that this schema is not efficient at detecting hierarchical features. In our model, we extract multiscale feature maps via utilizing the parallel convolutional branch with different kernel sizes. Furthermore, we additionally introduce the channel attention mechanism for effectively learning channelwise feature interdependencies. Thus, our model is more powerful for feature representation.

IV. Experimental Results

In this section, we first briefly depict the experimental implementation as well as the training and testing datasets; the ablation studies follow this step. Finally, we compare our network with the state-of-the-art models on four benchmark datasets and show the visual results on different scales.

TABLE I
Effects of Different Residual Structures Measured on the Set14 × 3 Dataset in 200 Epochs

Model	Baseline	RPB_1	RPB_2	DRPB
PSNR	29.671	30.115	30.152	**30.172**

TABLE II
Results of an Ablation Study on the Effect of the SE Block. The Evaluation Is on the Set5 and B100 Test Sets

Scale	SE Block	Set5	B100
2	×	36.97	31.04
	✓	37.85	32.05
3	×	33.20	28.17
	✓	34.14	28.98
4	×	31.16	26.77
	✓	32.02	27.47

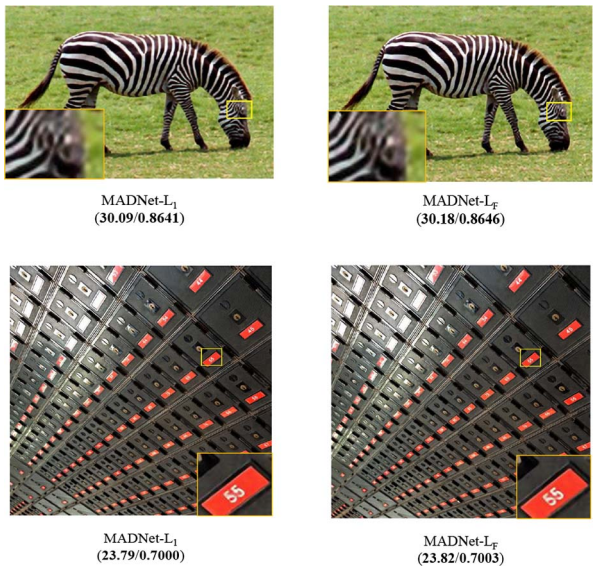


Fig. 7. Comparisons of the loss function for ×3 SR. On the top row, the "zebra" image from the Set14 dataset, the image processed with L_F has clear details in the area around the eye. On the bottom row, the "img006" image from the Urban100 benchmark, the L_F method super resolves the number "55" sharply.

TABLE III
Effect of the Loss Function With Scale Factors of ×2 and ×4 on the Set14 and Urban100 Benchmark Datasets, Respectively

Scale	Loss Function	Set14	Urban100
2	L_1	33.38/0.9161	31.62/0.9324
	L_F	33.39/0.9161	31.59/0.9233
4	L_1	31.95/0.8917	25.76/0.7746
	L_F	32.01/0.8925	25.77/0.7751

A. Training Details

As shown in Fig. 2, the input and output data of our network are RGB images. During training, in each mini-batch, we randomly crop 16 color patches with a specific size (i.e., 96×96 for ×2, 144×144 for ×3, and 192×192 for ×4) from the LR

TABLE IV
QUANTITATIVE COMPARISONS OF THE STATE-OF-THE-ART SUPER-RESOLUTION MODELS ON PUBLIC BENCHMARKS. RED/BLUE TEXT MEANS THE BEST/SECOND−BEST PERFORMANCE

Model	Scale	Params	Multi-adds	Set5	Set14	B100	Urban100
SRCNN [5]	2	57K	52.7G	36.66/0.9542	32.42/0.9063	31.36/0.8879	29.50/0.8946
FSRCNN [7]	2	12K	6.0G	37.00/0.9558	32.63/0.9088	31.53/0.8920	29.88/0.9020
VDSR [17]	2	665K	612.6G	37.53/0.9587	33.03/0.9124	31.90/0.8960	30.76/0.9140
DRCN [18]	2	1,774K	17,974G	37.63/0.9588	33.04/0.9118	31.85/0.8942	30.75/0.9133
LapSRN [20]	2	813K	29.9G	37.52/0.9590	33.08/0.9130	31.80/0.8950	30.41/0.9100
DRRN [33]	2	297K	6,796.9G	37.74/0.9591	33.23/0.9136	32.05/0.8973	31.23/0.9188
BTSRN [8]	2	410K	207.7G	37.75/-	33.20/-	32.05/-	31.63/-
MemNet [34]	2	677K	2,662.4G	37.78/0.9597	33.28/0.9142	32.08/0.8978	31.31/0.9195
SRMDNF [48]	2	1,513K	347.7G	37.79/0.9600	33.32/0.9150	32.05/0.8980	31.33/0.9200
IDN [15]	2	590K	174.10G	37.83/0.9600	33.30/0.9148	32.08/0.8985	31.27/0.9196
CARN [1]	2	1,592K	222.8G	37.76/0.9590	33.52/0.9166	32.09/0.8978	31.92/0.9256
CARN-M [1]	2	412K	91.2G	37.53/0.9583	33.26/0.9141	31.92/0.8960	31.23/0.9193
MADNet-L_1	2	878K	187.1G	37.85/0.9600	33.38/0.9161	32.04/0.8979	31.62/0.9233
MADNet-L_F	2	878K	187.1G	37.85/0.9600	33.39/0.9161	32.05/0.8981	31.59/0.9234
MADNet-$L_F{}^+$	2	878K	187.1G	37.94/0.9604	33.46/0.9167	32.10/0.8988	31.74/0.9246
SRCNN [5]	3	57K	52.7G	32.75/0.9090	29.28/0.8209	28.41/0.7863	26.24/0.7989
FSRCNN [7]	3	12K	5.0G	33.16/0.9140	29.43/0.8242	28.53/0.7910	26.43/0.8080
VDSR [17]	3	665K	612.6G	33.66/0.9213	29.77/0.8314	28.82/0.7976	27.14/0.8279
DRCN [18]	3	1,774K	17974G	33.82/0.9226	29.76/0.8311	28.80/0.7963	27.15/0.8276
DRRN [33]	3	297K	6796.9G	34.03/0.9244	29.96/0.8349	28.95/0.8004	27.53/0.8378
BTSRN [8]	3	410K	207.7G	34.03/-	29.90/-	28.97/-	27.75/-
MemNet [34]	3	677K	2662.4G	34.09/0.9248	30.00/0.8350	28.96/0.8001	27.56/0.8376
SRMDNF [48]	3	1,530K	156.3G	34.12/0.9250	30.04/0.8370	28.97/0.8030	27.57/0.8400
IDN [15]	3	590K	105.6G	34.11/0.9253	29.99/0.8354	28.95/0.8013	27.42/0.8359
CARN [1]	3	1,592K	118.8G	34.29/0.9255	30.29/0.8407	29.06/0.8034	28.06/0.8493
CARN-M [1]	3	412K	46.1G	33.99/0.9236	30.08/0.8367	28.91/0.8000	27.55/0.8385
MADNet-L_1	3	930K	88.4G	34.16/0.9253	30.21/0.8398	28.98/0.8023	27.77/0.8439
MADNet-L_F	3	930K	88.4G	34.14/0.9251	30.20/0.8395	28.98/0.8023	27.78/0.8439
MADNet-$L_F{}^+$	3	930K	88.4G	34.26/0.9262	30.29/0.8410	29.04/0.8033	27.91/0.8464
SRCNN [5]	4	57K	52.7G	30.48/0.8628	27.49/0.7503	26.90/0.7101	24.52/0.7221
FSRCNN [7]	4	12K	4.6G	30.71/0.8657	27.59/0.7535	26.98/0.7150	24.62/0.7280
VDSR [17]	4	665K	612.6G	31.35/0.8838	28.01/0.7674	27.29/0.7251	25.18/0.7524
DRCN [18]	4	1,774K	17974G	31.53/0.8854	28.02/0.7670	27.23/0.7233	25.14/0.7510
LapSRN [20]	4	813K	149.4G	31.54/0.8850	28.19/0.7720	27.32/0.7280	25.21/0.7560
DRRN [33]	4	297K	6796.9G	31.68/0.8888	28.21/0.7720	27.38/0.7284	25.44/0.7638
BTSRN [8]	4	410K	165.2G	31.85/-	28.20/-	27.47/-	25.74/-
MemNet [34]	4	677K	2,662.4G	31.74/0.8893	28.26/0.7723	27.40/0.7281	25.50/0.7630
SRMDNF [48]	4	1,555K	89.3G	31.96/0.8930	28.35/0.7770	27.49/0.7340	25.68/0.7730
IDN [15]	4	590K	81.87G	31.82/0.8903	28.25/0.7730	27.41/0.7297	25.41/0.7632
CARN [1]	4	1,592K	90.9G	32.13/0.8937	28.60/ 0.7806	27.58/0.7349	26.07/0.7837
CARN-M [1]	4	412K	32.5G	31.92/0.8903	28.42/0.7762	27.44/0.7304	25.63/0.7688
MADNet-L_1	4	1,002K	54.1G	31.95/0.8917	28.44/0.7780	27.47/0.7327	25.76/0.7746
MADNet-L_F	4	1,002K	54.1G	32.01/0.8925	28.45/0.7781	27.47/0.7327	25.77/0.7751
MADNet-$L_F{}^+$	4	1,002K	54.1G	32.11/0.8939	28.52/0.7799	27.52/0.7340	25.89/0.7782

images as input. We augment the training images via rotating by 90° and via horizontal flipping. Our model is trained by the ADAM optimizer [20] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The learning rate is initialized as $1e^{-3}$, and then reduced by half every 100 epochs for a total of 400 epochs. It takes about 15 h to train the proposed model for each magnification factor in this article. All experiments are implemented in the PyTorch framework on NVIDIA Tesla P100 with a single GPU.

B. Datasets

We train our model based on the DIV2K dataset [39], which includes 800 high-quality (2K resolution) images for the training set, and another 200 pictures for the validation and test set. During testing, we use four standard benchmark datasets: 1) Set5 [3]; 2) Set14 [53]; 3) B100 [2]; and 4) Urban100 [14], each of which has various characteristics. In detail, the Set5, Set14, and B100 datasets mainly contain images of person and natural landscapes in many

TABLE V
AVERAGE INFERENCE TIME (SECOND) AND RECONSTRUCTE PERFORMANCE. THE RESULTS ARE EVALUATED ON THE SET14, B100, AND DIV2K DATASETS FOR ×4 SR

Method	Scale	Multi-adds	Set14		B100		DIV2K	
			Time	PSNR/SSIM	Time	PSNR/SSIM	Time	PSNR/SSIM
CARN	×4	90.9G	0.0480	28.60/0.7806	0.0176	27.58/0.7349	0.1024	30.42/0.8373
CARN-M	×4	32.5G	0.0375	28.42/0.7762	0.0165	27.44/0.7304	0.0915	30.17/0.8317
MADNet-L_F	×4	54.1G	0.0455	28.45/0.7781	0.0162	27.47/0.7327	0.1117	30.26/0.8337

different scenes; the Urban100 set includes 100 urban building images in the real world. Both peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [46] results are calculated on the final SR images on the Y channel of the transformed YCbCr color space. The LR image is downscaled from the corresponding HR one using bicubic downsampling.

C. Ablation Study

To provide a better understanding of the proposed method, an ablation study is first conducted here from the following perspectives, that is, residual-path block, SE block, and loss function.

1) Study of the Residual-Path Block: Fig. 3 illustrates three different residual structures. We first conduct the ablation experiment on these structures and the corresponding results are presented in Fig. 6 and Table I. In Table I, the baseline is a plain structure without any shortcuts, the RPB_1 utilizes the residual learning between the first and last module, the RPB_2 connects the first two modules via adding shortcuts, and the DRPB is as illustrated in the previous section.

It can be seen that the block with residual learning shows better performance than the baseline because the residual path allows the earlier feature to pass into later layers. It also can be observed that the DRPB form depicts a better and stable performance as the training epochs increase. This result mainly occurs because the dual residual path effectively promotes the information propagation.

2) Study of the SE Block: To evaluate the performance of the SE block components in RMAM, we remove the SE block, such that the entire network does not take account of the attention mechanism. Observing the results shown in Table II, the attention schema can bring absolute improvements, and the PSNR value improves by approximately 0.9 and 0.8 dB on Set5 and B100, respectively.

3) Study of the Loss Function: To examine the effect of the mentioned loss functions, we trained two versions of our network. Expressed formally, let the first model be "L_1" (i.e., using L_1 loss for training) and other be "L_F" (i.e., using the enhanced L_F loss for training). We tried different linear combinations of L_1 and L_F with different weights. Moreover, it was found that $\lambda = 1e^{-5}$ achieves a tradeoff between PSNR and visual quality. Fig. 7 shows this perception that L_F loss leads to sharper images with more details. In addition, we test the performance on benchmarks. The corresponding results are illustrated in Table III. The L_F achieves better results with regard to both PSNR and SSIM. For example, L_F gains a PSNR improvement of 0.05 dB on the Set14 dataset with a scaling factor 4.

D. Comparison With State-of-the-Art Methods

We compare the proposed method with benchmark SR models on two commonly used image quality metrics, namely, PSNR and SSIM. Note that we use the number of parameters and multiadds to measure the model size. The multiadds is defined as follows [1], that is, the number of multiply accumulate operations and we assume the SR outputs size to 1280 × 720 to calculate multiadds. The geometric self-ensembling strategy [27], [41] is used for further evaluation and marked with "+" in this article. Note that we reimplement IDN [15] with PyTorch, and the official TensorFlow implementation is at https://github.com/Zheng222/IDN-tensorflow.

As shown in Fig. 1, we compare our model against the various state-of-the-art algorithms in terms of the multiadds on the Urban100 dataset with an upscaling factor of 3. Here, our MADNet method outperforms all state-of-the-art lightweight models that have less than 2M parameters. Specifically, MADNet has similar model size to those of DRCN [19], MemNet [38], and SRMDNF [54], while we achieve a better performance than all of them.

The quantitative comparisons with several state-of-the-art methods are listed in Table IV. Our model outperforms the existing models by a large margin on different scaling factors except for CARN [1]. It can be seen that although our method has quite a few parameters and multiadds, it gains completely similar performance or even better. Considering the GPU runtime, we mainly compare the proposed method with the latest CARN model and use the official codes to test their running time. As shown in Table V, our proposed model averagely spends 0.0455, 0.0162, and 0.1117 s to reconstruct an image on the Set14, B100, and DIV2K (100 validation pictures in total) datasets for scale factor 4, respectively, and totally running as fast as the CARN series.

Fig. 8 presents the visual comparisons on the B100 and Urban100 datasets for the ×4 scale. The figure shows that our method works better than other comparative ones, and the reconstructed SR images are closer to the HR ones in detail.

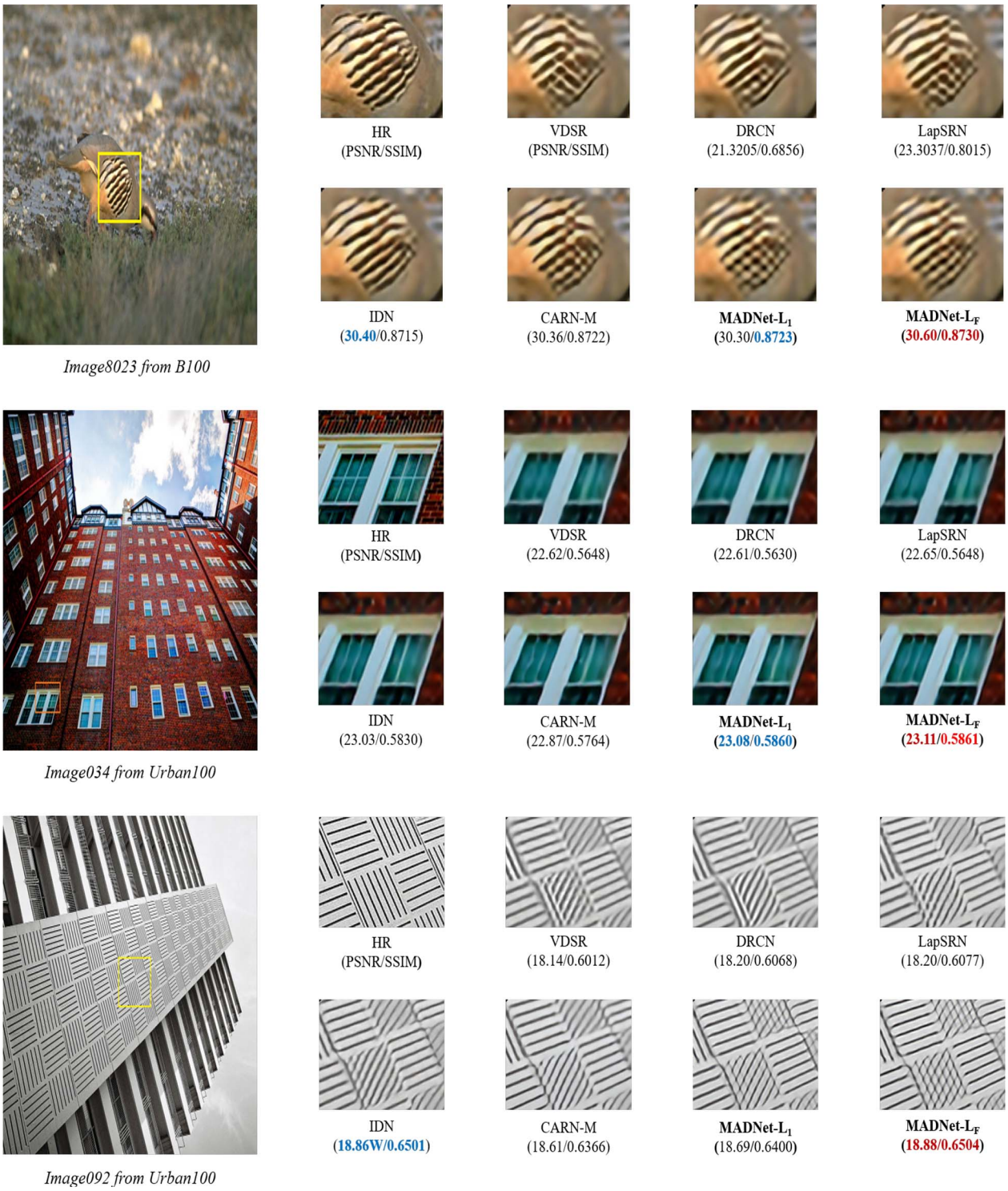


Fig. 8. Visual qualitative comparisons with the bicubic degradation model for ×4 SR on benchmarks.

V. Conclusion

We proposed a dense network of moderate size for fast and accurate image SR in this article. Specifically, the RMAM allows our model to capture the informative multiscale feature maps by embedding the channel attention operation. Furthermore, the DRPB utilizes the hierarchical features from the original LR images and allows the abundant low-frequency information to be bypassed through dense skip connections. Comprehensive benchmark evaluations show the effectiveness of our MADNet model in terms of the model size and reconstructed results.

Acknowledgment

The authors would like to greatly appreciate the reviewers' valued comments and suggestions which greatly improved this article.

REFERENCES

[1] N. Ahn, B. Kang, and K.-A. Sohn, "Fast, accurate, and lightweight super-resolution with cascading residual network," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 256–272.

[2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.

[3] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. A. Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 1–10.

[4] L. Chen, J. Pan, and Q. Li, "Robust face image super-resolution via joint learning of subdivided contextual model," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 5897–5909, Dec. 2019.

[5] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 184–199.

[6] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Jan. 2016.

[7] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Computer Vision—ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer Int., 2016, pp. 391–407.

[8] Y. Fan *et al.*, "Balanced two-stage residual networks for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops*, Jul. 2017, pp. 1157–1164.

[9] S. Gao, M. Cheng, K. Zhao, X. Zhang, M. Yang, and P. H. S. Torr, "Res2Net: A new multi-scale backbone architecture," *CoRR*, vol. abs/1904.01169, pp. 1–10, Sep. 2019. [Online]. Available: http://arxiv.org/abs/1904.01169

[10] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 1712–1722.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[12] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7132–7141.

[13] 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. 2261–2269.

[14] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5197–5206.

[15] Z. Hui, X. Wang, and X. Gao, "Fast and accurate single image super-resolution via information distillation network," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 723–731.

[16] K. Jiang, Z. Wang, P. Yi, G. Wang, T. Lu, and J. Jiang, "Edge-enhanced gan for remote sensing image super-resolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5799–5812, Aug. 2019.

[17] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 6, pp. 1153–1160, Dec. 1981.

[18] 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.

[19] 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.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2015.

[21] 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.*, 2017, pp. 624–632.

[22] R. Lan *et al.*, "Cascading and enhanced residual networks for accurate single image super-resolution," *IEEE Trans. Cybern.*, early access, doi: 10.1109/TCYB.2019.2952710

[23] R. Lan, Y. Zhou, Z. Liu, and X. Luo, "Prior knowledge-based probabilistic collaborative representation for visual recognition," *IEEE Trans. Cybern.*, early access, doi: 10.1109/TCYB.2018.2880290.

[24] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 105–114.

[25] B. Li, R. Liu, J. Cao, J. Zhang, Y.-K. Lai, and X. Liu, "Online low-rank representation learning for joint multi-subspace recovery and clustering," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 335–348, Jan. 2018.

[26] J. Li, F. Fang, K. Mei, and G. Zhang, "Multi-scale residual network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 527–542.

[27] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops*, Jul. 2017, pp. 1132–1140.

[28] S. Liu, D. Huang, and Y. Wang, "Receptive field block net for accurate and fast object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 404–419.

[29] A. Marquina and S. J. Osher, "Image super-resolution by TV-regularization and Bregman iteration," *J. Sci. Comput.*, vol. 37, no. 3, pp. 367–382, 2008.

[30] J. Pan *et al.*, "Learning dual convolutional neural networks for low-level vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 3070–3079.

[31] S. C. Park, K. K. Park, and M. G. Kang, "Super-resolution image reconstruction: A technical overview," *IEEE Signal Process. Mag.*, vol. 20, no. 3, pp. 21–36, May 2003.

[32] W. Shi *et al.*, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1874–1883.

[33] J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[34] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. ICLR Workshop*, 2016, pp. 4278–4284. [Online]. Available: https://arxiv.org/abs/1602.07261

[35] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[36] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[37] 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. 2790–2798.

[38] 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.

[39] R. Timofte *et al.*, "NTIRE 2017 challenge on single image super-resolution: Methods and results," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops*, Jul. 2017, pp. 1110–1121.

[40] R. Timofte, V. De Smet, and L. Van Gool, "A+: Adjusted anchored neighborhood regression for fast super-resolution," in *Computer Vision—ACCV 2014*, D. Cremers, I. Reid, H. Saito, and M.-H. Yang, Eds. Cham, Switzerlands: Springer Int., 2015, pp. 111–126.

[41] R. Timofte, R. Rothe, and L. Van Gool, "Seven ways to improve example-based single image super resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1865–1873.

[42] C. Wang, Z. Li, and J. Shi, "Lightweight image super-resolution with adaptive weighted learning network," 2019. [Online]. Available: arXiv:1904.02358.

[43] X. Wang, R. B. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7794–7803.

[44] Z. Wang *et al.*, "Multi-memory convolutional neural network for video super-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2530–2544, May 2019.

[45] Z. Wang, J. Chen, and S. C. H. Hoi, "Deep learning for image super-resolution: A survey," *CoRR*, vol. abs/1902.06068, pp. 1–24, Feb. 2019. [Online]. Available: http://arxiv.org/abs/1902.06068

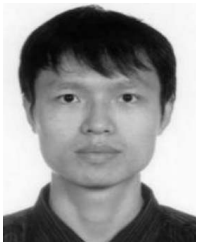
[46] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[47] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 3–19.

[48] B. Wronski *et al.*, "Handheld multi-frame super-resolution," *ACM Trans. Graph.*, vol. 38, no. 4, pp. 1–18, Jul. 2019. [Online]. Available: http://doi.acm.org/10.1145/3306346.3323024

[49] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.

[50] P. Yi, Z. Wang, K. Jiang, Z. Shao, and J. Ma, "Multi-temporal ultra dense memory network for video super-resolution," *IEEE Trans. Circuits Syst. Video Technol.*, early access, doi: 10.1109/TCSVT.2019.2925844.
[51] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. Int. Conf. Learn. Represent.*, 2016.
[52] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 2018–2025.
[53] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Curves and Surfaces*, J.-D. Boissonnat *et al.*, Eds. Heidelberg, Germany: Springer, 2012, pp. 711–730.
[54] K. Zhang, W. Zuo, and L. Zhang, "Learning a single convolutional super-resolution network for multiple degradations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 3262–3271.
[55] Y. Zhang, L. Sun, C. Yan, X. Ji, and Q. Dai, "Adaptive residual networks for high-quality image restoration," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3150–3163, Jul. 2018.
[56] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 294–310.
[57] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 2472–2481.
[58] L. Zhou, Z. Wang, Y. Luo, and Z. Xiong, "Separability and compactness network for image recognition and superresolution," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3275–3286, Nov. 2019.



Rushi Lan received the B.S. and M.S. degrees from the Nanjing University of Information Science and Technology, Nanjing, China, and the Ph.D. degree from the University of Macau, Macau, China.

He is currently an Associate Professor with the Guangxi Key Laboratory of Image and Graphic Intelligent Processing, Guilin University of Electronic Technology, Guilin, China. His research interests include image classification, image denoising, and metric learning.

Long Sun received the B.S. degree from the Yunnan University of Finance and Economics, Kunming, China, in 2018. He is currently pursuing the M.S. degree with the School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin, China.

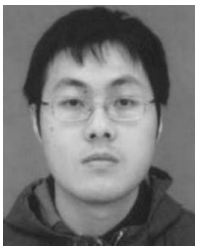
His current research interests include image/video restoration and computational photography.

Zhenbing Liu received the B.S. degree from Qufu Normal University, Qufu, China, and the M.S. and Ph.D. degrees from the Huazhong University of Science and Technology, Wuhan, China.

He was a Visiting Scholar with the Department of Radiology, University of Pennsylvania, Philadelphia, PA, USA, in 2015. He is currently a Professor and a Doctoral Supervisor with the School of Computer and Information Security, Guilin University of Electronic Technology, Guilin, China. His main research interests include image processing, machine learning, and pattern recognition.

Huimin Lu received the M.S. degrees in electrical engineering from the Kyushu Institute of Technology, Kitakyushu, Japan, and Yangzhou University, Yangzhou, China, in 2011, and the Ph.D. degree in electrical engineering from the Kyushu Institute of Technology in 2014.

From 2013 to 2016, he was a JSPS Research Fellow with the Kyushu Institute of Technology, where he is currently an Associate Professor. He is an Excellent Young Researcher with MEXT, Tokyo, Japan. His research interests include computer vision, robotics, artificial intelligence, and ocean observation.



Cheng Pang received the B.S. degree in computer science and the M.S. and Ph.D. degrees in computer technology from the Harbin Institute of Technology, Harbin, China, in 2011, 2013, and 2018, respectively.

He is currently a faculty with the Guilin University of Electronic Technology, Guilin, China. His interests include pattern recognition, image processing, machine learning, and computer vision.

Xiaonan Luo received the B.S. degree in computational mathematics from Jiangxi University, Nanchang, China, the M.S. degree in applied mathematics from Xidian University, Xi'an, China, and the Ph.D. degree in computational mathematics from the Dalian University of Technology, Dalian, China.

He is currently a Professor with the School of Computer and Information Security, Guilin University of Electronic Technology, Guilin, China. He was the Director of the National Engineering Research Center of Digital Life, Sun Yat-sen University, Guangzhou, China. He received the National Science Fund for Distinguished Young Scholars granted by the National Natural Science Foundation of China. His current research interests include computer graphics, machine learning, and pattern recognition.