Image Super-Resolution via Enhanced Multi-scale Residual Network

MengJie Wang[a], Xiaomin Yang[a,*], Marco Anisetti[b], Rongzhu Zhang[a], Marcelo Keese Albertini[c], Kai Liu[d]

[a] *College of Electronics and Information Engineering, Sichuan University, Chengdu, Sichuan, 610064, China*
[b] *Dipartimento di Informatica (DI), Università degli Studi di Milano, Via Celoria 18, Milano (MI) 20133, Italy*
[c] *Faculty of Computing, Federal University of Uberlandia, Uberlandia, Brazil*
[d] *College of Electrical and Engineering Information, Sichuan University, Chengdu, Sichuan, 610064, China*

Abstract

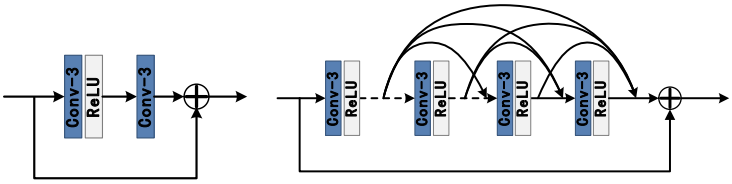
Recently, a very deep convolutional neural network (CNN) has achieved impressive results in image super-resolution (SR). In particular, residual learning techniques are widely used. However, the previously proposed residual block can only extract one single-level semantic feature maps of one single receptive field. Therefore, it is necessary to stack the residual blocks to extract higher-level semantic feature maps, which will significantly deepen the network. While a very deep network is hard to train and limits the representation for reconstructing the hierarchical information. Based on the residual block, we propose an enhanced multi-scale residual network (EMRN) to take advantage of hierarchical image features via dense connected enhanced multi-scale residual blocks (EMRBs). Specifically, the newly proposed residual block (EMRB) is capable of constructing multi-level semantic feature maps by a two-branch inception. The two-branch inception in our proposed EMRB consists of 2 convolutional layers and 4 convolutional layers in each branch respectively, therefore we have different ranges of receptive fields within one single EMRB. Meanwhile, the local feature fusion (LFF) is used in every EMRB to adaptively fuse the local feature maps extracted by the two-branch inception. Furthermore, global feature fusion (GFF) in EMRN is then used to obtain abundant useful features from previous EMRBs and subsequent ones in a holistic manner. Experiments on benchmark datasets suggest that our EMRN performs favorably over the state-of-the-art methods in reconstructing further superior super-resolution (SR) images.

Keywords:
Image super-resolution, Enhanced multi-scale residual network (EMRN), Enhanced multi-scale residual block (EMRB), A two-branch inception

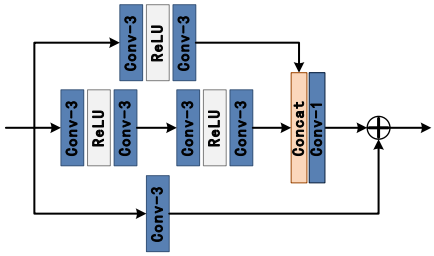
1. Introduction

A high-resolution (HR) image can be reconstructed from its correlated low-resolution (LR) observation by single image super-resolution (SISR). The SISR methods [3, 4, 5, 6, 7, 8, 9, 10]

*correspondingauthor
Email address: arielyang@scu.edu.cn (Xiaomin Yang)



(a) Residual block (b) Residual dense block



(c) Enhanced multi-scale residual block

Figure 1: Comparisons of prior residual blocks (a,b) and the proposed module EMRB (c). (a) Residual block in EDSR [1]. (b) Residual dense block in RDN [2]. (c) Our proposed enhanced multi-scale residual block.

can be well applied to various image restorations, such as image denoising, compression artifacts reduction, demosaicing, and super-resolution. While SISR is an inherently ill-posed procedure since a single LR input can reconstruct multiple HR outputs. As a result, the space of the possible functions of the mapping from LR to HR images becomes extremely large, which makes it hard to find a good solution [11]. To address this inverse problem, abundant deep neural networks for image super-resolution [2, 11, 12, 13, 14, 15, 16, 17] have been proposed. These networks aim to learn a non-linear mapping between LR and HR to reconstruct a HR image of good quality. Dong et al. [18] first developed a three-layer network, which achieved significant achievements over traditional algorithms. Kim et al. [13] first successfully used 20 layers to demonstrate that increasing depth significantly boosted performance with residual learning in VDSR. Meanwhile, Kim et al. applied recursive-supervision in DRCN [19] to make it easier to train a deeper network. An effective network model for image SR proves that the deeper the network, the better the reconstruction performance [7]. EDSR [1] built a very wide network and made a significant breakthrough in terms of SR performance by simplifying the network structure of the SRResNet [20]. The residual block in EDSR is shown in Fig 1(a). EDSR won the competition of NTIRE 2017 [21]. EDSR has about 43M parameters, 69 layers, and it takes 8 days to train this work. More recently, based on EDSR, Zhang et al. [2] introduced a residual dense network (RDN) (over 128 layers), which was built with the residual dense blocks (RDBs). RDB (Fig 1(b)) incorporated densely connected [22] convolutional layers into a residual block. Soon they proposed a very deep network RCAN [16] with more than 400 layers. Recently, Zhang et al. also proposed the residual non-local attention learning and then constructed the very deep residual non-local attention networks (RNAN) [8] for high-quality image restoration. More recently, Liu et al. [23] proposed a residual feature aggregation network (RFANet) for more efficient feature extraction. The RFANet is constructed by incorporating the proposed residual feature aggre-

Table 1: The main differences between our proposed EMRB and the residual modules proposed by several other methods

Method	The proposed residual module	Multi-scale	Residual blocks	Parameters
EDSR [1]	Residual block (RB)	×	32	43M
RDN [2]	Residual dense block (RDB)	×	16	22M
RCAN [16]	Residual channel attention block (RCAB)	×	200	16M
RFANet [23]	Residual feature aggregation (RFA) module	×	120	11M
EMRN(Ours)	Enhanced multi-scale residual block (EMRB)	✓	4	7M

gation (RFA) modules with the enhanced spatial attention (ESA) blocks. Especially, the RFA framework groups several residual modules together and adds skip connections to directly forward the features on each local residual branch [23]. Therefore, the RFA framework is able to aggregate these informative residual features to produce more powerful features. These methods have performed favorably in visual quality, but they also require a lot of time and massive graphics memory consumption in the training phases. The trend of current algorithms is to deepen convolutional neural networks (CNN) to obtain better performance [24]. However, deepening the network will make the training process difficult. Although the deep network models such as EDSR [1], RDN [2], and RCAN [16] can improve the SR performance, these methods still suffer from the large space issue of possible mapping functions and result in the limited performance [11]. More recently, Guo et al. [11] developed a dual regression scheme by introducing an additional constraint to reduce the space of the possible functions from LR to HR images. Thus, LR images can be reconstructed to enhance the performance of SR models.

As the depth of the network increases, the hierarchical information extracted by each convolutional layer will have different receptive fields. The receptive field is used to represent the range of the original images received by neurons at different locations within the convolutional neural network (CNN). The greater the value of the neuronal receptive field, the greater the range of original images it can access, which also means that it may contain more global and higher semantic information; and the smaller the value, it means the features it contains tend to be local and detailed. So the value of the receptive field can be roughly used to judge the abstraction level of each layer. Therefore, a residual block in EDSR with only one branch can only extract one single-level semantic information [25]. To get higher-level semantic information, it is necessary to stack residual blocks, which will sharply deepen the network. A very deep network can make the training process difficult, simultaneously limit the representation for reconstructing the hierarchical information.

To address these problems, we propose an enhanced multi-scale residual network (EMRN) with smaller depth to better utilize higher-level hierarchical information from LR images. Based on the residual dense block (RDB(Fig 1(b))) in RDN [2], we propose an enhanced multi-scale residual network (EMRN) with dense connected enhanced multi-scale residual blocks (EMRBs) (Fig 1(c)). Our EMRB consists of a two-branch inception and each branch in this inception is composed of 2 convolutional layers and 4 convolutional layers respectively. We have different receptive fields in one single EMRB, which is able to extract multi-level semantic information. Compared with some concurrent networks that improve multi-scale capabilities by extracting features with different resolutions, our proposed network refers to extracting multi-level features with different receptive fields in one single residual block. Table 1 shows the main differences between our proposed EMRB and the residual modules proposed by several other methods. EMRB also includes local feature fusion (LFF) and local residual learning (LRL) [2]. LFF can adaptively preserve the multi-level local feature maps extracted by the two-branch inception [25]. Moreover, LFF allows extremely high learning rates and experiments show that higher learning

rates can significantly improve the effectiveness of the network [2, 13]. Furthermore, we also use global feature fusion (GFF) [2] at the bottom of EMRN to adaptively preserve useful hierarchical information in a global manner [24]. The proposed framework EMRN aims to collect useful contextual information from a wide range of LR images so that we can better obtain sufficient knowledge to recover the details in HR images. In summary, the contributions of this article are as follows:

1. We propose an enhanced multi-scale residual network (EMRN) with smaller depth to reconstruct super-resolution images of high-quality in SISR with different scales ($\times 2$, $\times 3$, $\times 4$). Without deepening the network, the EMRN framework can also significantly make full use of hierarchical information. The proposed network EMRN converges much faster and performs favorably in reconstructing SR images with high visual quality.
2. We propose an enhanced multi-scale residual block (EMRB), which can extract multi-level semantic information with different receptive fields in one single EMRB. In the module EMRB, the concatenation of the outputs obtained by the two-branch inception is sent to a bottleneck layer, thereby the local feature maps with abundant high-level semantic information are adaptively preserved through the bottleneck layer. The proposed EMRBs can help build a wider network for stabling the training.

The remaining content is organized as follows. We briefly review the related work in Section 2. We present the architecture of the proposed network in Section 3. Experimental results and analysis are provided in Section 4.

2. Related Work

2.1. Single image super-resolution

Recently, deep learning-based methods have achieved great success against conventional ones. In this section, we only briefly review some works on single image super-resolution. Dong et al. [18] first proposed a super-resolution network (SRCNN). This network established an end-to-end mapping between the LR images and their HR counterparts. Inspired by this baseline, Kim et al. [13] proposed VDSR by stacking 20 convolutional layers with residual learning. Recursive learning was firstly introduced in DRCN [19] for parameter sharing. Later, Tai et al. introduced recursive blocks in DRRN [14] and memory blocks in Memnet [26] for deeper networks. These methods need to extract features from the interpolated LR images, which results in massive graphics memory consumption. To solve this issue, Shi et al. proposed an efficient sub-pixel convolutional layer in ESPCN [27], which was introduced to upscale the LR feature maps into the HR output at the end of the network. The efficient sub-pixel convolution layer was then adopted in many very deep networks, which have been proposed for a better performance. Lim et al. proposed a very wide network EDSR [1], which achieved a significant performance for SR by removing the batch normalization (BN) layers of the SRResNet [20]. Huang et al. introduced the dense connections between any two layers in DenseNet [22]. The dense connections were introduced among memory blocks [26] and dense blocks [15]. More recently, Zhang et al. [2] and Liu et al. [23] also used dense connections in RDN and RFANet to utilize all the hierarchical features from all the convolutional layers in the LR space.

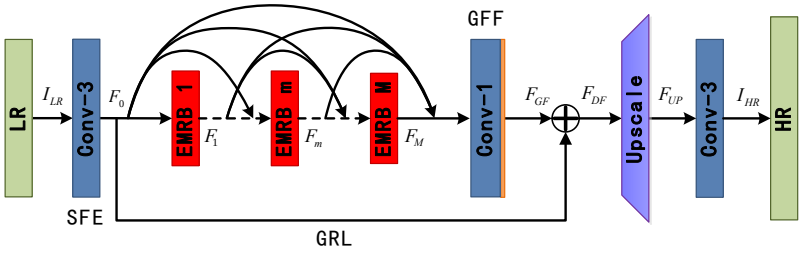


Figure 2: The architecture of our proposed enhanced multi-scale residual network (EMRN).

2.2. Multi-scale Representations

Multi-scale representation has exhibited dramatic success in a number of vision tasks [28, 29, 30, 31, 32, 33, 34]. Due to its strong robustness and generalization ability, multi-scale representation also plays an important role in the deep learning era. Lin et al. introduced feature pyramid in FPN [25] to fuse features from different depths at the end of the network for object detection tasks. PSP [35] proposed the pyramid pooling scheme to aggregate the global context information from region-based features for segmentation tasks. Sun et al. [36] proposed a well-designed network architecture that contains multiple branches where each branch has its own spatial resolution. Wang et al. adopted the similar idea in ELASTIC-Net [37] to design a replacement of residual block for ResNet [38] and thus the network is more effective to use. Multi-grid CNNs [39] proposed a multi-grid pyramid feature representation and defined the multi-grid convolutional layer (MG-Conv) operator as a replacement of convolution operator. MG-Conv is conceptually similar to OctConv[40] but is motivated for exploiting multi-scale features. Compared with MG-Conv [39], OctConv [40] adopts more efficient design to exchange inter-frequency information with higher performance.

3. EMRN for Image Restoration

3.1. Network Structure

The configuration of the proposed EMRN is depicted in Fig 2. EMRN can be constructed by four parts: shallow feature extraction (SFE), enhanced multi-scale residual blocks (EMRBs), dense feature fusion (DFF), upscale module (UPMod). Let's denote LR image I_{LR} as the input and SR image I_{SR} as the output. The low-resolution image I_{LR} is obtained by the bicubic interpolation of its corresponding high-resolution image I_{HR}. The SR image I_{SR} is the super-resolution version we want to reconstruct. According to the survey of [1, 16, 20], the shallow feature F_0 is extracted from the LR input by using only one 3×3 convolutional layer

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

where $H_{SF}(\cdot)$ represents convolution operation. F_0 is then used as the input of EMRBs and for global residual learning. Supposing EMRN contains M EMRBs, let F_{m-1} and F_m be the input and output of the m-th EMRB, and then the output F_m can be further obtained by

$$F_m = H_m(F_{m-1}) = H_m(H_{m-1}(\cdots(H_1(F_0))\cdots)), \tag{2}$$

where H_m indicates the operations of the m-th EMRB. H_m can be a composite function consisting of operations like convolution and rectified linear units (ReLU) [2]. We assume F_m consists of G_0 feature maps. $[F_0, F_1, \cdots, F_{m-1}]$ refers to the concatenation of the feature maps produced by the $(m-1)$-th EMRB. Enhanced multi-scale residual blocks $1,\cdots,(m-1)$, result in $G_0+(m-1)\times$ G feature maps (G is known as growth rate [2, 22]). In the proposed framework EMRN, short skip connections are used between an EMRB and every other EMRB. This operation preserves the feed-forward nature [2] and facilitates the flow of information. The feature reuse by short skip connections substantially reduces the number of parameters and requires less memory [15]. More details about the proposed EMRB will be shown in Section 3.2.

After we conduct a set of EMRBs to extract high-level semantic information, dense feature fusion (DFF) can be further utilized in a global manner. The DFF includes two parts: global feature fusion (GFF) and global residual learning (GRL). We utilize global feature fusion (GFF) in DFF [2] to obtain the global feature F_{GF} by adaptively fusing the output of the features from the final EMRB. And we utilize global residual learning (GRL) to take advantage of residual learning in a global way. By utilizing DFF, we can get richer semantic information. Therefore, F_{GF} can be formulated as

$$F_{GF} = W_m * H_m(H_{m-1}(\cdots H_1(F_0)\cdots)), \tag{3}$$

where W_m denotes the weight set to the 1×1 convolution of GFF, and we omit the bias term for simplicity. The weight set W_m is obtained by using the Xavier initialization method and the Xavier initialization method is a very effective method for initializing neural networks [41]. The high-level feature maps with multiple ranges of receptive fields are adaptively fused by this 1×1 convolutional layer.

Before conducting up-scaling, we utilize global residual learning (GRL) [2] in DFF to obtain the dense feature maps

$$F_{DF} = F_0 + F_{GF}, \tag{4}$$

where F_0 represents the shallow features. Before utilizing global residual learning (GRL), we conduct GFF to adaptively fuse the multi-level dense features with different receptive fields produced by the proposed EMRBs. Then the dense features F_{DF} are obtained by utilizing global residual learning (GRL). These hierarchical features with different receptive fields are then up-scaled by an upscale module

$$F_{UP} = H_{UP}(F_{DF}), \tag{5}$$

where F_{UP} denotes the upscaled features, and $H_{up}(\cdot)$ indicates an upscale module.

Inspired by [1, 2], we utilize ESPCN [27] in UPMod, which has been proven to be superior to previous up-scaling methods for SR in terms of computational complexity and obtaining better performance. Then the upscaled features are reconstructed by one 3×3 convolutional layer

$$I_{SR} = H_{REC}(F_{UP}) = H_{EMRN}(I_{LR}), \tag{6}$$

where H_{REC} and H_{EMRN} denote the final 3×3 convolutional layer of reconstruction and the operation of EMRN respectively.

Then we use L1 loss function to optimize EMRN. The L2 loss function is also one of the most widely-used optimization functions. Although it can achieve high PSNR/SSIM, the solution for

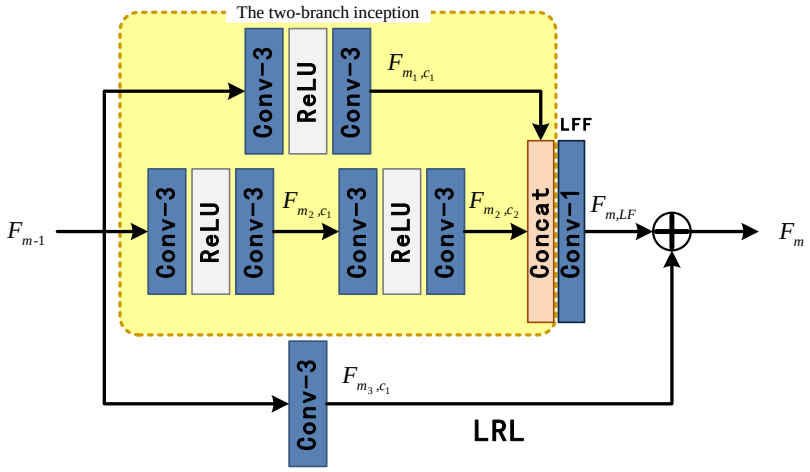


Figure 3: Details about the architecture of enhanced multi-scale residual block (EMRB).

L2 function is harder to converge in training and easier to lose detail texture information. For better and more effective results, we choose to optimize the proposed network EMRN with L1 loss function like most previous works [1, 2, 16, 42]. Given a training set $\left\{I_i^{LR}, I_i^{HR}\right\}_{i=1}^{N}$ that has N LR-HR counterparts. Thus, the loss function $L(\theta)$ of our EMRN can be defined as

$$L(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left\|H_{EMRN}(I_i^{LR}) - I_i^{HR}\right\|_1, \tag{7}$$

where θ represents the parameters of our EMRN. Then we will give more details of training in Section 4.2.

3.2. Enhanced Multi-scale Residual Block

We propose an enhanced multi-scale residual block (EMRB) for extracting multi-level semantic features with different receptive fields. More details about the architecture of EMRB are shown in Fig 3. The proposed module EMRB contains local feature fusion (LFF), and local residual learning (LRL) [2].

Local Feature Fusion. The proposed enhanced multi-scale residual block (EMRB) has a two-branch inception where each branch consists of 2 convolutional layers and 4 convolutional layers. Our EMRB is different from the previous residual block (Fig 1(a)) because the previous residual block has only one branch and can only extract one single-level semantic information. In the proposed EMRB, the low-level features extracted by the shorter branch and high-level features extracted by the longer branch are adaptively fused by a 1×1 convolutional layer at the bottom of the two-branch inception. We call this function performed in each EMRB as the local feature fusion (LFF) [2]. Therefore, EMRB can extract multi-level semantic information with different receptive fields in one single residual block. LFF can be obtained by

$$F_{m_1,c_1} = H_{m_1,c_1}(F_{m-1}), \tag{8}$$

$$F_{m_2,c_1} = H_{m_2,c_1}(F_{m-1}), \tag{9}$$

$$F_{m_2,c_2} = H_{m_2,c_2}(F_{m_2,c_1}), \tag{10}$$

$$F_{m,LF} = H_m^{LF}([F_{m_1,c_1}, F_{m_2,c_2}]), \tag{11}$$

where $H_{m,c}$ denotes a composite function consisting of convolution and ReLU. The subscript $m_i(i = 1, 2, 3)$ indicates the locations of the branches in Fig 3, and $c_i(i = 1, 2)$ denotes the number of the related operations performed in this row. $[F_{m_1,c_1}, F_{m_2,c_2}]$ refers to the concatenation operation.

Assume that the input of the first EMRB has G_0 feature maps. Due to the existence of dense connections between EMRBs, the input F_{m-1} of the m-th EMRB contains Q feature maps (Q = $G_0 + (m-1) \times G$), then the output F_{m_1,c_1} by the operation H_{m_1,c_1} of the first row has Q feature maps. The outputs F_{m_2,c_1}, F_{m_2,c_2} of the second row also have Q feature maps. After the outputs F_{m_1,c_1}, F_{m_2,c_2} are concatenated, one 1×1 convolution is used to adaptively fuse the multi-level features. We name the function of the 1×1 convolution as local feature fusion (LFF). These feature maps of concatenation contain redundant information, and if they are directly used as the input of the next EMRB, it will greatly increase the computational complexity. Therefore, the input 2Q feature maps of the 1×1 convolutional layer are reduced to the output G_0 feature maps. Meanwhile, this 1×1 convolution is extremely crucial for rebuilding a high-quality network by making full use of the multi-level features. The output of this 1×1 convolution is defined as $F_{m,LF}$ and this function is named H_m^{LF}. We find that as the network deepens, its spatial expression ability gradually decreases, but it extracts richer semantic information [24]. The proposed EMRB can extract multi-level features with different receptive fields in one single residual block and then the framework EMRN can obtain better experimental results without stacking a large number of EMRBs, which avoids a series of problems caused by deepening the network.

Local Residual Learning. The proposed module EMRB has a two-branch inception and each branch in this inception has several convolutional layers. The residual learning is used in EMRB to make the information flow better and we call this function in every EMRB as local residual learning (LRL) [2]. It can be defined as

$$F_m = F_{m_3,c_1} + F_{m,LF}, \tag{12}$$

where F_{m_3,c_1} denotes the output of the 3×3 convolution in the third row. The input F_{m-1} contains Q feature maps and is reduced into G_0 feature maps by the operation of this 3×3 convolution. This 3×3 convolution can help reduce the computational complexity caused by the Q feature maps, which has a large amount of information. F_m denotes the output of the EMRB and we get F_m by performing the element-wise addition. In other words, the implementation of LRL is performing an element-wise addition of the feature maps $F_{m,LF}$ extracted by the local feature fusion and the output F_{m_3,c_1} obtained by the 3×3 convolution in the third row. We find that the proposed network EMRN converges much faster with this local residual learning [13] and shows superior performance in the SISR performance.

Table 2: Quantitative evaluation of state-of-the-art SR methods: average PSNR/SSIM with scale factor ×2, ×3 and ×4 on datasets Set5, Set14, B100, Urban100, and Manga109. Best and second best results are highlighted and underlined.

Method	Scale	Set5 PSNR / SSIM	Set14 PSNR / SSIM	B100 PSNR / SSIM	Urban100 PSNR / SSIM	Manga109 PSNR / SSIM
Bicubic	×2	33.66 / 0.9299	30.24 / 0.8688	29.56 / 0.8431	26.88 / 0.8403	30.80 / 0.9339
SRCNN [43]	×2	36.66 / 0.9542	32.45 / 0.9067	31.36 / 0.8879	29.50 / 0.8946	35.60 / 0.9663
FSRCNN [44]	×2	37.00 / 0.9558	32.63 / 0.9088	31.53 / 0.8920	29.88 / 0.9020	36.67 / 0.9710
VDSR [13]	×2	37.53 / 0.9587	33.03 / 0.9124	31.90 / 0.8960	30.76 / 0.9140	37.22 / 0.9750
DRCN [19]	×2	37.63 / 0.9588	33.04 / 0.9118	31.85 / 0.8942	30.75 / 0.9133	37.55 / 0.9732
DRRN [14]	×2	37.74 / 0.9591	33.23 / 0.9136	32.05 / 0.8973	31.23 / 0.9188	37.88 / 0.9749
LapSRN [42]	×2	37.52 / 0.9591	32.99 / 0.9124	31.80 / 0.8952	30.41 / 0.9103	37.27 / 0.9740
MemNet [26]	×2	37.78 / 0.9597	33.28 / 0.9142	32.08 / 0.8978	31.31 / 0.9195	37.72 / 0.9740
EDSR-baseline [1]	×2	37.99 / 0.9604	33.57 / 0.9175	32.16 / 0.8994	31.98 / 0.9272	38.54 / 0.9769
SRMDNF [45]	×2	37.79 / 0.9601	33.32 / 0.9159	32.05 / 0.8985	31.33 / 0.9204	38.07 / 0.9761
EMRN (Ours)	×2	**38.07** / **0.9607**	**33.67** / **0.9177**	**32.21** / **0.8999**	**32.20** / **0.9291**	**38.56** / **0.9770**
Bicubic	×3	30.39 / 0.8682	27.55 / 0.7742	27.21 / 0.7385	24.46 / 0.7349	26.95 / 0.8556
SRCNN [43]	×3	32.75 / 0.9090	29.30 / 0.8215	28.41 / 0.7863	26.24 / 0.7989	30.48 / 0.9117
FSRCNN [44]	×3	33.18 / 0.9140	29.37 / 0.8240	28.53 / 0.7910	26.43 / 0.8080	31.10 / 0.9210
VDSR [13]	×3	33.66 / 0.9213	29.77 / 0.8314	28.82 / 0.7976	27.14 / 0.8279	32.01 / 0.9340
DRCN [19]	×3	33.82 / 0.9226	29.76 / 0.8311	28.80 / 0.7963	27.15 / 0.8276	32.24 / 0.9343
DRRN [14]	×3	34.03 / 0.9244	29.96 / 0.8349	28.95 / 0.8004	27.53 / 0.8378	32.71 / 0.9379
LapSRN [42]	×3	33.81 / 0.9220	29.79 / 0.8325	28.82 / 0.7980	27.07 / 0.8275	32.21 / 0.9350
MemNet [26]	×3	34.09 / 0.9248	30.00 / 0.8350	28.96 / 0.8001	27.56 / 0.8376	32.51 / 0.9369
EDSR-baseline [1]	×3	34.37 / 0.9270	30.28 / 0.8417	29.09 / **0.8052**	**28.15** / **0.8527**	33.45 / 0.9439
SRMDNF [45]	×3	34.12 / 0.9254	30.04 / 0.8382	28.97 / 0.8025	27.57 / 0.8398	33.00 / 0.9403
EMRN (Ours)	×3	**34.45** / **0.9273**	**30.34** / **0.8423**	**29.11** / **0.8052**	28.14 / 0.8519	**33.47** / **0.9442**
Bicubic	×4	28.42 / 0.8104	26.00 / 0.7027	25.96 / 0.6675	23.14 / 0.6577	24.89 / 0.7866
SRCNN [43]	×4	30.48 / 0.8628	27.50 / 0.7513	26.90 / 0.7101	24.52 / 0.7221	27.58 / 0.8555
FSRCNN [44]	×4	30.72 / 0.8660	27.61 / 0.7550	26.98 / 0.7150	24.62 / 0.7280	27.90 / 0.8610
VDSR [13]	×4	31.35 / 0.8838	28.01 / 0.7674	27.29 / 0.7251	25.18 / 0.7524	28.83 / 0.8870
DRCN [19]	×4	31.53 / 0.8854	28.02 / 0.7670	27.23 / 0.7233	25.14 / 0.7510	28.93 / 0.8854
DRRN [14]	×4	31.68 / 0.8888	28.21 / 0.7720	27.38 / 0.7284	25.44 / 0.7638	29.45 / 0.8946
LapSRN [42]	×4	31.54 / 0.8852	28.19 / 0.7720	27.32 / 0.7275	25.21 / 0.7562	29.09 / 0.8900
MemNet [26]	×4	31.74 / 0.8893	28.26 / 0.7723	27.40 / 0.7281	25.50 / 0.7630	29.42 / 0.8942
EDSR-baseline [1]	×4	32.09 / 0.8938	28.58 / 0.7813	27.57 / 0.7357	26.04 / 0.7849	30.35 / 0.9067
SRMDNF [45]	×4	31.96 / 0.8925	28.35 / 0.7787	27.49 / 0.7337	25.68 / 0.7731	30.09 / 0.9024
EMRN (Ours)	×4	**32.21** / **0.8950**	**28.61** / **0.7827**	**27.59** / **0.7369**	**26.07** / **0.7862**	**30.44** / **0.9085**

Figure 4: Visual comparisons for scale ×4 SR. The SR results are for image "86000" from B100.

4. Experiments

4.1. Datasets and metrics

Recently, a high-quality (2K resolution) dataset DIV2K [46] is widely used in image restoration tasks. DIV2K consists of 800 training images, 100 validation images, and 100 test images. In our experiments, we use 800 high-resolution training images from DIV2K as training set. For evaluation, we choose five standard benchmark datasets: Set5 [47], Set14 [48], B100 [49], Urban100 [50], and Manga109 [51]. The results of the super-resolution images are evaluated by PSNR and SSIM [52] metrics on Y channel of transformed YCbCr space.

4.2. Implementation details

During training, the LR images patches of size 48×48 with corresponding HR images are used as the input and the mini-batch size is set to 16. All images are pre-processed by subtracting the average RGB value of the DIV2K dataset. The parameters of ADAM optimizer [53] are setting as $\beta_1 = 0.9, \beta_2 = 0.999$, and $\varepsilon = 10^{-8}$. The initial learning rate is 10^{-4} and then decreases to half every 2×10^5 iterations. The total iterations are set to 10^6. We construct the proposed framework EMRN (benchmark model, M = 4) and EMRN_B8 (M = 8) with a scaling factor 1.0. The output of each EMRB has G = 64 feature maps. In EMRN, the convolution kernel size of all the convolutional layers is set to 3×3 except that in LFF and GFF, whose kernel size is 1×1. The shallow features at the beginning of EMRN are extracted by one 3×3 convolution. The bottleneck layers for local and global feature fusion have $G_0 = 64$ filters. We implement our models by applying PyTorch with NVIDIA GTX1080Ti. It roughly takes one day to train the proposed network.

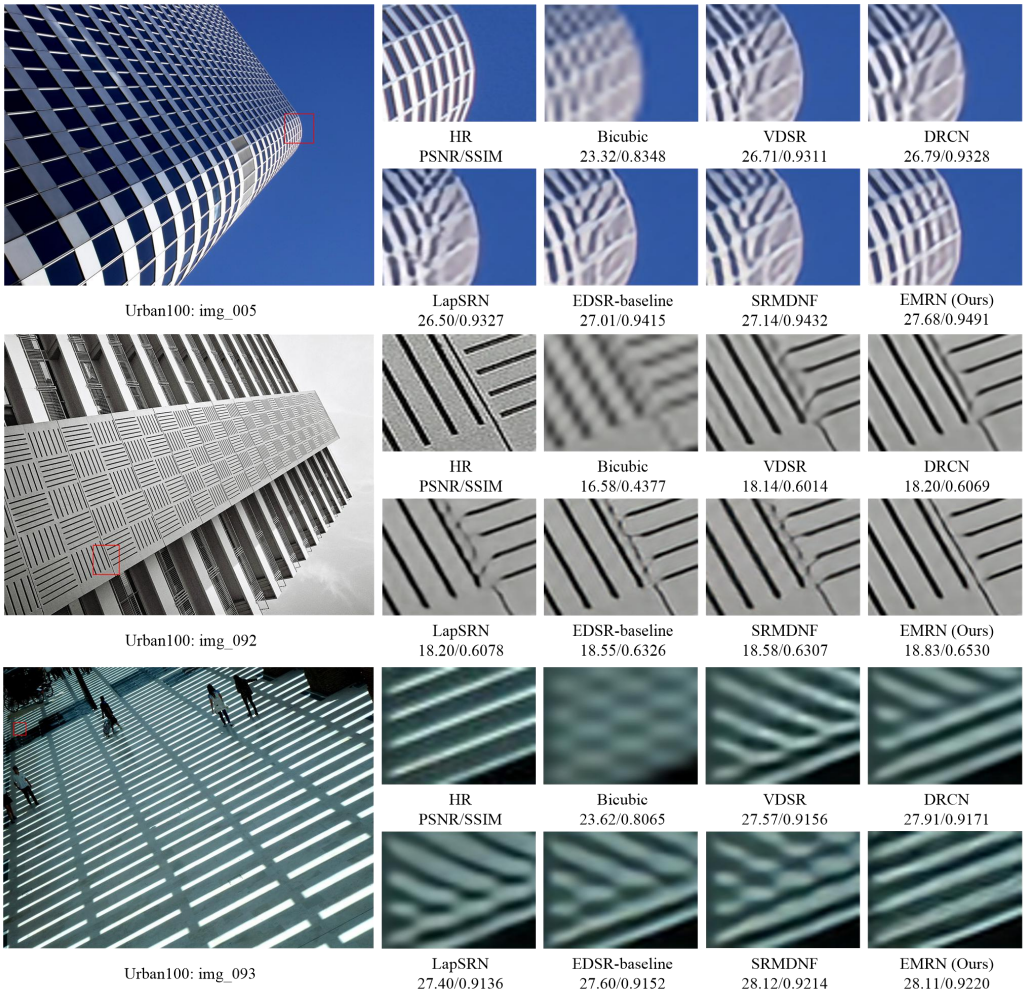


Figure 5: Visual comparisons for scale ×4 SR. The SR results are for image "img_005", "img_092" and "img_093" from Urban100 respectively.

4.3. Comparisons with the-state-of-the-arts

We compare EMRN with 9 state-of-the-art methods: SRCNN [43], FSRCNN [44], VDSR [13], DRCN [19], DRRN [14], LapSRN [42], MemNet [26], EDSR-baseline [1], SRMDNF [45].

Quantitative evaluation. Table 2 shows all the quantitative results for ×2, ×3, and ×4 SR. The results of SRCNN [43], FSRCNN [44], VDSR [13], DRCN [19], DRRN [14], LapSRN [42], MemNet [26], EDSR-baseline [1], and SRMDNF [45] are cited from IMDN [7]. In general, our EMRN outperforms the other compared methods on all the datasets with almost all scale factors. Especially for scale ×2 and ×4, EMRN achieves the best results on all the datasets. To further illustrate the effectiveness of the proposed framework, we compare the benchmark model EMRN with EDSR-baseline. When the scaling factor is ×4, the gains of our EMRN over EDSR-baseline significantly increase. For datasets Set5 and Manga109, the PSNR gains of EMRN over EDSR-baseline are 0.12dB and 0.09dB respectively. EDSR-baseline is more in-depth (37 v.s. 20), but our EMRN outperforms much better. The quantitative results prove that the proposed EMRN with dense connected EMRBs can gradually aggregate these hierarchical information to form more representative features without deepening the network, while EDSR-baseline has to deepen the network to obtain hierarchical information with multiple receptive fields by stacking residual blocks. EMRBs allow our network to provide richer semantic information and improve the performance for SR.

Visual analysis. Visual comparisons with scale factor ×4 are shown in Fig 4 and Fig 5. For img "86000", we find that most of the methods we compare cannot completely recover the grid of the window and would produce visual artifacts. However, our EMRN can better remove visual artifacts and recover the details of the grid. For "img_005", most of the methods we compare produce visible blurring artifacts at the top of the building and fail to recover the structures. Only the result produced by EMRN is closer to the ground truth image. For "img_092", we observe that at the junction of horizontal and vertical lines, all the methods we compare fail to recover the junction. In contrast, the image recovered by our EMRN is almost identical to the ground truth image. EMRN can alleviate the artifacts better. For "img_093", we can more clearly observe the effectiveness of EMRN. As we can see, all the other methods lose the right structures and produce the wrong structures, while the proposed EMRN can generate the right structures. The visual comparison results indicate that our EMRN can recover better visible structures. The multi-level semantic information extracted by these EMRBs can generate better details in reconstructing super-resolution images, and these reconstructed SR images often have better structural details and similarity.

4.4. Qualitative Analysis

Ablation Study. In this paper, the proposed framework EMRN consists of 4 dense connected enhanced multi-scale residual blocks (EMRBs). Different from the previous residual block, the proposed EMRB can extract different semantic information with multiple receptive fields in each EMRB. Therefore our EMRN can achieve comparable results even with smaller network depth. To verify the effectiveness of the proposed EMRB, the longest branch in the two-branch inception of EMRB is removed, and we call this module EMRB_NL. Then we train the framework EMRN_NL with 4 dense connected EMRB_NLs with the same environment as EMRN by 10^6 iterations. Table 3 and Fig 6 show the performance of ablation study on EMRB and EMRB_NL. Compared with EMRN_NL, EMRN achieves much higher PSNR on all scales, indicating that the two-branch inception in EMRB is very necessary and plays an important role in feature learning.

Comparison on Different Network Depths. It is well known that the deeper, the better. In our experiments, we increase the number of EMRBs to obtain better results. During calculating

Table 3: The performance of ablation study on EMRB and EMRB_NL. Average PSNR/SSIM on Set5, Set14, B100 with scale factor ×2, ×3 and ×4. Best results are highlighted.

Method	Scale	Set5 PSNR/SSIM	Set14 PSNR/SSIM	B100 PSNR/SSIM
EMRN_NL	×2	37.84/0.9598	33.38/0.9159	32.05/0.8981
EMRN		**38.07/0.9607**	**33.67/0.9177**	**32.21/0.8999**
EMRN_NL	×3	34.11/0.9247	30.13/0.8385	28.96/0.8019
EMRN		**34.45/0.9273**	**30.34/0.8243**	**29.11/0.8052**
EMRN_NL	×4	31.84/0.8903	28.41/0.7772	27.44/0.7315
EMRN		**32.21/0.8950**	**28.61/0.7827**	**27.59/0.7369**

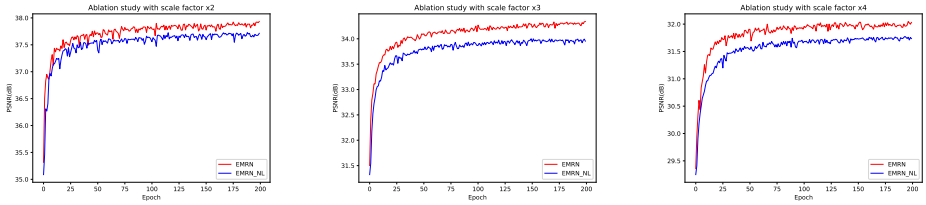


Figure 6: Ablation study of EMRB and EMRB_NL. The curves for EMRN and EMRN_NL represent the PSNR on Set5 with scale factor ×2, ×3 and ×4 in 200 epochs.

Table 4: Comparison on different network depths. Average PSNR/SSIM on Set5, Set14, B100 with scale factor ×2, ×3 and ×4. Best results are highlighted.

Method	Scale	Depth	Set5 PSNR/SSIM	Set14 PSNR/SSIM	B100 PSNR/SSIM
EMRN	×2	19	38.07/0.9607	33.67/0.9177	32.21/0.8999
EMRN_B8		35	**38.20/0.9611**	**33.85/0.9202**	**32.30/0.9011**
EMRN	×3	19	34.45/0.9273	30.34/0.8243	29.11/0.8052
EMRN_B8		35	**34.64/0.9289**	**30.48/0.8449**	**29.21/0.8080**
EMRN	×4	20	32.21/0.8950	28.61/0.7827	27.59/0.7369
EMRN_B8		36	**32.34/0.8967**	**28.75/0.7853**	**27.66/0.7391**

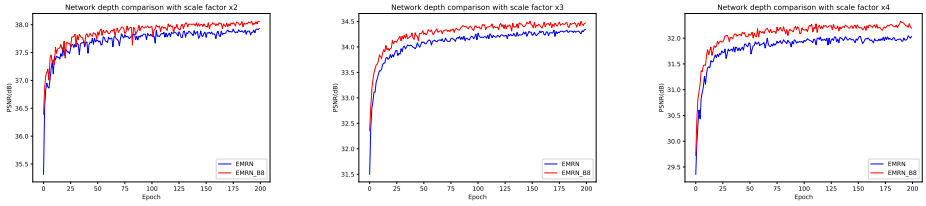


Figure 7: Comparison on different network depths. The curves for EMRN and EMRN_B8 represent the PSNR on Set5 with scale factor ×2, ×3 and ×4 in 200 epochs.

the network depth, we ignore the 1×1 convolutional layer. At the same time, because the EMRB module contains a two-branch inception, we take the longest branch as the depth of one EMRB. The experimental results show that we achieve great results when the number of EMRBs increases. Table 4 shows the average PSNR and SSIM values. The results verify that deeper is better. The PSNR gain of EMRN_B8 over EMRN is 0.13 dB with scale factor $\times 4$ on set5. Although the proposed framework EMRN with more EMRBs can achieve significant results, it will become more complicated to train as the network deepens. After weighing the performance and complexity of the network, we decide to build the most in-depth network EMRN_B8 with 8 EMRBs. Fig 7 shows the excellent results of EMRN_B8. The proposed EMRN_B8 makes a significant improvement in the SISR performance.

5. Conclusion

In this paper, we propose an enhanced multi-scale residual network (EMRN) for image SR. The EMRN framework effectively groups the enhanced multi-scale residual blocks (EMRBs) together, where the features of local residual blocks are sent directly to the end of the EMRN framework for fully utilizing these useful hierarchical features. The proposed EMRB is capable of adaptively extracting multi-level semantic information with different receptive fields. Meanwhile, the dense structure of the EMRN also allows reuse of hierarchical features from the previous EMRBs and subsequent EMRBs, which improves the flow of information between EMRBs. In the experiments, we build a more stable network with a scaling factor 1.0. The experimental results demonstrate the effectiveness of the proposed EMRN in terms of both quantitative and visual results for SR performance. In the future, we hope that the proposed network can be improved on building a lightweight network with a modest number of parameters and solving the SR problem of an arbitrary scale factor. For the future work, this approach may help to other image restoration tasks such as image-denoising and image-dehazing.

6. Acknowledgments

This work was supported by the National Natural Science Foundation of China (grant no. 61711540303 and 61701327).

References

[1] B. Lim, S. Son, H. Kim, S. Nah, K. Mu Lee, Enhanced deep residual networks for single image super-resolution, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2017, pp. 136–144.

[2] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, Y. Fu, Residual dense network for image super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2472–2481.

[3] H. Zhang, V. M. Patel, Density-aware single image de-raining using a multi-stream dense network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 695–704.

[4] H. Zhang, V. M. Patel, Densely connected pyramid dehazing network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 3194–3203.

[5] K. Li, Z. Wu, K.-C. Peng, J. Ernst, Y. Fu, Tell me where to look: Guided attention inference network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 9215–9223.

[6] H. Zhang, V. Sindagi, V. M. Patel, Image de-raining using a conditional generative adversarial network, IEEE transactions on circuits and systems for video technology (2019).

[7] Z. Hui, X. Gao, Y. Yang, X. Wang, Lightweight image super-resolution with information multi-distillation network, in: Proceedings of the 27th ACM International Conference on Multimedia, ACM, 2019, pp. 2024–2032.

[8] Y. Zhang, K. Li, K. Li, B. Zhong, Y. Fu, Residual non-local attention networks for image restoration, arXiv preprint arXiv:1903.10082 (2019).

[9] K. Zhang, L. V. Gool, R. Timofte, Deep unfolding network for image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3217–3226.
[10] Y.-S. Xu, S.-Y. R. Tseng, Y. Tseng, H.-K. Kuo, Y.-M. Tsai, Unified dynamic convolutional network for super-resolution with variational degradations, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 12496–12505.
[11] Y. Guo, J. Chen, J. Wang, Q. Chen, J. Cao, Z. Deng, Y. Xu, M. Tan, Closed-loop matters: Dual regression networks for single image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 5407–5416.
[12] Z. Hui, X. Wang, X. Gao, Fast and accurate single image super-resolution via information distillation network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 723–731.
[13] J. Kim, J. Kwon Lee, K. Mu Lee, Accurate image super-resolution using very deep convolutional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 1646–1654.
[14] Y. Tai, J. Yang, X. Liu, Image super-resolution via deep recursive residual network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 3147–3155.
[15] T. Tong, G. Li, X. Liu, Q. Gao, Image super-resolution using dense skip connections, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4799–4807.
[16] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, Y. Fu, Image super-resolution using very deep residual channel attention networks, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 286–301.
[17] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, J. Sun, Meta-sr: A magnification-arbitrary network for super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 1575–1584.
[18] C. Dong, C. C. Loy, K. He, X. Tang, Learning a deep convolutional network for image super-resolution, in: European conference on computer vision, Springer, 2014, pp. 184–199.
[19] J. Kim, J. Kwon Lee, K. Mu Lee, Deeply-recursive convolutional network for image super-resolution, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 1637–1645.
[20] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al., Photo-realistic single image super-resolution using a generative adversarial network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4681–4690.
[21] E. Agustsson, R. Timofte, Ntire 2017 challenge on single image super-resolution: Dataset and study, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 126–135.
[22] G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.
[23] J. Liu, W. Zhang, Y. Tang, J. Tang, G. Wu, Residual feature aggregation network for image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 2359–2368.
[24] J. Li, F. Fang, K. Mei, G. Zhang, Multi-scale residual network for image super-resolution, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 517–532.
[25] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117–2125.
[26] Y. Tai, J. Yang, X. Liu, C. Xu, Memnet: A persistent memory network for image restoration, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 4539–4547.
[27] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, Z. Wang, Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 1874–1883.
[28] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, in: Advances in neural information processing systems, 2015, pp. 91–99.
[29] A. Bulat, G. Tzimiropoulos, How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks), in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1021–1030.
[30] M. Najibi, P. Samangouei, R. Chellappa, L. S. Davis, Ssh: Single stage headless face detector, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 4875–4884.
[31] S. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, P. H. Torr, Res2net: A new multi-scale backbone architecture, IEEE transactions on pattern analysis and machine intelligence (2019).
[32] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE transactions on pattern analysis and machine intelligence 40 (2017) 834–848.
[33] J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng, J. Jiang, A simple pooling-based design for real-time salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3917–3926.
[34] J.-X. Zhao, Y. Cao, D.-P. Fan, M.-M. Cheng, X.-Y. Li, L. Zhang, Contrast prior and fluid pyramid integration for rgbd salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,

2019, pp. 3927–3936.
[35] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2881–2890.
[36] K. Sun, Y. Zhao, B. Jiang, T. Cheng, B. Xiao, D. Liu, Y. Mu, X. Wang, W. Liu, J. Wang, High-resolution representations for labeling pixels and regions, arXiv preprint arXiv:1904.04514 (2019).
[37] H. Wang, A. Kembhavi, A. Farhadi, A. L. Yuille, M. Rastegari, Elastic: Improving cnns with dynamic scaling policies, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2258–2267.
[38] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: European conference on computer vision, Springer, 2016, pp. 630–645.
[39] T.-W. Ke, M. Maire, S. X. Yu, Multigrid neural architectures, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6665–6673.
[40] Y. Chen, H. Fan, B. Xu, Z. Yan, Y. Kalantidis, M. Rohrbach, S. Yan, J. Feng, Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 3435–3444.
[41] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the thirteenth international conference on artificial intelligence and statistics, 2010, pp. 249–256.
[42] W.-S. Lai, J.-B. Huang, N. Ahuja, M.-H. Yang, Deep laplacian pyramid networks for fast and accurate super-resolution, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 624–632.
[43] C. Dong, C. C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE transactions on pattern analysis and machine intelligence 38 (2015) 295–307.
[44] C. Dong, C. C. Loy, X. Tang, Accelerating the super-resolution convolutional neural network, in: European conference on computer vision, Springer, 2016, pp. 391–407.
[45] K. Zhang, W. Zuo, L. Zhang, Learning a single convolutional super-resolution network for multiple degradations, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3262–3271.
[46] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, L. Zhang, Ntire 2017 challenge on single image super-resolution: Methods and results, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2017, pp. 114–125.
[47] M. Bevilacqua, A. Roumy, C. Guillemot, M. L. Alberi-Morel, Low-complexity single-image super-resolution based on nonnegative neighbor embedding (2012).
[48] R. Zeyde, M. Elad, M. Protter, On single image scale-up using sparse-representations, in: International conference on curves and surfaces, Springer, 2010, pp. 711–730.
[49] D. Martin, C. Fowlkes, D. Tal, J. Malik, et al., A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, Iccv Vancouver:, 2001.
[50] J.-B. Huang, A. Singh, N. Ahuja, Single image super-resolution from transformed self-exemplars, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5197–5206.
[51] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, K. Aizawa, Sketch-based manga retrieval using manga109 dataset, Multimedia Tools and Applications 76 (2017) 21811–21838.
[52] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, et al., Image quality assessment: from error visibility to structural similarity, IEEE transactions on image processing 13 (2004) 600–612.
[53] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).