Image Super-resolution via Residual Block Attention Networks

Tao Dai[1,2,§], Hua Zha[2], Yong Jiang[1,2], Shu-Tao Xia[1,2,‡]
[1]Graduate School at Shenzhen, Tsinghua University, Shenzhen, China
[2] PCL Research Center of Networks and Communications, Peng Cheng Laboratory, Shenzhen, China
{dait19, jiangy, xiast}@sz.tsinghua.edu.cn, zhah@pcl.ac.cn

Abstract

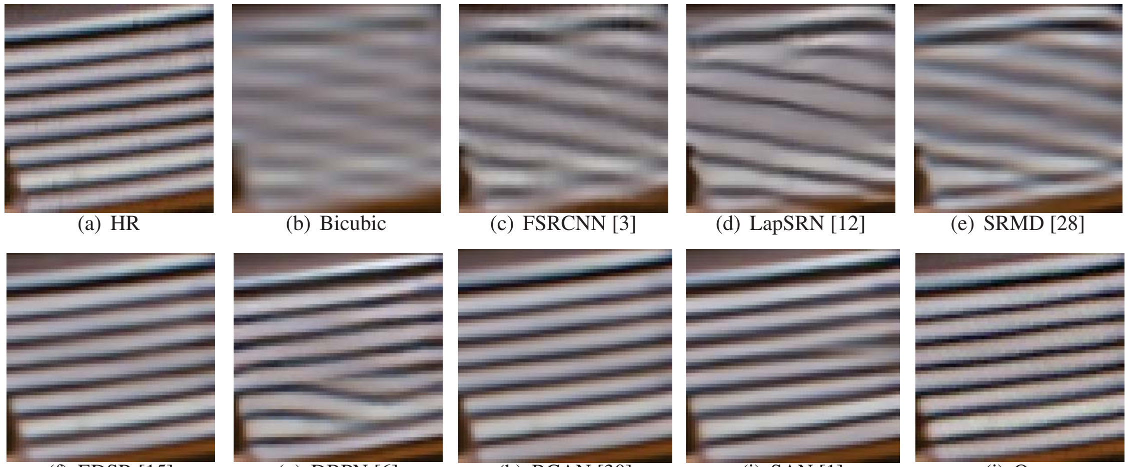
Recently, deep convolutional neural networks (CNNs) have been widely used in image super-resolution (SR). Most state-of-the-art CNN-based SR methods focus on improving the performance by designing deeper and wider networks. However, 1) using deeper networks makes the network difficult to train; 2) the relationships of features have not been thoroughly explored, therefore hindering the representational power of CNNs. In this paper, we investigate an effective end-to-end neural structure for more powerful feature expression and feature correlation learning. Specifically, we propose a residual block attention networks (RBAN) framework, which consists of two types of attention modules to efficiently exploit the feature correlations in spatial and channel dimensions for stronger feature expression. The proposed RBAN framework is constituted of a series of residual attention groups, which is further composed of several repeated residual block attention block to not only fully exploit the hierarchical features from different convolutional layers but also efficiently capture the contextual information and interdependencies among channels. Experimental results demonstrate the superiority of our RBAN network over state-of-the-art SR methods in terms of both quantitive and visual quality.

1. Introduction

Single image super-resolution (SISR) [5] recently has received much attention, whose goal is to produce a visually high-resolution (HR) output from its low-resolution (LR) input. SISR is widely used in a wide range of applications, such as medical imaging [19], face recognition [32], and depth map estimation [9]. However, image SR is an ill-posed problem, since multiple HR solutions can map to any LR input. To handle such an inverse problem, a great number of SR methods have been proposed, ranging from early interpolation-based [29] and reconstruction-based [4], to recent learning-based methods [23]. In recent years, due to the powerful feature representation and data inference ability, deep convolution neural networks (CNNs) [2, 10, 28, 31] have achieved significant performance improvement in image SR. These methods generally learn to map an interpolated or LR input to its HR output in an end-to-end manner. Most existing CNN-based SR methods focus on designing a deeper and wider neural network. Among them, Dong et al. [2] firstly introduced a shallow CNN with three layers into image SR and obtained impressive results over traditional SR methods. Later, deeper networks VDSR [10] and DRCN [11] were then proposed and obtained significant performance gain over SRCNN, mainly due to the deeper network depth (up to 20 layers). However, increasing the depth directly makes the network difficult to train. To ease this problem, He et al. [7] proposed effective residual learning strategy, which significantly eases the difficulty of training a very deep network. Such residual learning strategy was then widely used in many CNN-based SR algorithms [12, 15, 21]. Based on residual learning, Lim et al. [15] designed a very deep network EDSR (about 165 layers) by stacking simplified residual blocks. Zhang et al. [31] built a very deep residual dense network (RDN) to utilize the hierarchical features from different convolutional layers. The great performance gain on EDSR and RDN indicates the crucial importance of the depth representation for image SR.

On the other hand, many CNN-based methods exploit the effect of *attention* in CNNs to improve the performance [27, 24, 17, 25, 30, 8]. Wang et al. [24] proposed non-local neural network to compute the response at a position as a weighted sum of the features at all spatial positions. Liu et al. [16] later applied non-local neural network to image restoration task. Unlike these works that exploit the spatial correlations of features, some other work [8, 30, 25, 14] at-

‡Corresponding author: Shu-Tao Xia
§This work is supported in part by the National Key Research and Development Program of China under Grant 2018YFB1800204, the National Natural Science Foundation of China under Grant 61771273, the RD Program of Shenzhen under Grant JCYJ20180508152204044, and the research fund of PCL Future Regional Network Facilities for Large-scale Experiments and Applications PCL2018KP001).



(a) HR (b) Bicubic (c) FSRCNN [3] (d) LapSRN [12] (e) SRMD [28]

(f) EDSR [15] (g) DBPN [6] (h) RCAN [30] (i) SAN [1] (j) Ours

Figure 1. Zoom visual results for $4\times$ SR on "barbara" from Set14.

tempt to explore the channel correlation in CNNs. In [14], Li et al. proposed feedback blocks to improve feature expression. In [8], Hu et al. proposed squeeze-and-excitation (SE) block to model channel-wise relationships to obtain remarkable performance gain in image classification task. Later in [30], a very deep residual channel attention network (RCAN) based on SE block was proposed to obtain remarkable results. In [1], second-order channel attention was proposed to improve feature expression by exploiting feature statistics higher than first-order.

Inspired by the above methods, we investigate the effect of *attention* in CNNs, and propose a deep *residual block attention network (RBAN)* that explores feature correlations in both spatial and channel dimensions. Specifically, to fully utilize the representational power of CNNs, our RBAN network consists of a series of residual attention groups to formulate a deep network. To further enhance feature correlation learning, each residual attention group can be further composed of several repeated residual block attention modules to capture the spatial and channel-wise feature correlations efficiently. As shown in Fig. 1, our RBAN obtains sharper results and achieves better visual quality compared with other state-of-the-art SR methods.

2. Related Work

During the past decade, many image SISR methods have been proposed in the image processing community, including interpolation-based [29] and CNN-based methods [2, 20, 12, 11, 20, 13, 21, 31, 30]. Here, we briefly review works related to CNN-based SR methods and attention mechanism.

CNN-based SR models. In recent years, CNN-based methods have been widely studied in image SR, due to their strong power of feature expression. In general, such methods treat SR as a regression problem, and learn an end-to-end mapping from low resolution (LR) to high resolution (HR) directly. Most existing CNN-based methods mainly focus on designing a deeper or wider network structure [2, 10, 11, 6, 31, 30]. For example, Dong *et al*. [2] first introduced a shallow three-layer convolutional network (SRCNN) for image SR. Later, Kim *et al*. designed deeper VDSR [10] and DRCN [11] (more than 16 layers). To further improve the performance, Lim *et al*. [15] proposed very deep and wide networks EDSR/MDSR by stacking modified residual blocks. The significant performance gain indicates the depth and width of representation plays a key role in image SR. In addition to focusing on increasing the depth of the network, some other networks, such as NLRN [16] and RCAN [30], improve the performance by considering feature correlations in spatial or channel dimension.

Attention mechanism. *Attention* in human perception generally means that human visual systems adaptively process visual information and focus on salient regions. In recent years, several trials have embedded attention processing to improve the performance of CNNs for various tasks, such as classification tasks [8, 24]. Wang et al. [22] proposed residual attention network via trunk-and-mask attention for image classification. Another attention-based network , called non-local neural network [24], incorporates non-local operations for spatial attention in video classification. On the contrary, Hu *et al*. [8] proposed SENet to exploit channel-wise relationships to achieve significant performance gain for image classification. Some other works explore feature correlations in both spatial and channel dimensions for better performance. For example, Woo et al. [25] further proposed convolutional block attention module (CBAM) by exploiting both spatial and channel-wise feature relationships for better performance in object detection. Therefore, spatial and channel attention contribute to enhancing the discriminative ability of the network.

Motivated by the above observations, we propose a deep *residual block attention network (RBAN)* for better feature correlation for image SR.

3. Residual Block Attention Network

3.1. Network Framework

As shown in Fig. 2, our RBAN mainly consists of three parts: shallow feature extraction, residual attention group based deep feature extraction and upscaling part. Given $\mathbf{I}_{LR}$ and $\mathbf{I}_{SR}$ as the input and output of RBAN. As explored in [15, 30], we apply only one convolutional layer to extract the shallow features $\mathbf{F}_0$ from the LR input

$$\mathbf{F}_0 = H_{SF}(\mathbf{I}_{LR}), \quad (1)$$

where $H_{SF}(\cdot)$ stands for convolution operation. Then the extracted features $\mathbf{F}_0$ is used for deep feature extraction by residual attention group (RAG) based module. Thus the deep features can be obtained via

$$\mathbf{F}_{DF} = H_{RAG}(\mathbf{F}_0), \quad (2)$$

where H_{RAG} represents the RAG based deep feature extraction module, which consists of G RAG modules (see Fig. 2). The upscale module first upsamples the deep feature $\mathbf{F}_{DF}$, followed by reconstructing super-resolution (SR) image with a convolutional layer

$$\mathbf{I}_{SR} = H_R(H_{UP}(F_{DF})) = H_{RBAN}(\mathbf{I}_{LR}), \quad (3)$$

where $H_R(\cdot)$, $H_{UP}(\cdot)$ and H_{RBAN} represent the reconstruction layer, upsample layer and the function of the proposed RBAN, respectively.

During training, our RBAN is optimized with loss function. To verify the effectiveness of our RBAN, we adopt the same loss functions as previous works (e.g., L_1 loss function). Given a training set with N LR images and their HR counterparts denoted by $\{\mathbf{I}^i_{LR}, \mathbf{I}^i_{HR}\}^N_{i=1}$, the goal of training RBAN is to optimize the L_1 loss function

$$L(\Theta) = \frac{1}{N}\sum_{i=1}^{N} ||H_{RBAN(\mathbf{I}^i_{LR})} - \mathbf{I}^i_{HR}||_1, \quad (4)$$

where Θ denotes the parameter set of RBAN. The loss function is optimized by stochastic gradient descent algorithm. Our RBAN is composed of residual attention group (RAG) based deep feature extraction to enhance feature expression, which will be shown in the next section.

3.2. Residual Attention Group (RAG)

As shown in Fig. 2, residual attention group (RAG) is the basic component of RBAN, which is further composed of several repeated residual block attention modules (RBAM). A single RAG in the g-th group is represented as

$$\mathbf{F}_g = H_g(\mathbf{F}_{g-1}) = H_g(H_{g-1}(\cdots H_1(\mathbf{F}_0)\cdots)), \quad (5)$$

where H_g denotes the function of g-th RAG; $\mathbf{F}_g$ and $\mathbf{F}_{g-1}$ are the output and input of the g-th RAG. To stabilize the training of our deep network, we introduce residual learning via

$$\mathbf{F}_{DF} = \mathbf{F}_0 + \mathbf{F}_G. \quad (6)$$

To make better use of the abundant information from the LR inputs and intermediate features, we make a further step towards residual learning. As shown in Fig. 2, we stack M residual block attention modules (RBAM) in each RAG. The m-th RBAM in the g-th RAG is represented as

$$\mathbf{F}_{g,m} = H_{g,m}(\mathbf{F}_{g,m-1}) = H_{g,m}(H_{g,m-1}(\cdots H_{g,1}(\mathbf{F}_{g-1})\cdots)),$$

where $\mathbf{F}_{g,m}$, $\mathbf{F}_{g,m-1}$ and $H_{g,m}$ are the output, input and the corresponding function of the m-th RBAM in the g-th RAG. To make the network focus on more informative features, a skip connection is also introduced in each RAG, thus producing the output of g-th RAG as

$$\mathbf{F}_g = \mathbf{F}_{g-1} + \mathbf{F}_{g,M}. \quad (7)$$

With such skip connection, more abundant low-frequency information can be bypassed during training.

3.3. Spatial and Channel Attention Modules

Apart from depth and width, attention also plays a key role in the architecture design, which has been widely studied in previous works [22, 27, 30]. Attention not only tells where to focus, but also improves the representation of interests, i.e., concentrating on more informative features and suppressing unnecessary ones. For example, Zhang et al. [30] proposed a deep CNN-based SR method to explore channel-wise feature correlations. To further enhance discriminative learning of our network, we simultaneously exploit spatial and channel feature correlations with two types of attention modules.

Given an intermediate feature map $\mathbf{F} \in R^{H\times W\times C}$ of the m-th RBAM in the g-th RAG as input, our RBAM sequentially infers a spatial attention map $M_s \in R^{H\times W\times 1}$ and a channel attention map $M_c \in R^{1\times 1\times C}$. The main attention process can be formated as

$$\mathbf{F}^s = M_s(\mathbf{F}) \otimes \mathbf{F}, \qquad \mathbf{F}^{sc} = M_c(\mathbf{F}^s) \otimes \mathbf{F}^s, \quad (8)$$

where $\otimes$ denotes element-wise multiplication. Fig. 3 illustrates the computation process of each attention map.

Spatial attention (SA) module. We produce a spatial attention map $M_s(\mathbf{F})$ by utilizing the intra-spatial relationships of features. SA focuses on which part is more informative. It is known that using contextual information

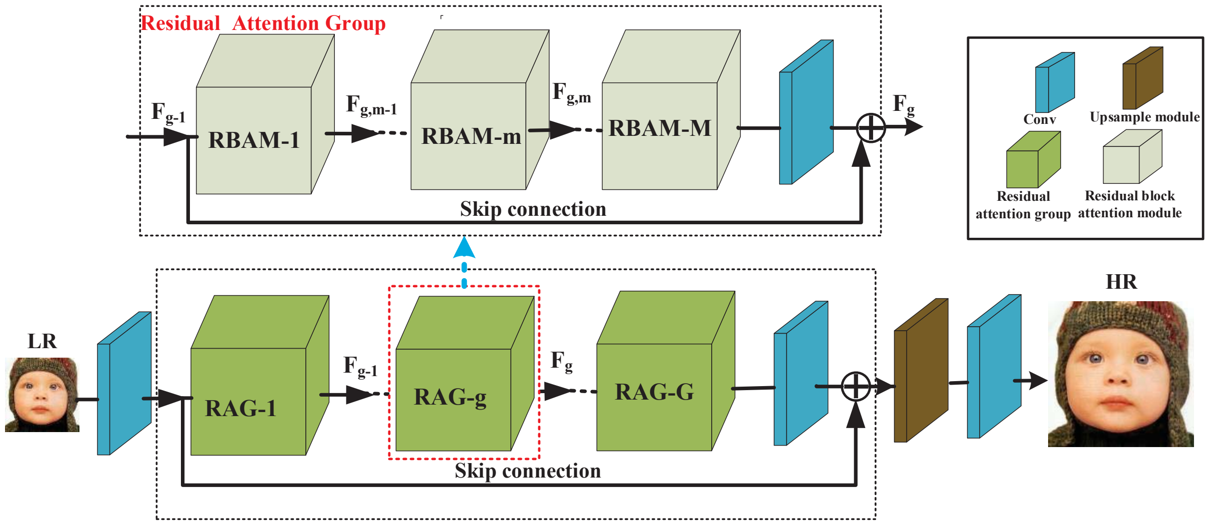


Figure 2. Network framework of our residual block attention network (RBAN)

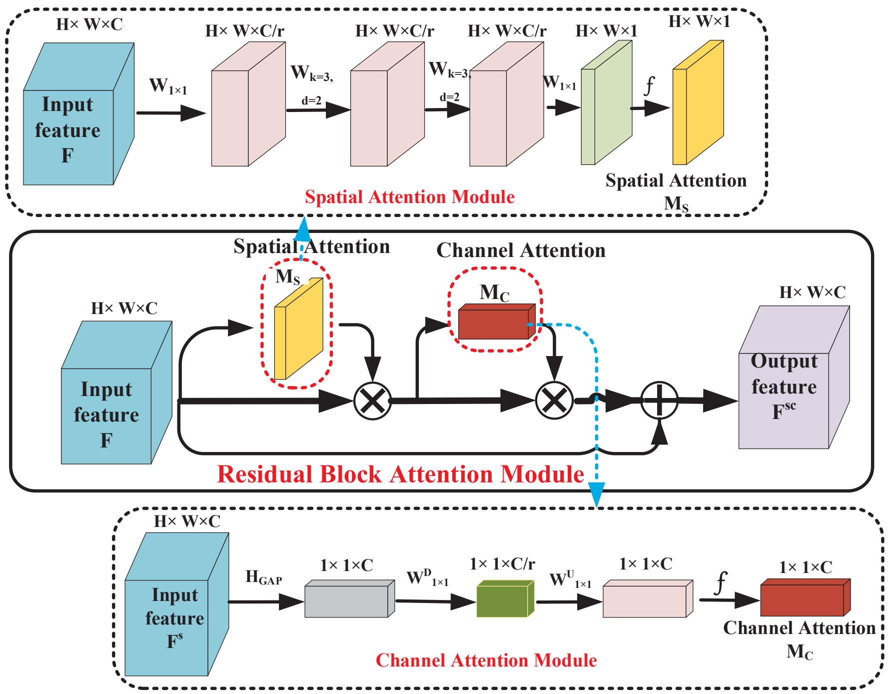


Figure 3. Diagram of residual block attention module (RBAM), which consists of a spatial and channel attention module.

is crucial to determine which spatial locations should be emphasized [26], and a large receptive field is helpful to extract much contextual information. Thus, dilated convolution is to have a larger receptive filed. To reduce the number of parameters and facilitate network training, we adopt the "bottleneck" structure as in ResNet [7]. Specifically, the feature $\mathbf{F} \in R^{H\times W\times C}$ is projected into a lower dimension $R^{H\times W\times C/r}$ with 1×1 convolution to combine and compress the feature map across the channel dimension. After the reduction, we use two 3×3 dilated convolution to acquire contextual information effectively. Finally, the features are then projected into our spatial attention map $R^{H\times W\times 1}$ with 1×1 convolution. In summary, the spatial attention map can be computed as

$$M_s(\mathbf{F}) = h_3^{1\times 1}(h_2^{3\times 3}(h_1^{3\times 3}(h_0^{1\times 1}(\mathbf{F})))), \tag{9}$$

where h represents a convolution, and the superscripts denotes the convolutional filter size.

Channel attention (CA) module. In our RBAN, the channel attention module takes the output $\mathbf{F}^s$ of spatial attention module as input. Unlike spatial attention focusing on exploiting intra-spatial relationship of features, channel attention module attempts to exploit inter-channel relationship of features. To compute the channel attention map efficiently, we first aggregate spatial information of a feature $\mathbf{F}_s$ by average pooling, thus generating a channel vector $\mathbf{F}_c \in R^{1\times1\times C}$. Then we take the channel vector $\mathbf{F}_c$ as the input of a multi-layer perceptron MLP) to estimate the channel attention map. To reduce the parameter overhead, the hidden activation size is set to $R^{1\times1\times C/r}$, where r is the reduction ratio. In summary, the channel attention map is formated as

$$M_c(\mathbf{F}) = f(\mathbf{W}_1(\mathbf{W}_0 AvgPool(\mathbf{F}_s))), \tag{10}$$

where f denotes the sigmoid function, $\mathbf{W}_1$ and $\mathbf{W}_0$ are the weight set of MLP.

Arrangement of two attention modules. Given an input image, our SA and CA modules produce complementary attention maps, which focus on intra-spatial and inter-channel relationships of features, respectively. In RBAN, we found that sequential arrangement of such attention modules achieves better results. More results about the effects of two attention modules can be found in the following sections.

4. Experiments

Table 1. Effects of different attention methods. We report the best PSNR (dB) values on Set5 ($4\times$) in 5.6×10^5 iterations.

a	b	c	d	e	f
CA	✗	✓	✗	✓	✓
SA	✗	✗	✓	✓	✓
CA→SA	✗	✗	✗	✓	✗
SA→CA	✗	✗	✗	✗	✓
	31.86	31.92	31.94	31.81	32.00

Datasets. Following [15, 31, 1], we use 800 images from DIV2K dataset [15] as training set. During training, we randomly crop RGB patches from the HR images and augment the HR patches with randomly rotation and horizontal flip. The LR patches are obtained by downsampling the HR patches with bicubic interpolation. For testing, we use five benchmark datasets: Set5, Set14, BSD100, and Urban100.

We set the number of RAG and RBAM as $R = 10, M = 10$. We train our model with Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The initial learning rate is set to 10^{-4}, and decreases to half every 200 epochs of back-propagation. Our model is implemented with Pytorch[18] framework with a NVIDIA 1080Ti GPU.

We compare our RBAN with several state-of-the-art SR methods: SRCNN [2], FSRCNN [3], VDSR[10], LapSRN [12], MemNet [21], EDSR [15], SRMD[28], DBPN [6], RDN [31], RCAN [30] and SAN [1]. The SR results are evaluated with PSNR and SSIM on Y channel of YCbCr space. Similar to [15, 30], we introduce self-ensembled one as RBAN+.

Table 2. Results of various SR methods. The best and second best values are highlighted in bold and underline in italic.

Method		Set5	Set14	BSD100	Urban100
		PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
Bicubic	×2	33.66/.9299	30.24/.8688	29.56/.8431	26.88/.8403
SRCNN	×2	36.66/.9542	32.45/.9067	31.36/.8879	29.50/.8946
FSRCNN	×2	37.05/.9560	32.66/.9090	31.53/.8920	29.88/.9020
VDSR	×2	37.53/.9590	33.05/.9130	31.90/.8960	30.77/.9140
LapSRN	×2	37.52/.9591	33.08/.9130	31.08/.8950	30.41/.9101
MemNet	×2	37.78/.9597	33.28/.9142	32.08/.8978	31.31/.9195
EDSR	×2	38.11/.9602	33.92/.9195	32.32/.9013	32.93/.9351
SRMD	×2	37.79/.9601	33.32/.9159	32.05/.8985	31.33/.9204
DBPN	×2	38.09/.9600	33.85/.9190	32.27/.9000	32.55/.9324
RDN	×2	38.24/.9614	34.01/.9212	32.34/.9017	32.89/.9353
RCAN	×2	38.27 .9614	34.12 .9216	32.41 .9027	33.34 .9384
SAN	×2	38.31 .9620	34.07 .9213	32.41 .9027	33.10 .9370
RBAN	×2	38.28/.9616	34.29/.9234	32.45/.9032	33.48/.9400
RBAN+	×2	**38.35/.9619**	**34.44/.9244**	**32.50/.9038**	**33.73/.9416**
Bicubic	×3	30.39/.8682	27.55/.7742	27.21/.7385	24.46/.7349
SRCNN	×3	32.75/.9090	29.30/.8215	28.41/.7863	26.24/.7989
FSRCNN	×3	33.18/.9140	29.37/.8240	28.53/.7910	26.43/.8080
VDSR	×3	33.67/.9210	29.78/.8320	28.83/.7990	27.14/.8290
LapSRN	×3	33.82/.9227	29.87/.8320	28.82/.7980	27.07/.8280
MemNet	×3	34.09/.9248	30.01/.8350	28.96/.8001	27.56/.8376
EDSR	×3	34.65/.9280	3.52/ .8462	29.25/.8093	28.80/.8653
SRMD	×3	34.12/.9254	30.04/.8382	28.97/.8025	27.57/.8398
RDN	×3	34.71/.9296	30.57/.8468	29.26/.8093	28.80/.8653
RCAN	×3	34.74 .9299	30.65 .8482	29.32 .8111	29.08 .8702
SAN	×3	34.75 .9300	30.59 .8476	29.33 .8112	28.93 .8671
RBAN	×3	34.83/.9302	30.66/.8485	29.32/.8111	29.09/.8703
RBAN+	×3	**34.89/.9306**	**30.77/.8498**	**29.38/.8121**	**29.29/.8730**
Bicubic	×4	28.42/.8104	26.00/.7027	25.96/.6675	23.14/.6577
SRCNN	×4	30.48/.8628	27.50/.7513	26.90/.7101	24.52/.7221
FSRCNN	×4	30.72/.8660	27.61/.7550	26.98/.7150	24.62/.7280
VDSR	×4	31.35/.8830	28.02/.7680	27.29/.0726	25.18/.7540
LapSRN	×4	31.54/.8850	28.19/.7720	27.32/.7270	25.21/.7560
MemNet	×4	31.74/.8893	28.26/.7723	27.40/.7281	25.50/.7630
EDSR	×4	32.46/.8968	28.80/.7876	27.71/.7420	26.64/.8033
SRMD	×4	31.96/.8925	28.35/.7787	27.49/.7337	25.68/.7731
DBPN	×4	32.47/.8980	28.82/.7860	27.72/.7400	26.38/.7946
RDN	×4	32.47/.8990	28.81/.7871	27.72/.7419	26.61/.8028
RCAN	×4	32.63 .9002	28.87 .7889	27.77 .7436	26.82 .8087
SAN	×4	32.64 .9003	28.92 .7888	27.78 .7436	26.79 .8068
RBAN	×4	32.64/.9003	28.93/.7907	27.80/.7447	27.03/.8132
RBAN+	×4	**32.70/.9013**	**29.05/.7921**	**27.86/.7457**	**27.23/.8169**
Bicubic	×8	24.40/.6580	23.10/.5660	23.67/.5480	20.74/.5160
SRCNN	×8	25.33/.6900	23.76/.5910	24.13/.5660	21.29/.5440
FSRCNN	×8	20.13/.5520	19.75/.4820	24.21/.5680	21.32/.5380
SCN	×8	25.59/.7071	24.02/.6028	24.30/.5698	21.52/.5571
VDSR	×8	25.93/.7240	24.26/.6140	24.49/.5830	21.70/.5710
LapSRN	×8	26.15/.7380	24.35/.6200	24.54/.5860	21.81/.5810
MemNet	×8	26.16/.7414	24.38/.6199	24.58/.5842	21.89/.5825
MSLap	×8	26.34/.7558	24.57/.6273	24.65/.5895	22.06/.5963
EDSR	×8	26.96/.7762	24.91/.6420	24.81/.5985	22.51/.6221
DBPN	×8	27.21/.7840	25.13/.6480	24.88/.6010	22.73/.6312
RCAN	×8	27.31/.7878	25.23/.6511	24.98/.6058	23.00/.6452
RBAN	×8	27.33/.7863	25.24/.6492	25.00/.6033	22.94/.6399
RBAN+	×8	**27.41/.7888**	**25.35/.6521**	**25.05/.6051**	**23.12/.6448**

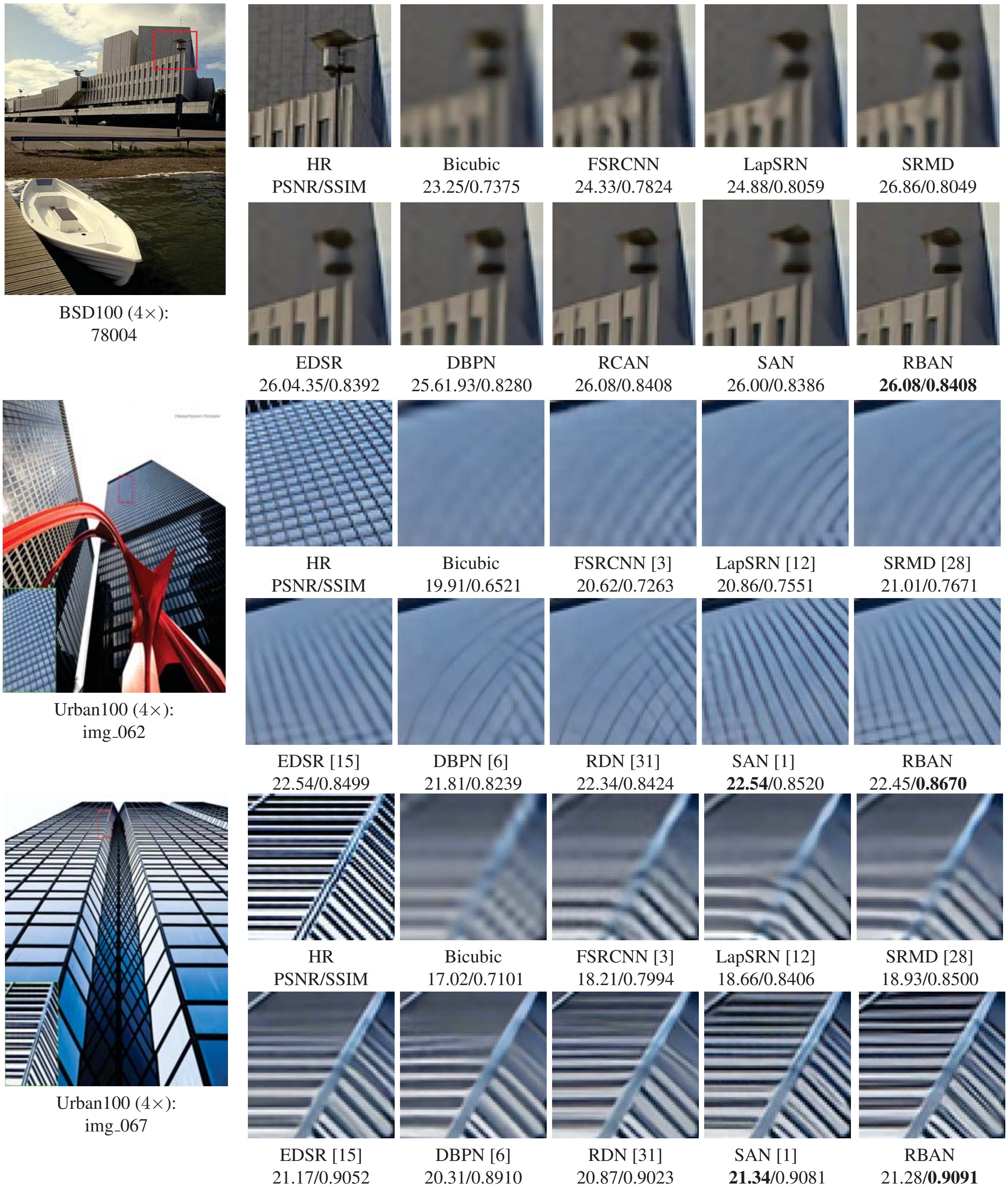


Figure 4. Visual comparison for $4\times$ SR with BI model on BSD100 and Urban100 datasets. The best results are **highlighted**

Effects of block attention. To verify the effect of the spatial attention (SA) and channel attention (CA) modules, we test our model with/without attention modules. In Table 1, when no attention modules are added, the PSNR value on Set ($4\times$) is relatively low. When only SA or CA is used, the performance of our model can be improved, which indicates the effect of attention modules. When CA and SA are used simultaneously, the arrangement of the two attention modules is crucial. As shown in Table 1, when SA is placed before CA ("e" in Table 1), the performance even decreases a little, when CA is placed before SA ("f" in Table 1), the performance is at its maximum. Thus, we use both SA and CA in our network, and arrange the SA and CA modules as "f" in Table 1. More results are reported in Table 2.

Quantitative Results. Table 2 shows the quantitative results of PSRN/SSIM values of various SR methods for $\times 2, \times 3, \times 4$, and $\times 8$ SR. From Table 2 we can see that RCAN, SAN and our RBAN obtain similar results and have much better performance than other SR methods. Specifically, Compared with previous RCAN and SAN, our RBAN+ obtains the best on all the datasets for all scaling factors. Even without self-ensemble, our RBAN achieves best results in most cases and outperforms RCAN and SAN. This is mainly because RCAN and SAN only explores feature correlations in channel dimension (channel attention), while our RBAN effectively exploits the feature correlations in both the spatial and channel dimensions for stronger feature expression (spatial and channel attention).

Visual results. In Fig. 4, we show zoomed visual results on scale $4\times$ on images "78004", "img_062" and "img_067" from BSD100 and Urban100 datasets, from which we can observe that most early proposed methods (e.g., FSRCNN and LapSRN and SRMD) suffer from blurring artifacts. The main reason is that these networks are shallow and do not explore the feature correlations. Recently developed attention-based methods (e.g., RCAN, SAN and RBAN) obtain much better visual quality, since these networks are very deep and consider inter-channel correlations with channel attention. Further compared with RCAN and SAN, our RBAN generates sharper output and recover more image details. For example, our RBAN restore the main outlines and have more faithful results (e.g., window regions in "img_62" and "img_67"), which is mainly because our RBAN explores both intra-spatial and inter-channel correlations simultaneously. These observations verify the effectiveness of our RBAN.

5. Conclusion

We propose deep residual block attention network (RBAN) for accurate image SR. Specifically, residual attention group (RAG) structure formulate our RBAN to be a very deep network. Meanwhile, RAG with skip connection allows abundant low-frequency information to be bypassed, making our RBAN concentrate on learning high-frequency information. For more powerful feature correlation learning, we propose spatial and channel attention modules to exploit intra-spatial and inter-channel correlations of features. Extensive experiments demonstrate the effectiveness of our RBAN.

References

[1] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *CVPR*, pages 11065–11074, 2019.

[2] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, pages 184–199. Springer, 2014.

[3] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *ECCV*, pages 391–407. Springer, 2016.

[4] Weisheng Dong, Lei Zhang, Guangming Shi, and Xiaolin Wu. Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *TIP*, 2011.

[5] William T Freeman, Egon C Pasztor, and Owen T Carmichael. Learning low-level vision. *IJCV*, 40(1):25–47, 2000.

[6] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. Deep backprojection networks for super-resolution. In *CVPR*, 2018.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[8] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018.

[9] Tak-Wai Hui, Chen Change Loy, and Xiaoou Tang. Depth map super-resolution by deep multi-scale guidance. In *ECCV*. Springer, 2016.

[10] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *CVPR*, pages 1646–1654, 2016.

[11] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *CVPR*, 2016.

[12] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate superresolution. In *CVPR*, 2017.

[13] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017.

[14] Zhen Li, Jinglei Yang, Zheng Liu, Xiaomin Yang, Gwanggil Jeon, and Wei Wu. Feedback network for image super-resolution. In *CVPR*, pages 3867–3876, 2019.

[15] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, volume 2, 2017.

[16] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. In *NIPS*, 2018.

[17] Jongchan Park, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Bam: Bottleneck attention module. *arXiv preprint arXiv:1807.06514*, 2018.

[18] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[19] Wenzhe Shi, Jose Caballero, Christian Ledig, Xiahai Zhuang, Wenjia Bai, Kanwal Bhatia, Antonio M Simoes Monteiro de Marvao, Tim Dawes, Declan O'Regan, and Daniel Rueckert. Cardiac image super-resolution with global correspondence using multi-atlas patchmatch. In *MICCA*, pages 9–16. Springer, 2013.

[20] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *CVPR*, 2017.

[21] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *CVPR*, pages 4539–4547, 2017.

[22] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *CVPR*, 2017.

[23] Shenlong Wang, Lei Zhang, Yan Liang, and Quan Pan. Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis. In *CVPR*, 2012.

[24] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018.

[25] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, 2018.

[26] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[27] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*, 2018.

[28] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *CVPR*, 2018.

[29] Lei Zhang and Xiaolin Wu. An edge-guided image interpolation algorithm via directional filtering and data fusion. *TIP*, 2006.

[30] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, 2018.

[31] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *CVPR*, 2018.

[32] Wilman WW Zou and Pong C Yuen. Very low resolution face recognition problem. *TIP*, 21(1):327–340, 2012.