Embedded Block Residual Network: A Recursive Restoration Model for Single-Image Super-Resolution

Yajun Qiu *
Yunnan University
qyjyun@gmail.com

Ruxin Wang *
Union Vision Innovation
rosinwang@gmail.com

Dapeng Tao †
Yunnan University
dapeng.tao@gmail.com

Jun Cheng
SIAT, Chinese Academy of Sciences
jun.cheng@siat.ac.cn

Abstract

Single-image super-resolution restores the lost structures and textures from low-resolved images, which has achieved extensive attention from the research community. The top performers in this field include deep or wide convolutional neural networks, or recurrent neural networks. However, the methods enforce a single model to process all kinds of textures and structures. A typical operation is that a certain layer restores the textures based on the ones recovered by the preceding layers, ignoring the characteristics of image textures. In this paper, we believe that the lower-frequency and higher-frequency information in images have different levels of complexity and should be restored by models of different representational capacity. Inspired by this, we propose a novel embedded block residual network (EBRN) which is an incremental recovering progress for texture super-resolution. Specifically, different modules in the model restores information of different frequencies. For lower-frequency information, we use shallower modules of the network to recover; for higher-frequency information, we use deeper modules to restore. Extensive experiments indicate that the proposed EBRN model achieves superior performance and visual improvements against the state-of-the-arts.

1. Introduction

Single-image super-resolution (SISR) has attracted extensive attention in both academia and industry. This technique aims at recovering a high-resolved (HR) image from a single low-resolved (LR) one, which offers an opportunity of overcoming resolution limitations in various computer vision applications, such as security, medical imaging [37], and object recognition [3]. The problem of SISR is ill-posed since there exist multiple HR solutions for any LR input. To overcome this issue, most methods such as those based on deep convolutional neural networks constrain the solution space by learning a mapping function from external low- and high-resolution exemplar pairs or by involving a priori knowledge on the HR feature space.

(a) HR (b) EDSR (c) Ours

Figure 1. The *Barbara* image from the Set14 dataset with an upscaling factor 4.

Learning-based methods are placed in the top performers in literatures, especially deep or wide convolutional neural networks because of their high representational capacity. With extensive parameters and a good learning process, the models have the ability of fitting on a large number of training data and that of exploiting the underlying structures of natural images. Most methods advocate to design an end-to-end learning process that facilitates both training and inference. The performance improvement in such a design comes from an increase of the parameter number and the elaboration of neural connection. However, the resultant complex models usually raise high consumption of computation and memory, which hinders their real-world applications.

The reason of the above issue is that the deep model-based methods fail to consider the frequency characteristics of images which is, however, widely used in conventional image processing techniques. The characteristics state

*These authors contributed equally to the work.
†D.Tao is the corresponding author.

that natural images consist of different frequency bands of information, with each band containing structures and textures of different complexity. Hence, different bands of information are extracted by using different base functions, as in wavelet analysis [6]. In image restoration tasks including SISR, the recovery of each band information requires a specific restoring function. Considering that the feature distribution varies across different frequency bands, lower-frequency information is composed of simpler structures and textures where simpler functions are needed for restoration; higher-frequency information consists of complex structures and textures where more complex restoring functions are expected.

At this point, the existing deep model-based methods do not distinguish the image frequency. The task of each layer in those models is to recover all information based on the features of the preceding layer. For shallow layers, the parameters may fit on the low-frequency information (which has simple textures), but underfit on the high-frequency information (which has complex textures). For deep layers, the parameters may fit on the high-frequency information, but overfit on the low-frequency information. The inconsistency between the model complexity and the frequency is a key issue that limits the performance of those deep CNN-based methods. While residual connection provides a way to split the information as those recovered and those not-recovered, the residual architectures have no correlation with the frequency-splitting principle. Instead, they advocate that the residual connection transfers the information of shallow layers to deep layers in a dense and direct way. To bridge the connection between the model architecture and the frequency bands, elaboration of the residual idea is required. We illustrate an example in Figure 1, from which we see that the textures on the book cannot be well restored by EDSR [29]. That method takes advantage of residual connection which, however, fails to recover the simple textures by using a very deep architecture. Instead, complex curves appear in the result. Our result exhibits better visual properties. This comparison validates the drawback of EDSR [29] that deep layers are easily over-fitted on the low-frequency information of the image.

Based on the above analyses, in this paper, we propose an embedded block residual network (EBRN) for single image super-resolution that restores the textures of different frequency by using sub-networks of different complexity. Specifically, the block residual module (BRM) is the basic module in our model, which splits the data flow as a super-resolution flow and a back-projection flow. The former flow restores most structures and textures of lower frequency, while the later flow calculates the information of higher frequency which is remained to be recovered by deeper layers. The whole model is an embedding of multiple BRMs. Each BRM is stacked on the back-projection flow of its antecedent BRM. In this way, a BRM is responsible for the recovery of information at lower frequency, passing the information of higher frequency to deeper BRMs. To fuse the outputs of all BRMs, we also propose a recurrent fusion technique that stabilizes the feature flow and the gradient flow in training and encourages a faster convergence rate of training. Extensive experiments on multiple SISR datasets illustrate the state-of-the-art performance of the proposed method and validate the correlation of the model complexity and the image frequency as discussed above. In summary, the main contributions of this work are as follows:

1. We propose a motivation that the information of different frequency in images should be restored by the models of different complexity. In a bad case, the information of lower frequency could be over-recovered by a deeper model while the information of higher frequency would be under-recovered by a shallower model.

2. We propose a block residual module (BRM) that tries to restore the image structures and textures while passing the hard-to-recovered information to deeper modules. This allows each BRM to focus on the information of proper frequency, which is important for ensuring the correlation of model complexity and image frequency.

3. We propose a novel technique for embedding multiple BRMs, which can effectively improve the final reconstruction quality based on the outputs of each module. We also empirically demonstrate that the proposed model is superior over the state-of-the-arts.

2. Related work

SISR is an active research field and has a long history. Existing literatures could be grouped into three categories: the interpolation-based methods [20, 9], the reconstruction-based methods [5, 38], and the learning-based methods. While the conventional methods have a long list, here we review the top performers, especially the deep learning-based methods, due to the limit of page length.

Dong *et al*. [7] introduced CNN [26] into the SR task and proposed the SRCNN model that was composed of a three-layer network to learn the mapping from LR images to HR images. This model achieved much better performance compared with the traditional algorithms. Kim *et al.* [21] proposed the VDSR model that used a very deep network with 20 layers which produced improved performance compared with SRCNN. A main contribution of this method is to employ residual learning which encourages a fast convergence rate in the training process. Lai *et al.* [25] proposed the lapSRN method that took the original LR images as input and progressively reconstructed the sub-band residuals of HR images. Kim *et al.* [22] proposed the DRCN method which was the first to involve recursive learning into SISR. To reuse the features of each layer in CNN, Tong *et al.* [42] developed the DenseNet [14] by increasing dense connections among the convolutional layers. Lim *et al.* [29] designed the EDSR model by removing unnecessary modules in conventional residual networks, which achieved the champion of the NTIRE2017 SR Challenge [40]. Sajjadi *et al.* [35] compared the performance of different combinations of loss functions in EnhanceNet, and empirically draw the conclusion that the combination of perceptual loss, texture matching loss, and anti-loss worked best. Li *et al.* [39] proposed MenNet to address the issue that a deep network lacks long-term memory. This method introduced a memory block which consisted of a recursive unit and a gate unit, to explicitly mine persistent memory through an adaptive learning process. In order to reduce model parameters and model practicability, Ahn *et al.* [1] proposed a cascading residual network (CARN) that achieved good performance by using a cascading mechanism with few parameters. Haris *et al.* [11] developed a novel architecture which was named as DBPN. This model exploited iterative up- and down- sampling layers, providing an error feedback mechanism for project errors at each stage. DBPN also improved super-resolution performance, yielding superior results and in particular establishing new state-of-the-art performance for large scale factors such as $\times 8$ on multiple datasets. The enhanced version D-DBPN achieved the best performance in $\times 8$ enlargement in NTIRE2018 [41] and won the championship of NTIRE2018 SR Challenge. Zhang *et al.* [48] proposed the RDN model which differed from other CNN models, *i.e.*, this model did not make full use of hierarchical features of LR images. Hui *et al.* [16] proposed the information distillation network (IDN) with lightweight architecture and low computational complexity. Zhang *et al.* [47] stated that previous SR models treated each channel equally, hindering the representational capacity of CNN. They proposed RCAN to solve the problem by introducing channel attention mechanism. Li *et al.* [28] proposed the MSRN model to explore the multi-scale information of LR images.

The above deep learning-based methods were proposed to improve the PSNR/SSIM indexes of the restored images. However, existing studies indicate that the solution of the L2 objective function is an averaged version of multiple real HR solutions. Regarding this, the perceptual loss [19] was investigated to recover visually pleasing results for textures. For example, Ledig *et al.* [27] proposed SRGAN which inferred photo-realistic natural images. The results did not yield a high PSNR value, but produced realistic visual effects by using a perceptual loss consisting of an adversarial loss and a content loss. Wang *et al.* [43] proposed the SFTGAN model which involved a spatial feature modulation layer that integrated a priori of semantic categories into the network, generating more realistic and visually pleasing textures. Park *et al.* [32] proposed the SRFeat model to alleviate the issue that GAN-based approaches tend to include less meaningful high-frequency noise which is irrelevant to the input image. This model involved an additional discriminator on the feature domain. Wang *et al.* [44] developed ESRGAN to remove the artificial artifacts in the results of SRGAN by compensating an improvement sub-network.

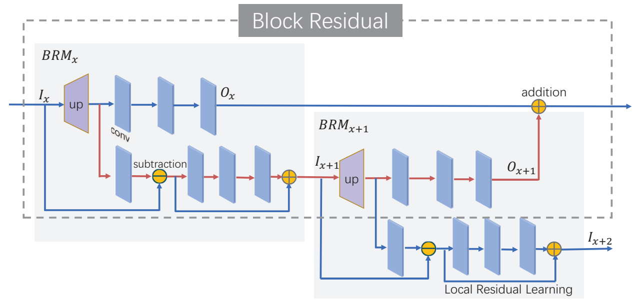

Figure 3. The block residual module.

The above literature review reveals that a significant improvement on SISR has been achieved by deep learning-based methods, especially CNNs and GANs. While the performance in the cases of scale factors $\times 2$, $\times 3$, and $\times 4$ may reach a bottleneck, the restoration of $\times 8$ becomes a main interest in recent publications. With the increase of the scale factor, we find that existing models have no concern about image frequency and model complexity, resulting in over-restoration of simple textures by using complex models, and under-restoration of complex textures by using simple models. Therefore, to alleviate this issue, this work starts from a different viewpoint, developing a proper architecture that associates the information of a frequency range with the model of appropriate complexity.

3. Proposed Method

In this section, we introduce the details of the proposed EBRN model and analyze how the information of different frequency is processed by the network. The architecture is illustrated in Figure 2, where the basic module is BRM which is presented in the following.

3.1. Block Residual Module

The block residual module (BRM) aims at restoring parts of the HR information while passing the remained signals to deeper modules for restoration. At this regard, the module contains two data flow: the super-resolution flow and the back-projection flow.

The super-resolution flow is a basic deconvolution network which takes the LR feature maps I_x as input and processes by using a stack of a deconvolutional layer (also

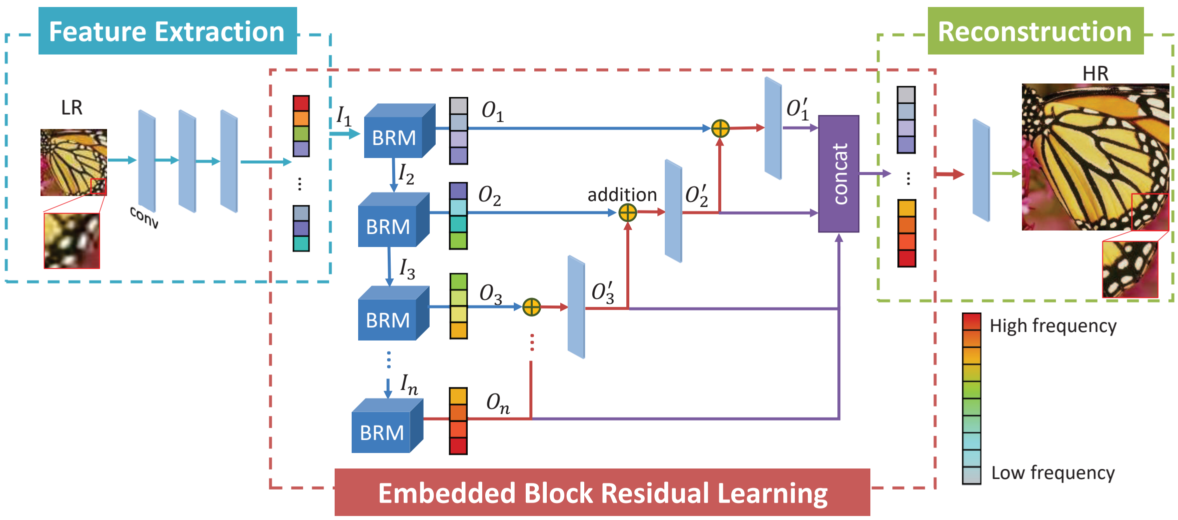

Figure 2. The architecture of the proposed embedded block residual network.

known as transposed convolution) and three convolutional layers. The output of this flow is the super-resolved feature maps O_x, where x is the index of BRM in the model. An alternative choice of the deconvolutional layer is the sub-pixel convolutional layer [36] which could improve the performance but yielding more parameters. Considering the tradeoff between performance and model efficiency, the deconvolutional layer is selected for up-scaling.

To compute the information that the super-resolution flow has not recovered, the back-projection flow employs an operation which first down-samples the deconvolved feature maps to the LR spatial size and then compute a minus between the down-sampled feature maps and the input LR feature maps of this module. The computed residual conveys the information that the super-resolution flow fails to recover. This residual is then processed by a local residual learning stage, outputting a set of encoded features I_{x+1} which forms the input of the next BRM.

The design of BRM is illustrated in Figure 3. All the convolutional layers utilize $3 \times 3 \times 64$ convolutional kernels. The layers except that for down-sampling are set with the stride of 1×1 and the padding size of 1×1. The parameters of the down-sampling layer are set according to the up-scaling factor, *i.e.*, the output feature maps have the same spatial size as the input feature maps. The local residual learning stage is to encourage a fast convergence rate of training, as in other residual learning methods. With such a design, we empirically find that the super-resolution flow could restore information of lower frequency, and the information of higher frequency which is difficult to be recovered is passed to later modules.

3.2. Embedded Block Residual Network

The embedded block residual network (EBRN) is composed of multiple BRMs, as shown in Figure 2. Before the first BRM, an initial feature extraction module is presented to formulate the shape of the feature maps. In this module, the first convolutional layer produces 256-channel feature maps, followed by which two convolutional layers are stacked with each outputting 64-channel feature maps. The convolutional kernel size in these layers is 3×3.

The BRMs are composed in an embedding way, instead of a simple stacking way. That is, the first BRM is stacked on the output of the initial feature extraction module, the second BRM is concatenated to the output of the back-projection flow of the first BRM, and so on. Each BRM is responsible for restoring the residual feature maps produced by the back-projection flow of its antecedent BRM. Note that the last BRM only contains the super-resolution flow where the back-projection flow is dropped. In this way, the information of lower frequency is only passed through the shallower BRMs which have low model complexity. The issue of overfitting on this part of information can be avoided. On the other hand, the information of higher frequency is flowed to deeper BRMs which have higher model complexity, where the underfitting problem can be alleviated. Therefore, a deeper BRM always tries to restore what has not been restored by shallower BRMs. This is consistent with our motivation. Another important point is that we associate the information of certain frequencies with a subnetwork of proper complexity. It is not required to fit a simple model on complex textures and also not to fit a complex model on simple structures. Hence, the number of the parameters in those sub-networks could be significantly re-

Figure 4. The residual module.

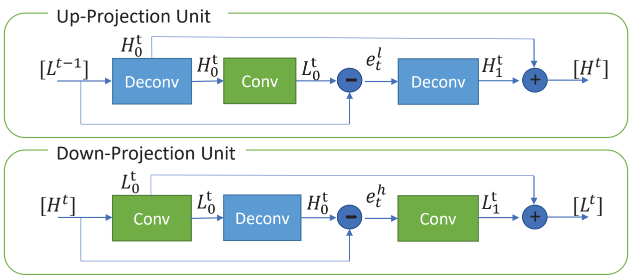

Figure 5. The up-projection and down-projection units in DBPN.

duced yet retaining a high restoration performance on the corresponding information.

To combine the outputs of all BRMs, we note that the recovered information by deep modules could help improve the restoration of shallow modules. Regarding this, we propose a recursive fusion technique instead of simple summation. Specifically, the outputs of the super-resolution flows of two adjacent BRMs are summed, which is then followed by a convolutional layer. Suppose that the output of the $(x+1)$-th module is O_{x+1}. The fusion process is

$$O'_x = f(O_x + O'_{x+1}), \tag{1}$$

where f is the function of the convolutional layer. Such a recursive fusion process is conducted until the first BRM is reached. Compared with simple summation, this technique allows us to process the outputs in a smooth way, resulting in better reconstruction. Moreover, to avoid the gradient vanishing issue in training, we propose to connect the output of each BRM directly to the image reconstruction module. As shown in Figure 2, we combine the outputs of all BRMs through a concatenation layer followed by a reconstruction sub-network. This design has two advantages: 1) the error propagation way to deep BRMs is shortened, encouraging a fast convergence rate in training, and 2) the intermediate feature maps of the model are reused for reconstruction, which is beneficial. The reconstruction sub-network utilizes $3 \times 3 \times 64$ convolutional kernels, while the last layer produces a 3-channel RGB image.

3.3. Loss Function

Existing loss functions for SISR include the L1 loss, the L2 loss, the adversarial loss [10], and the perceptual loss [19]. The current work aims at restoring an image that is as close as possible to the original HR image, and thus pixel-wise losses are selected. We follow the suggestion of Zhao *et al.* [49] which says that a SR model trained with the L2 loss function does not guarantee better PSNR/SSIM performance than with other loss functions. Here, we first select the L1 loss as the training objective of the proposed model, which is shown to speed up the convergence of training compared with the L2 loss. As a second step, we employ the L2 loss to finetune the model, which could result in higher PSNR performance. More details about training can be found in Section 5.2.

4. Discussions

In this section, we mainly discuss the difference between the proposed model and its related methods.

4.1. EBRN vs. Residual Network

Residual networks [13] have recently exhibited excellent performance in various computer vision tasks. In SISR, the first model using the residual learning idea is VDSR [21], which achieved superior performance compared with its competitors. The advantage of residual networks relative to traditional CNN models is that residual learning promotes the transmission of features in the network, and alleviates the gradient vanishing problem, making the network easier to train.

In this work, we exploit the residual learning idea, which is different from the conventional residual networks. For example, as shown in Figure 3, the proposed model does not use the batch normalization (BN) [17] layer since the BN layer limits the range flexibility of the intermediate features during feature normalization [29]. Another important difference comes from how the residual is computed and what the residual conveys. In the residual networks, the residual signal is the difference between the input and the output. In the proposed model, one type of residual signal is the information of a certain frequency range; another type of residual signal is the difference between the original LR features and the back-projected LR features. In each BRM, the second residual signal is important for SR since it explicitly conveys which information is to be recovered by the following BRM.

4.2. EBRN vs. Deep Back-Projection Network

A similar method to the current work is the deep back-projection network (DBPN) proposed by Haris *et al.* [11]. This method exploits iteratively up- and down-sampling layers, providing an error feedback mechanism for projecting errors at each stage. The errors can effectively improve the restoration by deep layers in the model.

The difference between the two methods comes from two aspects: 1) in each up- and down-projection unit, DBPN directly maps the LR residual to the HR space, whereas the LR residual in our model contains higher frequency information which is fed into deeper sub-networks for restora-

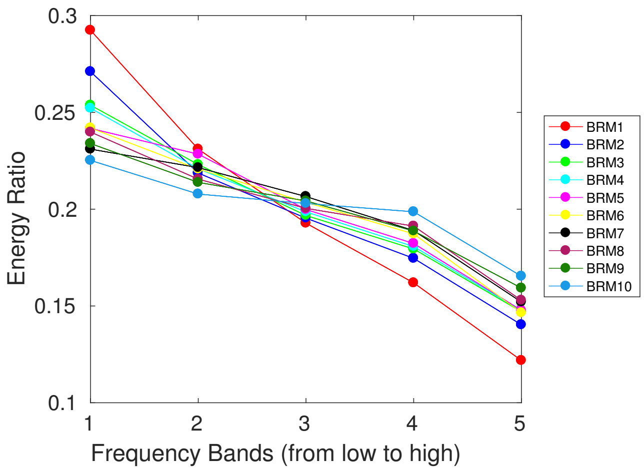

Figure 6. Energy distribution of different-BRM outputs across different frequency bands.

tion; and 2) DBPN exploits the LR residual and the HR residual with the goal that each up- and down-projection unit tries to minimize these residuals, whereas our method relates the residual signals to the information of different frequencies and each BRM is responsible for restoring the corresponding information. The difference of motivation results in that the proposed model has fewer parameters than DBPN, yet producing improved performance over DBPN.

5. Experiments

In this section, we present the experimental details and analyses to validate the effectiveness of the propose model.

5.1. Datasets

Following [29], we use the DIV2K [40] dataset for training, which is a high-quality (2K resolution) image restoration dataset containing 800 training images, 100 validation images, and 100 test images. The up-scaling factors including $\times 2$, $\times 4$, and $\times 8$ are used for training and model evaluation. 5 standard benchmark datasets are employed during testing, among which Set5 [4], Set14 [46], BSDS100 [2] consist of natural scenes, Urban100 [15] contains urban scenes with large amounts of regular texture patterns, and Manga109 [31] is a dataset of Japanese manga.

5.2. Implementation Details

To prepare the training data, we synthesize the LR images by down-sampling the training HR images using bicubic interpolation. Three training datasets are collected with each corresponding to one of the up-scaling factors, *i.e.,* $\times 2$, $\times 4$, and $\times 8$. Data augmentation techniques are utilized including horizontal, vertical flipping, and $90°$ rotation. Regarding the training details, the proposed model takes the

Method	Set5		Set14	
	PSNR	SSIM	PSNR	SSIM
EBRN(summation)	32.63	0.9018	28.89	0.7895
EBRN(recursive fusion)	32.79	0.9032	29.01	0.7903

Table 1. The comparison of different feature fusion techniques. Red indicates the best performance ($\times 4$).

Model	VDSR	DRCN	lapSRN	DRRN	MemNet	IDN	EBRN
time	0.071	0.984	0.023	4.4373	5.887	0.007	0.034

Table 2. Comparison of the running time (in seconds) on BSD100 ($\times 4$).

#BRMs	4	5	6	7	8	9	10
PSNR	32.05	32.23	32.35	32.44	32.51	32.65	32.79

Table 3. Performance *v.s.* number of BRMs on Set5 ($\times 4$).

RGB-channel images as input and output. The LR images are randomly cropped as 64×64 patch images which are then fed into the model with the batch size of 32. The sizes of the ground-truth HR patch images are determined by the up-scaling factor. The model weights are initialized using the method proposed in [12] and the biases are initialized as zero. The parametric rectified linear units (PRelu) [12] is used as the activation function. To ensure numeric stability during training, we scale the pixel range of LR and HR images to $[0, 1]$. The Adam [23] optimization algorithm is employed with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\varepsilon = 10^{-8}$. The learning rate is initially set to 10^{-4} and decreased by a factor of 10 at every 100 epochs. All experiments are implemented using the Pytorch [33] framework and evaluated on the NVIDIA TITAN X GPU devices.

5.3. Model Analyses

In this section, we conduct a series of experiments to validate the proposed motivation and investigate the effects of parameters on model performance.

Recall the motivation that the information of different frequency range should be processed by models of different complexity. To validate this, we illustrate the energy distributions of different-BRM outputs across different frequency bands in Figure 6. The energy distribution across different frequency bands is computed based on the wavelet coefficients of different levels. The result indicates that the outputs of shallower BRMs contain more lower-frequency information while the outputs of deeper BRMs tends to recover more higher-frequency information.

We also investigate the proposed model via ablation studies, including the correlation between the model performance and the feature fusion technique, and the correlation between the performance and the number of BRMs. Table 1 reveals the superiority of the proposed recursive feature fusion technique, compared with a simple summation op-

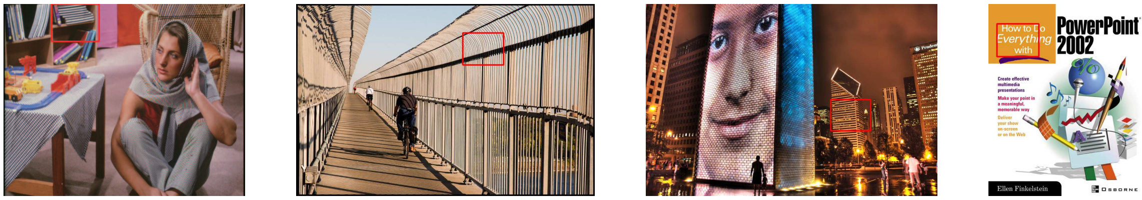

Figure 7. Four examples for the visualization of different restorations.

Method	Scale	Set5		Set14		BSD100		Urban100		Manga109	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	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 [7]	2	36.66	0.9542	32.45	0.9067	31.36	0.8879	29.50	0.8946	35.60	0.9663
VDSR [21]	2	37.53	0.9590	33.05	0.9130	33.05	0.8960	30.77	0.9140	37.22	0.9750
N^3Net [34]	2	37.57	-	-	-	31.98	-	30.80	-	-	-
DRCN [22]	2	37.63	0.9588	33.04	0.9118	31.85	0.8942	30.75	0.9133	37.57	0.9730
LapSRN [25]	2	37.52	0.9591	33.08	0.9130	31.08	0.8950	30.41	0.9101	32.27	0.9740
MemNet [39]	2	37.78	0.9597	33.28	0.9142	32.08	0.8978	31.31	0.9195	32.72	0.9740
EDSR [29]	2	38.11	0.9602	33.92	0.9195	32.32	0.9013	32.93	0.9351	39.10	0.9773
D-DBPN [11]	2	38.09	0.9600	33.85	0.9190	32.27	0.9000	32.55	0.9324	38.89	0.9775
IDN [16]	2	37.83	0.9600	33.30	0.9148	32.08	0.8985	31.27	0.9196	-	-
NLRN [30]	2	38.00	0.9603	33.46	0.9159	32.19	0.8992	31.81	0.9249	-	-
MSRN [28]	2	38.08	0.9605	33.74	0.9170	32.23	0.9013	32.22	0.9326	38.82	0.9868
CARN [1]	2	37.76	0.9590	33.52	0.9166	32.09	0.8978	31.92	0.9256	-	-
RDN [48]	2	38.24	0.9614	34.01	0.9212	32.34	0.9017	32.89	0.9353	39.18	0.9780
RCAN [47]	2	38.27	0.9614	34.12	0.9216	32.41	0.9027	33.34	0.9384	39.44	0.9786
EBRN(ours)	2	38.35	0.9620	34.24	0.9226	32.47	0.9033	33.52	0.9402	39.62	0.9802
Bicubic	4	28.42	0.8104	26.00	0.7027	25.96	0.6675	23.14	0.6577	24.89	0.7866
SRCNN [7]	4	30.48	0.8628	27.50	0.7513	26.90	0.7101	24.52	0.7221	27.58	0.8555
VDSR [21]	4	31.35	0.8830	28.02	0.7680	27.29	0.0726	25.18	0.7540	28.83	0.8870
N^3Net [34]	4	31.50	-	-	-	27.34	-	25.23	-	-	-
DRCN [22]	4	31.53	0.8854	28.02	0.7670	27.23	0.7233	25.14	0.7510	28.97	0.8860
LapSRN [25]	4	31.54	0.8850	28.19	0.7720	27.32	0.7270	25.21	0.7560	29.09	0.8900
MemNet [39]	4	31.74	0.8893	28.26	0.7723	27.40	0.7281	25.50	0.7630	29.42	0.8942
EDSR [29]	4	32.46	0.8968	28.80	0.7876	27.71	0.7420	26.64	0.8033	31.02	0.9148
D-DBPN [11]	4	32.47	0.8980	28.82	0.7860	27.72	0.7400	26.38	0.7946	30.91	0.9137
IDN [16]	4	31.82	0.8903	28.25	0.7730	27.41	0.7297	25.41	0.7632	-	-
NLRN [30]	4	31.92	0.8916	28.36	0.7745	27.48	0.7306	25.79	0.7729	-	-
MSRN [28]	4	32.07	0.8903	28.60	0.7751	27.52	0.7273	26.04	0.7896	30.17	0.9034
CARN [1]	4	32.13	0.8937	28.60	0.7806	27.58	0.7349	26.07	0.7837	-	-
RDN [48]	4	32.47	0.8990	28.81	0.7871	27.72	0.7419	26.61	0.8028	31.00	0.9151
PFF [24]	4	32.74	0.9021	28.98	0.7904	-	-	-	-	-	-
RCAN [47]	4	32.63	0.9002	28.87	0.7889	27.77	0.7436	26.82	0.8087	31.22	0.9173
EBRN(ours)	4	32.79	0.9032	29.01	0.7903	27.85	0.7464	27.03	0.8114	31.53	0.9198
Bicubic	8	24.40	0.6580	23.10	0.5660	23.67	0.5480	20.74	0.5160	21.47	0.6500
SRCNN [7]	8	25.33	0.6900	23.76	0.5910	24.13	0.5660	21.29	0.5440	22.46	0.6950
VDSR [21]	8	25.93	0.7240	24.26	0.6140	24.49	0.5830	21.70	0.5710	23.16	0.7250
LapSRN [25]	8	26.15	0.7380	24.35	0.6200	24.54	0.5860	21.81	0.5810	23.39	0.7350
MemNet [39]	8	26.16	0.7414	24.38	0.6199	24.58	0.5842	21.89	0.5825	23.56	0.7387
EDSR [29]	8	26.96	0.7762	24.91	0.6420	24.81	0.5985	22.51	0.6221	24.69	0.7841
MSRN [28]	8	26.59	0.7254	24.88	0.5961	24.70	0.5410	22.37	0.5977	24.28	0.7517
D-DBPN [11]	8	27.21	0.7840	25.13	0.6480	24.88	0.6010	22.73	0.6312	25.14	0.7987
RCAN [47]	8	27.31	0.7878	25.23	0.6511	24.98	0.6058	23.00	0.6452	25.24	0.8029
EBRN(ours)	8	27.45	0.7908	25.44	0.6542	25.12	0.6079	23.32	0.6498	25.51	0.8085

Table 4. The average performance of the state-of-the-art methods. Red font indicates the best performer and blue font indicates the second best performer.

eration. The two models have the same number of BRMs (*i.e.,* 10). Table 3 lists the performance of EBRN models with different sizes.

The number of model parameters is an important factor for SISR in real applications. As discussed in previous sections, the proposed embedding strategy could significantly reduce the number of parameters, which is validated here by comparing with the state-of-the-arts. As shown in Figure 9, the EBRN model with 10 BRMs exhibits better performance and fewer parameters than MDSR [29], D-DBPN [11], RCAN [47], and EDSR [29] which are the recently published SR methods. EBRN is also superior over the conventional small models including SRDenseNet [42], DRCN [22], LapSRN [25], VDSR [21], FSRCNN [8], and SRCNN [7]. These results indicate that the proposed EBRN performs well with limited amount of parameters, owing to its elaborate architecture.

5.4. Comparison with State-of-the-arts

In this section, we compare the proposed model with the state-of-the-arts including SRCNN [7], VDSR [21], DRCN [22], lapSRN [25], EDSR [29], RDN [48], IDN [16], M-

(a) HR (b) Bicubic (c) VDSR (d) DRCN (e) LapSRN (f) EDSR (g) IDN (h) MSRN (i) Our

Figure 8. Visualization of the selected parts restored by different methods. The up-scaling factor is 4.

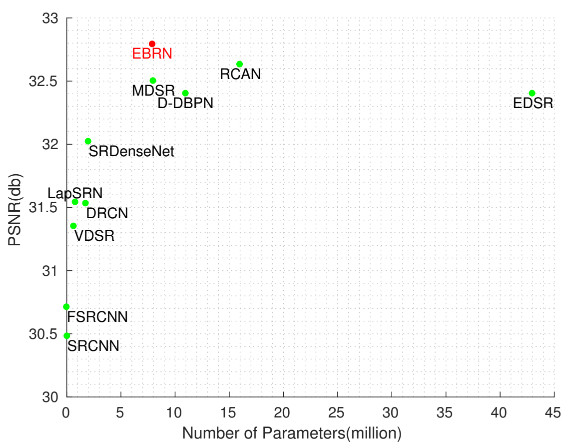

Figure 9. PSNR vs. the number of parameters. The comparison is conducted on Set5 with the $\times 4$ up-scaling factor.

SRN [28], D-DBPN [11], and RCAN [47]. The peak signal-to-noise ratio (PSNR) [18] and the structural similarity index (SSIM) [45] are employed as the evaluation metrics. Following a common setting and for fair comparison, we use the luminence channel (Y) of the transformed YCbCr space for quality measurement. While the proposed model takes RGB images as input, the Y-channel output is extracted after color conversion. The LR images are synthesized by using bicubic interpolation. Table 4 presents the $\times 2$, $\times 4$, and $\times 8$ performances of different methods, from which we see that the proposed method achieves the best PSNR and SSIM scores in all cases. Regarding the inference time, we use the published codes of the competitors which are evaluated on a server with 4.2GHz Intel i7 CPU, 32GB RAM, and a Nvidia TITANX GPU card. In Table 2, we show the comparison of running time of several efficient methods, indicating that the proposed model fullfills the requirements of real-time applications. We select four examples for visualization, as shown in Figure 7. The details of the examples are zoomed in and visualized in Figure 8, from which it is observed that the proposed method can synthesize more pleasing textures and structures. The competitors produce flawed textures which may be caused by underfitting of the model on complex textures or overfitting of the model on simple areas.

6. Conclusions

In this paper, we are motivated by that information of different frequency should be restored by models of different complexity, and propose an embedded block residual network for single image super-resolution. We advocate that the limitation of existing methods is caused by underfitting of the models on complex textures and overfitting on simple structures. As such, we develop a block residual module that could restore parts of the image information while passing the remained information to deeper layers. The modules are embedded to form a deep architecture. An elaborate sub-network is also designed for effective feature fusion. Using the proposed model, the information of lower frequency is restored by shallower BRMs while the information of higher frequency is recovered by deeper BRMs. Comprehensive experiments demonstrate the effectiveness of the proposed idea.

Acknowledgements This work was supported by National Natural Science Foundation of China under Grant 61772455, 61772508, and U1713213, Yunnan Natural Science Funds under Grant 2018FY001(-013), the Program for Excellent Young Talents of National Natural Science Foundation of Yunnan University under Grant 2018YDJQ004, Shenzhen Technology Project (JCYJ20170413152535587, JCYJ20180507182610734), and CAS Key Technology Talent Program.

References

[1] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *Proceedings of the European Conference on Computer Vision*, September 2018.

[2] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, 2011.

[3] Yancheng Bai, Yongqiang Zhang, Mingli Ding, and Bernard Ghanem. Finding tiny faces in the wild with generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–30, 2018.

[4] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012.

[5] Shengyang Dai, Mei Han, Wei Xu, Ying Wu, Yihong Gong, and Aggelos K Katsaggelos. Softcuts: a soft edge smoothness prior for color image super-resolution. *IEEE Transactions on Image Processing*, 18(5):969–981, 2009.

[6] Ingrid Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory*, 36(5):961–1005, 1990.

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

[8] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *Proceedings of the European Conference on Computer Vision*, pages 391–407. Springer, 2016.

[9] Claude E Duchon. Lanczos filtering in one and two dimensions. *Journal of applied meteorology*, 18(8):1016–1022, 1979.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[11] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1664–1673, 2018.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2016.

[14] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, July 2017.

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

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

[17] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[18] Michal Irani and Shmuel Peleg. Motion analysis for image enhancement: Resolution, occlusion, and transparency. *Journal of Visual Communication Image Representation*, 4(4):324–335, 1993.

[19] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of the European Conference on Computer Vision*, pages 694–711. Springer, 2016.

[20] Robert Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(6):1153–1160, 1981.

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

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

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[24] Shu Kong and Charless Fowlkes. Image reconstruction with predictive filter flow. *arXiv preprint arXiv:1811.11482*, 2018.

[25] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 624–632, 2017.

[26] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989.

[27] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan 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*, pages 4681–4690, 2017.

[28] Juncheng Li, Faming Fang, Kangfu Mei, and Guixu Zhang. Multi-scale residual network for image super-resolution. In

Proceedings of the European Conference on Computer Vision (ECCV), pages 517–532, 2018.

[29] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, July 2017.

[30] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 1673–1682, 2018.

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

[32] Seong-Jin Park, Hyeongseok Son, Sunghyun Cho, Ki-Sang Hong, and Seungyong Lee. Srfeat: Single image super-resolution with feature discrimination. In *Proceedings of the European Conference on Computer Vision*, pages 439–455, 2018.

[33] 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.

[34] Tobias Plötz and Stefan Roth. Neural nearest neighbors networks. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 1087–1098, 2018.

[35] Mehdi SM Sajjadi, Bernhard Scholkopf, and Michael Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4491–4500, 2017.

[36] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan 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*, pages 1874–1883, 2016.

[37] Wenzhe Shi, Jose Caballero, Christian Ledig, Xiahai Zhuang, Wenjia Bai, Kanwal Bhatia, Antonio M Simoes Monteiro de Marvao, Tim Dawes, Declan ORegan, and Daniel Rueckert. Cardiac image super-resolution with global correspondence using multi-atlas patchmatch. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 9–16. Springer, 2013.

[38] Jian Sun, Zongben Xu, and Heung-Yeung Shum. Image super-resolution using gradient profile prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

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

[40] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei 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*, July 2017.

[41] Radu Timofte, Shuhang Gu, Jiqing Wu, and Luc Van Gool. Ntire 2018 challenge on single image super-resolution: methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 852–863, 2018.

[42] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4799–4807, 2017.

[43] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 606–615, 2018.

[44] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision*, pages 63–79. Springer, 2018.

[45] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[46] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *Proceedings of the International conference on curves and surfaces*, pages 711–730. Springer, 2010.

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

[48] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2472–2481, 2018.

[49] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for neural networks for image processing. *arXiv preprint arXiv:1511.08861*, 2015.