Feedback Network for Image Super-Resolution

Zhen Li[1] Jinglei Yang[2] Zheng Liu[3] Xiaomin Yang[1*] Gwanggil Jeon[4] Wei Wu[1*]
[1]Sichuan University, [2]University of California, Santa Barbara, [3]University of British Columbia,
[4]Incheon National University

Abstract

Recent advances in image super-resolution (SR) explored the power of deep learning to achieve a better reconstruction performance. However, the feedback mechanism, which commonly exists in human visual system, has not been fully exploited in existing deep learning based image SR methods. In this paper, we propose an image super-resolution feedback network (SRFBN) to refine low-level representations with high-level information. Specifically, we use hidden states in a recurrent neural network (RNN) with constraints to achieve such feedback manner. A feedback block is designed to handle the feedback connections and to generate powerful high-level representations. The proposed SRFBN comes with a strong early reconstruction ability and can create the final high-resolution image step by step. In addition, we introduce a curriculum learning strategy to make the network well suitable for more complicated tasks, where the low-resolution images are corrupted by multiple types of degradation. Extensive experimental results demonstrate the superiority of the proposed SRFBN in comparison with the state-of-the-art methods. Code is avaliable at `https://github.com/Paper99/SRFBN_CVPR19`.

1. Introduction

Image super-resolution (SR) is a low-level computer vision task, which aims to reconstruct a high-resolution (HR) image from its low-resolution (LR) counterpart. It is inherently ill-posed since multiple HR images may result in an identical LR image. To address this problem, numerous image SR methods have been proposed, including interpolation-based methods[45], reconstruction-based methods[42], and learning-based methods [33, 26, 34, 15, 29, 6, 18].

Since Dong *et al*. [6] firstly introduced a shallow Convolutional Neural Network (CNN) to implement image SR, deep learning based methods have attracted extensive attention in recent years due to their superior reconstruction performance. The benefits of deep learning based methods mainly come from its two key factors, i.e., *depth* and *skip connections* (residual or dense) [18, 36, 31, 11, 47, 46, 37]. The first one provides a powerful capability to represent and establish a more complex LR-HR mapping, while preserving more contextual information with larger receptive fields. The second factor can efficiently alleviate the gradient vanishing/exploding problems caused by simply stacking more layers to deepen networks.

As the depth of networks grows, the number of parameters increases. A large-capacity network will occupy huge

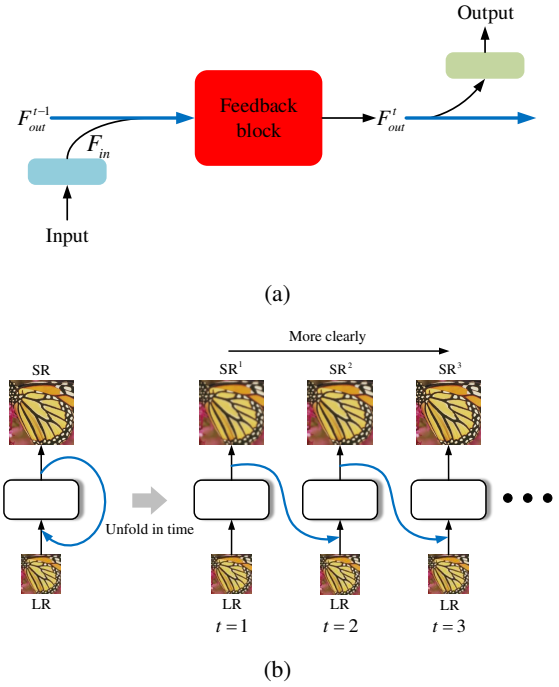


Figure 1. The illustrations of the feedback mechanism in the proposed network. Blue arrows represent the feedback connections. (a) Feedback via the hidden state at one iteration. The feedback block (FB) receives the information of the input F_{in} and hidden state from last iteration F_{out}^{t-1}, and then passes its hidden state F_{out}^{t} to the next iteration and output. (b) The principle of our feedback scheme.

*Corresponds to: {arielyang,wuwei}@scu.edu.cn

storage resources and suffer from the overfitting problem. To reduce network parameters, the recurrent structure is often employed. Recent studies [22, 10] have shown that many networks with recurrent structure (*e.g*. DRCN [19] and DRRN [31]) can be extrapolated as a single-state Recurrent Neural Network (RNN). Similar to most conventional deep learning based methods, these networks with recurrent structure can share the information in a feedforward manner. However, the feedforward manner makes it impossible for previous layers to access useful information from the following layers, even though skip connections are employed.

In cognition theory, feedback connections which link the cortical visual areas can transmit response signals from higher-order areas to lower-order areas [17, 9]. Motivated by this phenomenon, recent studies [30, 40] have applied the feedback mechanism to network architectures. The feedback mechanism in these architectures works in a top-down manner, carrying high-level information back to previous layers and refining low-level encoded information.

In this paper, we propose a novel network for image SR, namely the Super-Resolution Feedback Network (SRFBN), in order to refine low-level information using high-level one through feedback connections. The proposed SRFBN is essentially an RNN with a feedback block (FB), which is specifically designed for image SR tasks. The FB is constructed by multiple sets of up- and down-sampling layers with dense skip connections to generate powerful high-level representations. Inspired by [40], we use the output of the FB, i.e., a hidden state in an unfolded RNN, to achieve the feedback manner (see Fig. 1(a)). The hidden state at each iteration flows into the next iteration to modulate the input. To ensure the hidden state contains the information of the HR image, we connect the loss to each iteration during the training process. The principle of our feedback scheme is that the information of a coarse SR image can facilitate an LR image to reconstruct a better SR image (see Fig. 1(b)). Furthermore, we design a curriculum for the case, in which the LR image is generated by a complex degradation model. For each LR image, its target HR images for consecutive iterations are arranged from easy to hard based on the recovery difficulty. Such curriculum learning strategy well assists our proposed SRFBN in handling complex degradation models. Experimental results demonstrate the superiority of our proposed SRFBN against other state-of-the-art methods.

In summary, our main contributions are as follows:

- Proposing an image super-resolution feedback network (SRFBN), which employs a feedback mechanism. High-level information is provided in top-down feedback flows through feedback connections. Meanwhile, such recurrent structure with feedback connections provides strong early reconstruction ability, and requires only few parameters.
- Proposing a feedback block (FB), which not only efficiently handles feedback information flows, but also enriches high-level representations via up- and down-sampling layers, and dense skip connections.
- Proposing a curriculum-based training strategy for the proposed SRFBN, in which HR images with increasing reconstruction difficulty are fed into the network as targets for consecutive iterations. This strategy enables the network to learn complex degradation models step by step, while the same strategy is impossible to settle for those methods with only one-step prediction.

2. Related Work

2.1. Deep learning based image super-resolution

Deep learning has shown its superior performance in various computer vision tasks including image SR. Dong *et al*. [7] firstly introduced a three-layer CNN in image SR to learn a complex LR-HR mapping. Kim *et al*. [18] increased the depth of CNN to 20 layers for more contextual information usage in LR images. In [18], a skip connection was employed to overcome the difficulty of optimization when the network became deeper. Recent studies have adopted different kind of skip connections to achieve remarkable improvement in image SR. SRResNet[21] and EDSR[23] applied residual skip connections from [13]. SRDenseNet[36] applied dense skip connections from [14]. Zhang *et al*. [47] combined local/global residual and dense skip connections in their RDN. Since the skip connections in these network architectures use or combine hierarchical features *in a bottom-up way*, the low-level features can only receive the information from previous layers, lacking enough contextual information due to the limitation of small receptive fields. These low-level features are reused in the following layers, and thus further restrict the reconstruction ability of the network. To fix this issue, we propose a super-resolution feedback network (SRFBN), in which high-level information flows through feedback connections *in a top-down manner* to correct low-level features using more contextual information.

Meanwhile, with the help of skip connections, neural networks go deeper and hold more parameters. Such large-capacity networks occupy huge amount of storage resources and suffer from overfitting. To effectively reduce network parameters and gain better generalization power, the recurrent structure was employed[19, 31, 32]. Particularly, the recurrent structure plays an important role to realize the feedback process in the proposed SRFBN (see Fig. 1(b)).

2.2. Feedback mechanism

The feedback mechanism allows the network to carry a notion of output to correct previous states. Recently, the

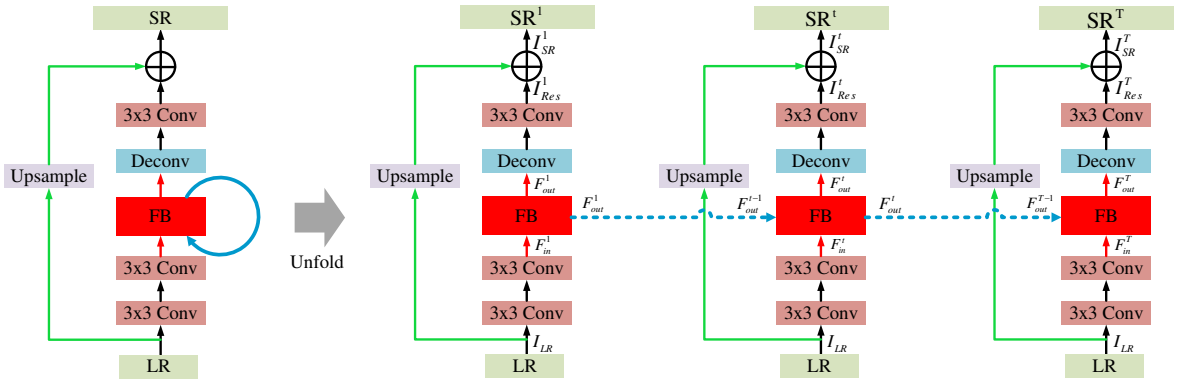


Figure 2. The architecture of our proposed super-resolution feedback network (SRFBN). Blue arrows represent feedback connections. Green arrows represent global residual skip connections.

feedback mechanism has been adopted by many network architectures for various vision tasks[5, 4, 40, 11, 10, 28].

For image SR, a few studies also showed efforts to introduce the feedback mechanism. Based on back-projection, Haris *et al*. [11] designed up- and down-projection units to achieve iterative error feedback. Han *et al*. [10] applied a delayed feedback mechanism which transmits the information between two recurrent states in a dual-state RNN. However, the flow of information from the LR image to the final SR image is still feedforward in their network architectures unlike ours.

The most relevant work to ours is [40], which transfers the hidden state with high-level information to the information of an input image to realize feedback in an convolutional recurrent neural network. However, it aims at solving high-level vision tasks, *e.g*. classification. To fit a feedback mechanism in image SR, we elaborately design a feedback block (FB) as the basic module in our SRFBN, instead of using ConvLSTM as in [40]. The information in our FB efficiently flows across hierarchical layers through dense skip connections. Experimental results indicate our FB has superior reconstruction performance than ConvLSTM[1] and thus is more suitable for image SR tasks.

2.3. Curriculum learning

Curriculum learning[2], which gradually increases the difficulty of the learned target, is well known as an efficient strategy to improve the training procedure. Early work of curriculum learning mainly focuses on a single task. Pentina *et al*. [27] extended curriculum learning to multiple tasks in a sequential manner. Gao *et al*. [8] utilized curriculum learning to solve the fixation problem in image restoration. Since their network is limited to a one-time prediction, they enforce a curriculum through feeding different training data in terms of the complexity of tasks as epoch increases during the training process. In the context of image SR, Wang *et al*. [38] designed a curriculum for the pyramid structure, which gradually blends a new level of the pyramid in previously trained networks to upscale an LR image to a bigger size.

While previous works focus on a single degradation process, we enforce a curriculum to the case, where the LR image is corrupted by multiple types of degradation. The curriculum containing easy-to-hard decisions can be settled for one query to gradually restore the corrupted LR image.

3. Feedback Network for Image SR

Two requirements are contained in a feedback system: (1) iterativeness and (2) rerouting the output of the system to correct the input in each loop. Such iterative cause-and-effect process helps to achieve the principle of our feedback scheme for image SR: high-level information can guide an LR image to recover a better SR image (see Fig. 1(b)). In the proposed network, there are three indispensable parts to enforce our feedback scheme: (1) tying the loss at each iteration (to force the network to reconstruct an SR image at each iteration and thus allow the hidden state to carry a notion of high-level information), (2) using recurrent structure (to achieve iterative process) and (3) providing an LR input at each iteration (to ensure the availability of low-level information, which is needed to be refined). Any absence of these three parts will fail the network to drive the feedback flow.

3.1. Network structure

As shown in Fig. 2, our proposed SRFBN can be unfolded to T iterations, in which each iteration t is temporally ordered from 1 to T. In order to make the hidden state in SRFBN carry a notion of output, we tie the loss for every iteration. The description of the loss function can be found in Sec. 3.3. The sub-network placed in each iteration t contains three parts: an LR feature extraction block (LRFB), a feedback block (FB) and a reconstruction block (RB). The

[1]Further analysis can be found in our supplementary material.

weights of each block are shared across time. The global residual skip connection at each iteration t delivers an upsampled image to bypass the sub-network. Therefore, the purpose of the sub-network at each iteration t is to recover a residual image I^t_{Res} while input a low-resolution image I_{LR}. We denote $Conv(s, n)$ and $Deconv(s, n)$ as a convolutional layer and a deconvolutional layer respectively, where s is the size of the filter and n is the number of filters.

The LR feature extraction block consists of $Conv(3, 4m)$ and $Conv(3, m)$. m denotes the base number of filters. We provide an LR input I_{LR} for the LR feature extraction block, from which we obtain the shallow features F^t_{in} containing the information of an LR image:

$$F^t_{in} = f_{LRFB}(I_{LR}), \tag{1}$$

where f_{LRFB} denotes the operations of the LR feature extraction block. F^t_{in} are then used as the input to the FB. In addition, F^1_{in} are regarded as the initial hidden state F^0_{out}.

The FB at the t-th iteration receives the hidden state from previous iteration F^{t-1}_{out} through a feedback connection and shallow features F^t_{in}. F^t_{out} represents the output of the FB. The mathematical formulation of the FB is:

$$F^t_{out} = f_{FB}(F^{t-1}_{out}, F^t_{in}), \tag{2}$$

where f_{FB} denotes the operations of the FB and actually represents the feedback process as shown in Fig. 1(a). More details of the FB can be found in Sec. 3.2.

The reconstruction block uses $Deconv(k, m)$ to upscale LR features F^t_{out} to HR ones and $Conv(3, c_{out})$ to generate a residual image I^t_{Res}. The mathematical formulation of the reconstruction block is:

$$I^t_{Res} = f_{RB}(F^t_{out}), \tag{3}$$

where f_{RB} denotes the operations of the reconstruction block.

The output image I^t_{SR} at the t-th iteration can be obtained by:

$$I^t_{SR} = I^t_{Res} + f_{UP}(I_{LR}), \tag{4}$$

where f_{UP} denotes the operation of an upsample kernel. The choice of the upsample kernel is arbitrary. We use a bilinear upsample kernel here. After T iterations, we will get totally T SR images $(I^1_{SR}, I^2_{SR}, ..., I^T_{SR})$.

3.2. Feedback block

As shown in Fig. 3, the FB at the t-th iteration receives the feedback information F^{t-1}_{out} to correct low-level representations F^t_{in}, and then passes more powerful high-level representations F^t_{out} as its output to the next iteration and the reconstruction block. The FB contains G projection groups sequentially with dense skip connections among them. Each projection group, which can project HR features

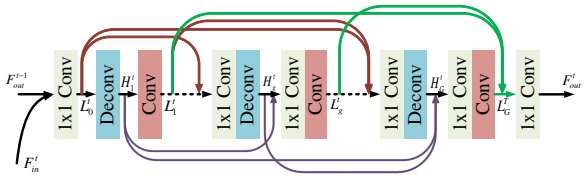


Figure 3. Feedback block (FB).

to LR ones, mainly includes an upsample operation and a downsample operation.

At the beginning of the FB, F^t_{in} and F^{t-1}_{out} are concatenated and compressed by $Conv(1, m)$ to refine input features F^t_{in} by feedback information F^{t-1}_{out}, producing the refined input features L^t_0:

$$L^t_0 = C_0([F^{t-1}_{out}, F^t_{in}]), \tag{5}$$

where C_0 refers to the initial compression operation and $[F^{t-1}_{out}, F^t_{in}]$ refers to the concatenation of F^{t-1}_{out} and F^t_{in}. Let H^t_g and L^t_g be the HR and LR feature maps given by the g-th projection group in the FB at the t-th iteration. H^t_g can be obtained by:

$$H^t_g = C^{\uparrow}_g([L^t_0, L^t_1, ..., L^t_{g-1}]), \tag{6}$$

where $C^{\uparrow}_g$ refers to the upsample operation using $Deconv(k, m)$ at the g-th projection group. Correspondingly, L^t_g can be obtained by

$$L^t_g = C^{\downarrow}_g([H^t_1, H^t_2, ..., H^t_g]), \tag{7}$$

where $C^{\downarrow}_g$ refers to the downsample operation using $Conv(k, m)$ at the g-th projection group. Except for the first projection group, we add $Conv(1, m)$ before $C^{\uparrow}_g$ and $C^{\downarrow}_g$ for parameter and computation efficiency.

In order to exploit useful information from each projection group and map the size of input LR features F^{t+1}_{in} at the next iteration, we conduct the feature fusion (green arrows in Fig. 3) for LR features generated by projection groups to generate the output of FB:

$$F^t_{out} = C_{FF}([L^t_1, L^t_2, ..., L^t_G]), \tag{8}$$

where C_{FF} represents the function of $Conv(1, m)$.

3.3. Curriculum learning strategy

We choose $L1$ loss to optimize our proposed network. T target HR images $(I^1_{HR}, I^2_{HR}, ..., I^T_{HR})$ are placed to fit in the multiple output in our proposed network. $(I^1_{HR}, I^2_{HR}, ..., I^T_{HR})$ are identical for the single degradation model. For complex degradation models, $(I^1_{HR}, I^2_{HR}, ..., I^T_{HR})$ are ordered based on the difficulty of tasks for T iterations to enforce a curriculum. The loss function in the network can be formulated as:

$$L(\Theta) = \frac{1}{T}\sum_{t=1}^{T} W^t \left\| I^t_{HR} - I^t_{SR} \right\|_1, \tag{9}$$

where Θ denotes to the parameters of our network. W^t is a constant factor which demonstrates the worth of the output at the t-th iterations. As [40] do, we set the value to 1 for each iteration, which represents each output has equal contribution. Details about settings of target HR images for complex degradation models will be revealed in Sec. 4.4.

3.4. Implementation details

We use PReLU[12] as the activation function following all convolutional and deconvolutional layers except the last layer in each sub-network. Same as [11], we set various k in $Conv(k, m)$ and $Deconv(k, m)$ for different scale factors to perform up- and down-sampling operations. For $\times 2$ scale factor, we set k in $Conv(k, m)$ and $Deconv(k, m)$ as 6 with two striding and two padding. Then, for $\times 3$ scale factor, we set $k = 7$ with three striding and two padding. Finally, for $\times 4$ scale factor, we set $k = 8$ with four striding and two padding. We take the SR image I^T_{SR} at the last iteration as our final SR result unless we specifically analysis every output image at each iteration. Our network can process both gray and color images, so c_{out} can be 1 or 3 naturally.

4. Experimental Results

4.1. Settings

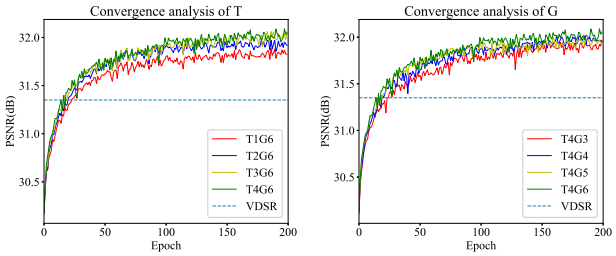
Datasets and metrics. We use DIV2K[1] and Flickr2K as our training data. To make full use of data, we adopt data augmentation as [23] do. We evaluate SR results under PSNR and SSIM[39] metrics on five standard benchmark datasets: Set5[3], Set14[41], B100[24], Urban100[15], and Manga109[25]. To keep consistency with previous works, quantitative results are only evaluated on luminance (Y) channel.

Degradation models. In order to make fair comparison with existing models, we regard bicubic downsampling as our standard degradation model (denoted as **BI**) for generating LR images from ground truth HR images. To verify the effectiveness of our curriculum learning strategy, we further conduct two experiments involving two other multi-degradation models as [47] do in Sec. 4.4 and 4.5.3. We define **BD** as a degradation model which applies Gaussian blur followed by downsampling to HR images. In our experiments, we use 7x7 sized Gaussian kernel with standard deviation 1.6 for blurring. Apart from the **BD** degradation model, **DN** degradation model is defined as bicubic downsampling followed by adding Gaussian noise, with noise level of 30.

Scale factor	$\times 2$	$\times 3$	$\times 4$
Input patch size	60×60	50×50	40×40

Table 1. The settings of input patch size.

Training settings. We train all networks with the batch-size of 16. To fully exploit contextual information from LR images, we feed RGB image patches with different patch size based on the upscaling factor. The settings of input patch size are listed in Tab. 1. The network parameters are initialized using the method in [12]. Adam[20] is employed to optimize the parameters of the network with initial learning rate 0.0001. The learning rate multiplies by 0.5 for every 200 epochs. We implement our networks with Pytorch framework and train them on NVIDIA 1080Ti GPUs.



(a) (b)

Figure 4. Convergence analysis of T and G on Set5 with scaling factor $\times 4$.

4.2. Study of T and G

In this subsection, we explore the influence of the number of iterations (denoted as T) and the number of projection groups in the feedback block (denoted as G). The base number of filters m is set to 32 in subsequent experiments. We first investigate the influence of T by fixing G to 6. It can be observed from Fig. 4(a) that with the help of feedback connection(s), the reconstruction performance is significantly improved compared with the network without feedback connections (T=1). Besides, as T continues to increase, the reconstruction quality keeps rising. In other words, our feedback block surely benefits the information flow across time. We then study the influence of G by fixing T to 4. Fig. 4(b) shows that larger G leads to higher accuracy due to stronger representative ability of deeper networks. In conclusion, choosing larger T or G both contribute to better results. It is worth noticing that small T and G still outperform VDSR[18]. In the following discussions, we use SRFBN-L (T=4, G=6) for analysis.

No. Prediction	1st	2nd	3rd	4th
SRFBN-L-FF	30.69	31.74	32.00	32.09
SRFBN-L	**31.85**	**32.06**	**32.11**	**32.11**

Table 2. The impact of feedback on Set5 with scale factor $\times 4$.

4.3. Feedback vs. feedforward

To investigate the nature of the feedback mechanism in our network, we compare the feedback network with feedforward one in this subsection.

We first demonstrate the superiority of the feedback mechanism over its feedforward counterpart. By simply disconnecting the loss to all iterations except the last one, the network is thus impossible to reroute a notion of output to low-level representations and is then degenerated to a feedforward one (however still retains its recurrent property), denoted as SRFBN-L-FF. SRFBN-L and SRFBN-L-FF both have four iterations, producing four intermediate output. We then compare the PSNR values of all intermediate SR images from both networks. The results are shown in Tab. 2. SRFBN-L outperforms SRFBN-L-FF at every iteration, from which we conclude that the feedback network is capable of producing high quality early predictions in contrast to feedforward network. The experiment also indicates that our proposed SRFBN does benefit from the feedback mechanism, instead of only rely on the power of the recurrent structure. Except for the above discussions about the necessity of early losses, we also conduct two more abalative experiments to verify other parts (discussed in Sec. 3) which form our feedback system. By turning off weights sharing across iterations, the PSNR value in the proposed network is decreased from 32.11dB to 31.82dB on Set5 with scale factor $\times 4$. By disconnecting the LR input at each iteration except the first iteration, the PSNR value is decreased by 0.17dB.

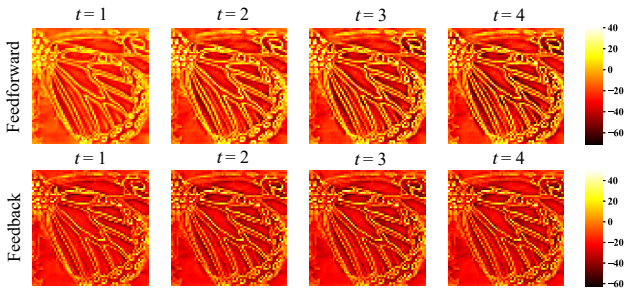


Figure 5. Average feature maps of feedforward and feedback networks.

To dig deeper into the difference between feedback and feedforward networks, we visualize the average feature map of every iteration in SRFBN-L and SRFBN-L-FF, illustrated in Fig. 5. Each average feature map is the mean of F_{out}^t in channel dimension, which roughly represents the output of the feedback block at the t-th iteration. Our network with global residual skip connections aims at recovering the residual image. In other words, the tasks of our network are to suppress the smooth area of the original input image[16] and to predict high-frequency components (*i.e.* edges and contours). From Fig. 5, we have two observations. First, compared with the feedforward network at early iterations, feature maps acquired from the feedback network contain more negative values, showing a stronger effect of suppressing the smooth area of the input image, which further leads to a more accurate residual image. To some extent, this illustration reflects the reason why the feedback network has more powerful early reconstruction ability than the feedforward one. The second observation is that the feedback network learns different representations in contrast to feedforward one when handling the same task. In the feedforward network, feature maps vary significantly from the first iteration (t=1) to the last iteration (t=4): the edges and contours are outlined at early iterations and then the smooth areas of the original image are suppressed at latter iterations. The distinct patterns demonstrate that the feedforward network forms a hierarchy of information through layers, while the feedback network is allowed to devote most of its efforts to take a self-correcting process, since it can obtain well-developed feature representations at the initial iteration. This further indicates that F_{out}^t containing high-level information at the t-th iteration in the feedback network will urge previous layers at subsequent iterations to generate better representations.

Model	from scratch		from pretrained	
	w/o CL	with CL	w/o CL	with CL
BD	29.78	29.96	29.98	**30.03**
DN	26.92	26.93	26.96	**26.98**

Table 3. The investigation of curriculum learning (CL) on **BD** and **DN** degradation models with scale factor $\times 4$. The average PSNR values are evaluated on Set5.

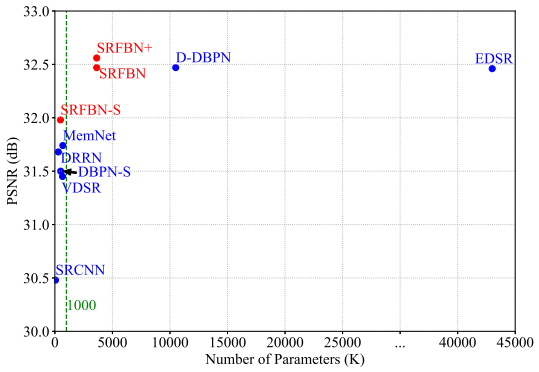


Figure 6. Performance and number of parameters. Results are evaluated on Set5 with scale factor $\times 4$. Red points represent our proposed networks.

4.4. Study of curriculum learning

As mentioned in Sec. 4.1, we now present our results for two experiments on two different degradation models, *i.e.* **BD** and **DN**, to show the effectiveness of our curriculum learning strategy.

We formulate the curriculum based on the recovery difficulty. For example, to guide the network to learn recovering a **BD** operator corrupted image step by step, we provide a Gaussian blurred HR image as (intermediate) ground truth

so that the network only needs to learn the inversion of a single downsampling operator at early iterations. Original HR image is provided at latter iterations as a senior challenge. Specifically, we empirically provide blurred HR images at first two iterations and original HR images at remaining two iterations for experiments with the **BD** degradation model. For experiments with the **DN** degradation model, we instead use noisy HR images at first two iterations and HR images without noise at last two iterations.

We also examine the compatibility of this strategy with two common training processes, *i.e*. training from scratch and fine-tuning on a network pretrained on the **BI** degradation model. The results shown in Tab. 3 infer that the curriculum learning strategy well assists our proposed SRFBN in handling **BD** and **DN** degradation models under both circumstances. We also observe that fine-tuning on a network pretrained on the **BI** degradation model leads to higher PSNR values than training from scratch.

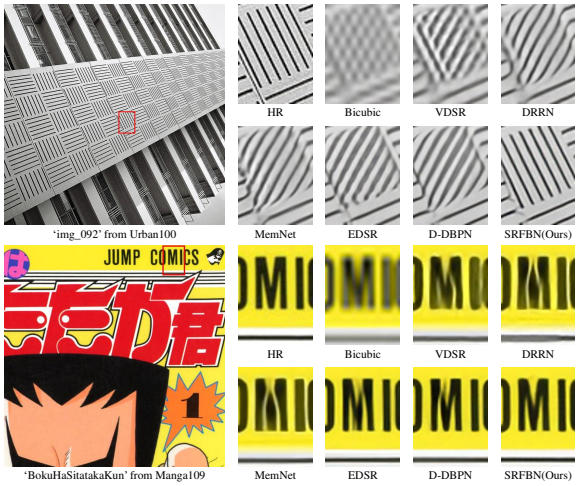


Figure 7. Visual results of **BI** degradation model with scale factor $\times 4$.

4.5. Comparison with the state-of-the-arts

The SRFBN with a larger base number of filters (m=64), which is derived from the SRFBN-L, is implemented for comparison. A self-ensemble method[35] is also used to further improve the performance of the SRFBN (denoted as SRFBN+). A lightweight network SRFBN-S (T=4, G=3, m=32) is provided to compare with the state-of-the-art methods, which are carried only few parameters.

4.5.1 Network parameters

The state-of-the-art methods considered in this experiment include SRCNN[7], VDSR[18], DRRN[31], MemNet[36], EDSR[23], DBPN-S[11] and D-DBPN[11]. The comparison results are given in Fig. 6 in terms of the network parameters and the reconstruction effects (PSNR). The SRFBN-S can achieve the best SR results among the networks with parameters fewer than 1000K. This demonstrates our method can well balance the number of parameters and the reconstruction performance. Meanwhile, in comparison with the networks with a large number of parameters, such as D-DBPN and EDSR, our proposed SRFBN and SRFBN+ can achieve competitive results, while only needs the 35% and 8% parameters of D-DBPN and EDSR, respectively. Thus, our network is lightweight and more efficient in comparison with other state-of-the-art methods.

4.5.2 Results with BI degradation model

For **BI** degradation model, we compare the SRFBN and SRFBN+ with seven state-of-the-art image SR methods: SRCNN[7], VDSR[18], DRRN[31], SRDenseNet[36], MemNet[36], EDSR[23], D-DBPN[11]. The quantitative results in Tab. 4 are re-evaluated from the corresponding public codes. Obviously, our proposed SRFBN can outperform almost all comparative methods. Compared with our method, EDSR utilizes much more number of filters (256 vs. 64), and D-DBPN employs more training images (DIV2K+Flickr2K+ImageNet vs. DIV2K+Flickr2K). However, our SRFBN can earn competitive results in contrast to them. In addition, it also can be seen that our SRFBN+ outperforms almost all comparative methods.

We show SR results with scale factor $\times 4$ in Fig. 7. In general, the proposed SRFBN can yield more convincing results. For the SR results of the 'BokuHaSitatakaKun' image from Manga109, DRRN and MemNet even split the 'M' letter. VDSR, EDSR and D-DBPN fail to recover the clear image. The proposed SRFBN produces a clear image which is very close to the ground truth. Besides, for the 'img_092' from Urban100, the texture direction of the SR images from all comparative methods is wrong. However, our proposed SRFBN makes full use of the high-level information to take a self-correcting process, thus a more faithful SR image can be obtained.

4.5.3 Results with BD and DN degradation models

As aforementioned, the proposed SRFBN is trained using curriculum learning strategy for **BD** and **DN** degradation models, and fine-tuned based on **BI** degradation model using DIV2K. The proposed SRFBN and SRFBN+ are compared with SRCNN[7], VDSR[18], IRCNN_G[43], IRCNN_C[43], SRMD(NF)[44], and RDN[47]. Because of degradation mismatch, SRCNN and VDSR are re-trained for **BD** and **DN** degradation models. As shown in Tab. 5, The proposed SRFBN and SRFBN+ achieve the best on almost all quantative results over other state-of-the-art methods.

Dataset	Scale	Bicubic	SRCNN [7]	VDSR [18]	DRRN [31]	MemNet [32]	SRFBN-S (Ours)	EDSR [23]	D-DBPN [11]	SRFBN (Ours)	SRFBN+ (Ours)
Set5	×2	33.66/0.9299	36.66/0.9542	37.53/0.9590	37.74/0.9591	37.78/0.9597	37.78/0.9597	38.11/0.9602	38.09/0.9600	38.11/0.9609	38.18/0.9611
	×3	30.39/0.8682	32.75/0.9090	33.67/0.9210	34.03/0.9244	34.09/0.9248	34.20/0.9255	34.65/0.9280	-/-	34.70/0.9292	34.77/0.9297
	×4	28.42/0.8104	30.48/0.8628	31.35/0.8830	31.68/0.8888	31.74/0.8893	31.98/0.8923	32.46/0.8968	32.47/0.8980	32.47/0.8983	32.56/0.8992
Set14	×2	30.24/0.8688	32.45/0.9067	33.05/0.9130	33.23/0.9136	33.28/0.9142	33.35/0.9156	33.92/0.9195	33.85/0.9190	33.82/0.9196	33.90/0.9203
	×3	27.55/0.7742	29.30/0.8215	29.78/0.8320	29.96/0.8349	30.00/0.8350	30.10/0.8372	30.52/0.8462	-/-	30.51/0.8461	30.61/0.8473
	×4	26.00/0.7027	27.50/0.7513	28.02/0.7680	28.21/0.7721	28.26/0.7723	28.45/0.7779	28.80/0.7876	28.82/0.7860	28.81/0.7868	28.87/0.7881
B100	×2	29.56/0.8431	31.36/0.8879	31.90/0.8960	32.05/0.8973	32.08/0.8978	32.00/0.8970	32.32/0.9013	32.27/0.9000	32.29/0.9010	32.34/0.9015
	×3	27.21/0.7385	28.41/0.7863	28.83/0.7990	28.95/0.8004	28.96/0.8001	28.96/0.8010	29.25/0.8093	-/-	29.24/0.8084	29.29/0.8093
	×4	25.96/0.6675	26.90/0.7101	27.29/0.7260	27.38/0.7284	27.40/0.7281	27.44/0.7313	27.71/0.7420	27.72/0.7400	27.72/0.7409	27.77/0.7419
Urban100	×2	26.88/0.8403	29.50/0.8946	30.77/0.9140	31.23/0.9188	31.31/0.9195	31.41/0.9207	32.93/0.9351	32.55/0.9324	32.62/0.9328	32.80/0.9341
	×3	24.46/0.7349	26.24/0.7989	27.14/0.8290	27.53/0.8378	27.56/0.8376	27.66/0.8415	28.80/0.8653	-/-	28.73/0.8641	28.89/0.8664
	×4	23.14/0.6577	24.52/0.7221	25.18/0.7540	25.44/0.7638	25.50//0.7630	25.71/0.7719	26.64/0.8033	26.38/0.7946	26.60/0.8015	26.73/0.8043
Manga109	×2	30.30/0.9339	35.60/0.9663	37.22/0.9750	37.60/0.9736	37.72/0.9740	38.06/0.9757	39.10/0.9773	38.89/0.9775	39.08/0.9779	39.28/0.9784
	×3	26.95/0.8556	30.48/0.9117	32.01/0.9340	32.42/0.9359	32.51/0.9369	33.02/0.9404	34.17/0.9476	-/-	34.18/0.9481	34.44/0.9494
	×4	24.89/0.7866	27.58/0.8555	28.83/0.8870	29.18/0.8914	29.42/0.8942	29.91/0.9008	31.02/0.9148	30.91/0.9137	31.15/0.9160	31.40/0.9182

Table 4. Average PSNR/SSIM values for scale factors ×2, ×3 and ×4 with **BI** degradation model. The best performance is shown in red and the second best performance is shown in blue.

Dataset	Model	Bicubic	SRCNN [7]	VDSR [18]	IRCNN_G [43]	IRCNN_C [43]	SRMD(NF) [44]	RDN [47]	SRFBN (Ours)	SRFBN+ (Ours)
Set5	**BD**	28.34/0.8161	31.63/0.8888	33.30/0.9159	33.38/0.9182	29.55/0.8246	34.09/0.9242	34.57/0.9280	34.66/0.9283	34.77/0.9290
	DN	24.14/0.5445	27.16/0.7672	27.72/0.7872	24.85/0.7205	26.18/0.7430	27.74/0.8026	28.46/0.8151	28.53/0.8182	28.59/0.8198
Set14	**BD**	26.12/0.7106	28.52/0.7924	29.67/0.8269	29.73/0.8292	27.33/0.7135	30.11/0.8364	30.53/0.8447	30.48/0.8439	30.64/0.8458
	DN	23.14/0.4828	25.49/0.6580	25.92/0.6786	23.84/0.6091	24.68/0.6300	26.13/0.6974	26.60/0.7101	26.60/0.7144	26.67/0.7159
B100	**BD**	26.02/0.6733	27.76/0.7526	28.63/0.7903	28.65/0.7922	26.46/0.6572	28.98/0.8009	29.23/0.8079	29.21/0.8069	29.28/0.8080
	DN	22.94/0.4461	25.11/0.6151	25.52/0.6345	23.89/0.5688	24.52/0.5850	25.64/0.6495	25.93/0.6573	25.95/0.6625	25.99/0.6636
Urban100	**BD**	23.20/0.6661	25.31/0.7612	26.75/0.8145	26.77/0.8154	24.89/0.7172	27.50/0.8370	28.46/0.8581	28.48/0.8581	28.68/0.8613
	DN	21.63/0.4701	23.32/0.6500	23.83/0.6797	21.96/0.6018	22.63/0.6205	24.28/0.7092	24.92/0.7362	24.99/0.7424	25.10/0.7458
Manga109	**BD**	25.03/0.7987	28.79/0.8851	31.66/0.9260	31.15/0.9245	28.68/0.8574	32.97/0.9391	33.97/0.9465	34.07/0.9466	34.43/0.9483
	DN	23.08/0.5448	25.78/0.7889	26.41/0.8130	23.18/0.7466	24.74/0.7701	26.72/0.8424	28.00/0.8590	28.02/0.8618	28.17/0.8643

Table 5. Average PSNR/SSIM values for scale factor ×3 with **BD** and **DN** degradation models. The best performance is shown in red and the second best performance is shown in blue.

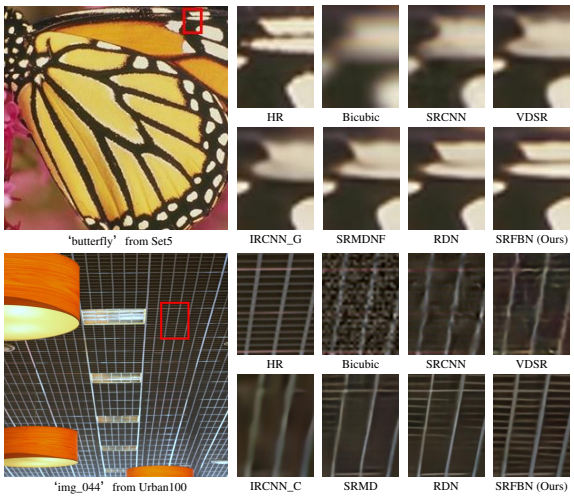


Figure 8. Visual results of **BD** and **DN** degradation models with scale factor ×3. The first set of images shows the results obtained from **BD** degradation model. The second set of images shows the results from **DN** degradation model.

In Fig. 8, we also show two sets of visual results with **BD** and **DN** degradation models from the standard benchmark datasets. Compared with other methods, the proposed SRFBN could alleviate the distortions and generate more accurate details in SR images. From above comparisions, we further indicate the robustness and effectiveness of SRFBN in handling **BD** and **DN** degradation models.

5. Conclusion

In this paper, we propose a novel network for image SR called super-resolution feedback network (SRFBN) to faithfully reconstruct a SR image by enhancing low-level representations with high-level ones. The feedback block (FB) in the network can effectively handle the feedback information flow as well as the feature reuse. In addition, a curriculum learning strategy is proposed to enable the network to well suitable for more complicated tasks, where the low-resolution images are corrupted by complex degradation models. The comprehensive experimental results have demonstrated that the proposed SRFBN could deliver the comparative or better performance in comparison with the state-of-the-art methods by using very fewer parameters.

Acknowledgement. The research in our paper is sponsored by National Natural Science Foundation of China (No.61701327 and No.61711540303), Science Foundation of Sichuan Science and Technology Department (No.2018GZ0178).

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *CVPRW*, 2017.

[2] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, 2009.

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

[4] Chunshui Cao, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, Deva Ramanan, and Thomas S. Huang. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *ICCV*, 2015.

[5] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *CVPR*, 2015.

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

[7] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *TPAMI*, 2016.

[8] Ruohan Gao and Kristen Grauman. On-demand learning for deep image restoration. In *ICCV*, 2017.

[9] Charles D Gilbert and Mariano Sigman. Brain states: top-down influences in sensory processing. *Neuron*, 2007.

[10] Wei Han, Shiyu Chang, Ding Liu, Mo Yu, Michael Witbrock, and Thomas S. Huang. Image super-resolution via dual-state recurrent networks. In *CVPR*, 2018.

[11] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *CVPR*, 2018.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, 2015.

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

[14] Gao Huang, Zhuang Liu, Van Der Maaten Laurens, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2016.

[15] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *CVPR*, 2015.

[16] Zheng Hui, Xiumei Wang, and Xinbo Gao. Fast and accurate single image super-resolution via information distillation network. In *CVPR*, 2018.

[17] J. M. Hupé, A. C. James, B. R. Payne, S. G. Lomber, P Girard, and J Bullier. Cortical feedback improves discrimination between figure and background by v1, v2 and v3 neurons. *Nature*, 1998.

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

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

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2014.

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

[22] Qianli Liao and Tomaso Poggio. Bridging the gaps between residual learning, recurrent neural networks and visual cortex. *arXiv preprint arXiv:1604.03640*, 2016.

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

[24] David R. Martin, Charless C. Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001.

[25] 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*, 2017.

[26] Tomer Peleg and Michael Elad. A statistical prediction model based on sparse representations for single image super-resolution. *TIP*, 2014.

[27] Anastasia Pentina, Viktoriia Sharmanska, and Christoph H. Lampert. Curriculum learning of multiple tasks. In *CVPR*, 2014.

[28] Deepak Babu Sam and R. Venkatesh Babu. Top-down feedback for crowd counting convolutional neural network. In *AAAI*, 2018.

[29] Samuel Schulter, Christian Leistner, and Horst Bischof. Fast and accurate image upscaling with super-resolution forests. In *CVPR*, 2015.

[30] Marijn F Stollenga, Jonathan Masci, Faustino Gomez, and Jürgen Schmidhuber. Deep networks with internal selective attention through feedback connections. In *NIPS*. 2014.

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

[32] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *ICCV*, 2017.

[33] Radu Timofte, Vincent De Smet, and Luc Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *ICCV*, 2013.

[34] Radu Timofte, Vincent De Smet, and Luc Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *ACCV*, 2015.

[35] Radu Timofte, Rasmus Rothe, and Luc Van Gool. Seven ways to improve example-based single image super resolution. In *CVPR*, 2016.

[36] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *ICCV*, 2017.

[37] 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 *ECCVW*, 2018.

[38] Yifan Wang, Federico Perazzi, Brian Mcwilliams, Alexander Sorkinehornung, Olga Sorkinehornung, and Christopher Schroers. A fully progressive approach to single-image super-resolution. In *CVPRW*, 2018.

[39] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *TIP*, 2004.

[40] Amir R. Zamir, Te-Lin Wu, Lin Sun, William B. Shen, Bertram E. Shi, Jitendra Malik, and Silvio Savarese. Feedback networks. In *CVPR*, 2017.

[41] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *Curves and Surfaces*, 2010.

[42] Kaibing Zhang, Xinbo Gao, Dacheng Tao, Xuelong Li, et al. Single image super-resolution with non-local means and steering kernel regression. *TIP*, 2012.

[43] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep CNN denoiser prior for image restoration. In *CVPR*, 2017.

[44] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *CVPR*, 2017.

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

[46] 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.

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