Feedback Network for Image Super-Resolution
– *Supplementary Material* –

Abstract

The following items are contained in the supplementary material:
1. Discussions on the feedback block.
2. More insights on the feedback mechanism.
3. Quantitative results using DIV2K training images.
4. Running time comparison.
5. More qualitative results.

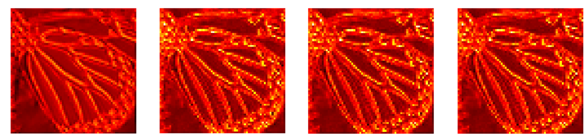

Figure 1. Average feature maps of refined low-level features from different iterations in the propose SRFBN (zoom for a better view). All average feature maps use the same colormap for better visualization.

1. Study of Feedback Block

More effective basic block could generate finer high-level representations and then benefits our feedback process. Thus, we explore the design of the basic block in this section. We still use SRFBN-L (T=4, G=6), which has a small base number of fiters (m=32) for analysis.

Ablation study mainly focuses on two components of our feedback block (FB): (1) up- and down-sampling layers (UDSL), (2) dense skip connecitons (DSC). To analysis the effect of UDSL in our proposed FB, we replace the up- and down-sampling layers with 3×3 sized convolutional layers (with one padding and one stridding). In Tab. 1, when UDSL is replaced with 3×3 sized convolutional layers in the FB, the PSNR value dramatically decreases. This indicates that up- and down-sampling operations carrying large kernel size can expliot abundant contextual information and are effective for image super-resolution (SR). After adding DSC to the FB, the reconstruction performance can be further improved, because the information efficiently flows through DSC across hierarchy layers and even across time.

Different combinations of UDSL and DSC				
UDSL	✗	✔	✗	✔
DSC	✗	✗	✔	✔
PSNR	31.41	32.05	31.62	**32.11**

Table 1. The investigation of up- and down-sampling layers (UDSL), and dense skip connection (DSC) with scale factor $\times 4$ on Set5.

Other basic blocks are considered in this experiment in comparison with our FB. We choose two superior basic blocks (*i.e.* projection units[2] and RDB[7]), which were designed for image SR task recently, and ConvLSTM from [5] for comparison. To keep consistency, the number of convolutional layers[1] and filters in each basic block is set to 12 and 32, respectively. In Tab. 2, we first see that all SR custom basic blocks outperform ConvLSTM by a large margin. The gate mechanisms in ConvLSTM influence the distribution and intensity of original images and thus are difficult to meet high fidelity needs in image SR tasks. Besides, high-level information is directly added to low-level information in ConvLSTM, causing the loss of enough contextual information for the next iteration. Noticeably, our proposed FB obtains the best quantitative results in comparison with other basic blocks. This further demonstrates the powerful representation ability of our proposed FB.

2. Additional Insights on Feedback Mechanism

For better understanding the feedback mechanism in the proposed network, we visualize the average feature map of L_0^t at each iteration t in Fig. 1. L_0^t actually represents the low-level representations refined by high-level features F_{out}^{t-1} from last iteration (see the main paper's Eq. 5). The initial state F_{out}^0 is set to F_{in}^1, hence the first iteration in the proposed network can't receive the feedback information. From Fig. 1, we observe that, except the first iteration (t=1), these average feature maps show bright activations in the contours and outline edges of the original image. It seems that the feedback connection adds the high-level representations to the initial feature maps. This further indicates that

[1] 1×1 sized convolutional layers are omitted.

	ConvLSTM	Projection units	RDB	Ours
PSNR	31.26	32.07	32.01	**32.11**

Table 2. The investigation of other block design with scale factor $\times 4$ on Set5.

	Params.	Set5	Set14	B100	Urban100	Manga109
MemNet-Pytorch	677K	31.75/0.889	28.31/0.775	27.37/0.729	25.54/0.766	29.65/0.897
D-DBPN [2]	10,426K	**32.40/0.897**	28.75/0.785	27.67/0.738	26.38/0.793	30.89/0.913
SRFBN-S (Ours)	483K	31.98/0.892	28.45/0.778	27.44/0.731	25.71/0.772	29.91/0.901
SRFBN (Ours)	3,631K	32.39/**0.897**	**28.77/0.786**	**27.68/0.740**	**26.47/0.798**	**30.96/0.914**

Table 3. Average PSNR/SSIM values for scaling factor $\times 4$ using **BI** degradation model. The networks used for comparison are all trained using DIV2K training images. The best performance is **highlighted**.

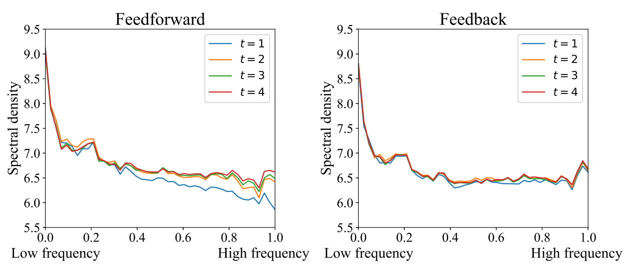

Figure 2. The spectral densities of the average feature map at each iteration t (zoom for a better view). From left to right of the vertical axis, frequency is normalized and ranged from low to high for better visualization.

initial low-level features, which lack enough contextual information, surely are corrected using high-level information through the feedback mechanism in the proposed network.

To further conduct differences between feedforward and feedback networks, we plot 1-D spectral densities of the average feature map at each iteration t in SRFBN-L (feedback) and SRFBN-L-FF (feedforward). As shown in Fig.5 of the main paper, each average feature map is the mean of F^t_{out}. To acquire 1-D spectral densities of the average feature map at each iteration t, we get the 2-D spectrum map through discrete Fourier transform, center the low-frequency component of the spectrum map, and place concentric annular regions to compute the mean of spectral densities for continuous frequency ranges. From Fig. 2, we can conclude that the feedback network can estimate more mid-frequency and high-frequency information than the feedforward network at early iterations. With the iteration t grows, the feedforward network gradually recovers mid-frequency and high-frequency components, while the feedback network pays attention to refine the well-developed information. For the feedback network, we also observe that, because of the help of the feedback mechanism (t >1), mid-frequency and high-frequency information of the average feature map at the second iteration (t=2) is more similiar to the final representations (t=4) in contrast to the first iteration (t=1).

3. Sanity Check

To purely investigate the effect of the network architecture design, we compare the quantitative results obtained from different networks using the same training dataset (DIV2K training images[1]). The choices of networks for comparison include D-DBPN (which is a state-of-the-art network with moderate parameters) and MemNet[4] (which is the leading network with recurrent structure). Because MemNet only reveals the results trained using 291 images, we re-train it using DIV2K on Pytorch framework. The results of D-DBPN are cited from their supplementary materials. Our SRFBN-S (T=4, G=3, m=32) and final SRFBN (T=4, G=6, m=64) are provided for this comparison. In Tab. 3, our SRFBN-S shows better quantitative results than MemNet with 71% fewer parameters. Moreover, the final SRFBN also gains competitive results in contrast to D-DBPN especially on Urban100 and Manga109 datasets, which mainly contain images with a large size. This comparison shows the effectiveness of the proposed SRFBN.

Model	Running time (s)	PSNR
MemNet-Pytorch	0.481	25.54
EDSR	1.218	26.64
D-DBPN	0.015	26.38
RDN	1.268	26.61
RCAN	1.130	26.82
SRFBN-S (Ours)	0.006	25.71
SRFBN (Ours)	0.011	26.60

Table 4. Average running time comparison on Urban100 with scale factor 4 on an NVIDIA 1080Ti GPU.

4. Running Time Comparison

We compare running time of our proposed SRFBN-S and SRFBN with five state-of-the-art networks: MemNet[4], EDSR[3], D-DBPN[2], RDN[7] and RCAN[6] on Urban100 with scale factor $\times 4$. Because the large memory consumption in Caffe, we re-implement MemNet on Pytorch for fair comparison. The running time of all networks is evaluated on the same machine with 4.2GHz Intel i7 CPU (16G RAM) and an NVIDIA 1080Ti GPU using their official codes. Tab. 4 shows that our SRFBN-S and SRFBN have the fastest evaluation time in comparison with other networks. This further reflects the effectiveness of our proposed networks. The quantitative results of our proposed networks are less comparable with RCAN, but RCAN mainly focuses on much deeper networks design (about 400 convolutional layers) to purchase more accurate SR results. In contrast, our SRFBN only has about 100 convolutional layers with 77% fewer parameters (3,631K vs. 15,592K) than RCAN.

5. More Qualitative Results

In Fig. 3-14, we provide more visual results of different degradation models to prove the superiority of the proposed network.

References

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

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

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

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

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

[6] 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.

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

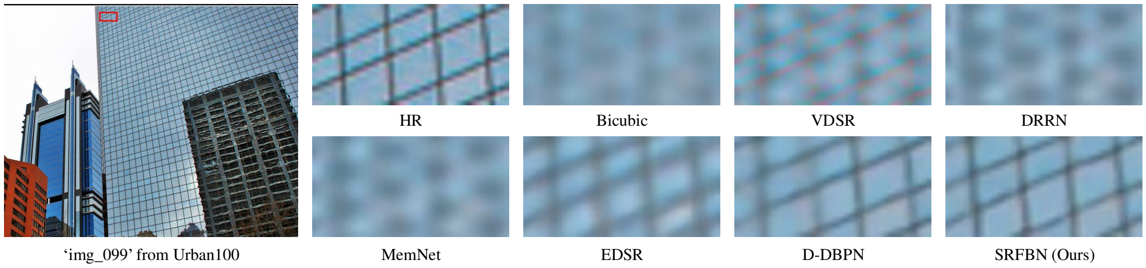

Figure 3. Visual results of **BI** degradation model with scale factor $\times 4$.

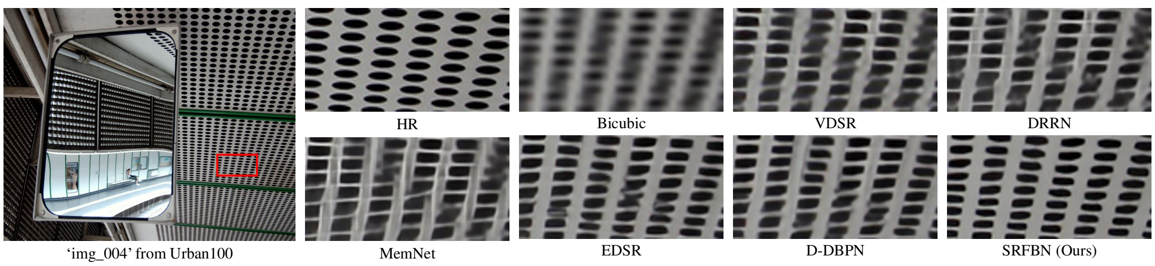

Figure 4. Visual results of **BI** degradation model with scale factor $\times 4$.

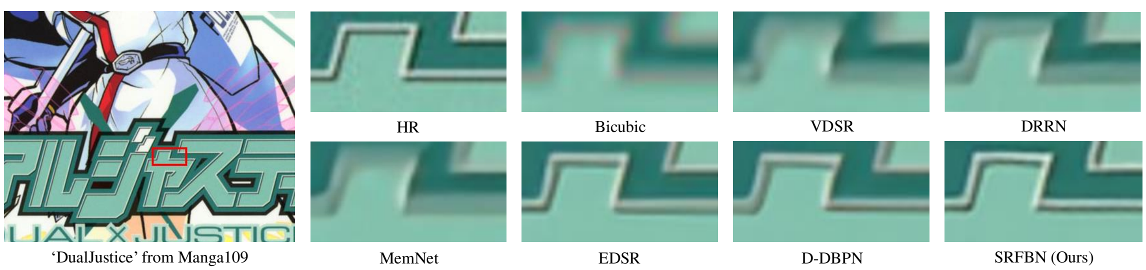

Figure 5. Visual results of **BI** degradation model with scale factor $\times 4$.

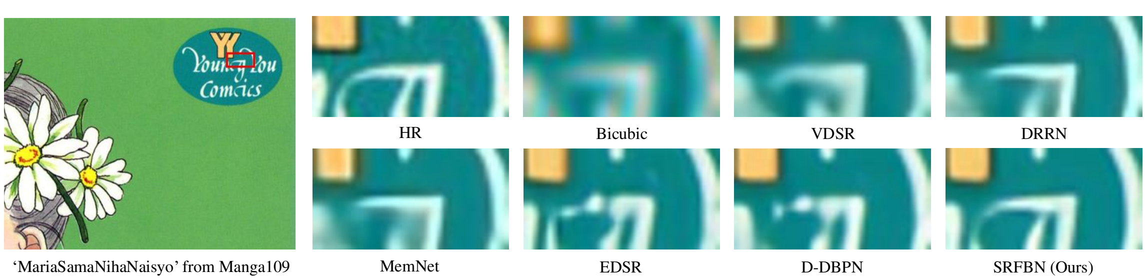

Figure 6. Visual results of **BI** degradation model with scale factor $\times 4$.

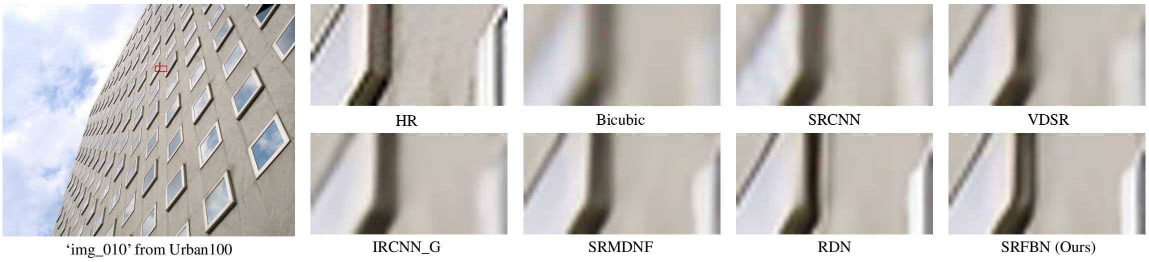

Figure 7. Visual results of **BD** degradation model with scale factor $\times 4$.

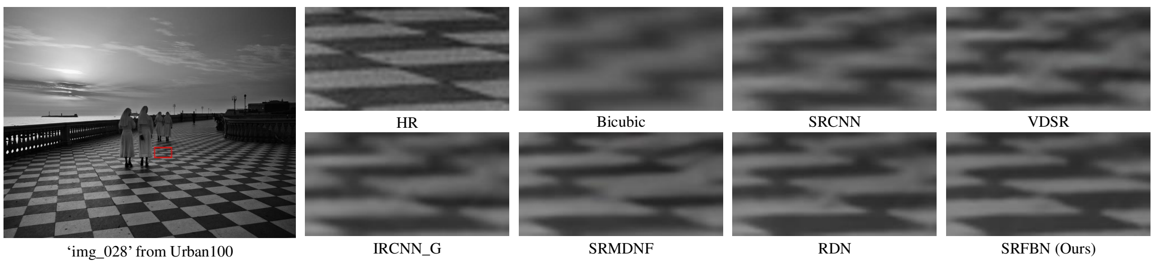

Figure 8. Visual results of **BD** degradation model with scale factor $\times 4$.

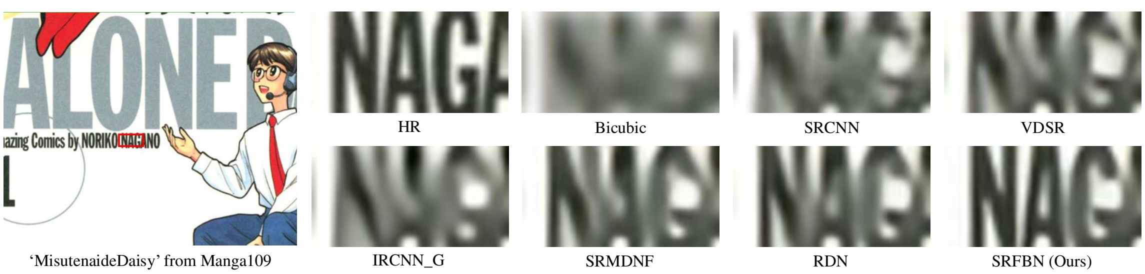

Figure 9. Visual results of **BD** degradation model with scale factor $\times 4$.

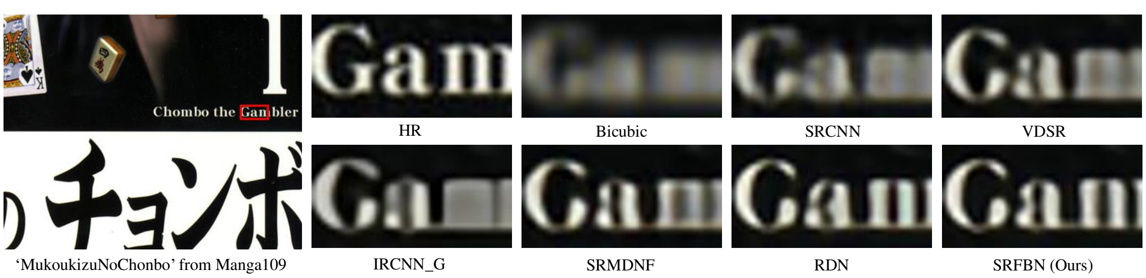

Figure 10. Visual results of **BD** degradation model with scale factor $\times 4$.

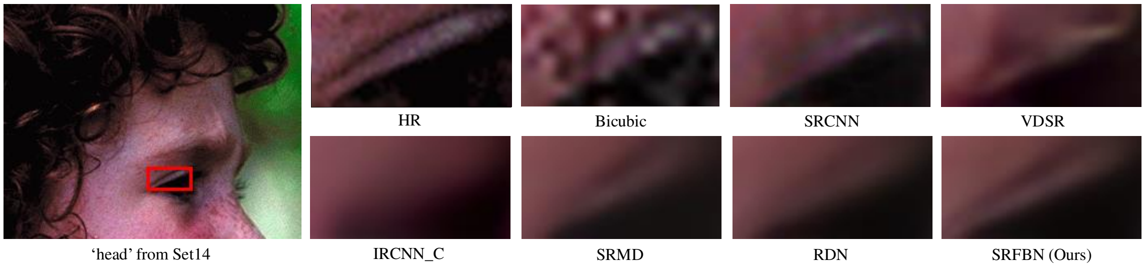

Figure 11. Visual results of **DN** degradation model with scale factor $\times 4$.

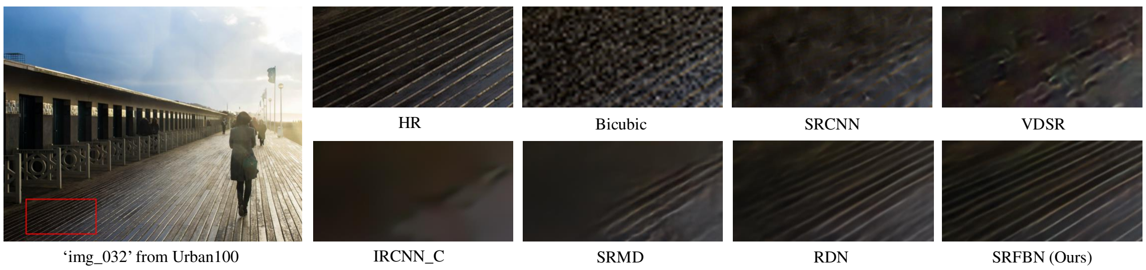

Figure 12. Visual results of **DN** degradation model with scale factor $\times 4$.

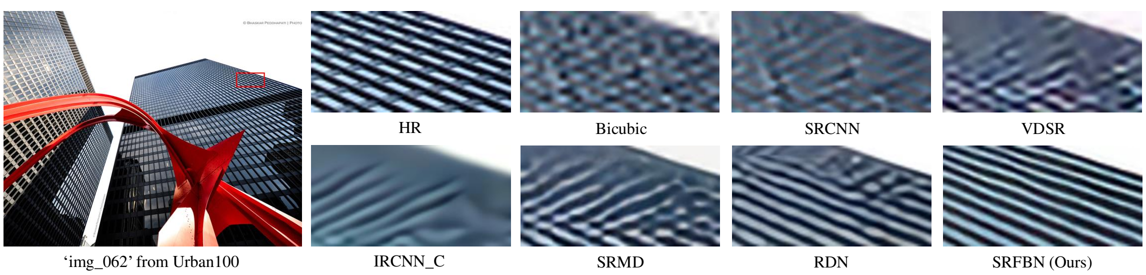

Figure 13. Visual results of **DN** degradation model with scale factor $\times 4$.

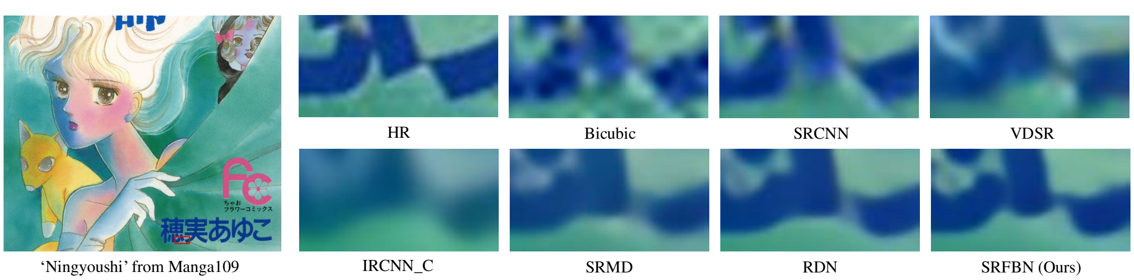

Figure 14. Visual results of **DN** degradation model with scale factor $\times 4$.