Deep Back-Projection Networks For Super-Resolution
— Supplementary Material —

Muhammad Haris[1], Greg Shakhnarovich[2], and Norimichi Ukita[1],
[1]Toyota Technological Institute, Japan [2]Toyota Technological Institute at Chicago, United States
{mharis, ukita}@toyota-ti.ac.jp, gregory@ttic.edu

Project Page

1. Overview

We provide additional materials for better understanding of our proposed networks. First, we provide the detailed architectures from the variants of DBPN. Second, we present additional analysis of DBPN. Last, we provide additional qualitative results from our networks compare to the state-of-the-arts methods.

2. Implementation Details of Networks Architecture

There are six variants of DBPN which is shown in the paper: DBPN-SS, DBPN-S, DBPN-M, DBPN-L, D-DBPN-L, and D-DBPN. The detailed architectures of those networks are shown in Table 1.

3. Additional Analysis

3.1. Sanity Check

We compare D-DBPN, VDSR [4] and EDSR [6]. Larger dataset is used for fair comparison which is DIV2K [8]. In Table 2, we report results of training with DIV2K, with total number 800 training images, on the x2, x4, and x8 super-resolution task. Here, we use our PyTorch implementation which is also publicly available in the internet. For reference, we also include original VDSR trained on BSDS200 [1] + T91 [9] dataset.

From Table 2, it's evident that our network has overall better performance than either version of VDSR.

3.2. Error Feedback

As stated in our manuscript, error feedback (EF) is used to guide the reconstruction in the early layer. Here, we analyze how error feedback can help for better reconstruction. For the scenario without EF, we replace up- and down-projection unit with single up- and down-sampling (deconvolution and convolution) layer.

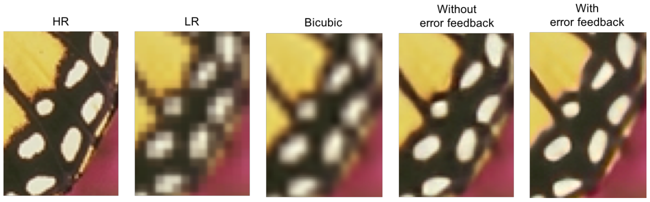

Figure 1. Qualitative comparisons of DBPN-S with EF and without EF on 4× enlargement.

We show PSNR of DBPN-S with EF and without EF in Table 3. The result with EF has 0.53 dB and 0.26 dB better than without EF on Set5 and Set14, respectively. In Fig. 1, we visually show how error feedback can construct better and sharper HR image especially in the white stripe pattern of the wing.

The performance of DBPS-S without EF is interestingly 0.57 dB and 0.35 dB better than SRCNN [2], FSRCNN [3], respectively, on Set5. These results show the effectiveness of our mutual-connected up- and downsampling layers which can demonstrate the LR-HR mutual dependency by mapping LR features to HR space, then project it back to the LR space.

3.3. Convergence Curve

In Fig. 2 and Fig. 3, we show the convergence curve of $4\times$ and $8\times$ enlargement from each proposed network in the manuscript. Our proposed networks have fast convergence speed especially for D-DBPN where the results of 50k iteration can outperform the state-of-the-art methods except for EDSR.

3.4. Filter Size

We analyze the size of filters which is used in the back-projection stage. In the manuscript, we stated that the choice of filter size in the back-projection stage is based on the preliminary results. For the $4\times$ enlargement, we show

Table 1. Network Architecture of DBPN variants. ”Feat0” and ”Feat1” refer to first and second convolutional layer in the initial feature extraction stages. Note: conv(f, n, st, pd) where f is filter size, n is number of filters, st is striding, and pd is padding

	Scale	DBPN-SS	DBPN-S	DBPN-M	DBPN-L	D-DBPN-L	D-DBPN
Input/Output		Luminance	Luminance	Luminance	Luminance	Luminance	RGB
Feat0		conv(3,64,1,1)	conv(3,128,1,1)	conv(3,128,1,1)	conv(3,128,1,1)	conv(3,128,1,1)	conv(3,256,1,1)
Feat1		conv(1,18,1,0)	conv(1,32,1,0)	conv(1,32,1,0)	conv(1,32,1,0)	conv(1,32,1,0)	conv(1,64,1,0)
Reconstruction		conv(1,1,1,0)	conv(1,1,1,0)	conv(1,1,1,0)	conv(1,1,1,0)	conv(1,1,1,0)	conv(3,3,1,1)
BP stages	2×	conv(6,18,2,2)	conv(6,32,2,2)	conv(6,32,2,2)	conv(6,32,2,2)	conv(6,32,2,2)	conv(6,64,2,2)
	4×	conv(8,18,4,2)	conv(8,32,4,2)	conv(8,32,4,2)	conv(8,32,4,2)	conv(8,32,4,2)	conv(8,64,4,2)
	8×	conv(12,18,8,2)	conv(12,32,8,2)	conv(12,32,8,2)	conv(12,32,8,2)	conv(12,32,8,2)	conv(12,64,8,2)
Parameters (k)	2×	106	337	779	1221	1230	5819
	4×	188	595	1381	2168	2176	10291
	8×	421	1332	3101	4871	4879	23071
Depth		12	12	24	36	40	52
No. of stage (T)		2	2	4	6	6	7
Dense connection		No	No	No	No	Yes	Yes

Table 2. Quantitative evaluation of state-of-the-art SR algorithms on DIV2K data sets: average PSNR/SSIM for scale factors 2×, 4× and 8×. Red indicates the best and blue indicates the second best performance.

		Set5		Set14		BSDS100		Urban100		Manga109	
Algorithm	Scale	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
VDSR [4]	2	37.53	0.958	32.97	0.913	31.90	0.896	30.77	0.914	37.16	0.974
VDSR-DIV2K	2	37.55	0.958	32.98	0.913	31.93	0.896	30.78	0.915	37.20	0.976
EDSR [6]	2	38.11	0.960	33.92	0.919	32.32	0.901	32.93	0.935	39.10	0.977
D-DBPN	2	38.05	0.960	33.79	0.919	32.25	0.900	32.51	0.932	38.81	0.976
VDSR [4]	4	31.35	0.882	28.03	0.770	27.29	0.726	25.18	0.753	28.82	0.886
VDSR-DIV2K	4	31.37	0.882	28.04	0.771	27.31	0.727	25.25	0.754	28.90	0.888
EDSR [6]	4	32.46	0.897	28.80	0.788	27.71	0.742	26.64	0.803	31.02	0.915
D-DBPN	4	32.40	0.897	28.75	0.785	27.67	0.738	26.38	0.793	30.89	0.913
VDSR [4]	8	25.72	0.711	24.21	0.609	24.37	0.576	21.54	0.560	22.83	0.707
VDSR-DIV2K	8	25.99	0.729	24.28	0.614	24.46	0.579	21.77	0.573	23.21	0.721
EDSR [6]	8	26.97	0.775	24.94	0.640	24.80	0.596	22.47	0.620	24.58	0.778
D-DBPN	8	27.25	0.785	25.14	0.649	24.91	0.602	22.72	0.630	25.14	0.798

Table 3. Analysis of EF using DBPN-S on 4× enlargement. Red indicates the best performance.

	Set5	Set14
SRCNN [2]	30.49	27.61
FSRCNN [3]	30.71	27.70
Without EF	31.06	27.95
With EF	31.59	28.21

that filter 8×8 is 0.08 dB and 0.09 dB better than filter 6×6 and 10×10, respectively, as shown in Table 4.

Table 4. Analysis of filter size in the back-projection stages on 4× enlargement from D-DBPN. Red indicates the best performance.

Filter size	Striding	Padding	Set5	Set14
6	4	1	32.39	28.78
8	4	2	32.47	28.82
10	4	3	32.38	28.79

3.5. Luminance vs RGB

In the final network (D-DBPN), we change input/output from luminance to RGB color channels. There is no significant improvement in the quality of the result as shown in Table 5. However, it might reduce the complexity and simplify the implementation by avoiding the use of another interpolation techniques, such as Bicubic, to process other channels.

Table 5. Analysis of input/output color channel using DBPN-L. Red indicates the best performance.

	Set5	Set14
RGB	31.88	28.47
Luminance	31.86	28.47

3.6. Performance gain on dense connections

Table 6 gives detailed information about performance gains derived from dense connections. We compare DBPN-L to the same architecture augmented with dense connec-

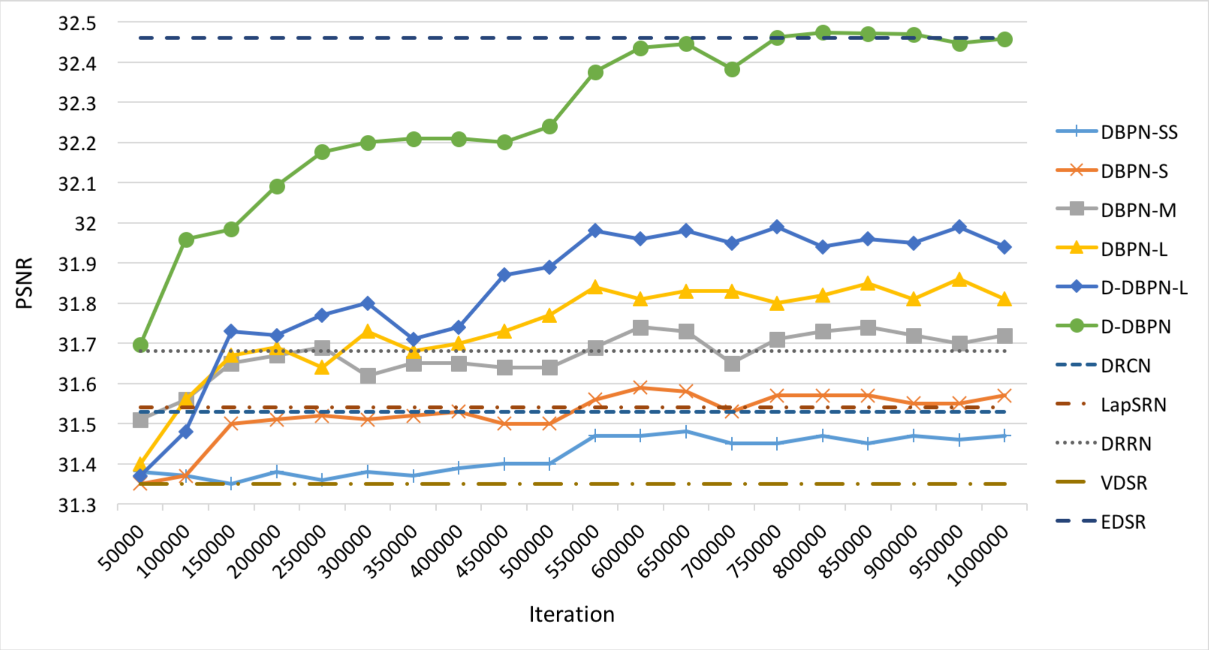

Figure 2. Convergence curve for 4× enlargement on Set5.

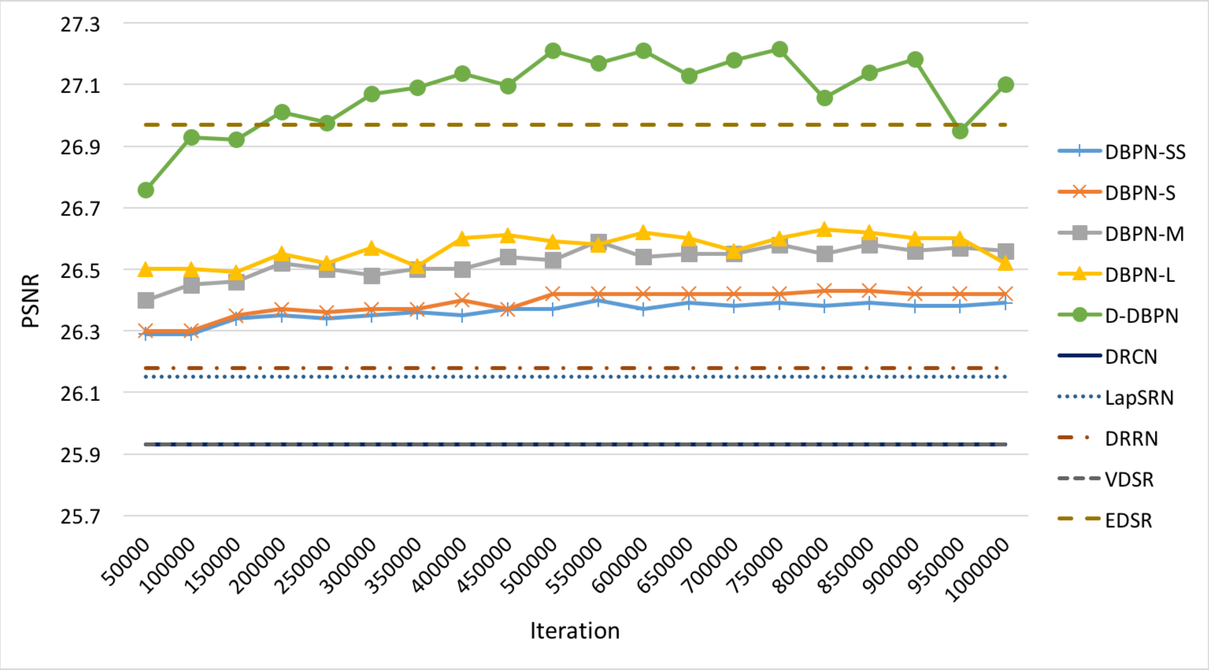

Figure 3. Convergence curve for 8× enlargement on Set5.

tions, D-DBPN-L. The results demonstrate that D-DBPN-L achieves superior performance on all test sets.

3.7. Runtime Evaluation

We present the runtime comparisons between our networks and 3 state-of-the-art networks: VDSR [4], DRRN [7], and EDSR [6]. The comparison must be done in fair settings. Therefore, we choose only three methods which have the same in nature with our implementation using Caffe. The runtime is calculated using python function `timeit` which encapsulating `forward` function in Caffe. For EDSR, we use original author code based on Torch and use `timer` function to obtain the runtime.

We evaluate each network using Nvidia TITAN X GPU (12G Memory). The input image size is 64× 64, then upscaled into 128× 128 (2×), 256× 256 (4×), and 512× 512

Table 6. Comparison of the DBPN-L and D-DBPN-L on 4× enlargement. (* indicates that the input is divided into four parts and calculated separately due to computation limitation of Caffe)

	DBPN-L		D-DBPN-L	
Algorithm	PSNR	SSIM	PSNR	SSIM
Set5	31.86	0.891	31.99	0.893
Set14	28.47	0.777	28.52	0.778
BSDS100	27.50	0.732	27.53	0.733
Urban100*	26.65	0.780	26.76	0.783

(8×). The results are the average of 10 times trials.

Table 7 shows the runtime comparisons on 2×, 4×, and 8× enlargement. It shows that our SS and S networks obtain the best and second best performance on 4× and 8× enlargement. On 2× enlargement, we did not construct the variants of our proposed network except for D-DBPN. Therefore, we cannot produce the runtime for SS, S, M, and L networks. Compare to EDSR, our final network (D-DBPN) show its effectiveness by having faster runtime with comparable quality on 2× and 4× enlargement. On 8× enlargement, the gap is bigger. It shows that D-DBPN has better results with lower runtime than EDSR.

Noted that input for VDSR and DRRN is only luminance channel and need preprocessing to create middle-resolution image. So that, the runtime should be added by additional computation of interpolation computation on preprocessing.

Table 7. Runtime evaluation with input size 64×64. Red indicates the best and blue indicates the second best performance, * indicates the calculation using function timer in Torch, and N.A. indicates that the algorithm runs out of GPU memory.

	2× (128×128)	4× (256×256)	8× (512×512)
VDSR [4]	0.02223	0.03225	0.06856
DRRN [7]	0.25413	0.32893	N.A.
*EDSR [6]	0.8579	1.2458	1.1477
DBPN-SS	-	0.01672	0.02692
DBPN-S	-	0.02073	0.03812
DBPN-M	-	0.04511	0.08106
DBPN-L	-	0.06971	0.12635
D-DBPN	0.15331	0.19396	0.31851

4. Additional Qualitative Results

In Fig. 4-16, we provide additional results for 8× enlargement to clearly show the effectiveness of our proposed network. The comparisons focus to compare between top-3 current state-of-the-art networks which are LapSRN [5], EDSR [6], and D-DBPN. The complete results on all datasets will be published in our website.

References

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2011. 1

[2] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2016. 1, 2

[3] C. Dong, C. C. Loy, and X. Tang. Accelerating the super-resolution convolutional neural network. In *European Conference on Computer Vision*, pages 391–407. Springer, 2016. 1, 2

[4] J. Kim, J. Kwon Lee, and K. 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, June 2016. 1, 2, 3, 4

[5] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *IEEE Conferene on Computer Vision and Pattern Recognition*, 2017. 4

[6] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee. Enhanced deep residual networks for single image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017. 1, 2, 3, 4

[7] Y. Tai, J. Yang, and X. Liu. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 3, 4

[8] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, L. Zhang, B. Lim, S. Son, H. Kim, S. Nah, K. M. Lee, et al. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 1110–1121. IEEE, 2017. 1

[9] J. Yang, J. Wright, T. S. Huang, and Y. Ma. Image super-resolution via sparse representation. *Image Processing, IEEE Transactions on*, 19(11):2861–2873, 2010. 1

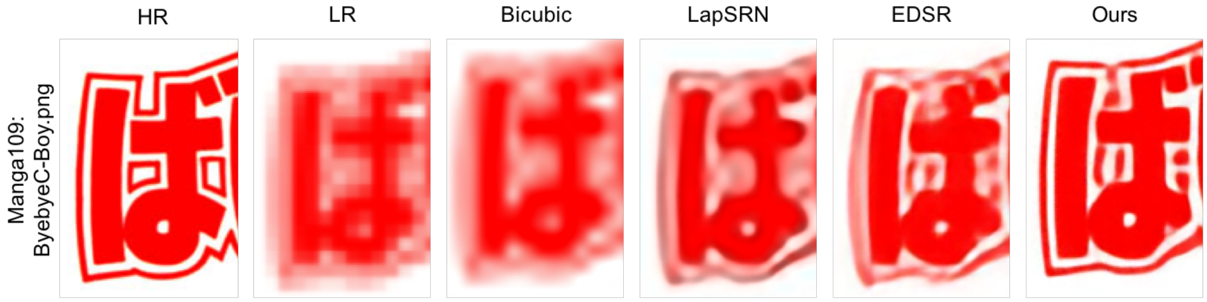

Figure 4. Visual comparison for 8× enlargement. D-DBPN is able to separate clearly between the hiragana word and outer stripe pattern.

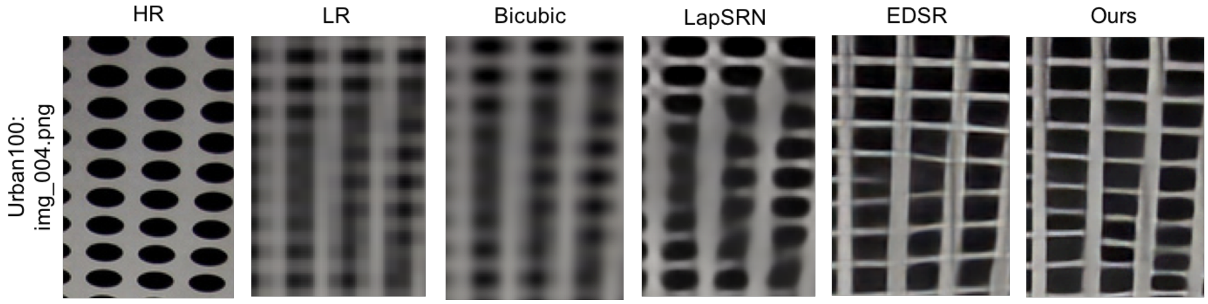

Figure 5. Visual comparison for 8× enlargement. All networks fail to keep the shape consistency from the HR image. However, the correct number of holes in the image is only achieved by D-DBPN.

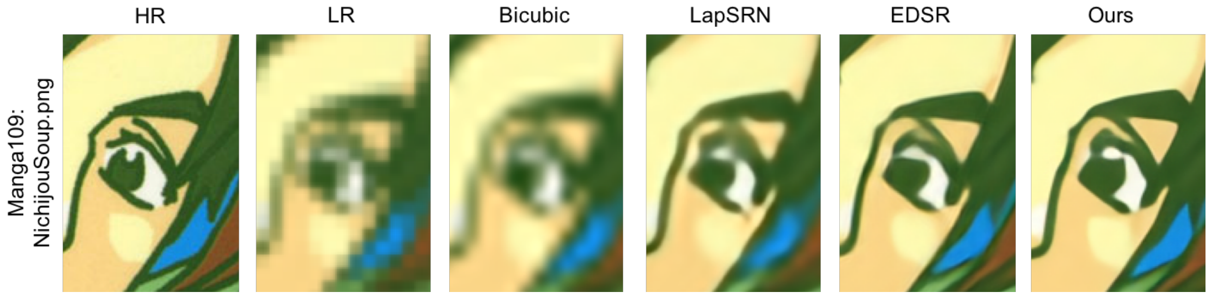

Figure 6. Visual comparison for 8× enlargement. D-DBPN is able to construct shaper eyelashes close to the ground truth.

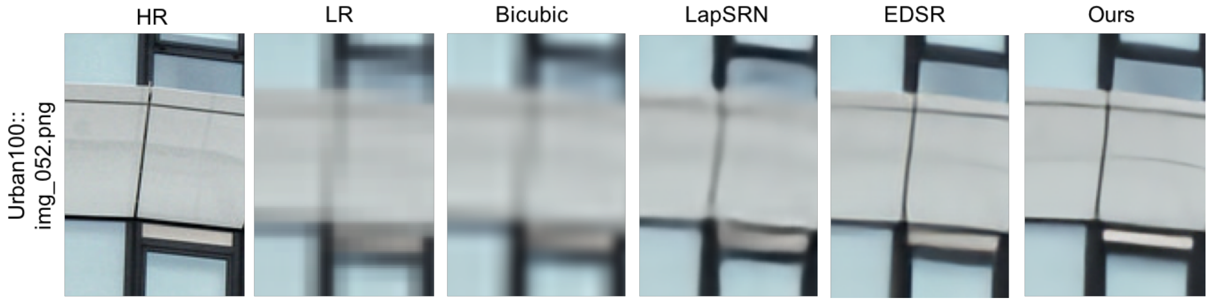

Figure 7. Visual comparison for 8× enlargement. D-DBPN is able to construct sharper edges. However, it also creates soft black stripes in the middle part of the wall.

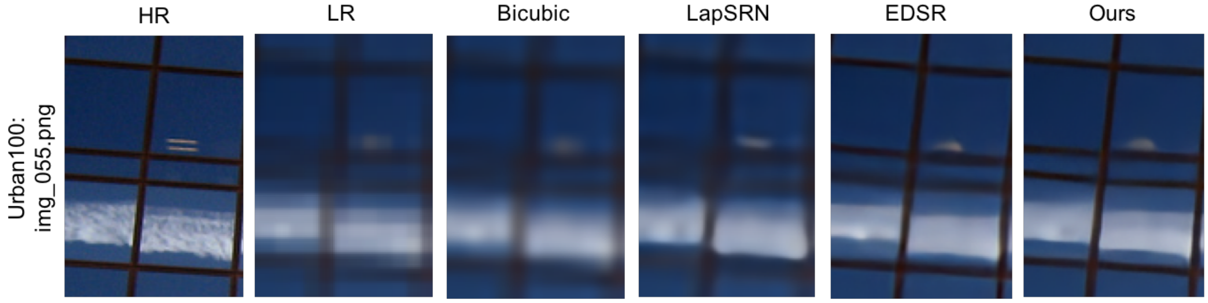

Figure 8. Visual comparison for 8× enlargement. D-DBPN is able to construct sharper edges from the windows.

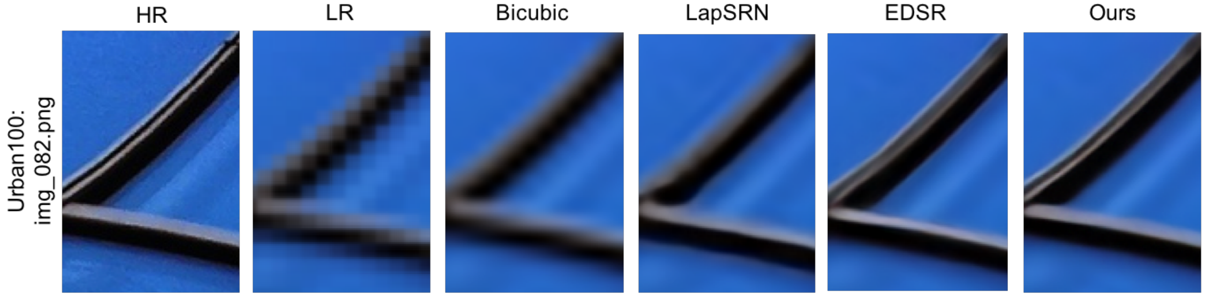

Figure 9. Visual comparison for 8× enlargement. D-DBPN is able to construct more detailed patterns compare to LapSRN and EDSR.

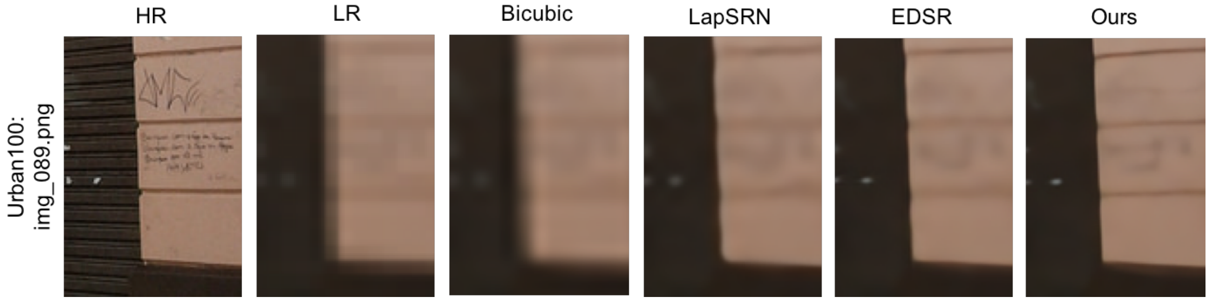

Figure 10. Visual comparison for 8× enlargement. D-DBPN is able to preserve the stripe pattern in the wall.

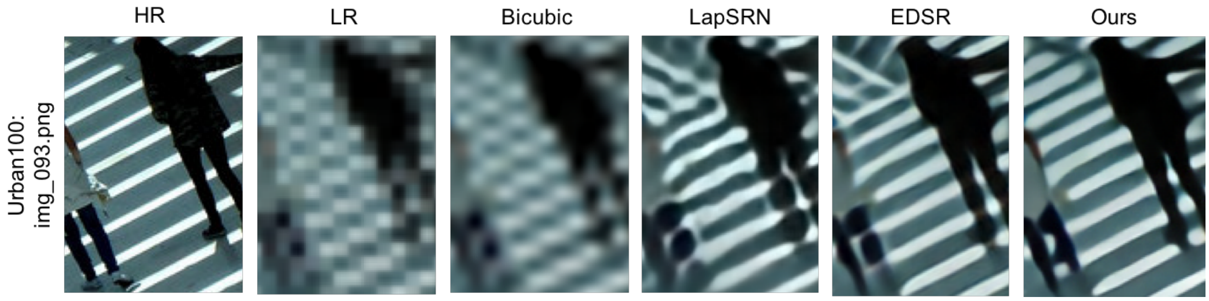

Figure 11. Visual comparison for 8× enlargement. D-DBPN is able to construct the white stripes better than LapSRN and EDSR.

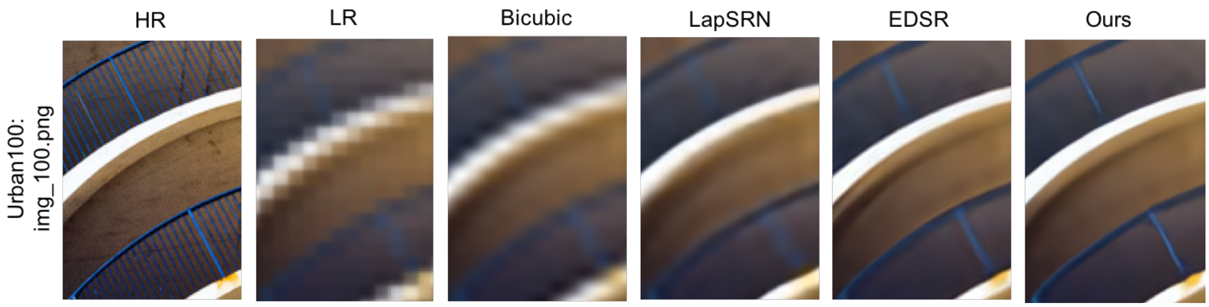

Figure 12. Visual comparison for 8× enlargement. D-DBPN is able to construct sharper the blue bars pattern.

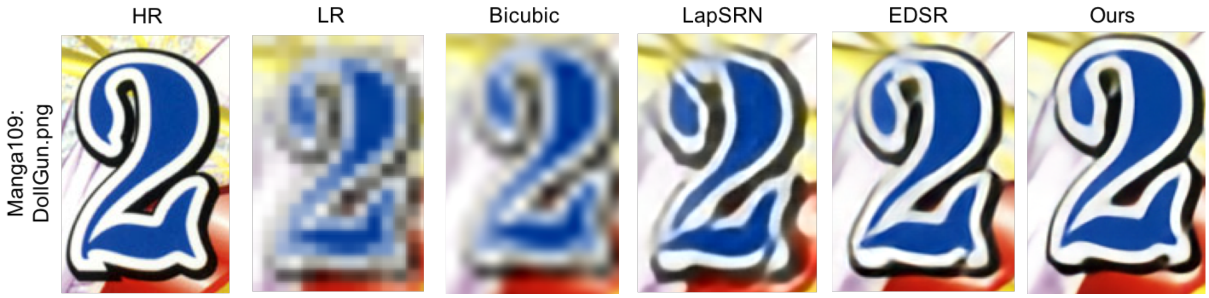

Figure 13. Visual comparison for 8× enlargement. D-DBPN is able to construct sharper pattern of ”2” than LapSRN and EDSR.

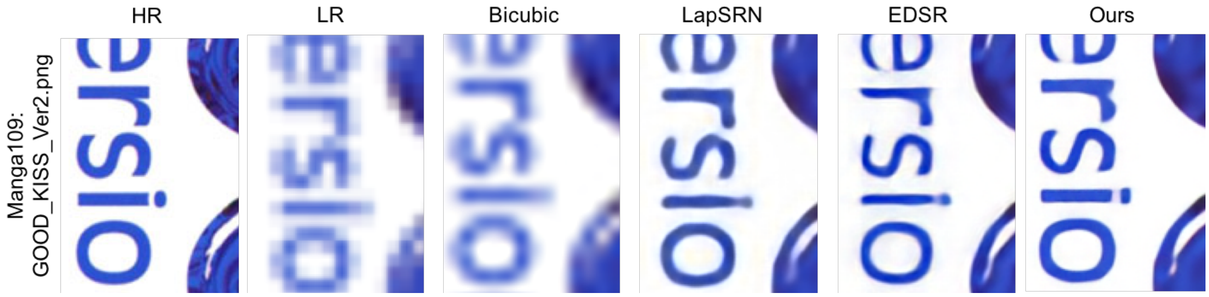

Figure 14. Visual comparison for 8× enlargement. D-DBPN is able to construct the characters sharper than LapSRN and EDSR.

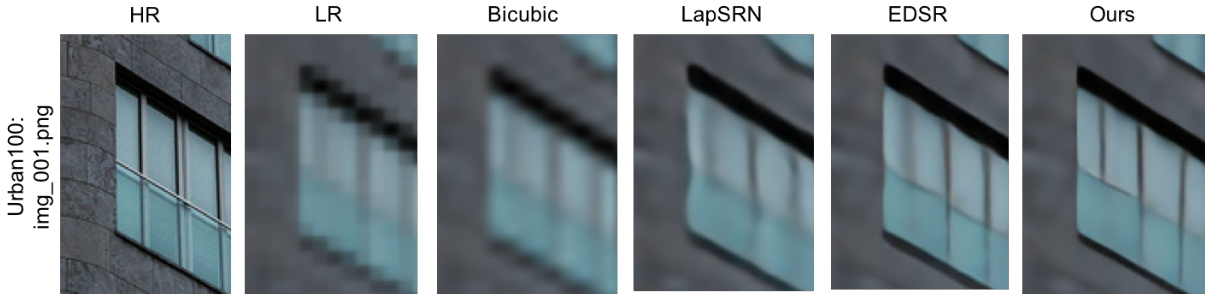

Figure 15. Visual comparison for 8× enlargement. D-DBPN is able to construct the bars in the window.

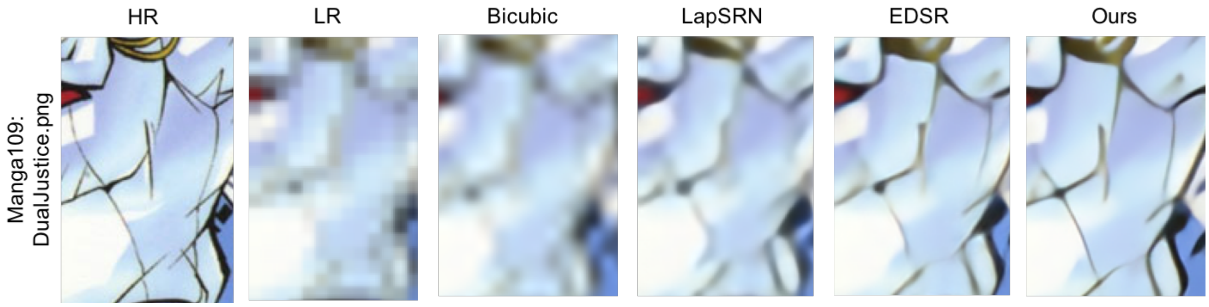

Figure 16. Visual comparison for 8× enlargement. D-DBPN is able to preserve the sketch pattern (light black stripes) in the image better than LapSRN and EDSR.