Recurrent Back-Projection Network for Video Super-Resolution
—Supplementary Materials—

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, greg@ttic.edu

1. Additional Experimental Results

1.1. Multiple scale factors

To enrich the evaluation of RBPN, we provide the results for multiple scaling factors (i.e., 2× and 8×) on Vimeo-90k [10], SPMCS-32 [9], and Vid4 [6] as shown in Table 1. Due to the limitation on other methods, the scores for other methods were copied from the respective publications. RBPN is superior to existing methods on all test sets except the SSIM score on Vid4 in scaling factor 2×. However, note that the difference between the best score (i.e., VSR-DUF) and our score is only 0.001.

1.2. Network size

We observe the performance of RBPN on different network sizes. The "original" RBPN use the same setup as in the main paper. We use DBPN [2] for $\texttt{Net}_{sisr}$, and Resnet [3] for $\texttt{Net}_{misr}$, $\texttt{Net}_{res}$, and $\texttt{Net}_{D}$. For $\texttt{Net}_{sisr}$, we construct three stages using 8×8 kernel with stride = 4 and pad by 2 pixels. For $\texttt{Net}_{misr}$, $\texttt{Net}_{res}$, and $\texttt{Net}_{D}$, we construct five blocks where each block consists of two convolutional layers with 3×3 kernel with stride = 1 and pad by 1 pixel. The up-sampling layer in $\texttt{Net}_{misr}$ and down-sampling layer in $\texttt{Net}_{D}$ use 8×8 kernel with stride = 4 and pad by 2 pixels. It also uses $c^l = 256, c^m = 256$, and $c^h = 64$.

RBPN-S uses $\texttt{Net}_{misr}$, $\texttt{Net}_{res}$, and $\texttt{Net}_{D}$ with three blocks, while RBPN-L uses deeper $\texttt{Net}_{sisr}$ with six stages. The other setup remain the same. Table 2 shows that RBPN/6 achieves the best performance. The performance of RBPN/6 is reported in detail in the main paper.

1.3. Residual Learning

We also investigate the use of residual learning on RBPN. First, the target frame is interpolated using Bicubic, then RBPN only produces the residual image. Finally, the interpolated and residual images are combined to produce an SR image. Unfortunately, the current hyper-parameters show that residual learning is not effective to improve RBPN as shown in Table 3.

1.4. Complexity Analysis

We report computational time, no. of parameter, and no. of FLOPS of our proposal (and competition) in Table 4.

1.5. Additional Qualitative Results

Here, we show additional results on several test sets and scaling factors. Figures 1, 2, and 3 show qualitative results for the 4× scaling factor on Vid4 [6], SPMCS-32 [9], and Vimeo-90k [10], respectively. RBPN/6-PF obtains reconstruction that appears most similar to the GT, more pleasing and sharper than reconstructions with other methods. We have highlighted regions in which this is particularly notable.

We also provide the results on a larger scaling factor (i.e., 8×) in Fig. 4. However, no results were provided by other methods on 8×, so we only compare ours with DBPN and Bicubic. It shows that RBPN/6 successfully generates the best results.

References

[1] Jose Caballero, Christian Ledig, Andrew P Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

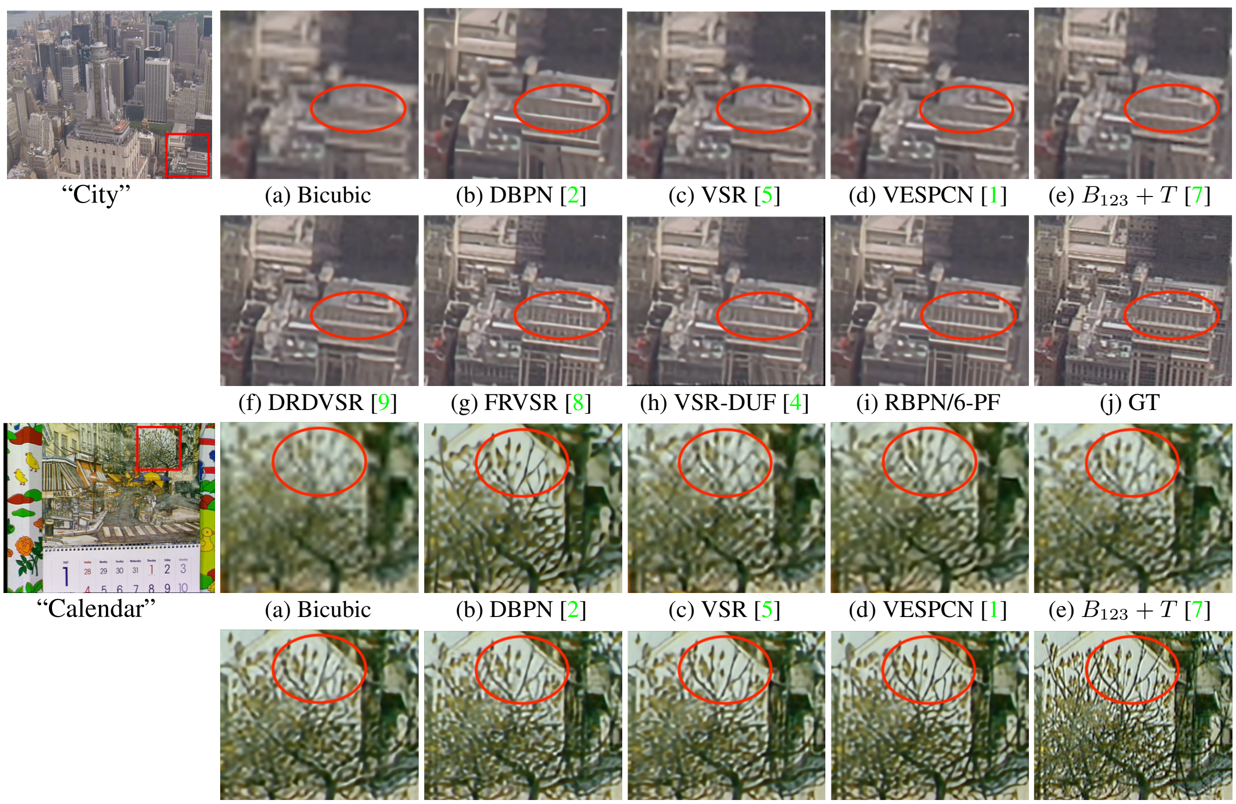
[2] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 2, 4

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. 1

[4] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3224–3232, 2018. 2, 3

Scale	Algorithm	Vimeo-90k [10] Slow	Vimeo-90k [10] Medium	Vimeo-90k [10] Fast	SPMCS-32 [9]	Vid4 [6]
2×	Bicubic	34.47/0.939	36.96/0.956	40.18/0.969	32.39/0.919	28.43/0.866
	DBPN [2]	39.69/0.973	42.33/0.979	45.12/0.984	37.58/0.966	32.30/0.934
	VSR [5]*	-	-	-	-	31.30/0.929
	DRDVSR [9]*	-	-	-	36.62/0.960	-
	VSR-DUF [4]*	-	-	-	-	33.73/0.955
	RBPN/6	40.68/0.978	43.65/0.985	45.63/0.987	39.26/0.977	33.73/0.954
8×	Bicubic	25.81/0.715	27.23/0.766	29.33/0.816	24.23/0.607	21.36/0.465
	DBPN [2]	28.40/0.795	30.28/0.837	32.74/0.872	25.70/0.682	22.39/0.547
	RBPN/6	28.94/0.809	31.35/0.858	33.91/0.890	26.31/0.708	23.04/0.588

Table 1. Additional quantitative evaluation (PSNR/SSIM) of state-of-the-art SR algorithms. (*the values have been copied from the respective publications.)



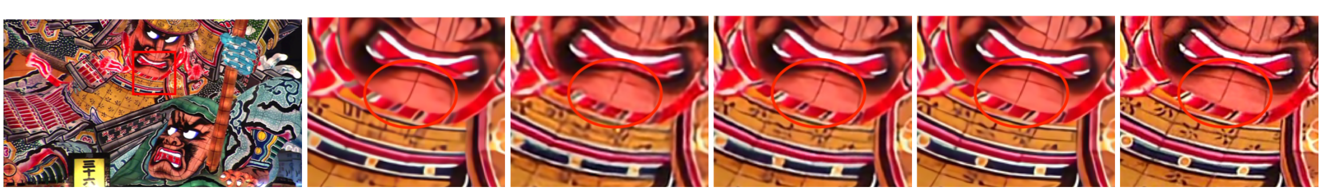
"City" (a) Bicubic (b) DBPN [2] (c) VSR [5] (d) VESPCN [1] (e) $B_{123} + T$ [7]

(f) DRDVSR [9] (g) FRVSR [8] (h) VSR-DUF [4] (i) RBPN/6-PF (j) GT

"Calendar" (a) Bicubic (b) DBPN [2] (c) VSR [5] (d) VESPCN [1] (e) $B_{123} + T$ [7]

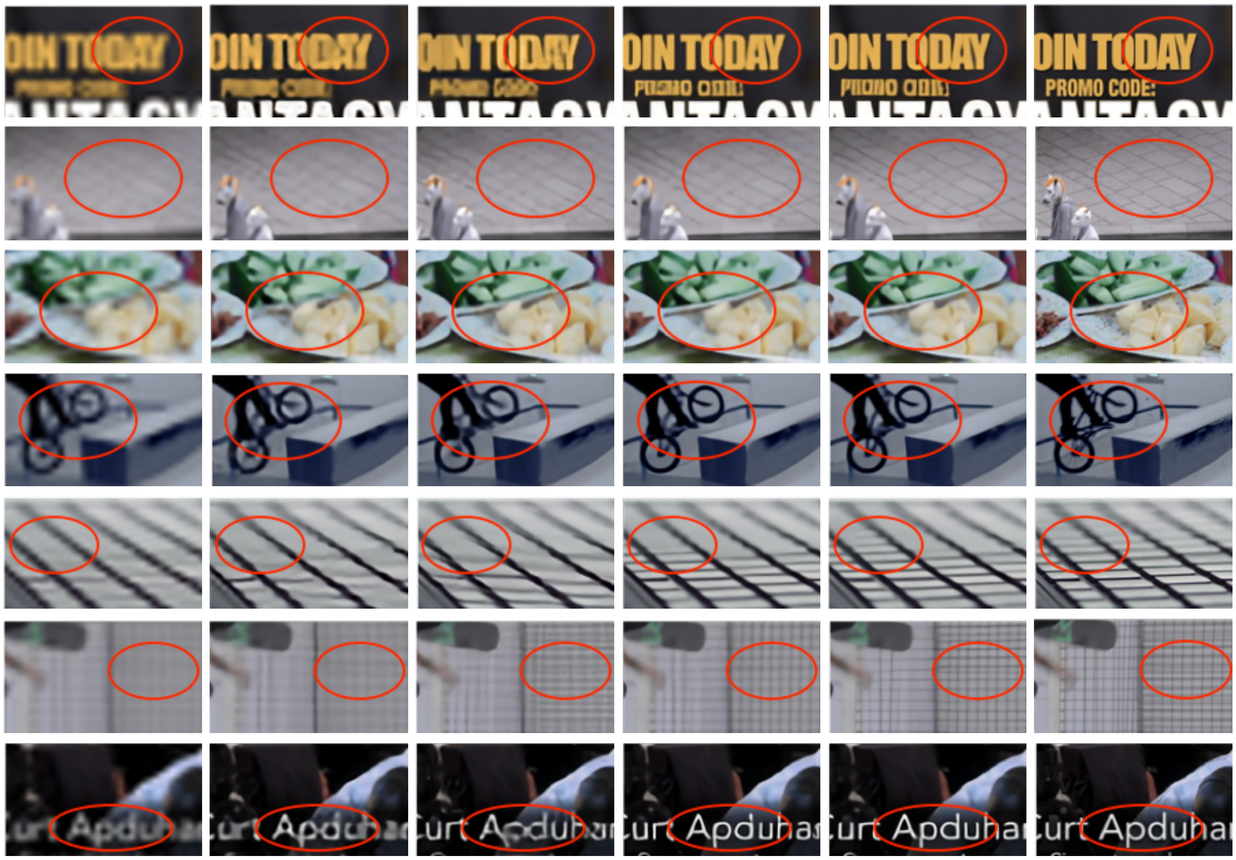
(f) DRDVSR [9] (g) FRVSR [8] (h) VSR-DUF [4] (i) RBPN/6-PF (j) GT

Figure 1. Visual results on Vid4 for $4\times$ scaling factor.



(a) DBPN [2] (b) DRDVSR [9] (c) VSR-DUF [4] (d) RBPN/6-PF (e) GT

Figure 2. Visual results on SPMCS for $4\times$ scaling factor.



(a) Bicubic (b) TOFlow [10] (c) VSR-DUF [4] (d) RBPN/3-P (e) RBPN/6-PF (f) GT

Figure 3. Visual results on Vimeo-90k for $4\times$ scaling factor.

	RBPN/6-S	RBPN/6	RBPN/6-L
PSNR/SSIM	31.39/0.878	31.64/0.883	31.58/0.882
# of Parameter	8,538k	12,771k	17,619k

Table 2. Network size analysis on SPMCS-32 (PSNR/SSIM).

	RBPN/6	
	w/	w/o
PSNR/SSIM	31.57/0.882	31.64/0.883

Table 3. Residual analysis on SPMCS-32 (PSNR/SSIM).

	RBPN/4-PF	RBPN/6-PF	VSR-DUF [4]	DRDVSR [9]
Time (s)	0.058	0.141	0.128	0.108
# of param (M)	12.7	12.7	6.8*	0.7*
# of FLOPS (G)	1650	2475	-	-
PSNR (dB)	29.75	30.10	29.42	28.82

Table 4. Computational complexity on $4\times$ SR with input size 120×160. *Counted manually from model definitions described in the papers.

[5] Armin Kappeler, Seunghwan Yoo, Qiqin Dai, and Aggelos K Katsaggelos. Video super-resolution with convolutional neural networks. *IEEE Transactions on Computational Imaging*, 2(2):109–122, 2016. 2

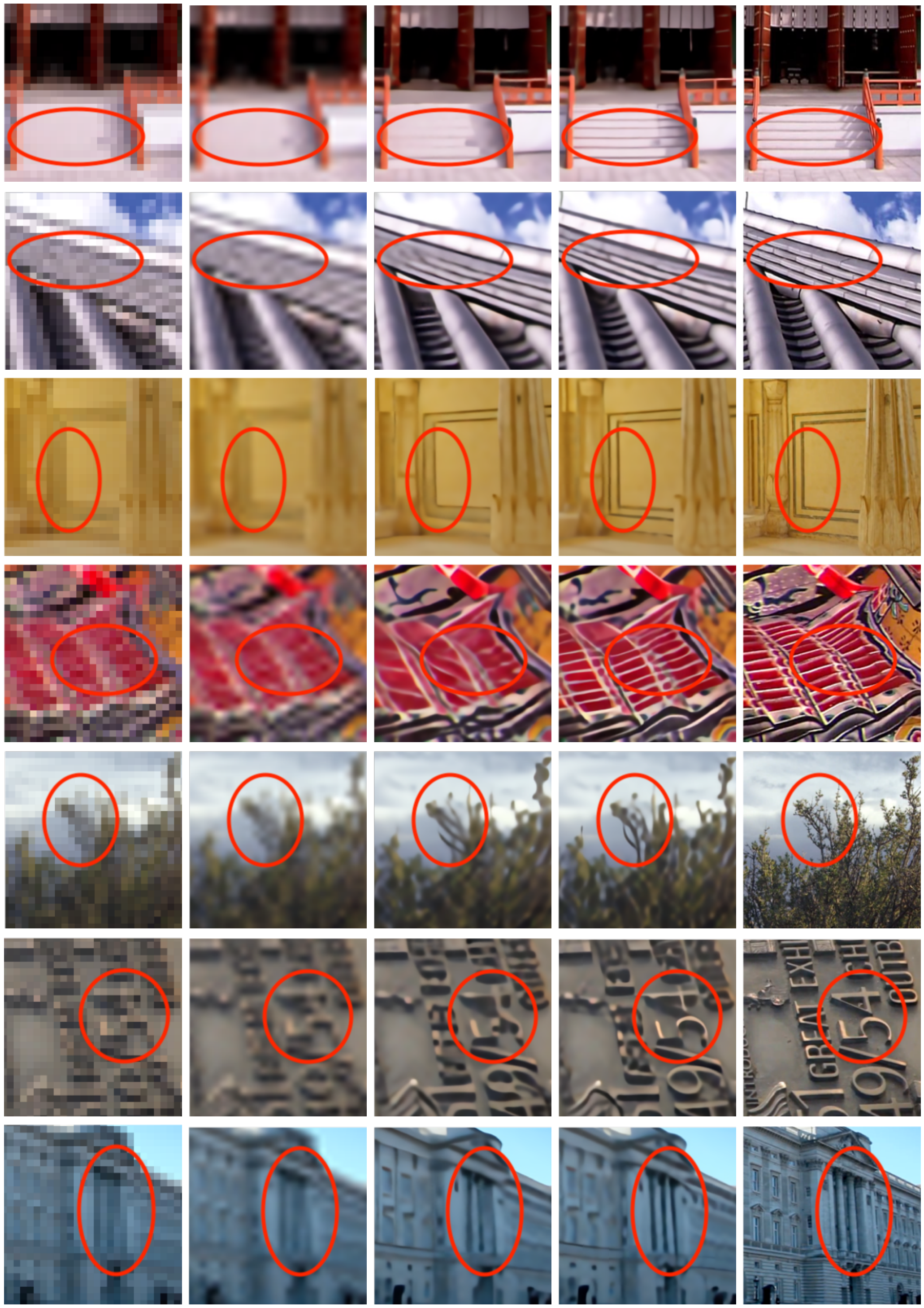
[6] Ce Liu and Deqing Sun. A bayesian approach to adaptive video super resolution. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 209–216. IEEE, 2011. 1, 2

[7] Ding Liu, Zhaowen Wang, Yuchen Fan, Xianming Liu, Zhangyang Wang, Shiyu Chang, and Thomas Huang. Robust video super-resolution with learned temporal dynamics. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2526–2534. IEEE, 2017. 2

[8] Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-recurrent video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6626–6634, 2018. 2

[9] Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. Detail-revealing deep video super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy*, pages 22–29, 2017. 1, 2, 3

[10] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *arXiv preprint arXiv:1711.09078*, 2017. 1, 2, 3



(a) LR (b) Bicubic (c) DBPN [2] (d) RBPN/6 (e) GT

Figure 4. Visual results on SPMCS for $8\times$ scaling factor.