

Supplementary: Practical Single-Image Super-Resolution Using Look-Up Table

Younghyun Jo Seon Joo Kim
Yonsei University

A. More Comparisons

With RAISR RAISR [9] works in two-stage that consists of a global and a local enhancement, with the latter part using a hash table. They compute image gradient and SVD for each patch of the input image for hash table keys, and this would take more computation time than our interpolation step. We note that a direct comparison is difficult with RAISR as the code is not publicly available. We can estimate the runtime of RAISR as roughly 97ms when running on our desktop computer (Intel Xeon CPU E3- 1230 v3 @ 3.30GHz with 32GB RAM), by comparing the reported runtime of A+ in RAISR paper and our paper. In addition, an unofficial implementation of RAISR¹ takes 2.6s on the same machine.

With Other Deep SR Models Some methods which have not been compared with are – PAMS [7], BSRN [14], FALSr [4], ESRN [10] and TPSR [6]. They made valuable efforts to improve the efficiency of deep SR models but they still consist of a number of convolutional layers. We show some quantitative comparisons in Table 1. It is difficult to compare runtimes directly because the codes are not available (except for PAMS), however, we can roughly estimate the runtimes of each method based on the model size and the number of multiplication-addition operations. Based on the runtime of FSRCNN, we can conclude that our method is much faster and efficient than the others. However, the deep SR methods show good PSNR.

B. More Results

We show some examples for natural textures and faces in Fig. 1, which show very comparable methods to ANR [11] and A+ [12]. The images are from Set14 and CelebA [8] testsets. The results show similar visual impression in the texture images, but our method shows partly unsmooth results in the face images due to the limited RF size. This may disappear if trained using the images in CelebA, or considering more RF would be a key factor for better performance in the future. However, our method runs much faster than ANR and A+. In addition, we expect that our method will

Method	Ours-S	FSRCNN	TPSR	PAMS	FALSr	ESRN	BSRN
Runtime	91ms	371ms	-	-	-	-	-
Ops	276M	6G	3.6G	-	74.7G	85G	20.7G
Size	1.274MB	12K [†]	61K [†]	484K [†]	326K [†]	324K [†]	1216K [†]
Set5	29.82	30.71	29.60	31.59	-	31.99	31.35

Table 1. Quantitative comparisons with other deep SR models. We can roughly estimate the runtimes of each method by comparing the number of multiplication-addition operations and the model size. [†]: The number of parameters of DNN.

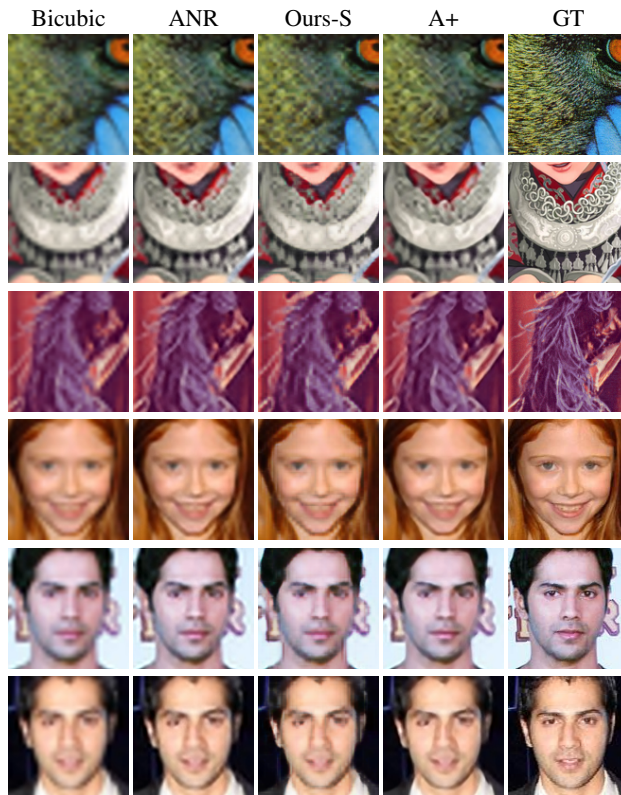


Figure 1. Visual comparisons with the comparable methods on textures and face images.

¹<https://github.com/JalaliLabUCLA/Jalali-Lab-Implementation-of-RAISR>

also perform better than the bicubic interpolation in a real SR data [2].

We also show quantitative results for the upscaling factor $r = 2$ and 3 in Table 2 and Table 3 respectively. Note that we report the results of RCAN [16] instead of RRDB [13] because RRDB has no reference model for $r = 2$ and 3. In addition, visual results for the upscaling factor $r = 2$ and 3 is shown in Fig. 2 and Fig. 3 respectively. Ours-F results show better performance in terms of PSNR and SSIM, but Ours-S results show better visual quality. A noticeable difference is not shown in Fig. 2, however, the difference looks more noticeable as the upscale factor gets larger.

References

- [1] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *ECCV*, pages 252–268, 2018. 3
- [2] Jianrui Cai, Hui Zeng, Hongwei Yong, Zisheng Cao, and Lei Zhang. Toward real-world single image super-resolution: A new benchmark and a new model. In *ICCV*, 2019. 2
- [3] Hong Chang, Dit-Yan Yeung, and Yimin Xiong. Super-resolution through neighbor embedding. In *CVPR*, 2004. 3
- [4] Xiangxiang Chu, Bo Zhang, Hailong Ma, Ruijun Xu, and Qingyuan Li. Fast, accurate and lightweight super-resolution with neural architecture search. *arXiv preprint arXiv:1901.07261*, 2019. 1
- [5] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *ECCV*, pages 391–407. Springer, 2016. 3, 4
- [6] Royson Lee, Łukasz Dudziak, Mohamed Abdelfattah, Stylianos I Venieris, Hyeji Kim, Hongkai Wen, and Nicholas D Lane. Journey towards tiny perceptual super-resolution. In *ECCV*, 2020. 1
- [7] Huixia Li, Chenqian Yan, Shaohui Lin, Xiawu Zheng, Baochang Zhang, Fan Yang, and Rongrong Ji. Pams: Quantized super-resolution via parameterized max scale. In *ECCV*, 2020. 1
- [8] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, December 2015. 1
- [9] Yaniv Romano, John Isidoro, and Peyman Milanfar. Rair: rapid and accurate image super resolution. *IEEE Transactions on Computational Imaging*, 3(1):110–125, 2016. 1
- [10] Dehua Song, Chang Xu, Xu Jia, Yiyi Chen, Chunjing Xu, and Yunhe Wang. Efficient residual dense block search for image super-resolution. In *AAAI*, pages 12007–12014, 2020. 1
- [11] Radu Timofte, Vincent De Smet, and Luc Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *ICCV*, pages 1920–1927, 2013. 1, 3, 4
- [12] Radu Timofte, Vincent De Smet, and Luc Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *ACCV*, pages 111–126. Springer, 2014. 1, 3, 4
- [13] 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 *ECCV Workshops*, September 2018. 2
- [14] Jingwei Xin, Nannan Wang, Xinrui Jiang, Jie Li, Heng Huang, and Xinbo Gao. Binarized neural network for single image super resolution. In *ECCV*, 2020. 1
- [15] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730. Springer, 2010. 3
- [16] 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*, pages 286–301, 2018. 2, 3

	Method	Size	Set5		Set14		BSDS100		Urban100		Manga109	
			PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Interpolation	Nearest	-	30.82	0.8991	28.51	0.8446	28.39	0.8239	25.62	0.8199	28.12	0.9089
	Bilinear	-	32.12	0.9106	29.15	0.8384	28.65	0.8090	25.95	0.8077	29.13	0.9115
	Bicubic	-	33.63	0.9292	30.23	0.8681	29.53	0.8421	26.86	0.8394	30.78	0.9338
SR-LUT	Ours-V	1MB	34.79	0.9420	31.32	0.8907	30.34	0.8692	27.95	0.8689	33.23	0.9535
	Ours-F	77KB	35.64	0.9475	31.88	0.8991	30.77	0.8787	28.49	0.8784	34.43	0.9600
	Ours-S	1.274MB	35.46	0.9466	31.73	0.8958	30.64	0.8750	28.50	0.8777	33.87	0.9579
Sparse coding	NE + LLE [3]	1.434MB	35.79	0.9491	31.82	0.8996	30.77	0.8787	28.48	0.8803	33.95	0.9590
	Zeyde <i>et al.</i> [15]	1.434MB	35.79	0.9494	31.87	0.8989	30.77	0.8771	28.47	0.8794	34.06	0.9599
	ANR [11]	1.434MB	35.85	0.9500	31.86	0.9006	30.82	0.8800	28.49	0.8807	33.94	0.9597
	A+ [12]	15.171MB	36.57	0.9545	32.34	0.9056	31.21	0.8860	29.23	0.8938	35.32	0.9670
DNN	FSRCNN [5]	12K [†]	37.00	0.9555	32.69	0.9086	31.49	0.8902	29.87	0.9007	36.61	0.9704
	CARN-M [1]	412K [†]	37.42	0.9583	33.17	0.9136	31.88	0.8960	31.23	0.9192	37.60	0.9740
	RCAN [16]	15445K [†]	38.17	0.9604	34.03	0.9202	32.37	0.9016	33.30	0.9376	39.32	0.9777

Table 2. Quantitative comparisons on 5 common single-image SR testsets for $r = 2$. Best values are shown in bold among our models. [†]: The number of parameters of DNN.

	Method	Size	Set5		Set14		BSDS100		Urban100		Manga109	
			PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Interpolation	Nearest	-	27.93	0.8123	26.00	0.7330	26.17	0.7065	23.34	0.6992	25.04	0.8157
	Bilinear	-	29.54	0.8504	26.96	0.7526	26.77	0.7177	23.99	0.7135	26.15	0.8372
	Bicubic	-	30.40	0.8678	27.55	0.7736	27.20	0.7379	24.45	0.7343	26.94	0.8554
SR-LUT	Ours-V	1MB	31.29	0.8861	28.33	0.7984	27.68	0.7625	25.13	0.7641	28.72	0.8882
	Ours-F	77KB	31.88	0.8947	28.72	0.8088	27.97	0.7734	25.46	0.7751	29.44	0.8977
	Ours-S	1.274MB	31.95	0.8969	28.73	0.8057	27.92	0.7690	25.53	0.7750	29.32	0.8970
Sparse coding	NE + LLE [3]	1.434MB	31.87	0.8958	28.64	0.8085	27.92	0.7727	25.41	0.7755	28.70	0.8889
	Zeyde <i>et al.</i> [15]	1.434MB	31.93	0.8969	28.70	0.8079	27.95	0.7715	25.45	0.7761	28.85	0.8920
	ANR [11]	1.434MB	31.95	0.8970	28.69	0.8102	27.96	0.7745	25.45	0.7768	28.78	0.8900
	A+ [12]	15.171MB	32.63	0.9090	29.16	0.8190	28.28	0.7832	26.04	0.7974	29.90	0.9099
DNN	FSRCNN [5]	12K [†]	33.19	0.9139	28.61	0.8007	28.50	0.7889	26.42	0.8062	31.07	0.9198
	CARN-M [1]	412K [†]	34.00	0.9235	29.99	0.8357	28.90	0.8001	27.55	0.8384	32.82	0.9385
	RCAN [16]	15445K [†]	34.71	0.9287	30.55	0.8463	29.30	0.8101	29.07	0.8694	34.40	0.9489

Table 3. Quantitative comparisons on 5 common single-image SR testsets for $r = 3$. [†]: The number of parameters of DNN.

