Supplementary Material:
AccelIR: Task-aware Image Compression for Accelerating Neural Restoration

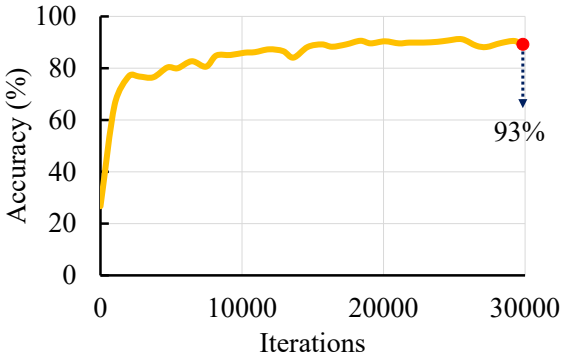

Figure 1. Adjacent accuracy of QP-NET on validation dataset on training. (dataset: DIV2K)

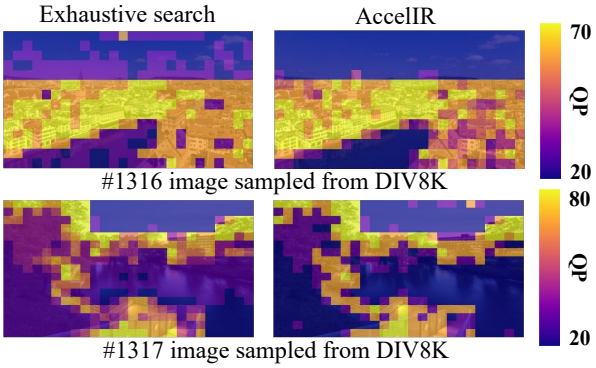

Figure 2. Comparison between the QP allocation method of AccelIR and the optimal version based on exhaustive search.

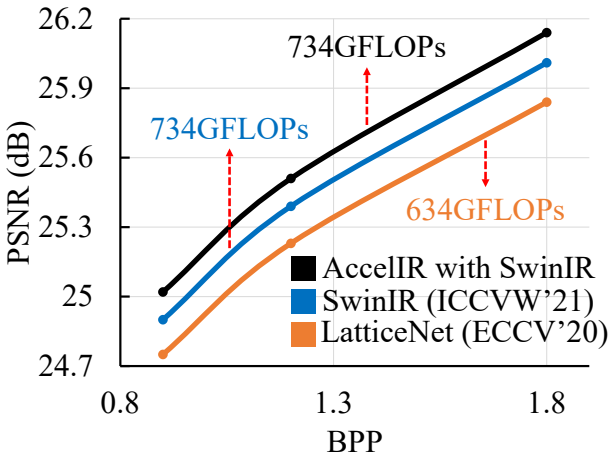

Figure 3. The rate-distortion curve of AccelIR and baselines.

In the supplementary material, we provide the followings: 1) We introduce the detailed experimental setup for the evaluation of the main paper, 2) We present the analysis of individual system components including QP-NET and the QP allocation method, and 3) We provide the supported and additional experiment results.

1. Details of Experimental Setup

Datasets. We use the DIV2K dataset [1] (index 0001-0800) for training and the DIV8K dataset [6] (index 1301-1400). Especially, the test dataset is downsampled to 4K as used in other works [8].

IR networks. We use total nine different IR networks: super-resolution, de-noising, de-blurring. We adjust the IR network capacity by changing the number of channels to ensure that the resulting quality is consistent across all methods. Specifically, the channel configurations used in the evaluation are FSRCNN [5] (56, 36 (-35%)), CARN [2] (64, 48(-42%), 40(59%)), EDSR [10] (64, 40(-60%), 32(-74%)), LatticeNet [11] (64, 48(-21%), 42(-32%)), SwinIR [9] (60, 36(-63%)), DnCNN [13] (64, 56(-24%)), FFDNet [15] (64, 52(-34%)), IRCNN [14] (64, 52(-34%)) and MIMO-UNet [4](32, 28(-24%), 26(-34%).

2. Component-wise Analysis

AccelIR allocates QPs across image blocks in two steps. First, the framework runs QP-NET to predict the best-fitted IR utility group for each block, retrieving the IR quality and size profiles of the selected group. Next, the framework finds near-optimal QPs for each block based on the enhanced A-star algorithm. Therefore, both the performance of QP-NET and QP-allocation method critically affect the amount of IR computation saving.

Accuracy of QP-NET. Figure 1 shows the validation accuracy of QP-NET over the training iterations using the DIV2K dataset [1]. We measure adjacent accuracy that is the top-3 accuracy neared to label. As shown in figure, QP-NET achieves 93% adjacent accuracy in predicting the IR utility group.

Accuracy of QP allocation method. Figure 2 shows the comparison between our QP allocation method and the optimal QP allocation based on exhaustive search. The result shows that the QP allocation of AccelIR is similar to the optimal one. Note that our algorithm runs in the linear time complexity while the optimal method requires the exponential time complexity, which is infeasible for real-world IR applications.

Comparison with direct IR quality prediction The minimal computing overhead is one of the main contributions of AccelIR. But, accurate and direct IR quality prediction requires a heavy computation comparable to IR network. To demonstrate how difficult direct prediction is, we train the neural net used in AccelIR (QP-NET) to directly predict IR quality and size and compare with AccelIR's QP-NET on 1000 random image blocks. Adjacent accuracy of direct prediction is 72.8%, which is 18% lower than AccelIR's QP-NET (90.1%).

3. Extended Evaluation

IR quality gain & Compression gain. AccelIR significantly accelerates IR networks under the same IR quality and image size. Also, AccelIR can deliver a large quality gain when using the same IR network capacity as shown in Table 1. Additionally, AccelIR can improve compression efficiency under the same model capacity, reducing storage or networking cost. As shown in Figure 3, AccelIR shows a better rate-distortion curve and improves BD-PSNR [3] by 7.4% compared to the same IR network without AccelIR. This performance gain is comparable (9.6% BD-PSNR) to that from designing the new IR network with a similar computational cost; LatticeNet [11] to SwinIR [9].

Other datasets. We additionally evaluate AccelIR on various resolutions and datasets: 1) 8K (6720 $\times$ 3072) images sampled from the DIV8K dataset (index 1201-1300) [6], 2) 2K (2040 $\times$ 1416) images sampled from the Flickr2K (index 2551-2650) [12], and 3) images below 2K (1024 $\times$ 1024) sampled from the FFHQ (index 2001-2100) [7]. Table 2 shows the quality and computing overhead of five different IR networks [2, 5, 9–11]. The results show that AccelIR consistently delivers a large gain in IR computation saving on various datasets and resolutions (1K to 8K).

Other scaling factors for SR. We additionally evaluate AccelIR using SR networks with various scaling factor ($2\times$, $3\times$, $4\times$). As shown in Table 3, AccelIR reduces the computing overhead by 74%, 69%, and 69% for the scaling factor of $\times 2$, $\times 3$ and $\times 4$, respectively.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 1122–1131, 2017. 1

[2] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. arXiv preprint arXiv:1803.08664, 2018. 1, 2, 3

[3] G. BJONTEGAARD. Calculation of average psnr differences between rd-curves. ITU SG16 Doc. VCEG-M33, 2001. 2

[4] S. Cho, S. Ji, J. Hong, S. Jung, and S. Ko. Rethinking coarse-to-fine approach in single image deblurring. In 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pages 4621–4630, Los Alamitos, CA, USA, oct 2021. IEEE Computer Society. 1

[5] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, Computer Vision – ECCV 2016, pages 391–407, Cham, 2016. Springer International Publishing. 1, 2, 3

[6] Shuhang Gu, Andreas Lugmayr, Martin Danelljan, Manuel Fritsche, Julien Lamour, and Radu Timofte. Div8k: Diverse 8k resolution image dataset. In 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pages 3512–3516. IEEE, 2019. 1, 2

[7] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 4396–4405, 2019. 2

[8] Xiangtao Kong, Hengyuan Zhao, Yu Qiao, and Chao Dong. Classsr: A general framework to accelerate super-resolution networks by data characteristic. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12016–12025, 2021. 1

[9] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. arXiv preprint arXiv:2108.10257, 2021. 1, 2, 3

[10] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pages 136–144, 2017. 1, 2, 3

[11] Radu Alexandru Rosu, Peer Schütt, Jan Quenzel, and Sven Behnke. Latticenet: Fast point cloud segmentation using permutohedral lattices. In Proc. of Robotics: Science and Systems (RSS), 2020. 1, 2, 3

[12] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, Lei Zhang, Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, Kyoung Mu Lee, Xintao Wang, Yapeng Tian, Ke Yu, Yulun Zhang, Shixiang Wu, Chao Dong, Liang Lin, Yu Qiao, Chen Change Loy, Woong Bae, Jaejun Yoo, Yoseob Han, Jong Chul Ye, Jae-Seok Choi, Munchurl Kim, Yuchen Fan, Jiahui Yu, Wei Han, Ding Liu, Haichao Yu,

Model	Encoder	0.9bpp PNSR / FLOPs	1.2bpp PSNR / FLOPs	1.8bpp PSNR / FLOPs
FSRCNN [5]	JPEG	24.44dB / 191G	24.88dB / 191G	25.41dB / 191G
	AccelIR	24.44dB / 124G **(-35%)**	24.89dB / 124G **(-35%)**	25.41dB / 124G **(-35%)**
		24.49dB **(+0.05dB)** / 191G	24.94dB **(+0.06dB)** / 191G	25.47dB **(+0.06dB)** / 191G
CARN [2]	JPEG	24.72dB / 484G	25.17dB / 484G	25.77dB / 484G
	AccelIR	24.72dB / 279G **(-42%)**	25.17dB / 198G **(-59%)**	25.79dB / 279G **(-42%)**
		24.79dB **(+0.07dB)** / 484G	25.27dB **(+0.10dB)** / 484G	25.87dB **(+0.10dB)** / 484G
EDSR [10]	JPEG	24.84dB / 1165G	25.28dB / 1165G	25.87dB / 1165G
	AccelIR	24.86dB / 462G **(-60%)**	25.30dB / 298G **(-74%)**	25.89dB / 298G **(-74%)**
		24.91dB **(+0.07dB)** / 1165G	25.38dB **(+0.10dB)** / 1165G	25.95dB **(+0.08dB)** / 1165G
LatticeNet [11]	JPEG	24.75dB / 634G	25.23dB / 634G	25.84dB / 634G
	AccelIR	24.75dB / 502G **(-21%)**	25.23dB / 431G **(-32%)**	25.84dB / 431G **(-32%)**
		24.77dB **(+0.02dB)** / 634G	25.26dB **(+0.03dB)** / 634G	25.25.87dB **(+0.03dB)** / 634G
SwinIR [9]	JPEG	24.91dB / 734G	25.40dB / 734G	26.02dB / 734G
	AccelIR	24.96dB / 270G **(-63%)**	25.44dB / 270G **(-63%)**	26.06dB / 270G **(-63%)**
		25.00dB **(+0.09dB)** / 734G	25.49dB **(+0.09dB)** / 734G	26.10dB **(+0.08dB)** / 734G

Table 1. Computation saving & PSNR gain of 5 different SR networks.

Model	Encoder	DIV8K (6720 × 3072) PNSR / FLOPs	Flickr2K (2040 × 1416) PSNR / FLOPs	FFHQ (1024 × 1024) PSNR / FLOPs
FSRCNN [5]	JPEG	26.06dB / 802G	24.05dB / 70G	30.83dB / 30G
	AccelIR	26.06dB / 522G **(-34%)**	24.06dB / 46G **(-34%)**	30.83dB / 19G **(-34%)**
		26.11dB **(+0.05dB)** / 802G	24.12dB **(+0.06dB)** / 70G	30.91dB **(+0.08dB)** / 30G
CARN [2]	JPEG	26.28dB / 2035G	24.37dB / 179G	31.30dB / 76G
	AccelIR	26.28dB / 1175G **(-42%)**	24.37dB / 103G **(-42%)**	31.34dB / 51G **(-34%)**
		26.35dB **(+0.07dB)** / 2035G	24.45dB **(+0.08dB)** / 179G	31.40dB **(+0.1dB)** / 76G
EDSR [10]	JPEG	26.45dB / 4888G	24.55dB / 430G	31.64dB / 183G
	AccelIR	26.47dB / 1270G **(-74%)**	24.55dB / 112G **(-74%)**	31.64dB / 73G **(-60%)**
		26.55dB **(+0.1dB)** / 4888G	24.64dB **(+0.09dB)** / 430G	31.76dB **(+0.12dB)** / 183G
LatticeNet [11]	JPEG	26.38dB / 2668G	24.50dB / 234G	31.52dB / 100G
	AccelIR	26.37dB / 2114G **(-21%)**	24.50dB / 185G **(-21%)**	31.52dB / 79G **(-21%)**
		26.40dB **(+0.02dB)** / 2668G	24.53dB **(+0.03dB)** / 234G	31.58dB **(+0.06dB)** / 100G
SwinIR [9]	JPEG	26.54dB / 3090G	24.69dB / 271G	31.77dB / 115G
	AccelIR	26.53dB / 797G **(-74%)**	24.75dB / 100G **(-63%)**	31.83dB / 42G **(-63%)**
		26.66dB **(+0.12dB)** / 3090G	24.80dB **(+0.11dB)** / 271G	31.89dB **(+0.12dB)** / 115G

Table 2. AccelIR accelerates SR networks on various datasets and resolutions under the same SR quality and image size. Also, AccelIR enhances the SR quality under the same computation and image size.

Model	Scale	Encoder	DIV8K (6720 × 3072) PNSR / FLOPs	Flickr2K (2040 × 1416) PSNR / FLOPs	FFHQ (1024 × 1024) PSNR / FLOPs
EDSR [10]	×2	JPEG	30.58dB / 11013G	28.38dB / 966G	35.07dB / 411G
		AccelIR	30.57dB / 2791G **(-74%)**	28.38dB / 251G **(-74%)**	35.08dB / 104G **(-74%)**
			30.70dB **(+0.12dB)** / 11013G	28.54dB **(+0.16dB)** / 966G	35.23dB **(+0.16dB)** / 411G
	×3	JPEG	27.88dB / 6028G	25.99dB / 567G	32.82dB / 221G
		AccelIR	27.90dB / 1567G **(-74%)**	25.99dB / 147G **(-74%)**	32.86dB / 88G **(-60%)**
			28.00dB **(+0.12dB)** / 6028G	26.11dB **(+0.11dB)** / 567G	32.93dB **(+0.11dB)** / 221G
	×4	JPEG	26.45dB / 4888G	24.55dB / 430G	31.64dB / 183G
		AccelIR	26.47dB / 1270G **(-74%)**	24.55dB / 112G **(-74%)**	31.64dB / 73G **(-60%)**
			26.55dB **(+0.1dB)** / 4888G	24.64dB **(+0.09dB)** / 430G	31.76dB **(+0.12dB)** / 183G

Table 3. AccelIR still delivers a large benefit on various scaling factors (×2, ×3, ×4) for SR (SR Network: EDSR).

Zhangyang Wang, Honghui Shi, Xinchao Wang, Thomas S. Huang, Yunjin Chen, Kai Zhang, Wangmeng Zuo, Zhimin Tang, Linkai Luo, Shaohui Li, Min Fu, Lei Cao, Wen Heng, Giang Bui, Truc Le, Ye Duan, Dacheng Tao, Ruxin Wang, Xu Lin, Jianxin Pang, Jinchang Xu, Yu Zhao, Xiangyu Xu, Jinshan Pan, Deqing Sun, Yujin Zhang, Xibin Song, Yuchao Dai, Xueying Qin, Xuan-Phung Huynh, Tiantong Guo, Hojjat Seyed Mousavi, Tiep Huu Vu, Vishal Monga, Cristovao Cruz, Karen Egiazarian, Vladimir Katkovnik, Rakesh Mehta, Arnav Kumar Jain, Abhinav Agarwalla, Ch V. Sai Praveen, Ruofan Zhou, Hongdiao Wen, Che Zhu, Zhiqiang Xia, Zhengtao Wang, and Qi Guo. Ntire 2017 challenge on single image super-resolution: Methods and results. In 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 1110–1121, 2017. 2

[13] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. IEEE Transactions on Image Processing, 26(7):3142–3155, 2017. 1

[14] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang. Learning deep cnn denoiser prior for image restoration. In IEEE Conference on Computer Vision and Pattern Recognition, pages 3929–3938, 2017. 1

[15] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for CNN based image denoising. IEEE Transactions on Image Processing, 2018. 1