Transcending the Limit of Local Window: Advanced Super-Resolution Transformer with Adaptive Token Dictionary

Supplementary Material

In this supplementary material, we present more implementation details and additional visual results. We first provide training details of our ATD and ATD-light model in Sec. A. Then, additional experimental results are shown in Sec. B to verify the efficacy of ATD. Finally, we present more illustrations of AC-MSA and visual examples by different models in Sec. C.

A. Training Details

ATD. We follow previous works [3, 10] and choose DF2K (DIV2K [15] + Flickr2K [11]) as the training dataset for ATD. Then, we implement the training of ATD in two stages. In the first stage, we randomly crop low-resolution (LR) patches of shape 64×64 and the corresponding high-resolution (HR) image patches for training. The batch size is set to 32, while commonly used data augmentation tricks including random rotation and horizontal flipping are adopted in our training stage. We adopt AdamW [12] optimizer with $\beta_1 = 0.9, \beta_2 = 0.9$ to minimize L_1 pixel loss between HR estimation and ground truth. For the case of $\times 2$ zooming factor, we train the model from scratch for 300k iterations. The learning rate is initially set as 2×10^{-4} and halved at 250k iteration milestone. In the second stage, we increase the patch size to 96×96 for another 250k training iterations to better explore the potential of AC-MSA. We initialize the learning rate as 2×10^{-4} and halve it at [150k, 200k, 225k, 240k] iteration milestones. We omit the first stage for $\times 3$ and $\times 4$ models to save time, only adopting the second stage for finetuning these models based on the well-trained $\times 2$ model. To ensure a smooth training process for the token dictionary, we set warm-up iterations at the beginning of each stage. During this period, the learning rate gradually increases from zero to the initial learning rate.

ATD-light. To make fair comparisons with previous SOTA methods, we only employ DIV2K as training dataset. Same as ATD and previous methods, we train the $\times 2$ model from scratch and finetune the $\times 3$ and $\times 4$ models from the $\times 2$ one. Specifically, we train the $\times 2$ ATD-light model for 500k iterations from scratch and finetune the $\times 3$, $\times 4$ model for 250k iterations based on the well-trained $\times 2$ model. The larger patch size used for ATD is not applied to ATD-light. The initial learning rate and iteration milestone for halving learning rate are set as 5×10^{-4}, [250k, 400k, 450k, 475k, 490k] for $\times 2$ model and 2×10^{-4}, [150k, 200k, 225k, 240k] for $\times 3$, $\times 4$ models. The rest of the training settings are kept the same as ATD.

Table B.1. Model size and computational burden comparisons between ATD and recent state-of-the-art methods.

Model	Params	FLOPs	Urban100		Manga109	
			PSNR	SSIM	PSNR	SSIM
CAT-A	16.6M	360G	27.89	0.8339	32.39	0.9285
HAT	20.8M	412G	27.97	0.9368	32.48	0.9292
ATD	20.3M	417G	28.17	0.8404	32.63	0.9306

B. More Experimental Results

B.1. Analysis on Model Size and Computational Burden.

In this subsection, we analyze the model size of the proposed ATD model. As shown in Tab. B.1, we present the accuracy of image restoration (PSNR), model size (number of parameters) and computational burden (FLOPs) comparison between ATD and recent state-of-the-art models on image SR task. Results in the table clearly demonstrate that the proposed ATD model helps the network achieve a better trade-off between restoration accuracy and model size. Our ATD method achieves better SR results with comparable model size and complexity to HAT. Furthermore, ATD outperforms CAT-A by up to 0.22 dB with only 10% more parameters and FLOPs.

B.2. Image Super-Resolution Results.

In this subsection, we present the complete image super-resolution results in Tab. B.2 and Tab. B.3. In addition to producing remarkable results in $\times 2$ and $\times 4$ SR, ATD and ATD-light also achieve impressive improvements of up to 0.38 dB in the $\times 3$ case.

C. More Visual Examples

C.1. More Visualization of AC-MSA.

In this subsection, we provide illustrations of the Categorize operation and more visual examples of categorization results in Fig. C.1 and Fig. C.2. We visualize only a few categories for each input image for simplicity. In the Categorize operation, pixels are first sorted and classified into $\boldsymbol{\theta}^1, \boldsymbol{\theta}^2, \cdots, \boldsymbol{\theta}^M$ based on the value of attention map. Then, each category is flattened and concatenated sequentially. Although certain pixels not belonging to the same category may be assigned to the same sub-category, it has almost no impact on performance. This is because the number of misassignments will not exceed the dictionary size

Table B.2. Quantitative comparison (PSNR/SSIM) with state-of-the-art methods on **classical SR** task. The best and second best results are colored with red and blue.

Method	Scale	Params	Set5		Set14		BSD100		Urban100		Manga109	
			PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
EDSR [11]	×2	42.6M	38.11	0.9602	33.92	0.9195	32.32	0.9013	32.93	0.9351	39.10	0.9773
RCAN [19]	×2	15.4M	38.27	0.9614	34.12	0.9216	32.41	0.9027	33.34	0.9384	39.44	0.9786
SAN [6]	×2	15.7M	38.31	0.9620	34.07	0.9213	32.42	0.9028	33.10	0.9370	39.32	0.9792
HAN [14]	×2	63.6M	38.27	0.9614	34.16	0.9217	32.41	0.9027	33.35	0.9385	39.46	0.9785
IPT [2]	×2	115M	38.37	-	34.43	-	32.48	-	33.76	-	-	-
SwinIR [10]	×2	11.8M	38.42	0.9623	34.46	0.9250	32.53	0.9041	33.81	0.9433	39.92	0.9797
EDT [9]	×2	11.5M	38.45	0.9624	34.57	0.9258	32.52	0.9041	33.80	0.9425	39.93	0.9800
CAT-A [4]	×2	16.5M	38.51	0.9626	34.78	0.9265	32.59	0.9047	34.26	0.9440	40.10	0.9805
ART [17]	×2	16.4M	38.56	0.9629	34.59	0.9267	32.58	0.9048	34.30	0.9452	40.24	0.9808
HAT [3]	×2	20.6M	38.63	0.9630	34.86	0.9274	32.62	0.9053	34.45	0.9466	40.26	0.9809
ATD (ours)	×2	20.1M	38.61	0.9629	34.92	0.9275	32.64	0.9054	34.73	0.9476	40.35	0.9810
EDSR [11]	×3	43.0M	34.65	0.9280	30.52	0.8462	29.25	0.8093	28.80	0.8653	34.17	0.9476
RCAN [19]	×3	15.6M	34.74	0.9299	30.65	0.8482	29.32	0.8111	29.09	0.8702	34.44	0.9499
SAN [6]	×3	15.9M	34.75	0.9300	30.59	0.8476	29.33	0.8112	28.93	0.8671	34.30	0.9494
HAN [14]	×3	64.2M	34.75	0.9299	30.67	0.8483	29.32	0.8110	29.10	0.8705	34.48	0.9500
IPT [2]	×3	116M	34.81	-	30.85	-	29.38	-	29.49	-	-	-
SwinIR [10]	×3	11.9M	34.97	0.9318	30.93	0.8534	29.46	0.8145	29.75	0.8826	35.12	0.9537
EDT [9]	×3	11.6M	34.97	0.9316	30.89	0.8527	29.44	0.8142	29.72	0.8814	35.13	0.9534
CAT-A [4]	×3	16.6M	35.06	0.9326	31.04	0.8538	29.52	0.8160	30.12	0.8862	35.38	0.9546
ART [17]	×3	16.6M	35.07	0.9325	31.02	0.8541	29.51	0.8159	30.10	0.8871	35.39	0.9548
HAT [3]	×3	20.8M	35.07	0.9329	31.08	0.8555	29.54	0.8167	30.23	0.8896	35.53	0.9552
ATD (ours)	×3	20.3M	35.15	0.9331	31.15	0.8556	29.58	0.8175	30.52	0.8924	35.64	0.9558
EDSR [11]	×4	43.0M	32.46	0.8968	28.80	0.7876	27.71	0.7420	26.64	0.8033	31.02	0.9148
RCAN [19]	×4	15.6M	32.63	0.9002	28.87	0.7889	27.77	0.7436	26.82	0.8087	31.22	0.9173
SAN [6]	×4	15.9M	32.64	0.9003	28.92	0.7888	27.78	0.7436	26.79	0.8068	31.18	0.9169
HAN [14]	×4	64.2M	32.64	0.9002	28.90	0.7890	27.80	0.7442	26.85	0.8094	31.42	0.9177
IPT [2]	×4	116M	32.64	-	29.01	-	27.82	-	27.26	-	-	-
SwinIR [10]	×4	11.9M	32.92	0.9044	29.09	0.7950	27.92	0.7489	27.45	0.8254	32.03	0.9260
EDT [9]	×4	11.6M	32.82	0.9031	29.09	0.7939	27.91	0.7483	27.46	0.8246	32.05	0.9254
CAT-A [4]	×4	16.6M	33.08	0.9052	29.18	0.7960	27.99	0.7510	27.89	0.8339	32.39	0.9285
ART [17]	×4	16.6M	33.04	0.9051	29.16	0.7958	27.97	0.7510	27.77	0.8321	32.31	0.9283
HAT [3]	×4	20.8M	33.04	0.9056	29.23	0.7973	28.00	0.7517	27.97	0.8368	32.48	0.9292
ATD (ours)	×4	20.3M	33.14	0.9061	29.25	0.7976	28.02	0.7524	28.22	0.8414	32.65	0.9308

Table B.3. Quantitative comparison (PSNR/SSIM) with state-of-the-art methods on **lightweight SR** task. The best and second best results are colored with red and blue.

Method	Scale	Params	Set5		Set14		BSD100		Urban100		Manga109	
			PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
CARN [1]	×2	1,592K	37.76	0.9590	33.52	0.9166	32.09	0.8978	31.92	0.9256	38.36	0.9765
IMDN [7]	×2	694K	38.00	0.9605	33.63	0.9177	32.19	0.8996	32.17	0.9283	38.88	0.9774
LAPAR-A [8]	×2	548K	38.01	0.9605	33.62	0.9183	32.19	0.8999	32.10	0.9283	38.67	0.9772
LatticeNet [13]	×2	756K	38.15	0.9610	33.78	0.9193	32.25	0.9005	32.43	0.9302	-	-
SwinIR-light [10]	×2	910K	38.14	0.9611	33.86	0.9206	32.31	0.9012	32.76	0.9340	39.12	0.9783
ELAN [18]	×2	582K	38.17	0.9611	33.94	0.9207	32.30	0.9012	32.76	0.9340	39.11	0.9782
SwinIR-NG [5]	×2	1181K	38.17	0.9612	33.94	0.9205	32.31	0.9013	32.78	0.9340	39.20	0.9781
OmniSR [16]	×2	772K	38.22	0.9613	33.98	0.9210	32.36	0.9020	33.05	0.9363	39.28	0.9784
ATD-light (Ours)	×2	753K	38.29	0.9616	34.10	0.9217	32.39	0.9023	33.27	0.9375	39.52	0.9789
CARN [1]	×3	1,592K	34.29	0.9255	30.29	0.8407	29.06	0.8034	28.06	0.8493	33.50	0.9440
IMDN [7]	×3	703K	34.36	0.9270	30.32	0.8417	29.09	0.8046	28.17	0.8519	33.61	0.9445
LAPAR-A [8]	×3	544K	34.36	0.9267	30.34	0.8421	29.11	0.8054	28.15	0.8523	33.51	0.9441
LatticeNet [13]	×3	765K	34.53	0.9281	30.39	0.8424	29.15	0.8059	28.33	0.8538	-	-
SwinIR-light [10]	×3	918K	34.62	0.9289	30.54	0.8463	29.20	0.8082	28.66	0.8624	33.98	0.9478
ELAN [18]	×3	590K	34.61	0.9288	30.55	0.8463	29.21	0.8081	28.69	0.8624	34.00	0.9478
SwinIR-NG [5]	×3	1190K	34.64	0.9293	30.58	0.8471	29.24	0.8090	28.75	0.8639	34.22	0.9488
OmniSR [16]	×3	780K	34.70	0.9294	30.57	0.8469	29.28	0.8094	28.84	0.8656	34.22	0.9487
ATD-light (ours)	×3	760K	34.74	0.9300	30.68	0.8485	29.32	0.8109	29.17	0.8709	34.60	0.9506
CARN [1]	×4	1,592K	32.13	0.8937	28.60	0.7806	27.58	0.7349	26.07	0.7837	30.47	0.9084
IMDN [7]	×4	715K	32.21	0.8948	28.58	0.7811	27.56	0.7353	26.04	0.7838	30.45	0.9075
LAPAR-A [8]	×4	659K	32.15	0.8944	28.61	0.7818	27.61	0.7366	26.14	0.7871	30.42	0.9074
LatticeNet [13]	×4	777K	32.30	0.8962	28.68	0.7830	27.62	0.7367	26.25	0.7873	-	-
SwinIR-light [10]	×4	930K	32.44	0.8976	28.77	0.7858	27.69	0.7406	26.47	0.7980	30.92	0.9151
ELAN [18]	×4	582K	32.43	0.8975	28.78	0.7858	27.69	0.7406	26.54	0.7982	30.92	0.9150
SwinIR-NG [5]	×4	1201K	32.44	0.8980	28.83	0.7870	27.73	0.7418	26.61	0.8010	31.09	0.9161
OmniSR [16]	×4	792K	32.49	0.8988	28.78	0.7859	27.71	0.7415	26.65	0.8018	31.02	0.9151
ATD-light (Ours)	×4	769K	32.63	0.8998	28.89	0.7886	27.79	0.7440	26.97	0.8107	31.48	0.9198

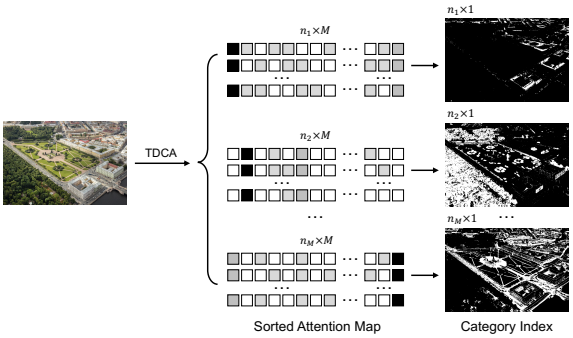

Figure C.1. An illustration of the Categorize operation. With the attention map obtained by TDCA operation, we assign a category index to each pixel based on the highest similarity between the pixel and the token dictionary.

$M = 128$, which is much less than the number of sub-categories HW/n_s.

The following visual examples further demonstrate that the categorize operation is capable of grouping similar textures together. We can see that the categorize operation performs well on various types of image, including either natural or cartoon images.

C.2. More Visual Comparisons.

In this subsection, we provide more visual comparisons between our ATD models and state-of-the-art methods. As shown in Fig. C.3 and Fig. C.4, ATD and ATD-light both yield better visual results. Specifically, ATD recovers sharper edges in img011 and img027, while the output of other methods remains blurry. Moreover, most existing methods fail to reconstruct correct shape of the black blocks in Donburakokko. In contrast, the output of ATD-light is more accurate to the rectangular shape in the ground truth.

References

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

[2] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer, 2020. 2

[3] Xiangyu Chen, Xintao Wang, Jiantao Zhou, Yu Qiao, and Chao Dong. Activating more pixels in image super-resolution transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22367–22377, 2023. 1, 2

[4] Zheng Chen, Yulun Zhang, Jinjin Gu, Yongbing Zhang, Linghe Kong, and Xin Yuan. Cross aggregation transformer for image restoration. In *NeurIPS*, 2022. 2

[5] Haram Choi, Jeongmin Lee, and Jihoon Yang. N-gram in swin transformers for efficient lightweight image super-resolution, 2022. 2

[6] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[7] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the 27th ACM International Conference on Multimedia*, 2019. 2

[8] Wenbo Li, Kun Zhou, Lu Qi, Nianjuan Jiang, Jiangbo Lu, and Jiaya Jia. Lapar: Linearly-assembled pixel-adaptive regression network for single image super-resolution and beyond, 2020. 2

[9] Wenbo Li, Xin Lu, Jiangbo Lu, Xiangyu Zhang, and Jiaya Jia. On efficient transformer and image pre-training for low-level vision. *arXiv preprint arXiv:2112.10175*, 2021. 2

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

[11] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017. 1, 2

[12] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018. 1

[13] Xiaotong Luo, Yuan Xie, Yulun Zhang, Yanyun Qu, Cuihua Li, and Yun Fu. *LatticeNet: Towards Lightweight Image Super-Resolution with Lattice Block*, page 272–289. 2020. 2

[14] Ben Niu, Weilei Wen, Wenqi Ren, Xiangde Zhang, Lianping Yang, Shuzhen Wang, Kaihao Zhang, Xiaochun Cao, and Haifeng Shen. *Single Image Super-Resolution via a Holistic Attention Network*, page 191–207. 2020. 2

[15] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, Lei Zhang, Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, Kyoung Mu Lee, et al. Ntire 2017 challenge on single image super-resolution: Methods and results. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017. 1

[16] Hang Wang, Xuanhong Chen, Bingbing Ni, Yutian Liu, and Liu jinfan. Omni aggregation networks for lightweight image super-resolution. In *Conference on Computer Vision and Pattern Recognition*, 2023. 2

[17] Jiale Zhang, Yulun Zhang, Jinjin Gu, Yongbing Zhang, Linghe Kong, and Xin Yuan. Accurate image restoration with attention retractable transformer. In *ICLR*, 2023. 2

[18] Xindong Zhang, Hui Zeng, Shi Guo, and Lei Zhang. Efficient long-range attention network for image super-resolution. In *European Conference on Computer Vision*, 2022. 2

[19] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. *Image Super-Resolution Using Very Deep Residual Channel Attention Networks*, page 294–310. 2018. 2

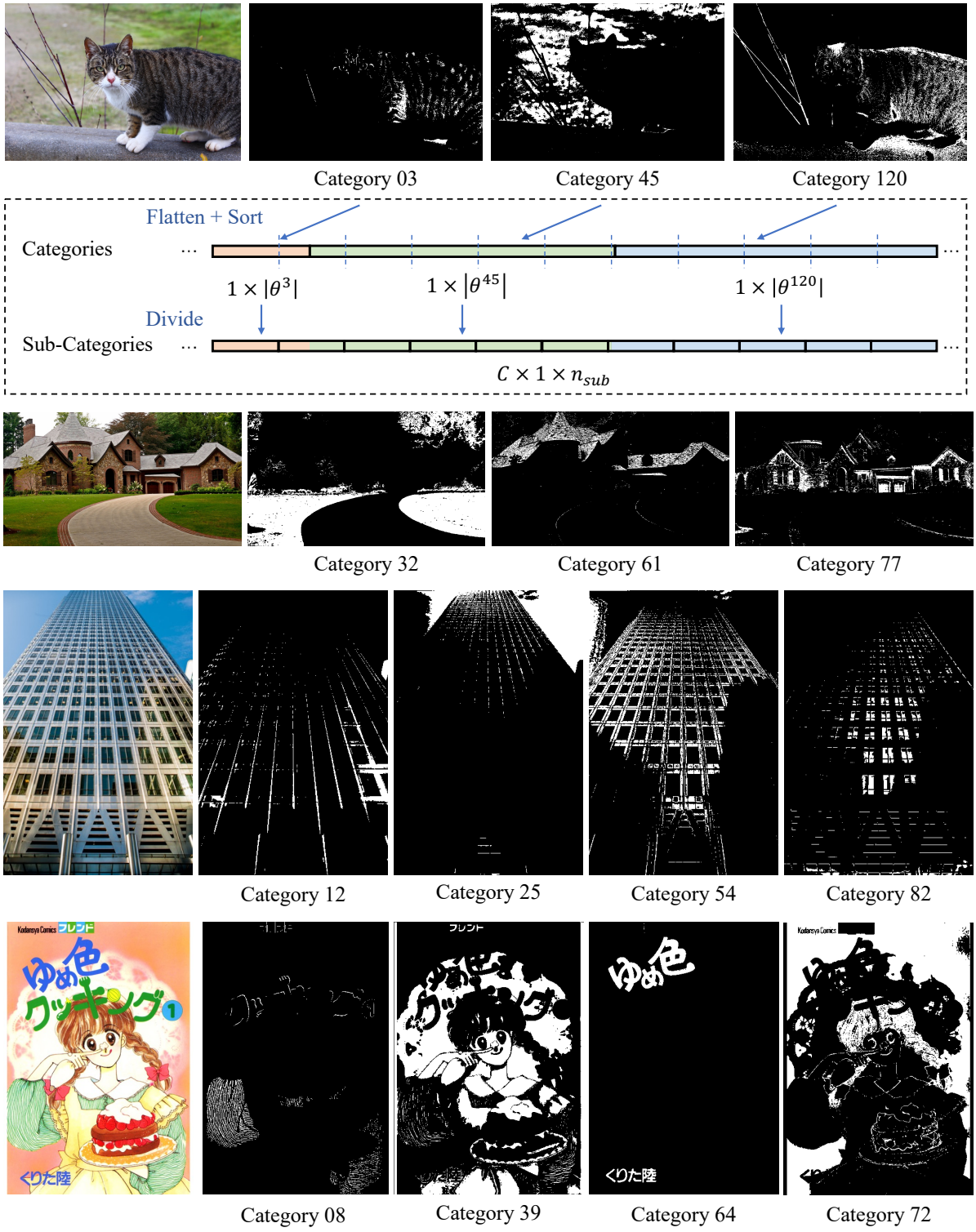

Figure C.2. An illustration of the Categorize operation, along with several visual examples of the categorization results. The white area in each binarized image represents a single category. Pixels in each category are flattened and then sorted for further dividing into sub-categories. These categorization results indicate that our AC-MSA is capable of dividing the image by the class of each pixel. Therefore, areas with similar texture (for example, sky, grass, roof) are grouped into the same category.

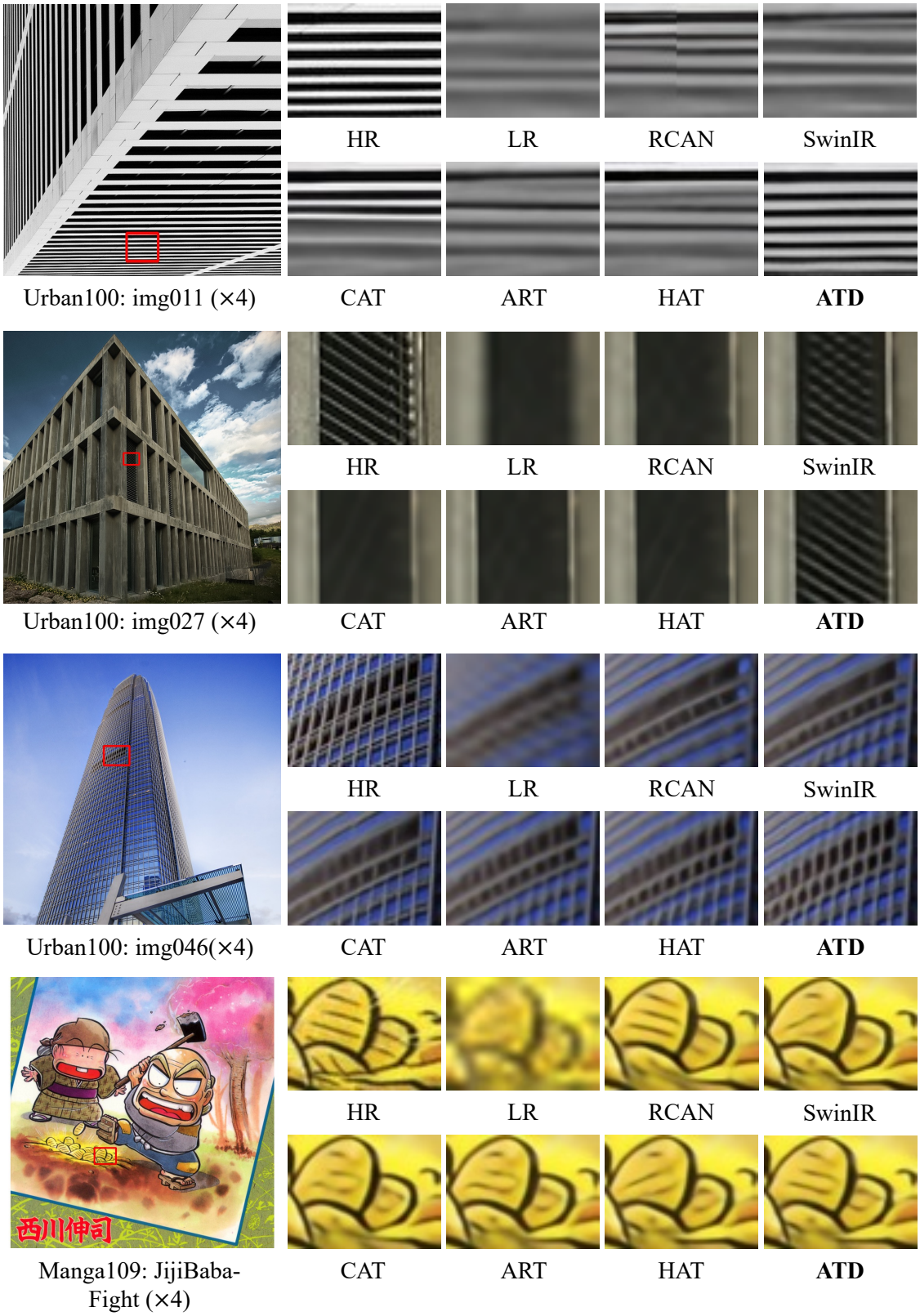

Figure C.3. Visual comparisons between ATD and state-of-the-art classical SR methods.

Figure C.4. Visual comparisons between ATD-light and state-of-the-art lightweight SR methods.