Learning RAW-to-sRGB Mappings with Inaccurately Aligned Supervision (Supplementary Material)

Zhilu Zhang[1], Haolin Wang[1], Ming Liu[1], Ruohao Wang[1], Jiawei Zhang[2], Wangmeng Zuo[1,3] (✉)
[1]Harbin Institute of Technology, [2]SenseTime Research, [3]Pazhou Lab, Guangzhou
{cszlzhang, Why_cs, csmliu, rhwangHIT}@outlook.com {zhjw1988}@gmail.com wmzuo@hit.edu.cn

A. Content

The content of this supplementary material involves:

- Network structure of GCM, LiteISPNet and the discriminator of LiteISPGAN in Sec. B.
- Visual results of alignment in Sec. C.
- Qualitative results of ablation study in Sec. D
- Implementation details on SR-RAW dataset in Sec. E.
- Quantitative results for re-splitting the train/test set on ZRR dataset in Sec. F.
- Additional visual comparison results on SR-RAW and ZRR dataset in Sec. G.

B. Network Structure

Global color mapping (GCM) module involves two components: spatially preserving network (SPN) and GuideNet. SPN stacks 1×1 convolutional layers to guarantee spatial independence of the mapping and GuideNet generates a conditional guidance vector from the target sRGB to modulate SPN features. The detailed structure of GCM are shown in Table A.

The structure configuration of LiteISPNet are shown in Table B. LiteISPNet is a U-Net [7] based multi-level wavelet ISP network. In each residual group, we only apply 4 residual channel attention blocks (RCABs).

The discriminator structure of LiteISPGAN are shown in Table C. We apply 54 × 54 PatchGAN [11], which distinguishes whether the image patch is real or fake.

C. Visual Results of Alignment

We show the demosaicked raw image ($\hat{\mathbf{x}}$), GCM output ($\tilde{\mathbf{y}}$), LiteISPNet output ($\hat{\mathbf{y}}$), warped target sRGB image ($\mathbf{y}^w$) and the original target sRGB image ($\mathbf{y}$) in Fig. A.

It can be seen that the color of $\tilde{\mathbf{y}}$ is consistent with $\mathbf{y}$. Although GCM model cannot perform local operations (*e.g.*, denoising), benefiting from PWC-Net [8], we can still align $\mathbf{y}$ with $\tilde{\mathbf{y}}$ robustly. Under the supervision of well aligned training data, LiteISPNet output has almost no pixel shift. In short, our method achieves the joint of image alignment and RAW-to-sRGB mapping.

D. Qualitative Results on Ablation Study

We show more qualitative results of different alignment strategies in Fig. B.

Due to the limitation of space in the submitted manuscript, we only showed the outputs of GCM model in Sec. 5.2. Here, by visualizing the illuminance ratio between the output of GCM and ground-truth (GT), we show the influence of each component and the dark corner phenomenon more clearly in Fig. C.

E. Implementation Details on SR-RAW Dataset

In each image pair of the SR-RAW dataset, the short focal-length raw image is used as input, while the long focal-length sRGB image is adopted as the ground-truth. In order to align high-resolution (HR) sRGB images with low-resolution (LR) raw images, we adopt downsampled HR sRGB image $\mathbf{y}_{\downarrow}$ to generate the conditional guidance vector in the GCM model. Then the optical flow between the GCM output $\tilde{\mathbf{y}}$ and $\mathbf{y}_{\downarrow}$ is estimated. Note that the size of optical flow is a quarter of the HR sRGB image. Thus, we upsample the optical flow to get the warped HR sRGB image. Finally, the warped HR sRGB image is utilized to supervise the learning of the backbone (SRResNet [5]).

Following [10], we use 400 scenes of images for training, 50 for validation, and the rest 50 for testing, and report the performance on 35/150 mm pairs in Table 2 of the main text. For a comprehensive comparison, we further show the performance on all 24/100, 35/150 and 50/240 test pairs in Table D. It can be seen that our method can still achieve better quantitative performance against all competing methods.

Table A: Structure configuration of GCM model. GCM involves two components: SPN (left column) and GuideNet (right column). Except for the stride of the first convolutional layer in GuideNet is 2, the stride of other convolutional layers is 1.

Spatially Preserving Network (SPN)				GuideNet			
Layer	Output size	Kernel size	Filter	Layer	Output size	Kernel size	Filter
Conv, ReLU	448×448	1×1	$5 \to 64$	Conv	222×222	7×7	$8 \to 32$
[Conv, ReLU] $\times 3$	448×448	1×1	$64 \to 64$	Conv, ReLU, Conv	222×222	3×3	$32 \to 32$
Conv	448×448	1×1	$64 \to 3$	Global Average Pooling	1×1	-	-
				Conv	1×1	1×1	$32 \to 64$

Table B: Structure configuration of LiteISPNet. DWT and IWT denote discrete wavelet transform and inverse wavelet transform, respectively. RG denotes the residual group containing 4 residual channel attention blocks (RCABs).

LiteISPNet		
Layer	Output size	Filter
Conv	224×224	$4 \to 64$
RG	224×224	$64 \to 64$
DWT	112×112	$64 \to 256$
Conv	112×112	$256 \to 64$
RG	112×112	$64 \to 64$
DWT	56×56	$64 \to 256$
Conv	56×56	$256 \to 128$
RG	56×56	$128 \to 128$
DWT	28×28	$128 \to 512$
Conv	28×28	$512 \to 128$
RG	28×28	$128 \to 128$
RG	28×28	$128 \to 128$
Conv	28×28	$128 \to 512$
IWT	56×56	$512 \to 128$
RG	56×56	$128 \to 128$
Conv	56×56	$128 \to 256$
IWT	112×112	$256 \to 64$
RG	112×112	$64 \to 64$
Conv	112×112	$64 \to 256$
IWT	224×224	$256 \to 64$
RG	224×224	$64 \to 64$
Conv	224×224	$64 \to 64$
Conv	224×224	$64 \to 256$
PixelShuffle	448×448	$256 \to 64$
Conv	448×448	$64 \to 3$

F. Quantitative results for re-splitting the train/test set on ZRR dataset

For the ZRR dataset, we follow the official division to train our LiteISPNet with 46,839 pairs, and report the quantitative results on the remaining 1,204 pairs in the main text. Here we conducted an experiment by re-splitting the dataset at approximately $9 : 1$, *i.e.*, 43,200 pairs for training and the rest 4,843 pairs for testing. Table E shows the quantitative results, and it can be seen that our LiteISPNet also exceeds the competing methods.

Table C: Structure configuration of the discriminator. The kernel size of all convolutional layers is 4×4. The stride of the first three convolutional layers is 2, while the stride of the last two convolutional layers is 1.

Discriminator		
Layer	Output size	Filter
Conv, LeakyReLU	224×224	$3 \to 64$
Conv, BatchNorm, LeakyReLU	112×224	$64 \to 128$
Conv, BatchNorm, LeakyReLU	56×56	$128 \to 256$
Conv, BatchNorm, LeakyReLU	55×55	$256 \to 512$
Conv, BatchNorm, LeakyReLU	54×54	$512 \to 1$

Table D: Average results on all 24/100, 35/150 and 50/240 test pairs of SR-RAW dataset. Methods taking LR sRGB image as input are marked with †. The metrics are computed by *Align GT with result*.

Method	PSNR↑ / SSIM↑ / LPIPS↓
SRGAN† [5]	21.72 / 0.6917 / 0.394
ESRGAN† [9]	21.85 / 0.6904 / 0.393
SPSR† [6]	21.75 / 0.6692 / 0.427
RealSR† [4]	21.89 / 0.6918 / 0.388
Zhang *et al.* [10]	21.97 / 0.7360 / 0.357
Ours	22.50 / **0.7369** / 0.329
Ours (GAN)	**22.56** / 0.7341 / **0.323**

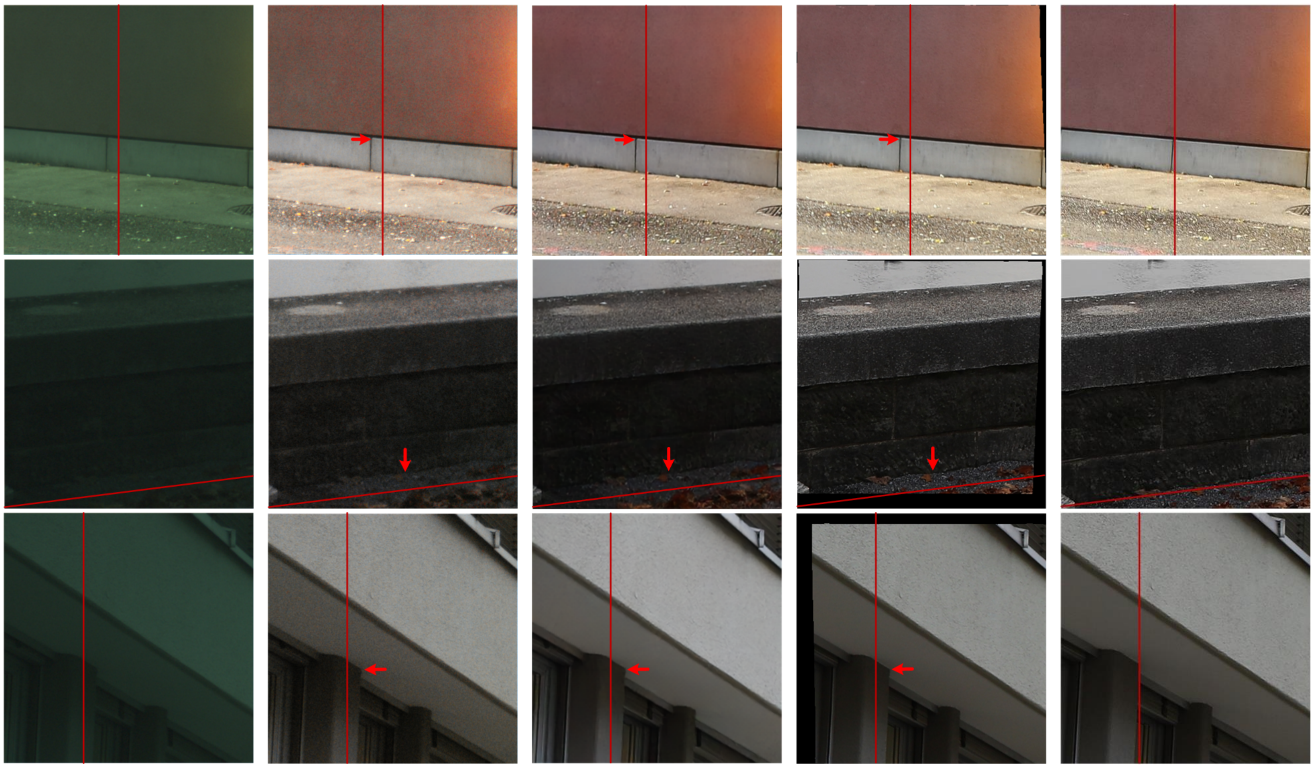
Table E: Quantitative results for re-splitting the train/test set of ZRR dataset. The metrics are computed by *Align GT with result*.

Method	PSNR↑ / SSIM↑ / LPIPS↓
PyNet [3]	22.67 / 0.8535 / 0.149
AWNet (raw) [1]	22.83 / 0.8513 / 0.160
AWNet (demosaicked) [1]	22.68 / 0.8447 / 0.173
MWISPNet [2]	23.00 / 0.8530 / 0.166
Ours (LiteISPNet)	**23.31 / 0.8747 / 0.131**

G. Additional visual comparison results on SR-RAW and ZRR dataset

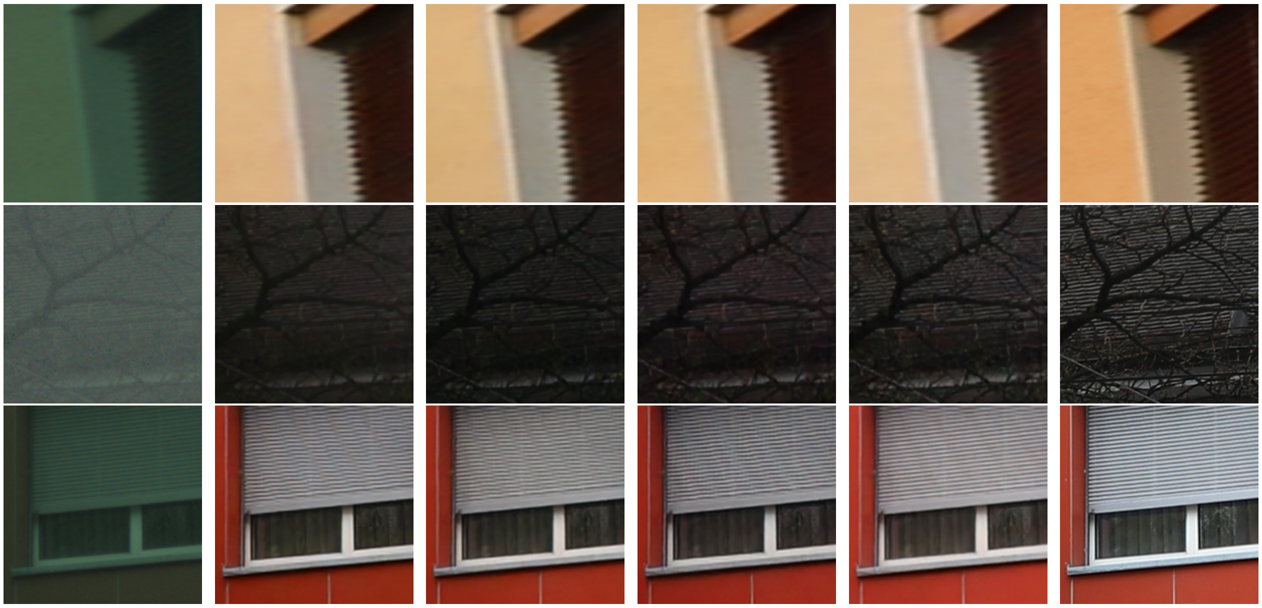
In Fig. D∼ F, we show more qualitative comparison results generated by SRGAN [5], ESRGAN [9], SPSR [6], RealSR [4], Zhang *et al.* [10] and our models on the SR-RAW dataset.

In Fig. G∼ I, we show more qualitative results generated by PyNet [3], AWNet [1], MWISPNet [2] and our models on the ZRR dataset.



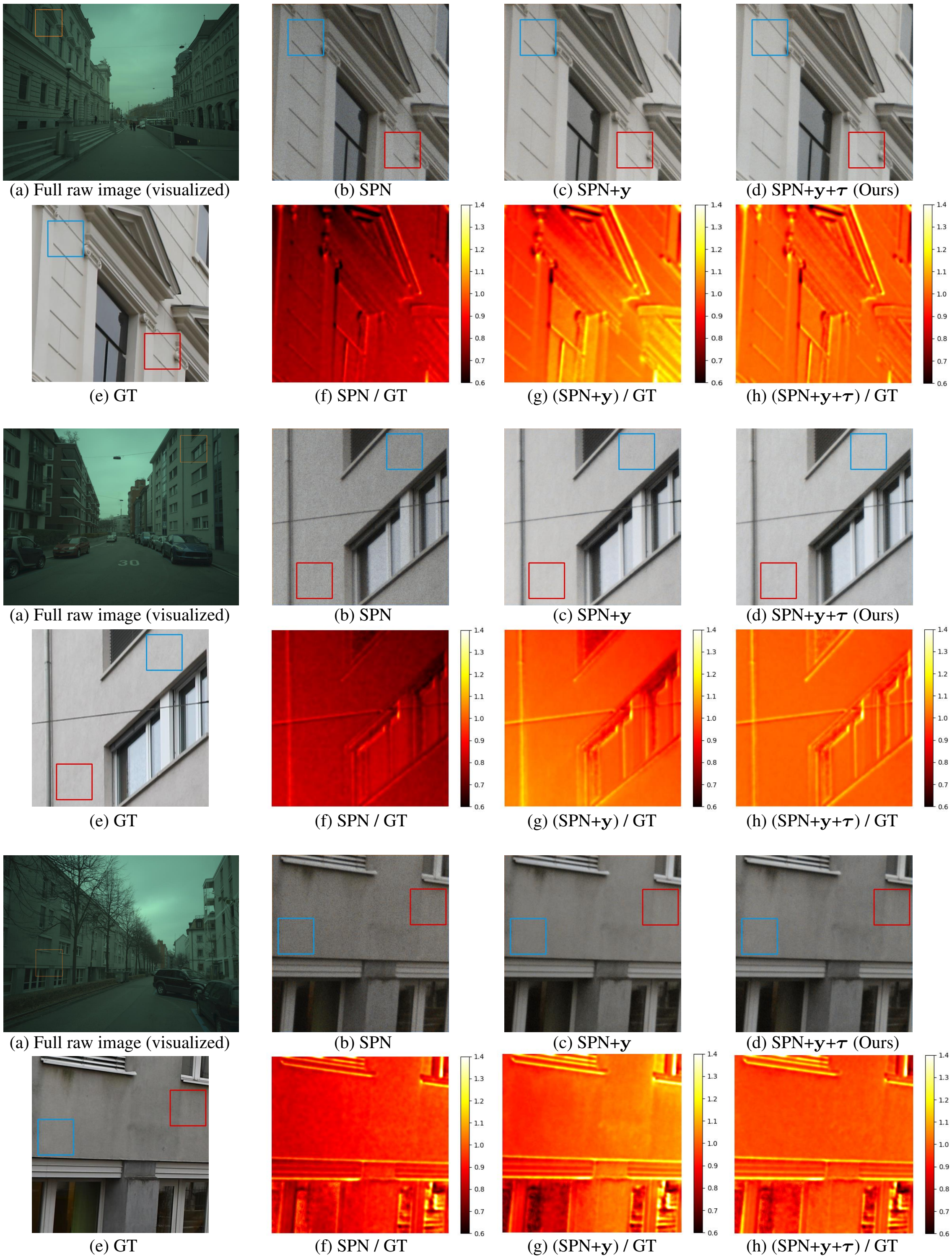
(a) Demosaicked raw ($\hat{\mathbf{x}}$) (b) GCM output ($\tilde{\mathbf{y}}$) (c) LiteISPNet output ($\hat{\mathbf{y}}$) (d) Warped target sRGB ($\mathbf{y}^w$) (e) GT ($\mathbf{y}$)

Figure A: Alignment visual results obtained by our joint learning framework. With the reference line, it can be observed our method obtains the well aligned data pairs while the demosaicked raw is not aligned with GT.



(a) Raw image (b) SIFT (baseline) (c) Align $\mathbf{y}$ to $\hat{\mathbf{y}}$ (d) Align $\mathbf{y}$ to $\hat{\mathbf{x}}$ (e) Align $\mathbf{y}$ to $\tilde{\mathbf{y}}$ (Ours) (f) GT

Figure B: Visual results of LiteISPNet output $\hat{\mathbf{y}}$. (b)∼(e) denote different alignment strategies. Our method (e) performs favorably against other alignment strategies.



(a) Full raw image (visualized) (b) SPN (c) SPN+$\mathbf{y}$ (d) SPN+$\mathbf{y}$+$\boldsymbol{\tau}$ (Ours)

(e) GT (f) SPN / GT (g) (SPN+$\mathbf{y}$) / GT (h) (SPN+$\mathbf{y}$+$\boldsymbol{\tau}$) / GT

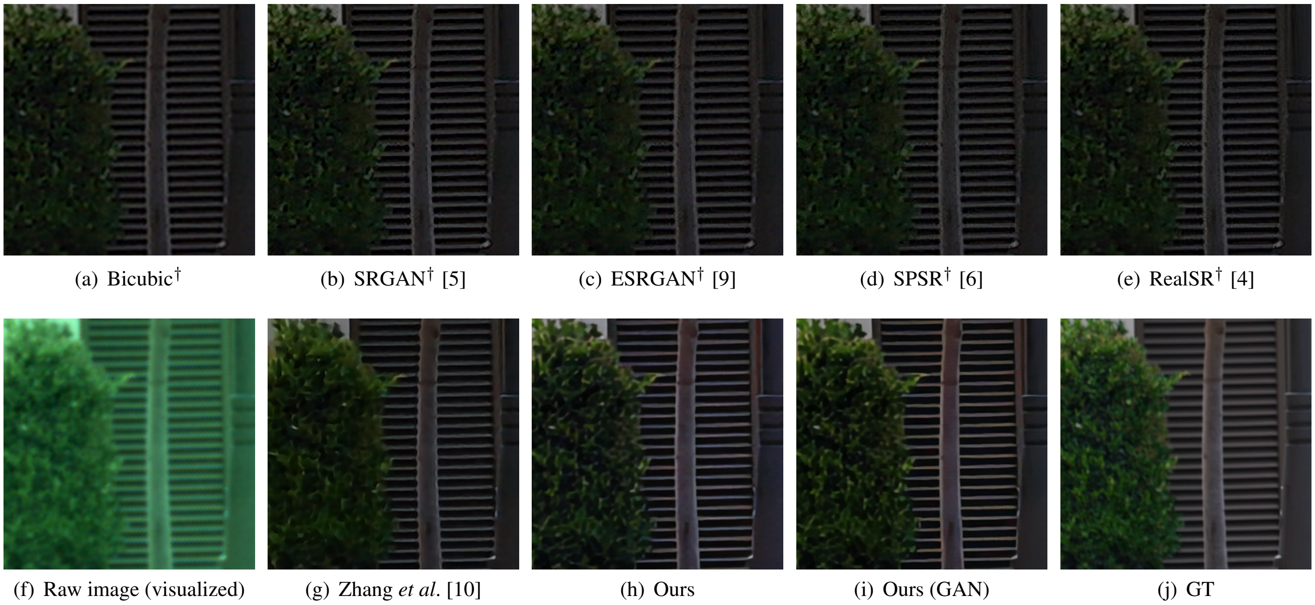
(a) Full raw image (visualized) (b) SPN (c) SPN+$\mathbf{y}$ (d) SPN+$\mathbf{y}$+$\boldsymbol{\tau}$ (Ours)

(e) GT (f) SPN / GT (g) (SPN+$\mathbf{y}$) / GT (h) (SPN+$\mathbf{y}$+$\boldsymbol{\tau}$) / GT

(a) Full raw image (visualized) (b) SPN (c) SPN+$\mathbf{y}$ (d) SPN+$\mathbf{y}$+$\boldsymbol{\tau}$ (Ours)

(e) GT (f) SPN / GT (g) (SPN+$\mathbf{y}$) / GT (h) (SPN+$\mathbf{y}$+$\boldsymbol{\tau}$) / GT

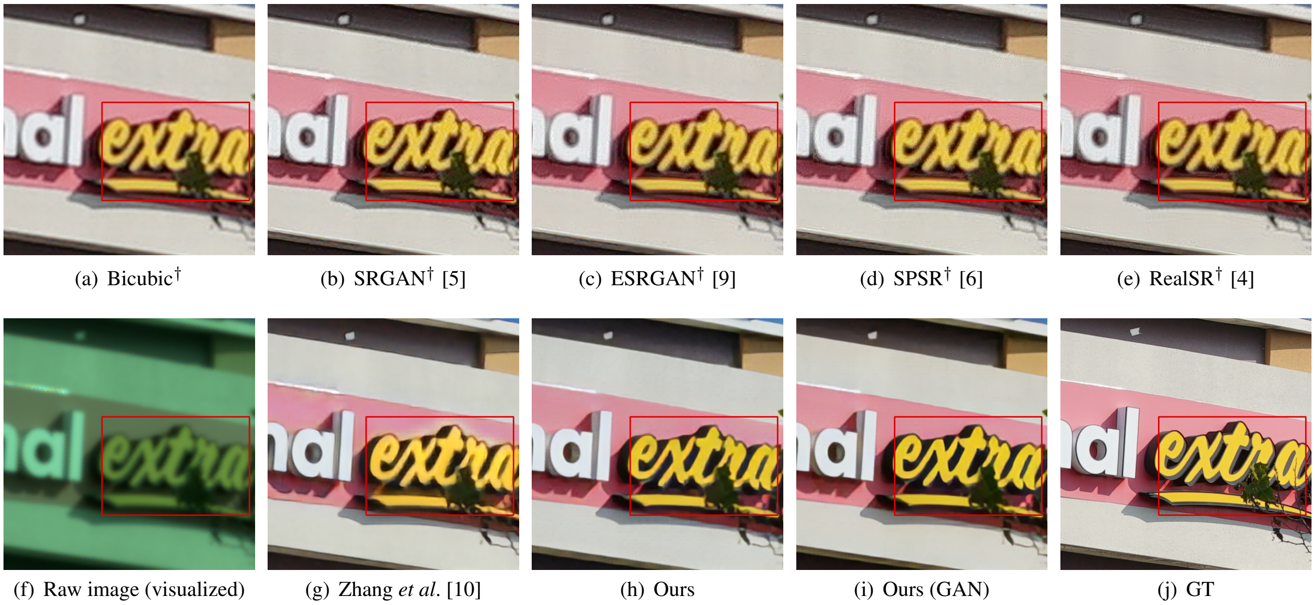
Figure C: Visual results of GCM output $\tilde{\mathbf{y}}$. (b)$\sim$(d) denote the results using different GCM components. (f)$\sim$(h) denote the illumination ratio between (b)$\sim$(d) and GT, respectively. With the guidence of $\mathbf{y}$, the color of GCM output $\tilde{\mathbf{y}}$ in (c) and (d) is closer to the target sRGB image. Dark corner can be observed in (b) and (c). In (b) and (c), the patch in the blue box is darker than the patch in the red box. But in (d) and (e), the patch in different boxes has similar illumination. The phenomenon can be seen more clearly in (f)$\sim$(h).



(a) Bicubic† (b) SRGAN† [5] (c) ESRGAN† [9] (d) SPSR† [6] (e) RealSR† [4]

(f) Raw image (visualized) (g) Zhang *et al*. [10] (h) Ours (i) Ours (GAN) (j) GT

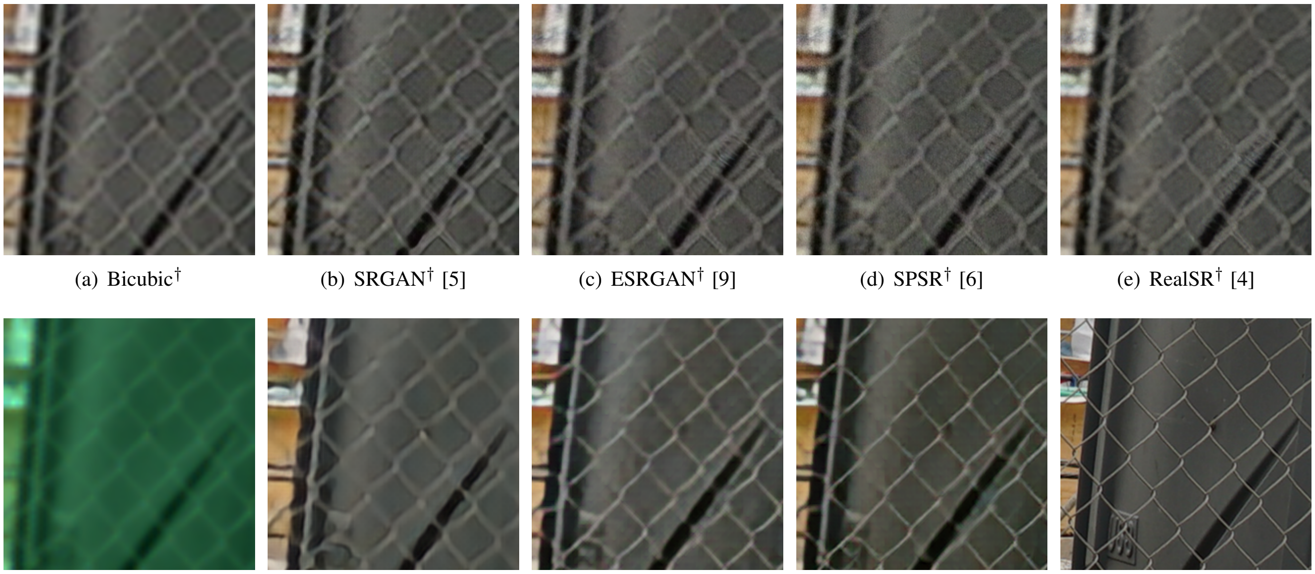
Figure D: Visual comparison on SR-RAW dataset. † means that the result is obtained given LR sRGB image as input. Our results have more textures on the leaves.



(a) Bicubic† (b) SRGAN† [5] (c) ESRGAN† [9] (d) SPSR† [6] (e) RealSR† [4]

(f) Raw image (visualized) (g) Zhang *et al*. [10] (h) Ours (i) Ours (GAN) (j) GT

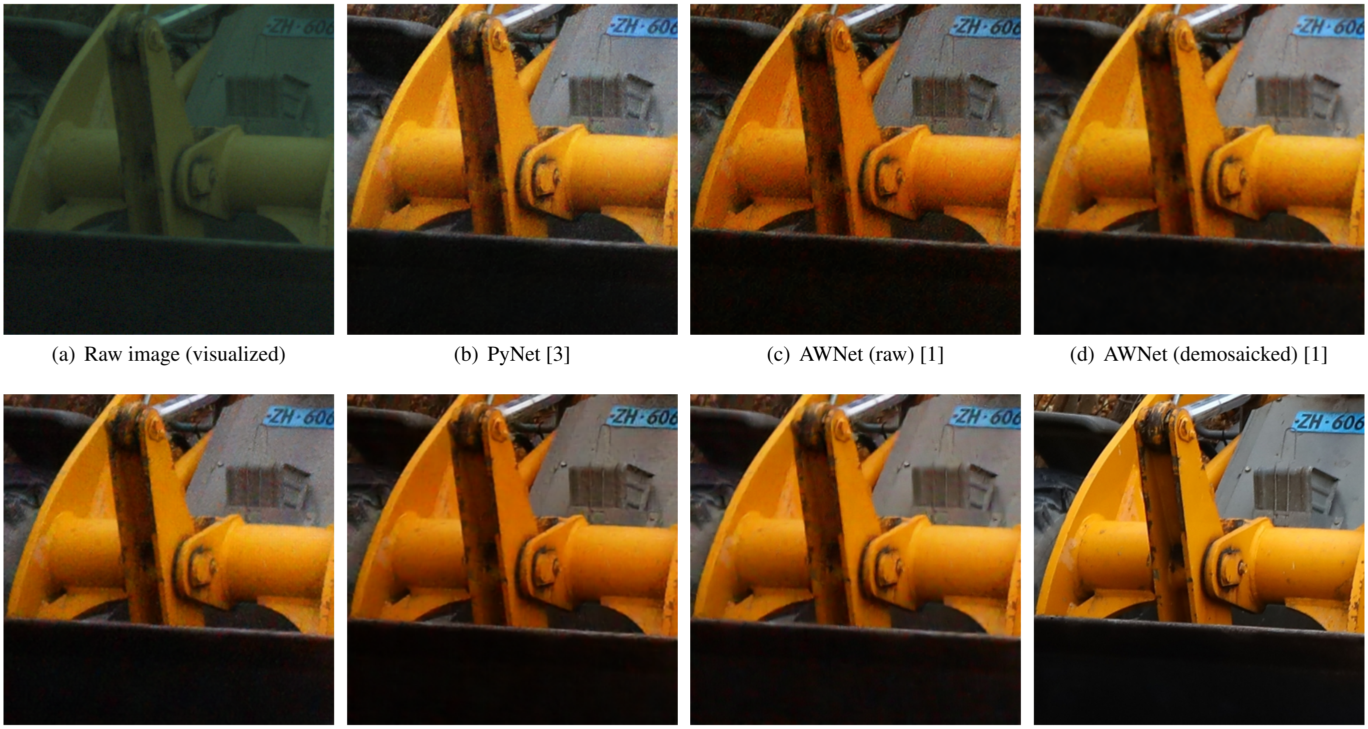
Figure E: Visual comparison on SR-RAW dataset. † means that the result is obtained given LR sRGB image as input. The edges of our results are sharper. It can be clearly observed in the red box.



(a) Bicubic† (b) SRGAN† [5] (c) ESRGAN† [9] (d) SPSR† [6] (e) RealSR† [4]

(f) Raw image (visualized) (g) Zhang *et al*. [10] (h) Ours (i) Ours (GAN) (j) GT

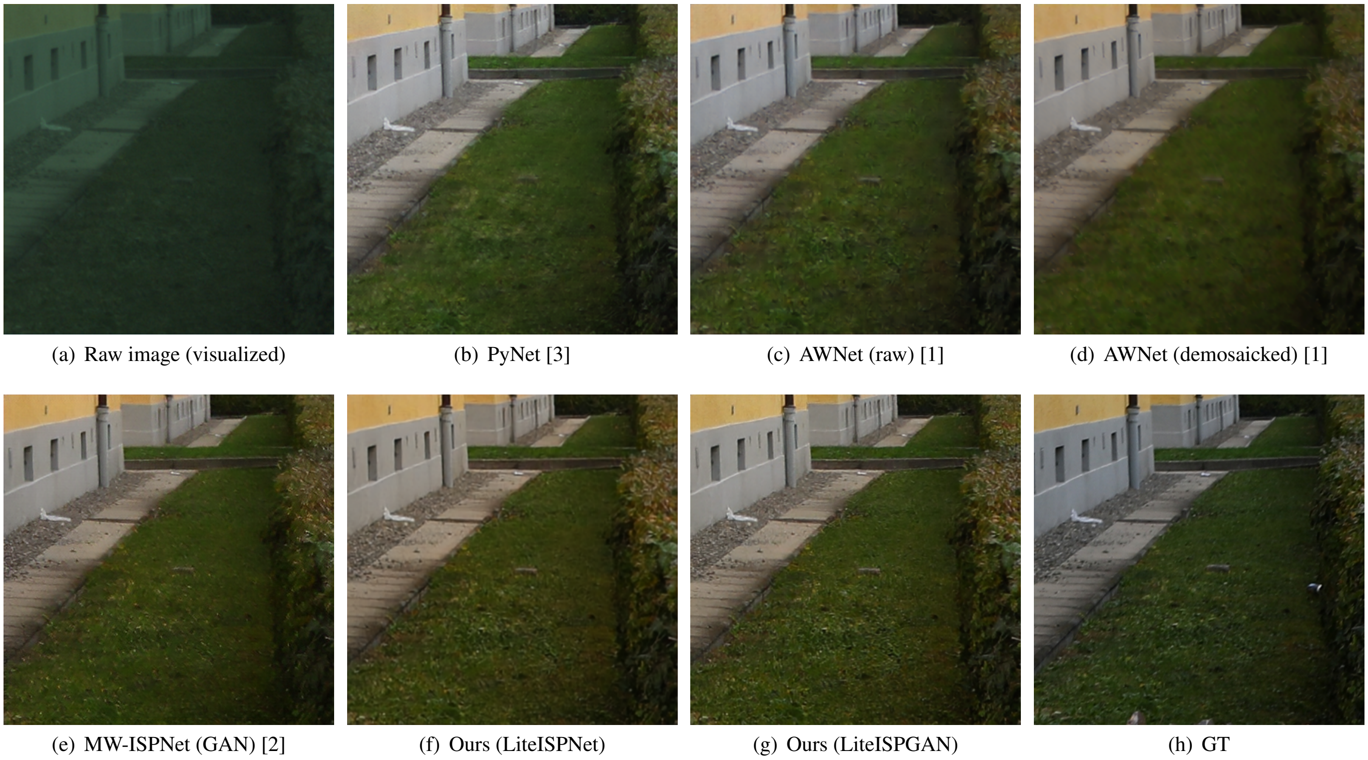
Figure F: Visual comparison on SR-RAW dataset. † means that the result is obtained given LR sRGB image as input. The edges of our results are sharper.



(a) Raw image (visualized) (b) PyNet [3] (c) AWNet (raw) [1] (d) AWNet (demosaicked) [1]

(e) MW-ISPNet (GAN) [2] (f) Ours (LiteISPNet) (g) Ours (LiteISPGAN) (h) GT

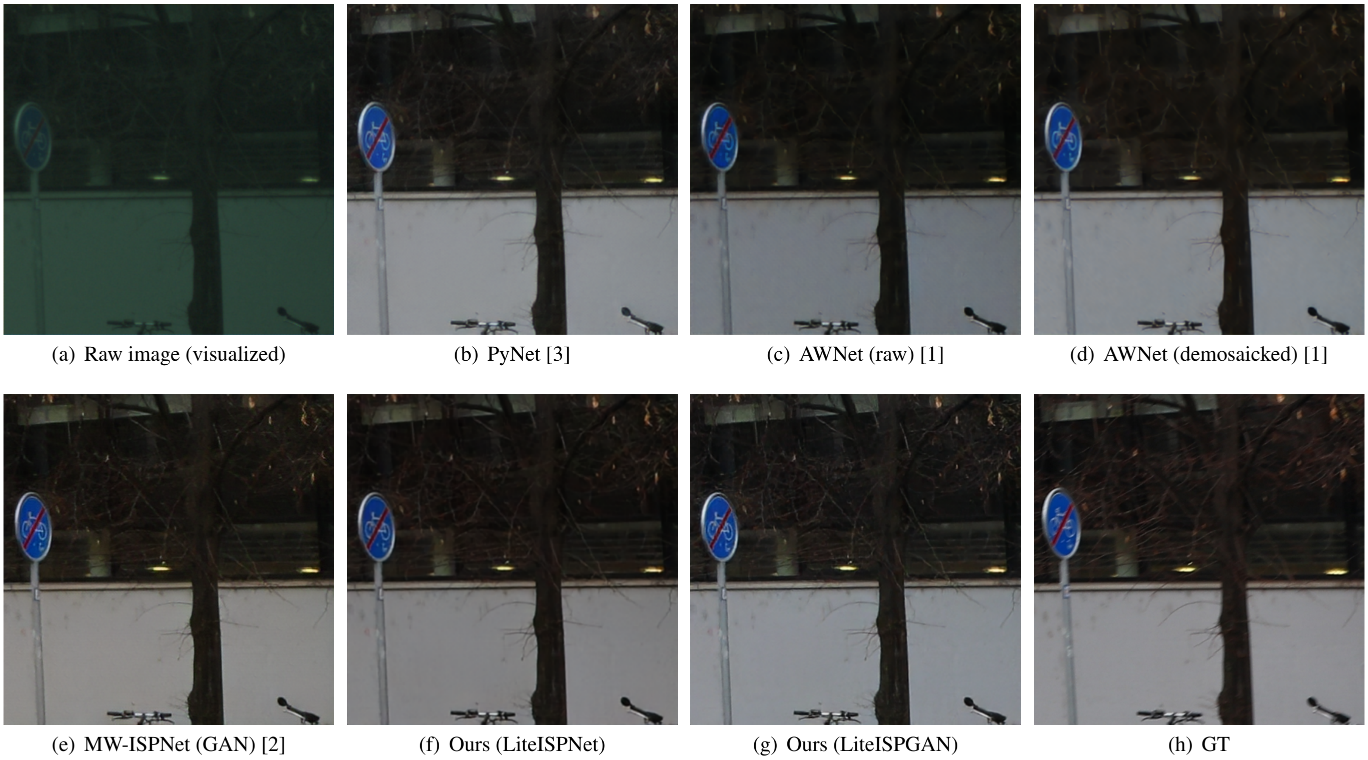
Figure G: Visual comparisons on ZRR dataset. Our results have less noise.



(a) Raw image (visualized) (b) PyNet [3] (c) AWNet (raw) [1] (d) AWNet (demosaicked) [1]

(e) MW-ISPNet (GAN) [2] (f) Ours (LiteISPNet) (g) Ours (LiteISPGAN) (h) GT

Figure H: Visual comparisons on ZRR dataset. Our results have richer textures on the grass.



(a) Raw image (visualized) (b) PyNet [3] (c) AWNet (raw) [1] (d) AWNet (demosaicked) [1]

(e) MW-ISPNet (GAN) [2] (f) Ours (LiteISPNet) (g) Ours (LiteISPGAN) (h) GT

Figure I: Visual comparisons on ZRR dataset. The tree branches in our results are clearer.

References

[1] Linhui Dai, Xiaohong Liu, Chengqi Li, and Jun Chen. Awnet: Attentive wavelet network for image isp. In *European Conference on Computer Vision Workshops (ECCVW)*, 2020.

[2] Andrey Ignatov, Radu Timofte, et al. Aim 2020 challenge on learned image signal processing pipeline. In *European Conference on Computer Vision Workshops (ECCVW)*, 2020.

[3] Andrey Ignatov, Luc Van Gool, and Radu Timofte. Replacing mobile camera isp with a single deep learning model. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 536–537, 2020.

[4] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 466–467, 2020.

[5] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4681–4690, 2017.

[6] Cheng Ma, Yongming Rao, Yean Cheng, Ce Chen, Jiwen Lu, and Jie Zhou. Structure-preserving super resolution with gradient guidance. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7769–7778, 2020.

[7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer Assisted Intervention*, pages 234–241. Springer, 2015.

[8] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. PWC-net: Cnns for optical flow using pyramid, warping, and cost volume. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8934–8943, 2018.

[9] 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 *European Conference on Computer Vision Workshops (ECCVW)*, pages 0–0, 2018.

[10] Xuaner Zhang, Qifeng Chen, Ren Ng, and Vladlen Koltun. Zoom to learn, learn to zoom. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3762–3770, 2019.

[11] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2223–2232, 2017.