Exploring Efficient Asymmetric Blind-Spots for Self-Supervised Denoising in Real-World Scenarios

Supplementary Material

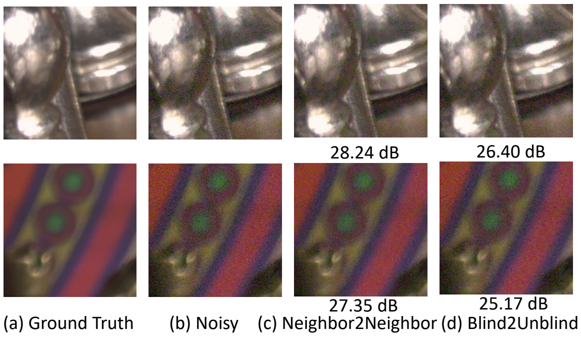
S1. Implementation Details

Training Settings. We adapt UNet-like BSN as our network. We set $k_a = 9$ and $k_b = 3$ for training and inference, respectively. For the BSN training, we follow settings of previous work [5]. The input images are cropped into 256×256 patches with a batch size of 8 and augmented with random flipping and rotation. The network is optimized with an Adam optimizer with a learning rate of 3×10^{-4} and $[\beta_1, \beta_2]$ of $[0.9, 0.999]$. The BSN is trained for 400k iterations, and the learning rate is decreased to zero with cosine annealing scheduler. For multi teacher distillation, we sample blind-spots from $Ks \in \{0, 1, 3, 5, 7, 9, 11\}$ for SIDD dataset, and $Ks \in \{0, 1, 3, 5, 7, 9\}$ for DND dataset as the noise intensity in the DND dataset is relatively small. We crop the input images into 128×128 patches with a batch size of 8. We set the initial learning rate to 3×10^{-4} and decrease it to zero with cosine annealing scheduler during 100k iterations.

Network Structure. We use a non-blind-spot network (NBSN) during multi teacher distillation, which don't use any shift or rotation operations and remove the final 1×1 convolutional layers. Specifically, We show the performance of student networks with three different parameters after distillation in main paper Tab.3. The student A is a lightweight network consisting of 2 downsampling and 2 upsampling layers, each containing only 1 convolution layer. The student B is simillar to A, but uses 3 convolution layers in the downsampling layer and 4 convolution layers in the upsampling layer. The student C increase both the upsampling and downsampling layers to 5, and uses 2 convolution layer and 1 convolution layers in them, respectively. We can find that even small networks can significantly benefit from multi teacher distillation. In addition, increasing the number of downsampling layers or convolutional layers to increase network depth can further improve performance, with the latter achieving better performance.

Methods	Downsampling based	Patch-masked conv based	Feature-shifted based (Ours)
Symmetric	33.40	35.64	36.35 ($k_{a/b} = 9/9$)
Asymmetric	**34.86**	**37.32**	**36.80** ($k_{a/b} = 9/3$)

Table S1. Comparison between symmetric and asymmetric designs on SIDD validation dataset.



(a) Ground Truth (b) Noisy (c) Neighbor2Neighbor (d) Blind2Unblind

Figure S1. Qualitative comparisons of Neighbor2Neighbor and Blind2Unblind on SIDD validation dataset.

Details of Distillation. Our multi-teacher distillation training process exhibits lower overall complexity compared to training process of AT-BSN. As illustrated in Fig.4 and Sec.3.4, only the meta-teacher (a shifted UNet) needs to compute four directional features once, then produce 7 offset features with negligible shift operations. These features, considered as outputs from potential teachers with different blind spots, undergo final shallow 1×1 Convs to yield 7 distillation labels. Thus the total distillation cost is $4{*}MAC_{shifted\_unet}{+}7{*}MAC_{1\times1}{+}MAC_{light\_unet}$, not $4{*}7{*}(MAC_{shifted\_unet}{+}MAC_{1\times1}){+}MAC_{light\_unet}$. All experiments were done on a RTX 3080. Following the distillation settings above ($batch\_size$=8, $patch$=128, $total\_iterations$=100k), the distillation finished in around 2.5 hours with speed of 11.1 iters/s and 2.4G GPU memory. In contrast, the Shifted UNet, with the same settings, requires around 3.2 hours to finish 100k iterations with speed of 8.3 iters/s and 5.7G GPU memory. Notably, during distillation, no gradient computation is needed for meta-teacher, and features in four directions are unnecessary for student. Furthermore, in practice, we can speed up the distillation process by loading part of the weights from the trained BSN to initialize the student C as they are almost identical, except for the absence of the last 1×1 convolutions and the difference in the output channel on the last upsampling layer. It should be noted that the shift and rotation operations within BSN only affect the feature maps and do not impact the application of BSN weights to NBSN.

S2. Importance of Asymmetric Design

Fig.7 compares symmetric and asymmetric BSN, where asymmetric BSN turns to symmetric one with consistent k

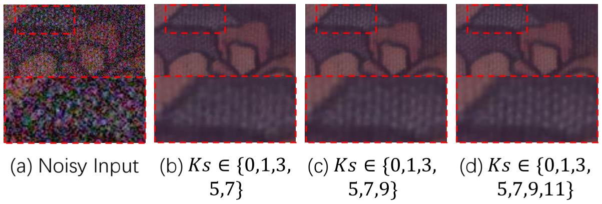

(a) Noisy Input (b) $Ks \in \{0,1,3,5,7\}$ (c) $Ks \in \{0,1,3,5,7,9\}$ (d) $Ks \in \{0,1,3,5,7,9,11\}$

Figure S2. Fusion results of teachers.

in training and inference. Optimal asymmetric BSN performance requires careful use of k. Tab.S1 contrasts three asymmetric designs in Fig.2 with their symmetric counterparts (results are from AP-BSN, LG-BPN, and Fig.7), demonstrating the importance of asymmetric design.

S3. Neighbor2Neighbor and Blind2Unblind

Neighbor2Neighbor and Blind2Unblind are two methods that do not consider the spatial correlation of noise, and in this section we will demonstrate their performance degradation in real-world noise scenarios.

Neighbor2Neighbor [1] proposes to subsample the noisy input images to obtain noisy pairs for Noise2Noise-like training. Blind2Unblind [7] proposes a global-aware mask mapper and re-visible loss to fully excavate the information in the blind-spot for Noise2Void-like training. Nevertheless, these methods rely on the pixel-wise independent noise assumption [2, 4], which is not satisfied in real-world scenarios. To be specific, the central pixel can be inferred using the neighboring noisy pixels as clues. We retrain both methods on the SIDD-Medium sRGB dataset, and report PSNR values of 25.98 dB and 23.10 dB, respectively. We find that both methods learn an approximate identity mapping that is close to the noisy input itself, as illustrated in Fig. S1

S4. Mean Teacher Distillation and Multi Teacher Distillation

Fig. S2 presents the fusion results of teachers, corresponding to 2-4 columns in Tab.2. One can find gradually improved visual effects as more teachers are fused. Moreover, from Fig. S3, we can find that result of mean teacher distillation is smoother. Although both mean teacher distillation and multi teacher distillation tend to learn the average image, but we consider all teachers to contribute equally, and the mean teacher cannot capture all the details of teacher distribution, especially when there are significant differences between teachers (e.g. $k_b = 1$ and $k_b = 11$). On the contrary, we utilize multi teacher distillation to learn different knowledge equally from multiple teacher networks, thereby further improving performance. We avoid explicit region partitioning in distillation loss of [5], ensuring our method not affected by the unstable induction bias of such process, that is, the partitioning heavily relies on initial results. To support this, we imitate the partitioning and utilize outputs of k=9 & k=0 to distill flat and texture area respectively, leading to 37.27 dB, weaker than 37.59 dB with our multi-teacher approach.

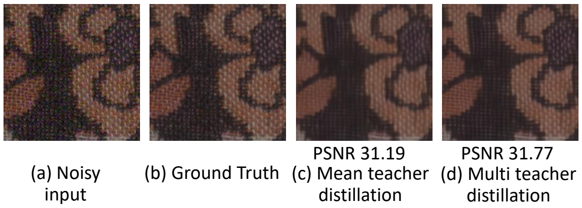

(a) Noisy input (b) Ground Truth PSNR 31.19 (c) Mean teacher distillation PSNR 31.77 (d) Multi teacher distillation

Figure S3. Qualitative comparisons of mean teacher distillation and multi teacher distillation.

S5. Additional Qualitative Results

S5.1. Different Combinations of Blind-Spots

To investigate the impact of different combinations of blind-spot sizes k during training and inference, we conduct additional qualitative experiments. Please note that for the setting of $k_a = 7$, we use early stopping to avoid overfitting.

We select a set of images with rich textures and another set with relatively flat textures. From Fig. S4, We find that under various k_a, for flat images, the PSNR of the results is approximately positively correlated with k_b. This is because larger blind-spots during testing can effectively suppress noise correlation, and the recovery of flat areas is insensitive to the loss of local information. For images with more textures, the PSNR of the results is approximately negatively correlated with k_b. This is because larger blind-spots during testing will lead to the loss of local spatial information, making it difficult to recover local texture details.

Interestingly, we also find that when k_a is larger during training, the destructive effect of larger k_b on texture information becomes smaller. This is because the network trained on larger k_a has stronger ability to find clues from farther places to recover the current pixel, so it can still maintain a certain degree of texture information at larger k_b.

Visually, the flat area gradually becomes cleaner with the increase of k_b, while the texture area gradually becomes more blurred with the increase of k_b. This once again shows that different blind-spots have different denoising effects on flat/texture areas. From the above analysis, we can see that the removal of noise and the preservation of texture details is a dilemma. Our multi-teacher distillation can learn from multiple teacher networks with different blind-spots, thereby achieving a balance in suppressing noise space correlation and maintaining local textures, that is, achieving a balance in denoising flat areas and texture areas, thereby greatly improving performance.

S5.2. More Results on DND and SIDD Datasets

Fig. S5, Fig. S6, Fig. S7 and Fig. S8, Fig. S9, Fig. S10 present qualitative comparisons between our proposed method and other approaches on the SIDD and DND benchmark datasets. We apply models trained on the SIDD Medium dataset directly to SIDD benchmark to demonstrate the generalization ability of these methods. For DND dataset, we utilize models trained directly on it to show the advantage of the full self-supervised methods.

We observe that our method outperforms other methods on both benchmark datasets, producing evidently better denoising results while preserving more texture details and less blur. One can find that although AP-BSN(R^3) [3] attempts to use R^3 [3] to eliminate the aliasing effect, the aliasing effect cannot be completely eliminated. In addition, although SDAP(E) [6] performs well in flat areas, it tends to over-smooth in texture areas, losing many details. SpatiallyAdaptive [5] and LG-BPN [8] can retain more details because they do not downsample the input image. However, the restoration of texture details is still not as good as our method. We also notice that LG-BPN produces cross artifacts when dealing with larger images in DND datasets (512×512), which we believe is caused by its downsampling operation in the feature domain. It should also be noted that the large kernel operation and post-processing operation of LG-BPN make its computational complexity extremely high, as shown in main paper Tab.4. Our AT-BSN can produce sharper images, while AT-BSN (D) is slightly smoother, achieving a balance between noise removal and texture detail preservation, thus the overall visual effect is better.

References

[1] Tao Huang, Songjiang Li, Xu Jia, Huchuan Lu, and Jianzhuang Liu. Neighbor2Neighbor: Self-supervised denoising from single noisy images. In *CVPR*, 2021. 2

[2] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2Void-learning denoising from single noisy images. In *CVPR*, 2019. 2

[3] Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. Ap-bsn: Self-supervised denoising for real-world images via asymmetric pd and blind-spot network. In *CVPR*, pages 17725–17734, 2022. 3

[4] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2Noise: Learning image restoration without clean data. In *ICML*, 2018. 2

[5] Junyi Li, Zhilu Zhang, Xiaoyu Liu, Chaoyu Feng, Xiaotao Wang, Lei Lei, and Wangmeng Zuo. Spatially adaptive self-supervised learning for real-world image denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9914–9924, 2023. 1, 2, 3

[6] Yizhong Pan, Xiao Liu, Xiangyu Liao, Yuanzhouhan Cao, and Chao Ren. Random sub-samples generation for self-supervised real image denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12150–12159, 2023. 3

[7] Zejin Wang, Jiazheng Liu, Guoqing Li, and Hua Han. Blind2unblind: Self-supervised image denoising with visible blind spots. In *CVPR*, pages 2027–2036, 2022. 2

[8] Zichun Wang, Ying Fu, Ji Liu, and Yulun Zhang. Lg-bpn: Local and global blind-patch network for self-supervised real-world denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18156–18165, 2023. 3

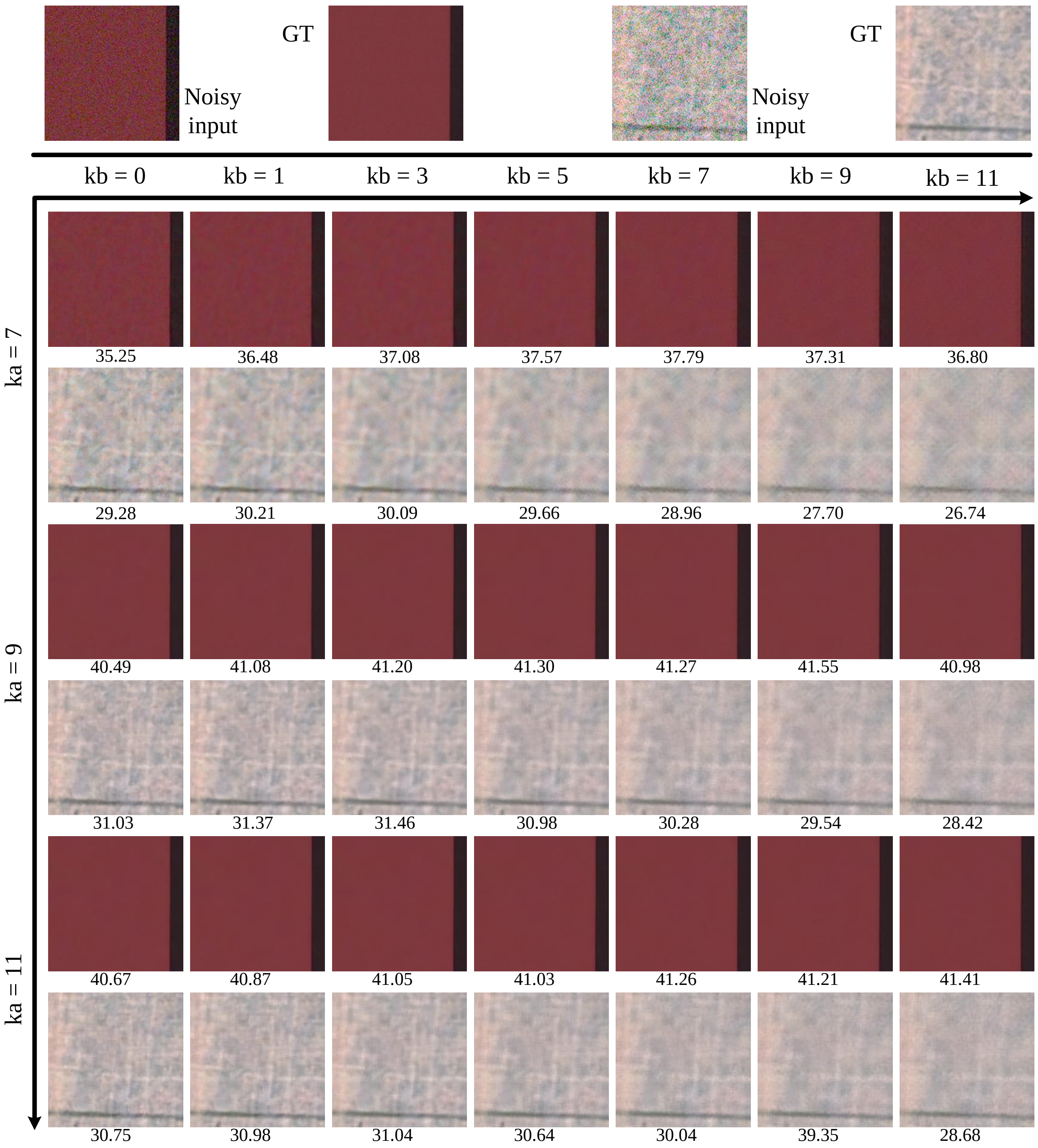
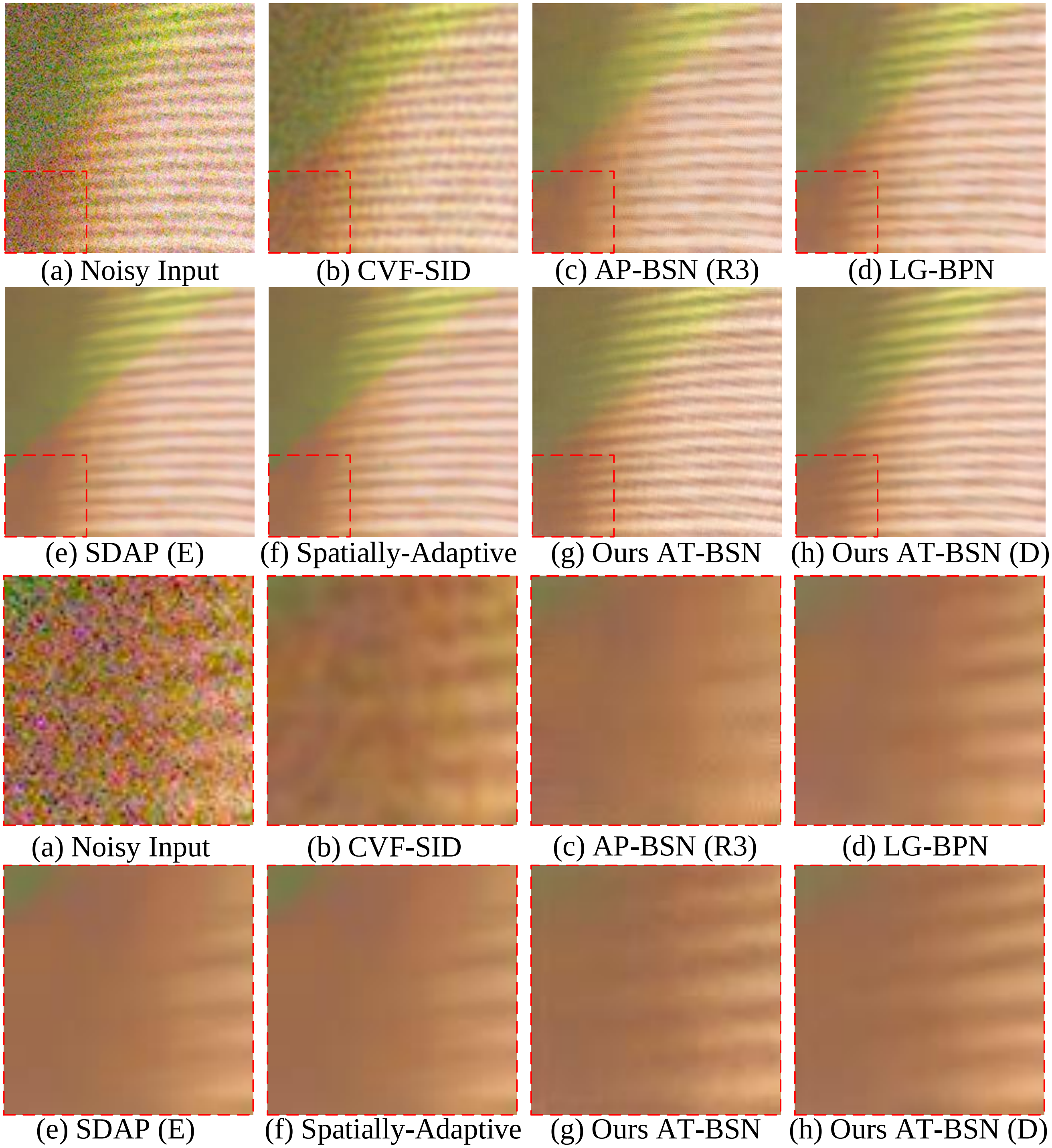


Figure S4. Qualitative results of different combinations of blind-spots during training (k_a) and inference (k_b) on SIDD validation dataset.



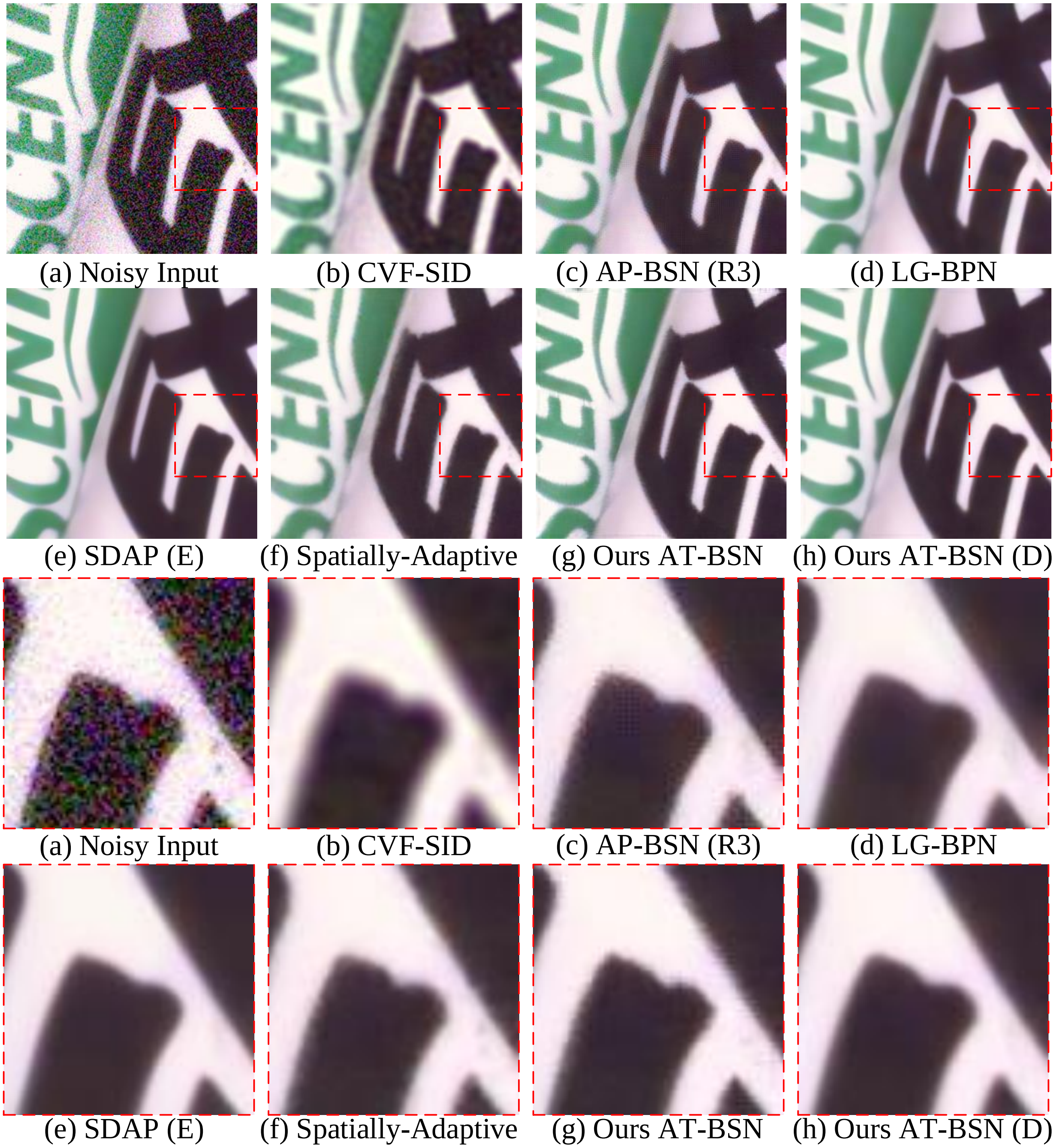
(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

Figure S5. Additional qualitative comparisons on SIDD benchmark dataset.



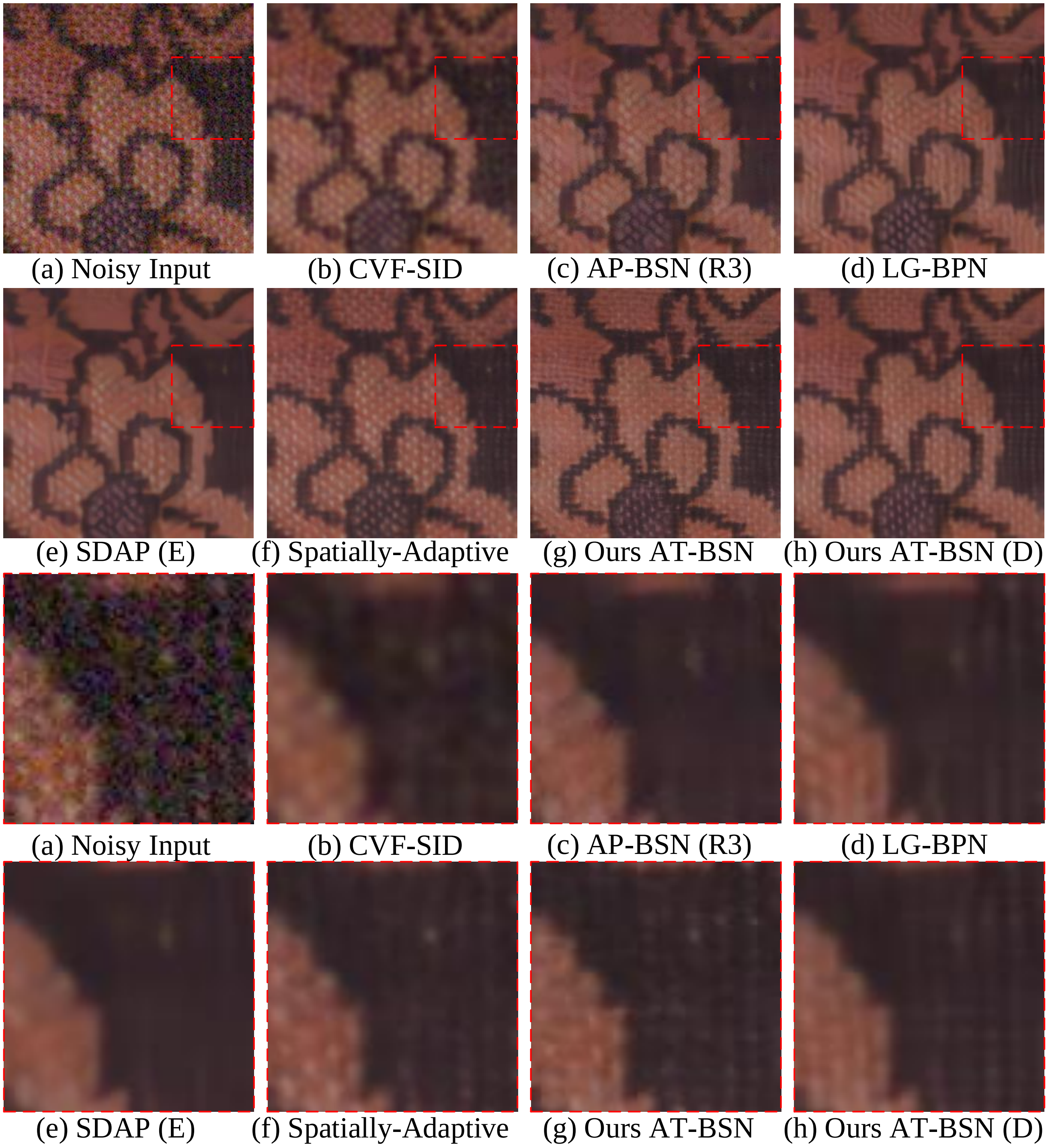
(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

Figure S6. Additional qualitative comparisons on SIDD benchmark dataset.



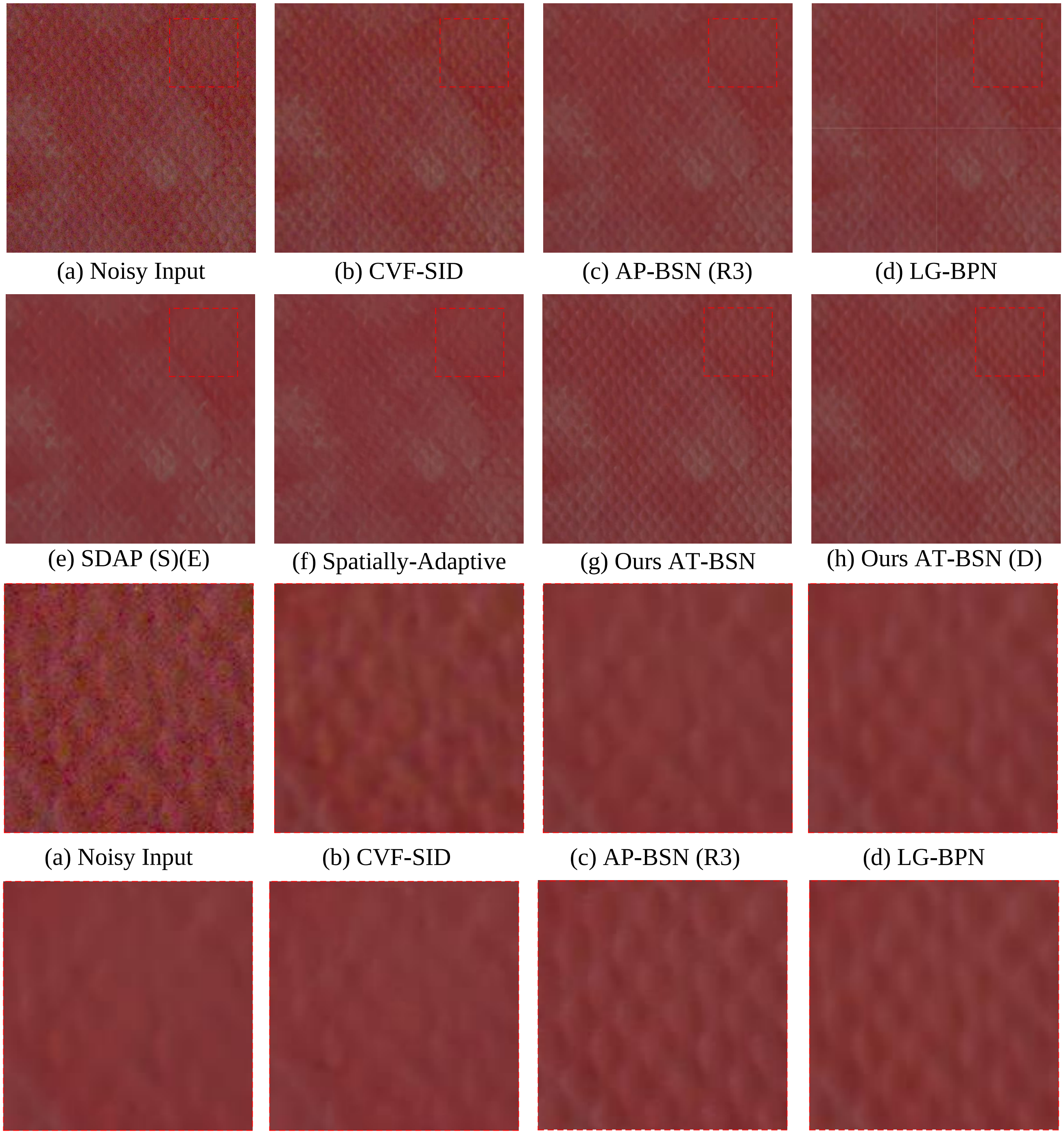
(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

Figure S7. Additional qualitative comparisons on SIDD benchmark dataset.



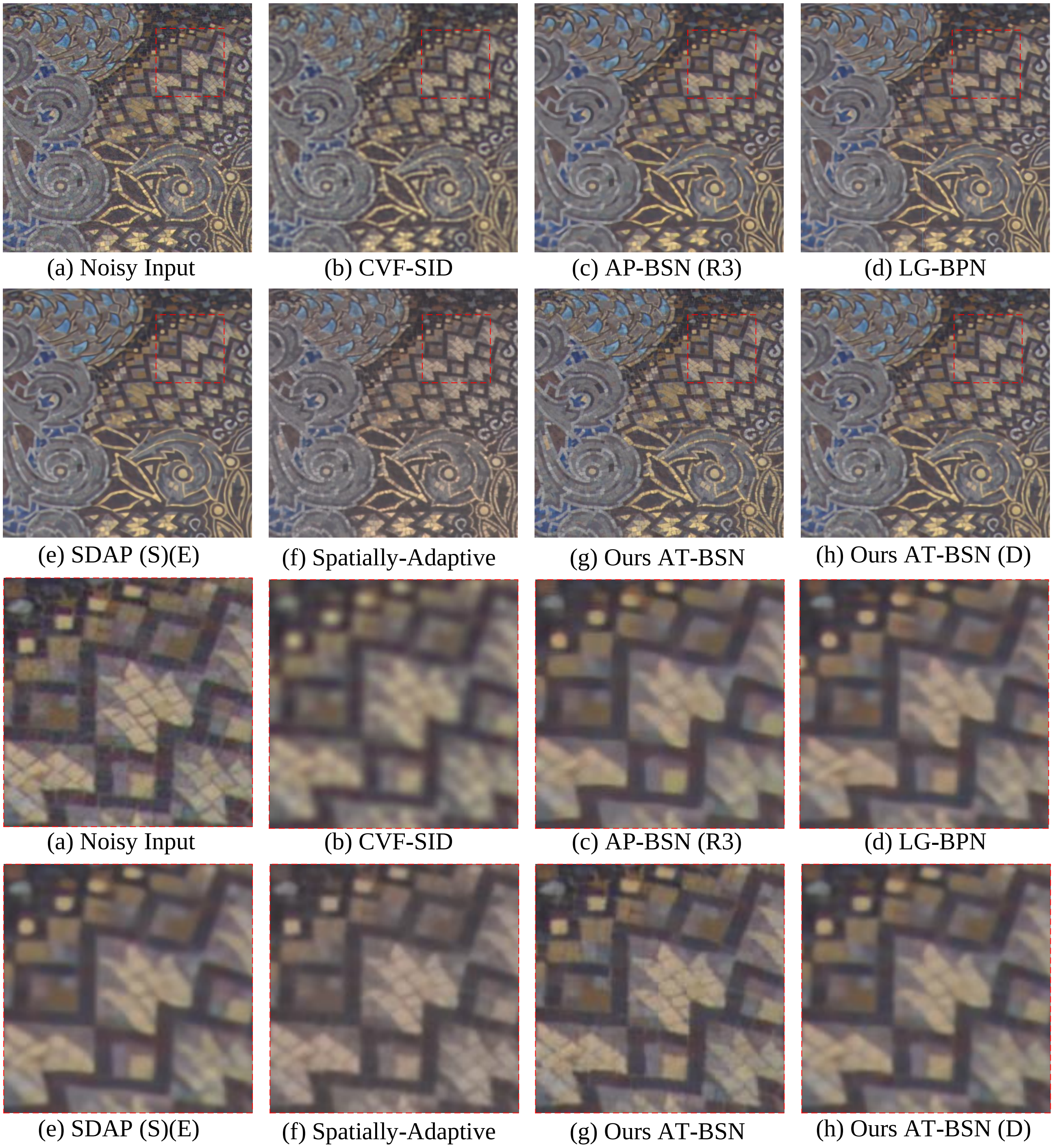
(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (S)(E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (S)(E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

Figure S8. Additional qualitative comparisons on DND benchmark dataset.



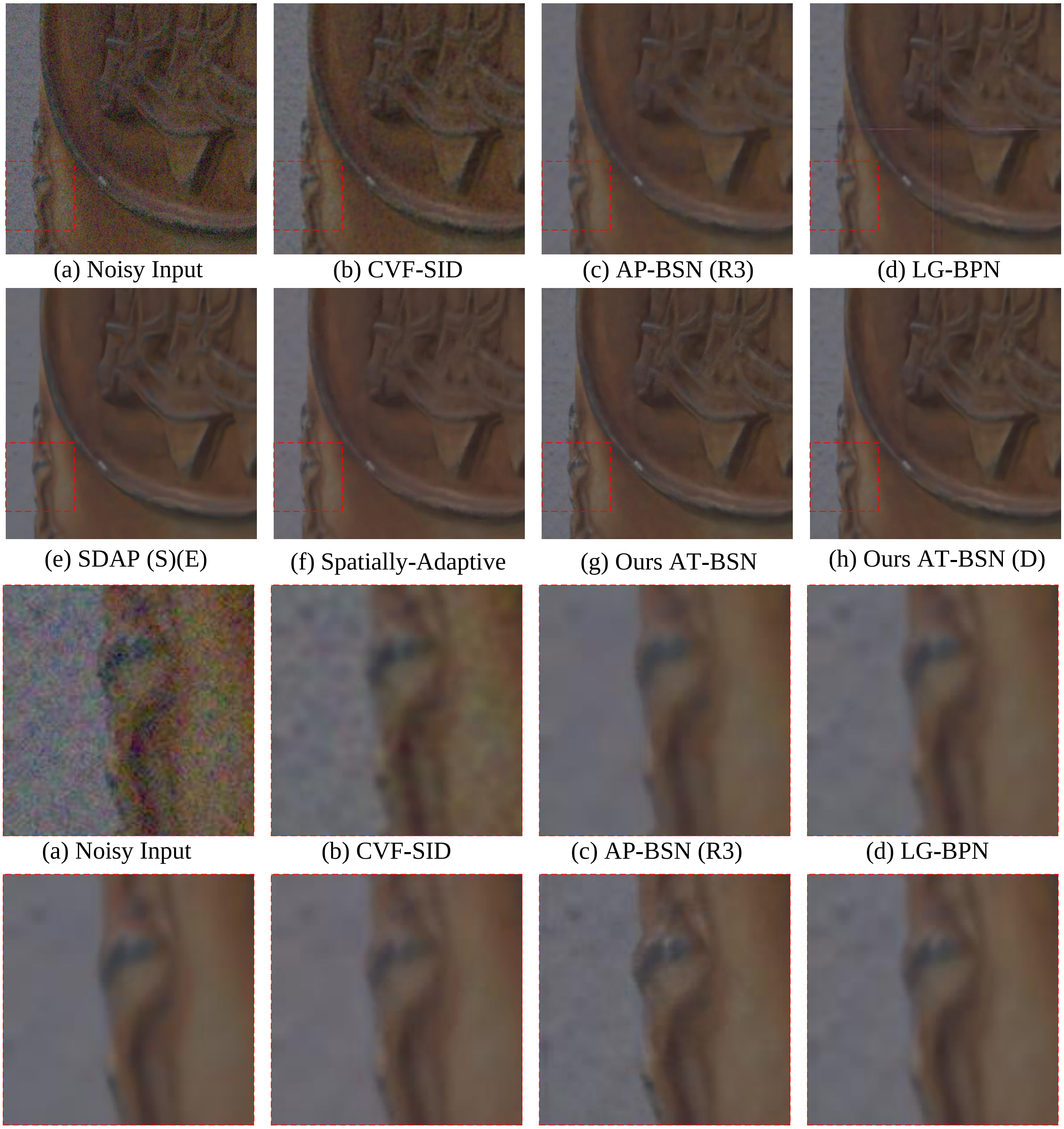
(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (S)(E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (S)(E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

Figure S9. Additional qualitative comparisons on DND benchmark dataset.



(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (S)(E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

(a) Noisy Input (b) CVF-SID (c) AP-BSN (R3) (d) LG-BPN

(e) SDAP (S)(E) (f) Spatially-Adaptive (g) Ours AT-BSN (h) Ours AT-BSN (D)

Figure S10. Additional qualitative comparisons on DND benchmark dataset.