2-Shots in the Dark: Low-Light Denoising with Minimal Data Acquisition

Supplementary Material

This appendix is organized as follows:

For optimal visual comparisons, we recommend viewing the results on a screen.

6. Method Details

6.1. Spectral Sampling Algorithm

The full procedure of our spectral sampling algorithm for dark-frame synthesis is summarized in Algorithm 1.

Algorithm 1: Spectral Sampling with Iterative Refinement for Dark Frame Synthesis

Input: Reference dark frame $I_{\text{dark}} \in \mathbb{R}^{C\times H\times W}$, Gaussian blur parameter σ, number of iterations K
Output: Synthesized dark frame $\tilde{I}_{\text{dark}}$
Step 1: Fixed-pattern removal
$S \leftarrow \mathcal{G}_\sigma(I_{\text{dark}})$; // Extract low-frequency pattern
$R' \leftarrow I_{\text{dark}} - S$; // Isolate stochastic residual
$\mu^R \leftarrow \mathbb{E}[R']$; // Compute channel-wise mean
$R \leftarrow R' - \mu^R$; // Center to zero mean
Step 2: Spectral prior estimation
$\widehat{R} \leftarrow \mathcal{F}\{R\}$; // Compute reference spectrum
Step 3: Phase randomization
$\xi^0 \sim \mathcal{D}\big([-\pi,\pi]^{(H,W)}\big)$; // Uniform random phase
$\xi \leftarrow \text{replicate}(\xi^0, C)$; // Copy across channels
$\widehat{N} \leftarrow |\widehat{R}| \odot \exp\Big(i\big(\theta_{\widehat{R}} + \xi\big)\Big)$; // Phase Randomization
$N^{(0)} \leftarrow \frac{1}{\sqrt{HW}}\mathcal{F}^{-1}\{\widehat{N}\}$; // Initial synthesis
Step 4: Iterative histogram-spectral refinement
for $k = 0$ **to** $K-1$ **do**
  $N^{\prime(k)}_{\text{hist}} \leftarrow \mathcal{H}(N^{(k)}, R)$; // Match Hist.
  $\mu^{(k)}_{hist} \leftarrow \mathbb{E}[N^{\prime(k)}_{\text{hist}}]$
  $N^{(k)}_{\text{hist}} \leftarrow N^{\prime(k)}_{\text{hist}} - \mu^{(k)}_{hist}$
  $\widehat{N}^{(k)}_{\text{hist}} \leftarrow \mathcal{F}\{N^{(k)}_{\text{hist}}\}$
  $\widehat{N}^{(k)}_{\text{corrected}} \leftarrow |\widehat{R}| \odot \exp(i\theta_{\widehat{N}^{(k)}_{\text{hist}}})$; // Restore spectrum
  $N^{(k+1)} \leftarrow \frac{1}{\sqrt{HW}}\mathcal{F}^{-1}\{\widehat{N}^{(k)}_{\text{corrected}}\} + \mu^{(k)}_{hist}$
end
Step 5: Final reconstruction
$\tilde{I}_{\text{dark}} \leftarrow N^{(K)} + S + \mu^R$; // Restore fixed pattern and mean
return $\tilde{I}_{\text{dark}}$

6.2. Photon Noise Parameter Estimation

In Sec. 3.2 of the main paper, we have shown that the noisy observation y can be modeled as:

$$y = g\mathcal{P}(x) + n_{\text{other}}. \tag{16}$$

For a Poisson random variable, the variance equals its mean, i.e., $\text{Var}(\mathcal{P}(x)) = x$. Thus the variance of the noisy observation is:

$$\begin{aligned}\text{Var}(y) &= g^2\text{Var}(\mathcal{P}(x)) + \text{Var}(n_{\text{other}}) \\ &= g^2 x + \text{Var}(n_{\text{other}}) \\ &= g(gx) + \text{Var}(n_{\text{other}}).\end{aligned} \tag{17}$$

Consequently, the system gain g can be estimated by performing a linear fit between the variance of the noisy observation y and the amplified clean signal gx, using a set of clean–noisy pairs. If only a single noisy image is available,

a rough estimate of g can still be obtained. Specifically, we unfold the noisy image into overlapping 3×3 patches. For each patch, we apply a Gaussian blur to obtain a pseudo-clean value, and all pixels within the patch serve as noisy observations from which the noise variance is computed. Patches sharing the same mean value are then grouped together to produce more robust variance estimates.

Of course, more sophisticated approaches can be used for pseudo-clean estimation when needed, and additional data can possibly further improve accuracy. For example, when flat-field frames are available (*e.g.*, a uniform calibration card illuminated by a single light source), a more reliable estimation of g can be obtained by following the procedures described in [12, 20, 38].

7. More Implementation Details

7.1. How to Choose σ and K

Two parameters need to be set manually in our method. The first is the σ value used to estimate the dark shading from a single noisy image. To choose an appropriate σ, we assume that dark shadings are spatially smooth. Therefore, σ can be selected based on a smoothness measure of the blurred dark frame, *e.g.* the normalized total variation:

$$\widetilde{\mathrm{TV}}(\sigma) = \frac{\mathrm{TV}(G_\sigma * I_{\text{dark}})}{\mathrm{TV}(G_{\sigma_0} * I_{\text{dark}})}.$$

Starting from $\sigma_0 = 5$ and increasing $\sigma_i = \sigma_{i-1} + 5$, we select

$$\sigma = \min\{\sigma_i : |\widetilde{\mathrm{TV}}(\sigma_i) - \widetilde{\mathrm{TV}}(\sigma_{i-1})| < 10^{-2}\}.$$

Intuitively, as σ increases, the Gaussian blur progressively removes high-frequency noise while preserving the low-frequency structure of the dark shading. Once most of the high-frequency noise has been smoothed out, further increasing σ produces only negligible changes in the normalized total variation. The stopping criterion therefore identifies the point at which the estimated dark shading becomes sufficiently smooth and stable.

The second parameter is K, the number of iterations used for histogram matching. In practice, the iterations can be terminated once the KL divergence between the synthetic and ground-truth dark frames falls below 5×10^{-4}.

7.2. Data Synthesis Details

In the LRID dataset [12], two types of dark frames of ISO=6400 are provided: one captured under normal sensor temperature and the other in the sensor's hot mode. For noise synthesis, we use a single dark frame from each setting to generate the corresponding synthetic dark frames. The synthesis results are shown in Fig. 14.

7.3. Training Details

We use the same U–Net architecture as [9] for all denoising experiments. The networks are trained for 500 epochs using the Adam optimizer. The initial learning rate is set to 2×10^{-4}, reduced by half at epoch 250, and fixed to 1×10^{-5} after epoch 400. During training, we use a batch size of 4 and randomly crop 256×256 patches with random horizontal flips for data augmentation. We train the denoisers using the L_1 loss between the network output and the clean reference image.

7.4. Evaluation Details

Since Peak Signal-to-Noise Ratio (PSNR) and Structural SIMilarity (SSIM) are sensitive to global illumination, even small brightness shifts can cause large metric variations. During denoising inference, we apply the illumination correction method from the ELD paper [38] to the denoised images. We apply this illumination correction to all compared methods before evaluating PSNR and SSIM. We compute the PSNR and SSIM between the denoised and reference images in the RAW domain.

8. More Discussions

8.1. Effect of the Fixed-Pattern Removal Before Sampling

Dark frames contain both deterministic fixed patterns and stochastic noise components. Directly applying spectral analysis to raw dark frames would conflate these two components: the low-frequency fixed patterns would dominate the power spectrum, biasing the estimated noise statistics and leading to synthesized noise that incorrectly replicates deterministic sensor artifacts as if they were stochastic variations.

To isolate the stochastic component for spectral modeling, we apply Gaussian blur (Eq. 1) to extract the fixed pattern, then work with the residual. Figure 7 demonstrates the effect of fixed-pattern removal for noise synthesis. Given a real dark frame, without fixed-pattern removal, the synthesized noise (bottom-left) incorrectly produces random patterns of the deterministic fixed pattern, failing to produce faithful results. With appropriate blur kernel sizes, the method successfully separates the two components. We set $\sigma = 50$ in our experiments.

8.2. Importance of Inter-Channel Correlation

Fig. 10 presents additional ablation results for the inter-channel correlation ICC. Without ICC, the denoised images exhibit noticeable residual banding artifacts, whereas our full method produces cleaner results.

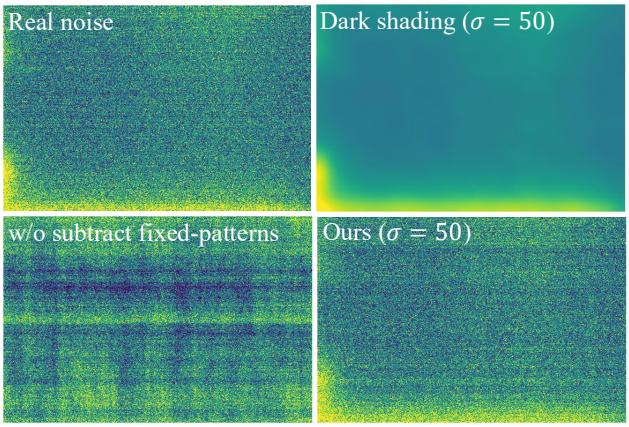

Figure 7. Effect of fixed-pattern removal before noise synthesis. Left: Real dark frame at ISO 12800 (top) and synthesized noise without fixed-pattern removal (bottom), which incorrectly reproduces deterministic row/column structure. Right: The estimated fixed-pattern component (top) and the synthesized noise after removing it prior to spectral sampling (bottom), resulting in a more realistic stochastic noise pattern.

8.3. Effect of Iterative Histogram Matching

In the main paper, we mentioned that we perform K iterations of histogram matching. Here, we study how the choice of K influences the accuracy of the synthesized noise. In Fig. 8, we show the relationship between the Kullback–Leibler divergence (KLD) between the synthetic and real noise and the value of K. As K increases, the synthetic noise progressively converges toward the real noise distribution, reflected by the decreasing KLD. We set $K = 10$ in our experiments.

In the main paper, we showed that iterative histogram matching (IHM) reduces color distortions in denoised results. Beyond this, we also find that IHM helps suppress saturated malfunctioning pixels. As shown in Fig. 11, without IHM, noticeable white spots appear in the denoised outputs due to saturated pixels in the noisy inputs. With our full method, this issue is effectively mitigated.

8.4. Different Ways to Leverage a Single Dark Frame

Given one single dark frame, we study the three different ways to leverage it for data synthesis: *DirectAdd*, *RandomCrop*, and *Ours*. In Fig. 12, we show two denoising examples, our method produces cleaner results with fewer artifacts.

8.5. Realism Validation of Synthetic Data

To further evaluate the realism of our synthetic noise, we train a denoising network (*RealDF*) using all available real dark frames from each ISO level in the SID dataset. For fair comparison, we generate one synthetic dark frame per real dark frame, ensuring identical data volume and ISO distribution, and train another denoising network (*SynDF*). The same protocol is applied to the LRID dataset. Fig. 9 shows the PSNR difference between *RealDF* and *SynDF* on the SID, ELD and LRID test set across different exposure ratios. The differences are consistently close to zero across all datasets and ratios, indicating that our method can synthesize noise distributions nearly indistinguishable from real sensor noise.

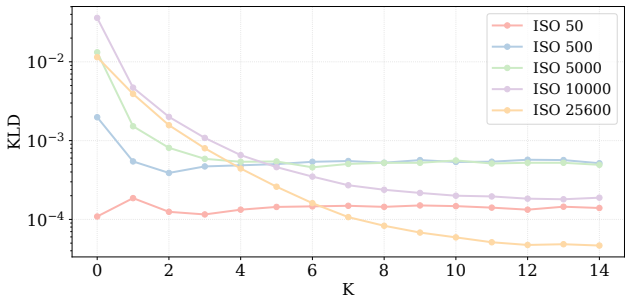

Figure 8. Relationship between the Kullback–Leibler divergence (KLD) between the synthetic and real noise and the histogram matching iteration number K, across different ISO settings. Increasing K generally reduces the KLD.

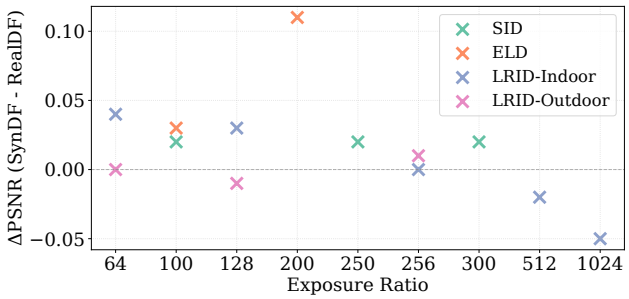

Figure 9. PSNR differences between *RealDF* and *SynDF* across different exposure ratios for the SID, ELD, and LRID test sets. Each × denotes the result of a subset of each test set, e.g., the leftmost × denotes that the PSNR difference between *RealDF* and *SynDF* on the LRID-Indoor dataset with exposure Ratio = 64 is almost 0, indicating that our method can synthesize noise distributions nearly indistinguishable from real sensor noise.

8.6. Photon Noise Parameter Estimation with Limited Data

In our experiments, we assume only a single noisy image per ISO is available for system gain estimation. To assess the effect of data availability, we compare denoising performance on the SID and ELD test sets when g is estimated using either one noisy image or 16 clean–noisy pairs.

As shown in Table 5, using a single noisy image leads to a slight drop in performance under some settings compared to using multiple clean–noisy pairs. While our study

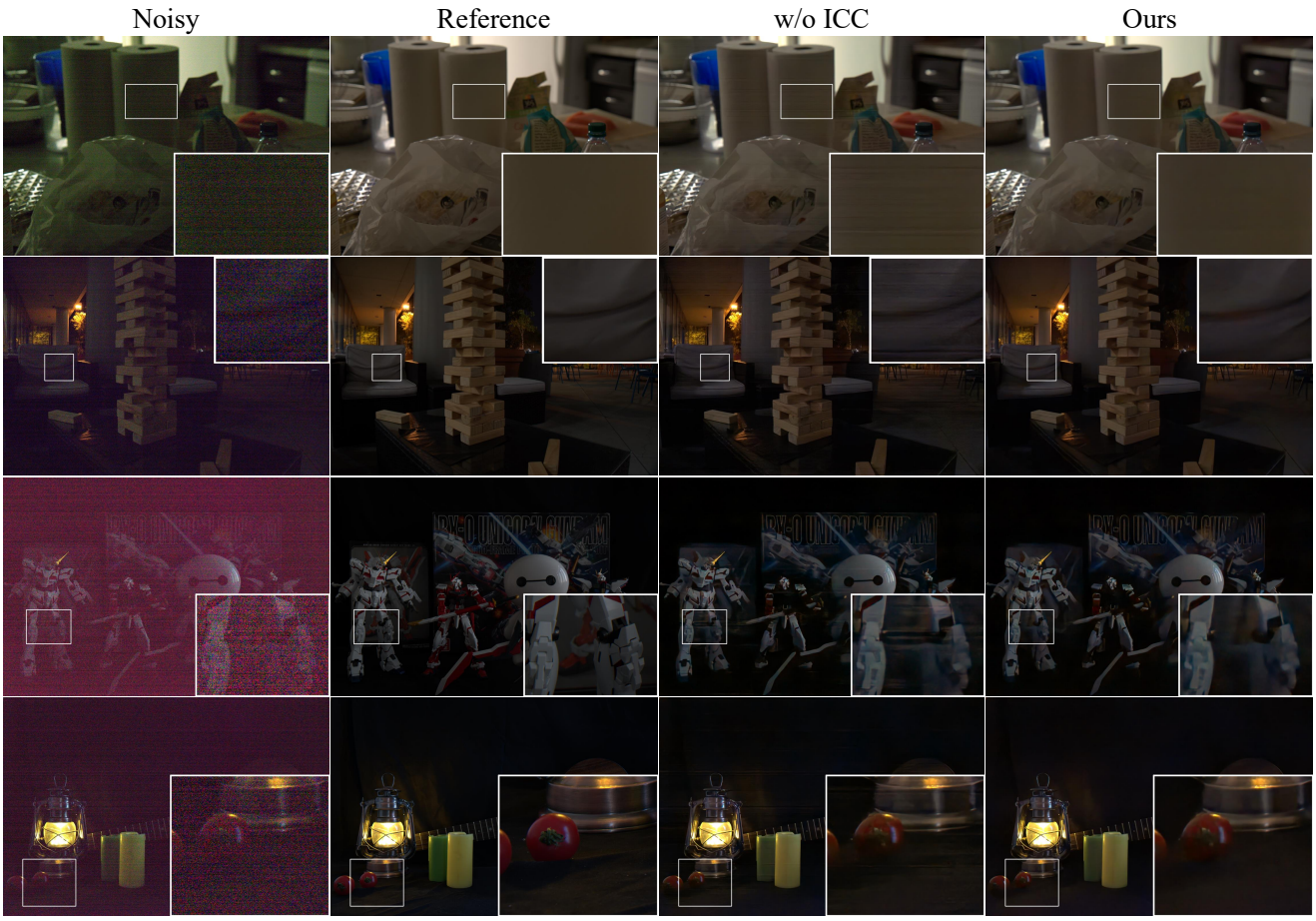

Figure 10. Ablation study on inter-channel correlation (ICC). Without ICC, the denoised results show residual banding artifacts, while our full method produces clean outputs. The first two examples are from the SID test set, and the third and fourth are from the ELD test set. Best viewed when zoomed in.

Table 5. Denoising performance when the system gain g is estimated from either a single noisy image or 16 clean–noisy pairs. Results are reported in PSNR / SSIM.

Dataset	Ratio	16 pairs	1 noisy
SID	×100	43.72 / 0.961	43.57 / 0.961
	×250	41.30 / 0.944	41.24 / 0.945
	×300	37.86 / 0.929	37.77 / 0.929
ELD	×100	47.14 / 0.986	47.13 / 0.986
	×200	44.78 / 0.966	44.89 / 0.969

focuses on the single noisy image scenario, access to additional data, such as multiple clean–noisy pairs or flat-field frames, could potentially improve the accuracy of gain estimation and denoising results.

8.7. Influence of Dark Shading Correction

To ensure a fair comparison, we apply dark shading correction (DSC) during denoising inference following [12, 29, 40]. This component is not our contribution. However, we report its influence on denoising performance here.

Table 6 shows the influence of DSC on denoising performance. *RealDF* and *SynDF* denote models trained with real and synthetic dark frames (DFs), respectively. *EDS* estimates the dark shading (DS) via Gaussian smoothing of a single dark frame, *ADS* averages 400 dark frames, and *w/o DS* disables DSC. Comparing (b) vs. (c) (and similarly (e) vs. (f)), removing DSC reduces denoising performance. *Note that all baselines in the paper use DSC, ensuring fair comparisons*. Comparing (a) vs. (b) (and (d) vs. (e)), using a dark shading estimated from a single dark frame causes only minor degradation ($\leq$ 0.1 dB), demonstrating the robustness of our method. The comparable performance of (a) and (d) further validates the realism of the synthetic noise. Note that in this ablation study, we use the Poisson noise parameters estimated from 16 clean-noisy pairs.

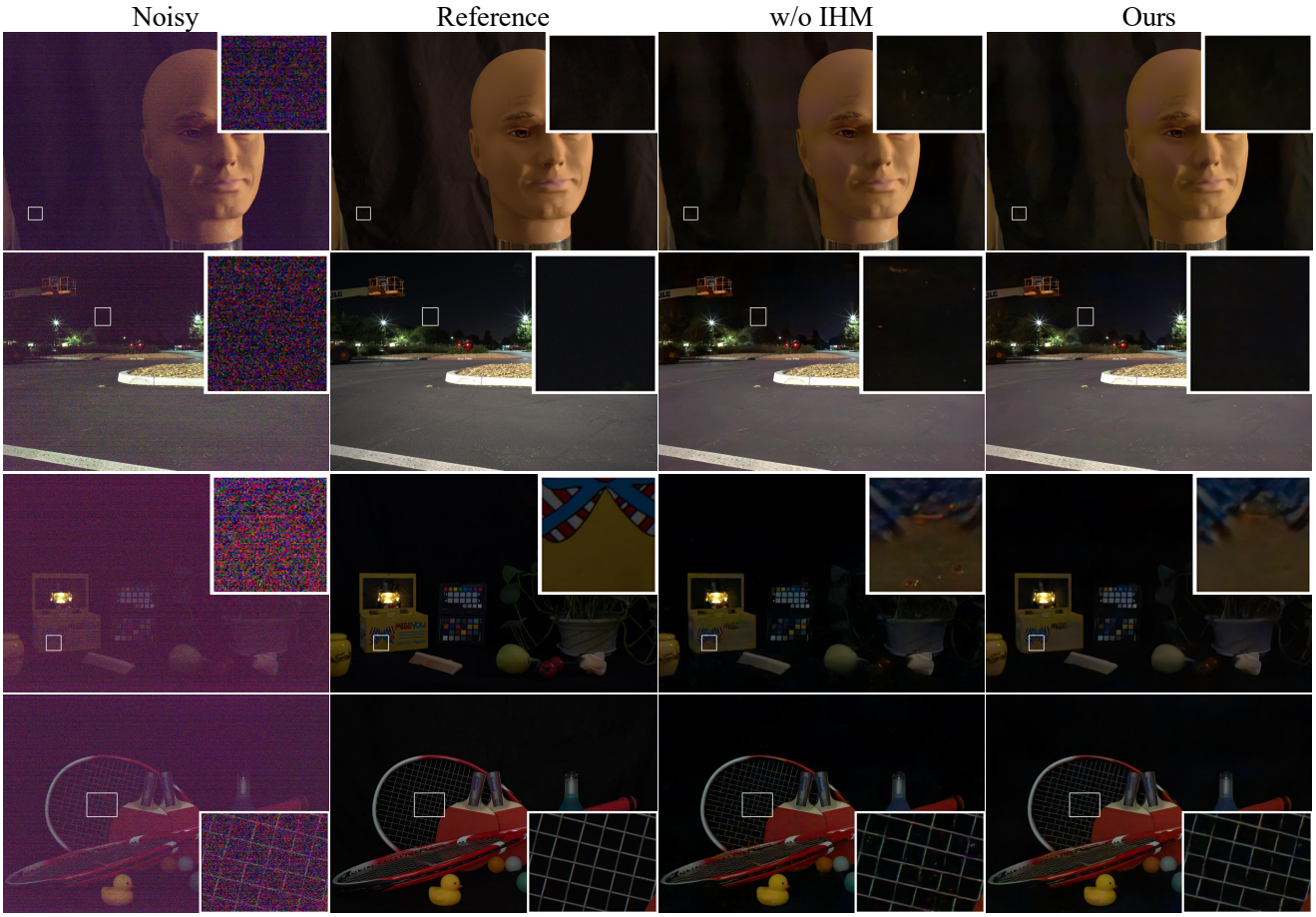

Figure 11. Ablation study on the iterative histogram matching (IHM). Without IHM, noticeable white spots appear in the denoised outputs due to saturated pixels in the noisy inputs. With our full method, this issue is effectively mitigated. The first two examples are from the SID test set while the third and fourth are from the ELD test set. Best viewed zoomed in.

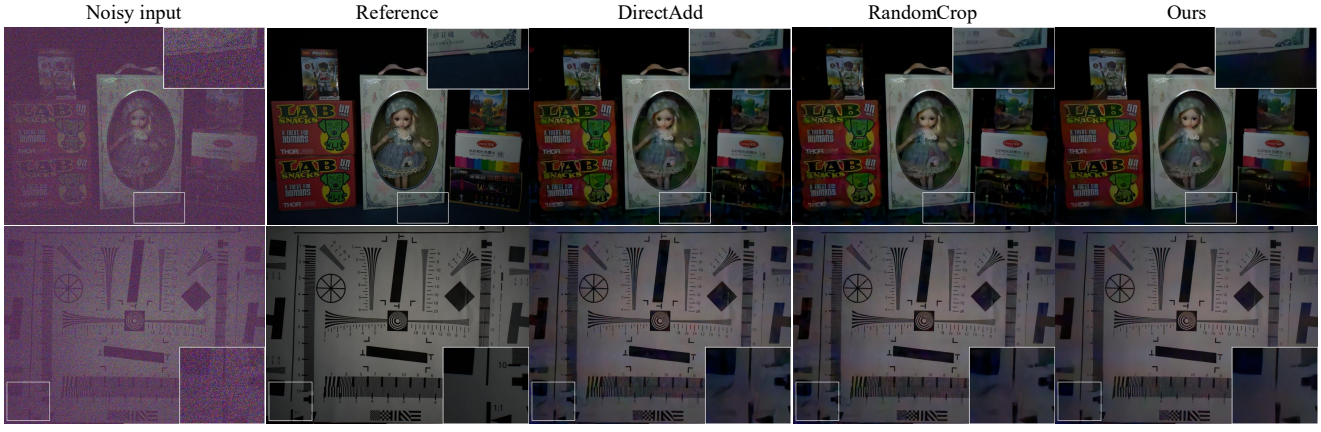

Figure 12. Denoising results of the network trained with different data sources synthesized using only one real dark frame per ISO setting. Our method produces cleaner results with fewer artifacts.

Table 6. Influence of dark shading correction (DSC) on denoising performance on the SID (top) and ELD (bottom) test sets.

Ratio	(a) RealDF+EDS	(b) RealDF+ADS	(c) RealDF+w/oDS	(d) SynDF+EDS	(e) SynDF+ADS	(f) SynDF+w/oDS
×100	43.74 / 0.961	43.75 / 0.962	43.06 / 0.957	43.72 / 0.961	43.72 / 0.961	43.16 / 0.957
×250	41.31 / 0.947	41.40 / 0.948	40.38 / 0.941	41.30 / 0.944	41.40 / 0.946	40.29 / 0.939
×300	37.89 / 0.930	37.96 / 0.931	37.28 / 0.922	37.86 / 0.929	37.95 / 0.930	37.21 / 0.921
×100	47.10 / 0.985	47.12 / 0.986	45.74 / 0.976	47.14 / 0.986	47.17 / 0.986	45.78 / 0.978
×200	44.68 / 0.965	44.75 / 0.967	42.50 / 0.925	44.78 / 0.966	44.86 / 0.968	43.41 / 0.949

Table 7. Influence of temperature mismatch between dark frames and noisy images on denoising performance on the LRID outdoor test set.

Ratio	hot I_n / hot I_{dark}	hot I_n / normal I_{dark}	normal I_n / normal I_{dark}	normal I_n / hot I_{dark}
×64	45.95 / 0.987	45.81 / 0.986	46.49 / 0.989	45.83 / 0.977
×128	44.18 / 0.975	44.15 / 0.976	45.05 / 0.983	43.29 / 0.936
×256	41.43 / 0.932	41.89 / 0.955	43.36 / 0.971	40.46 / 0.870

8.8. Influence of Temperature Mismatch Between Dark Frames and Noisy Images

We have acknowledged in the limitation section that some signal-independent noise components are influenced by factors beyond ISO, such as sensor temperature. To the best of our knowledge, this issue has not been analyzed in prior noise synthesis works. Here we provide a preliminary study of temperature mismatch on the LRID dataset [12], which provides two types of dark frames: one captured under normal temperature and one under hot sensor mode.

In Table 7, we analyze the temperature mismatch between the reference dark frame I_{dark} and the test noisy images I_n, where *hot* and *normal* denote different sensor temperatures. When the temperatures match, denoising performance remains stable, whereas a mismatch deteriorates performance. However, our method remains practical: a single dark frame per representative temperature can serve as an anchor from which additional dark frames are synthesized, substantially reducing the effort compared to collecting large numbers of dark frames for every setting.

9. Computational Cost

Our method is quite practical and does not involve complex deep networks. In Table 8, we report the computational cost of our noise synthesis method for different noise image resolutions. We evaluate our method on an NVIDIA A100-SXM4 GPU and an AMD EPYC 7543 CPU (32 cores, 64 threads).

Table 8. Computational cost of our noise synthesis method.

Resolution	Runtime (ms)	FLOPs (G)	Peak GPU memory (G)
$4 \times 512 \times 512$	20.45	0.03	0.42
$4 \times 1024 \times 1024$	39.43	0.13	0.71
$4 \times 2048 \times 2048$	122.84	0.54	1.87

10. Synthetic Noise Visualization Across Different Sensors

In this section, we visualize examples of real dark frames from various sensors alongside our synthesized dark frames. Figures 13 and 14 show examples from the Sony A7S2 sensor and the IMX686 sensor in the Redmi K30 smartphone, respectively. We also captured dark frames using a Fujifilm X-M5 camera, whose RAW images follow the X-Trans pattern instead of the Bayer pattern. Fig. 15 presents the real dark frames and our synthetic results. It can be observed that our method successfully generates realistic dark frames across different sensors, demonstrating its generalizability.

11. More Visual Comparisons

In this section, we provide more visual comparisons of denoising results from different denoisers trained on data synthesized using different methods. Fig. 16, Fig. 17 and Fig. 18 are examples from the SID test set. Fig. 19 are examples from the ELD test set. Fig. 20 and Fig. 21 are examples from the LRID test set.

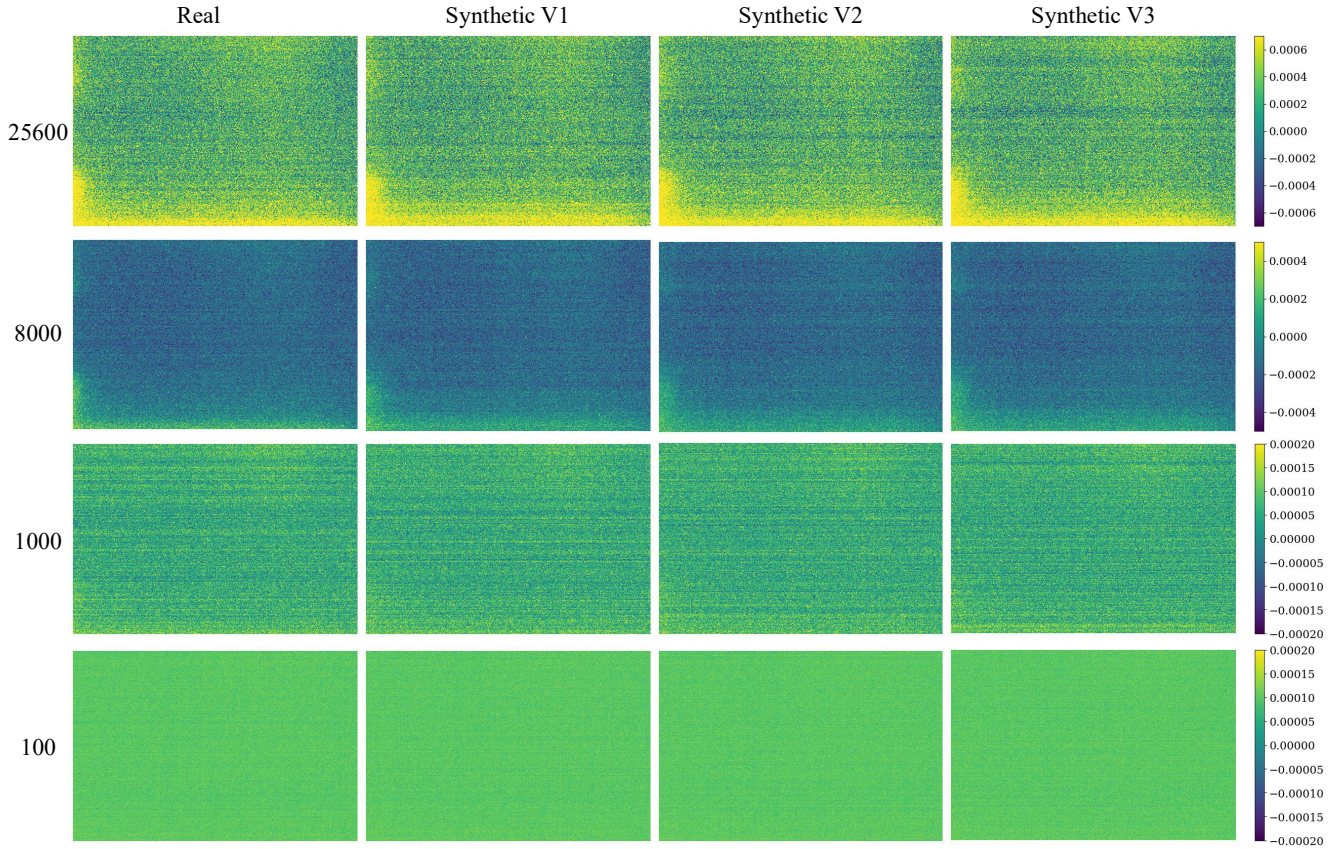

Figure 13. Real dark frames and three different synthetic dark frame realizations (generated with different random seeds) across various ISO settings (ISO=25600, 8000, 1000, 100) of the Sony A7S2 sensor.

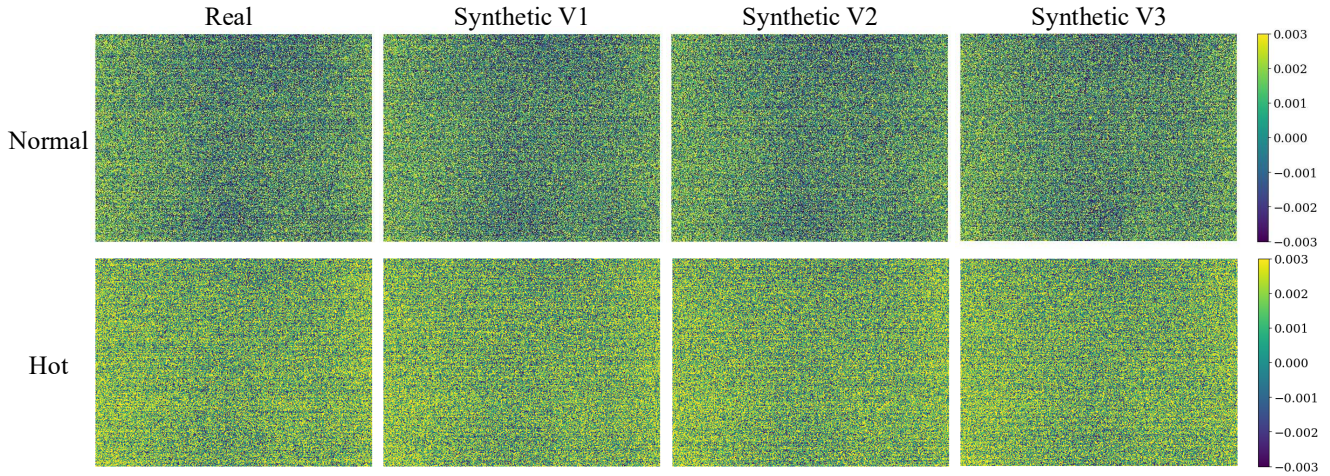

Figure 14. Real dark frames and three different synthetic dark frame realizations (generated with different random seeds) with ISO=6400 of the IMX686 sensor. The first example is captured under normal sensor temperature and the second in the sensor's hot mode.

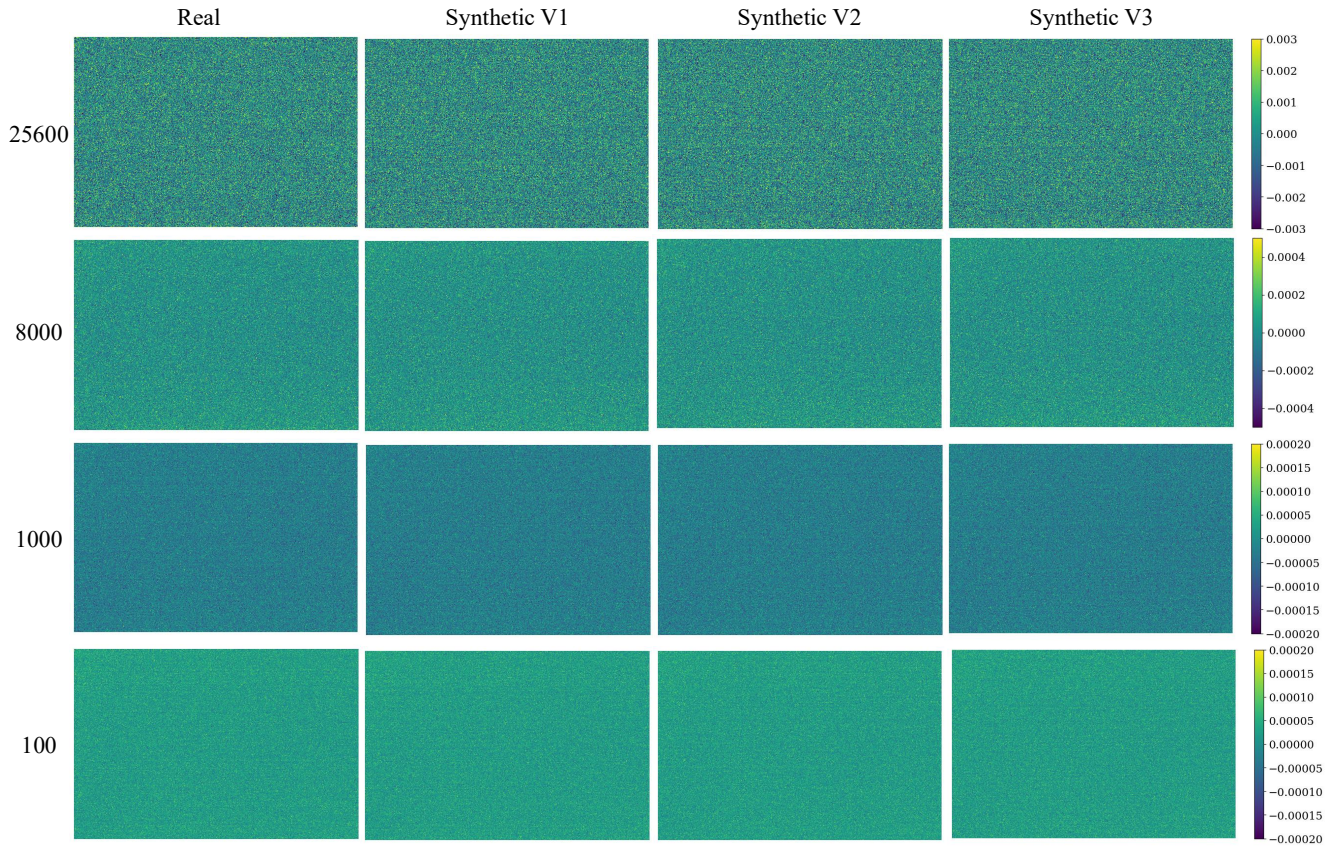

Figure 15. Real dark frames and three different synthetic dark frame realizations (generated with different random seeds) across various ISO settings (ISO=25600, 8000, 1000, 100) of the Fujifilm X-M5 sensor.

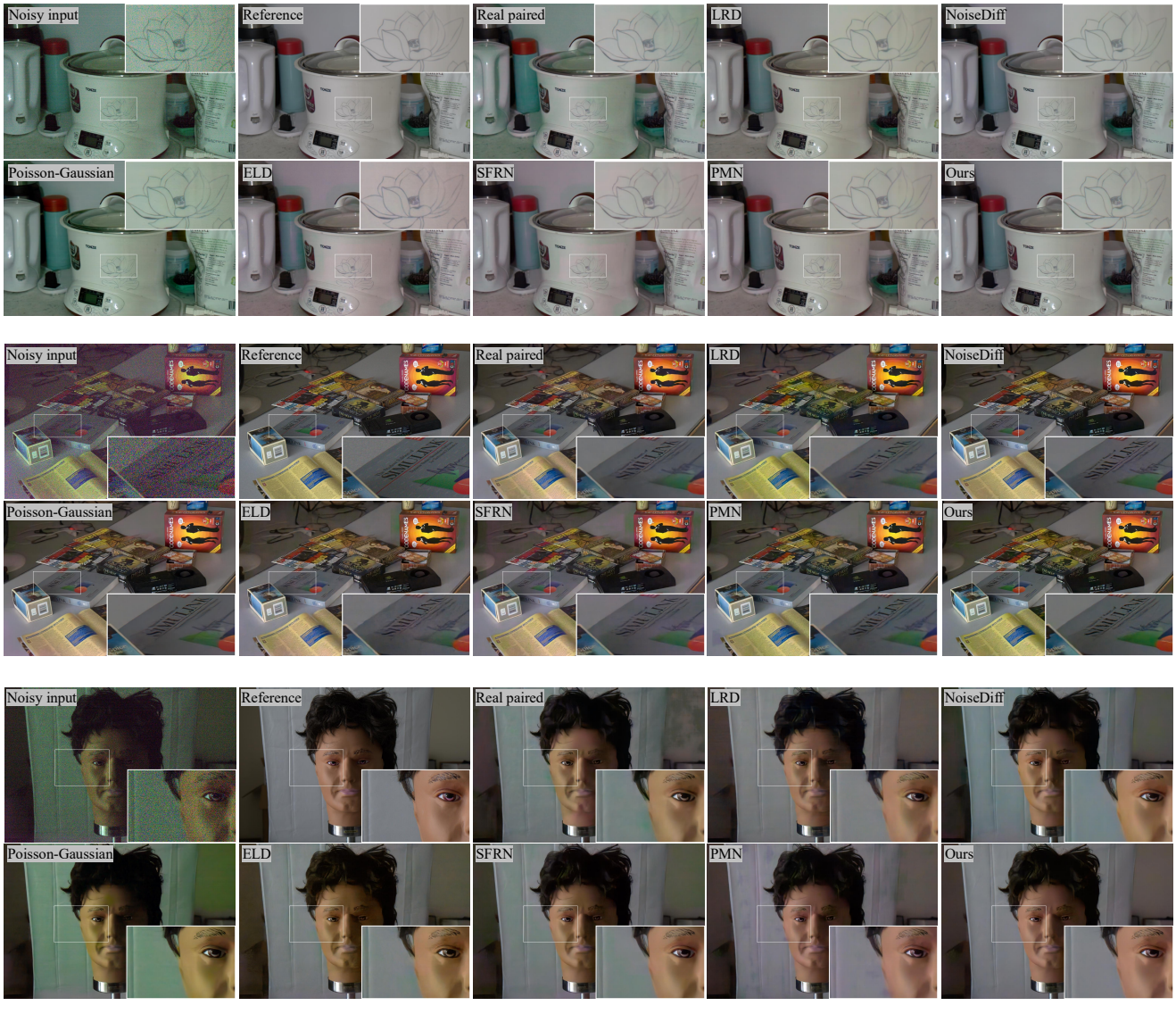

Figure 16. Result comparison of denoisers trained on data synthesized using different methods on the SID test set.

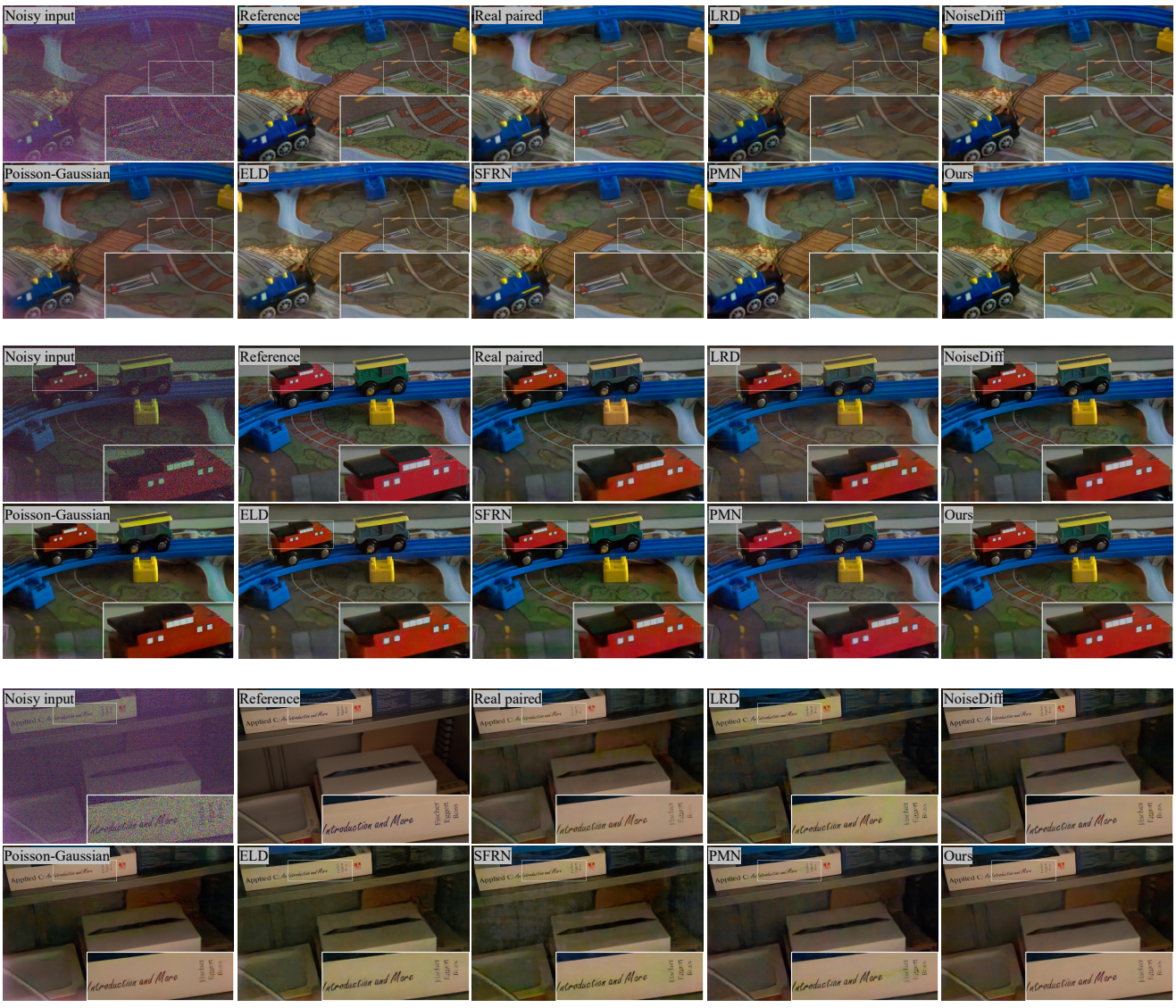

Figure 17. Result comparison of denoisers trained on data synthesized using different methods on the SID test set.

Figure 18. Result comparison of denoisers trained on data synthesized using different methods on the SID test set.

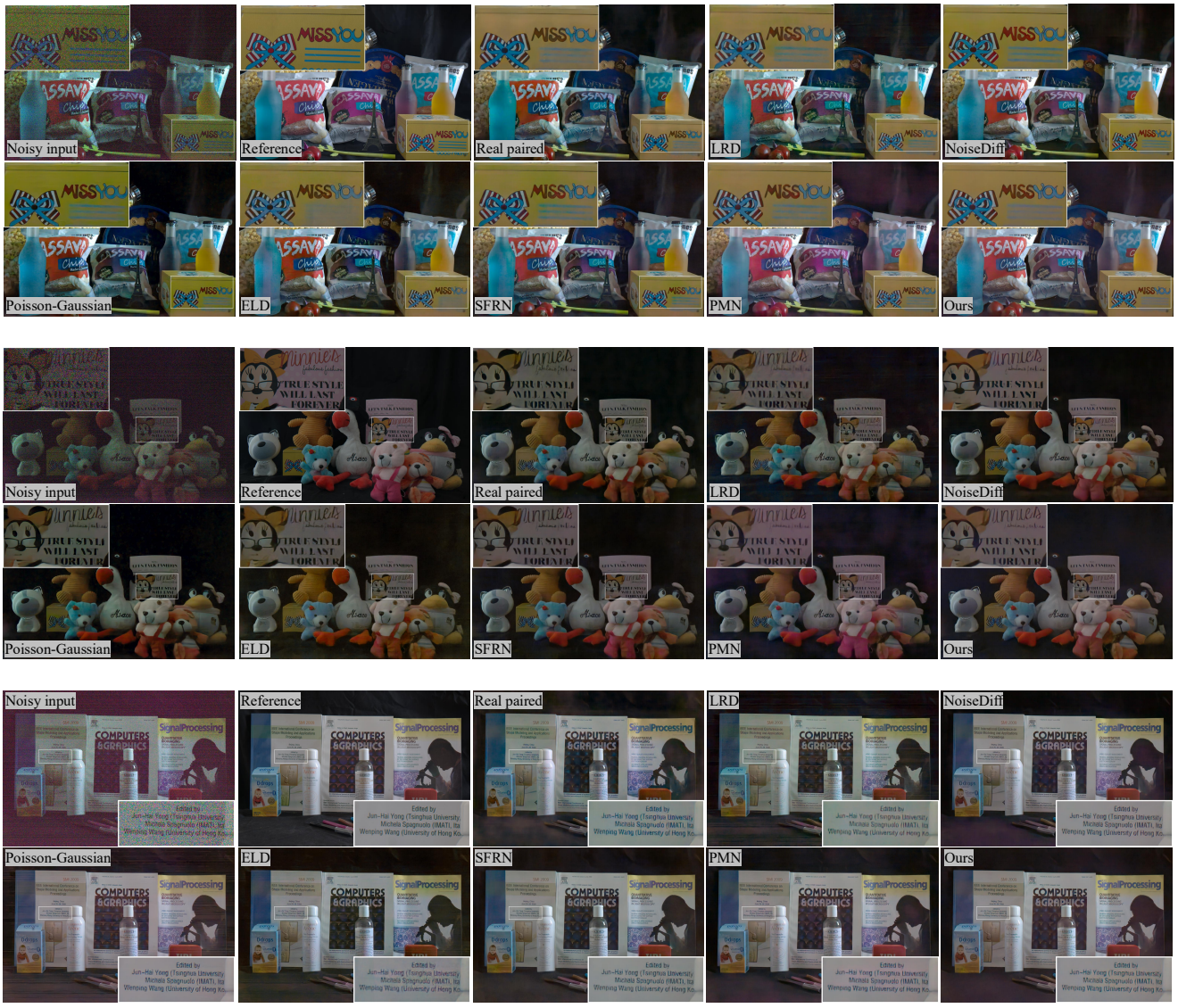

Figure 19. Result comparison of denoisers trained on data synthesized using different methods on the ELD test set. The original images are too dim to clearly show details, so the images presented here were further processed using gamma correction ($\gamma = 1.4$) for improved visibility.

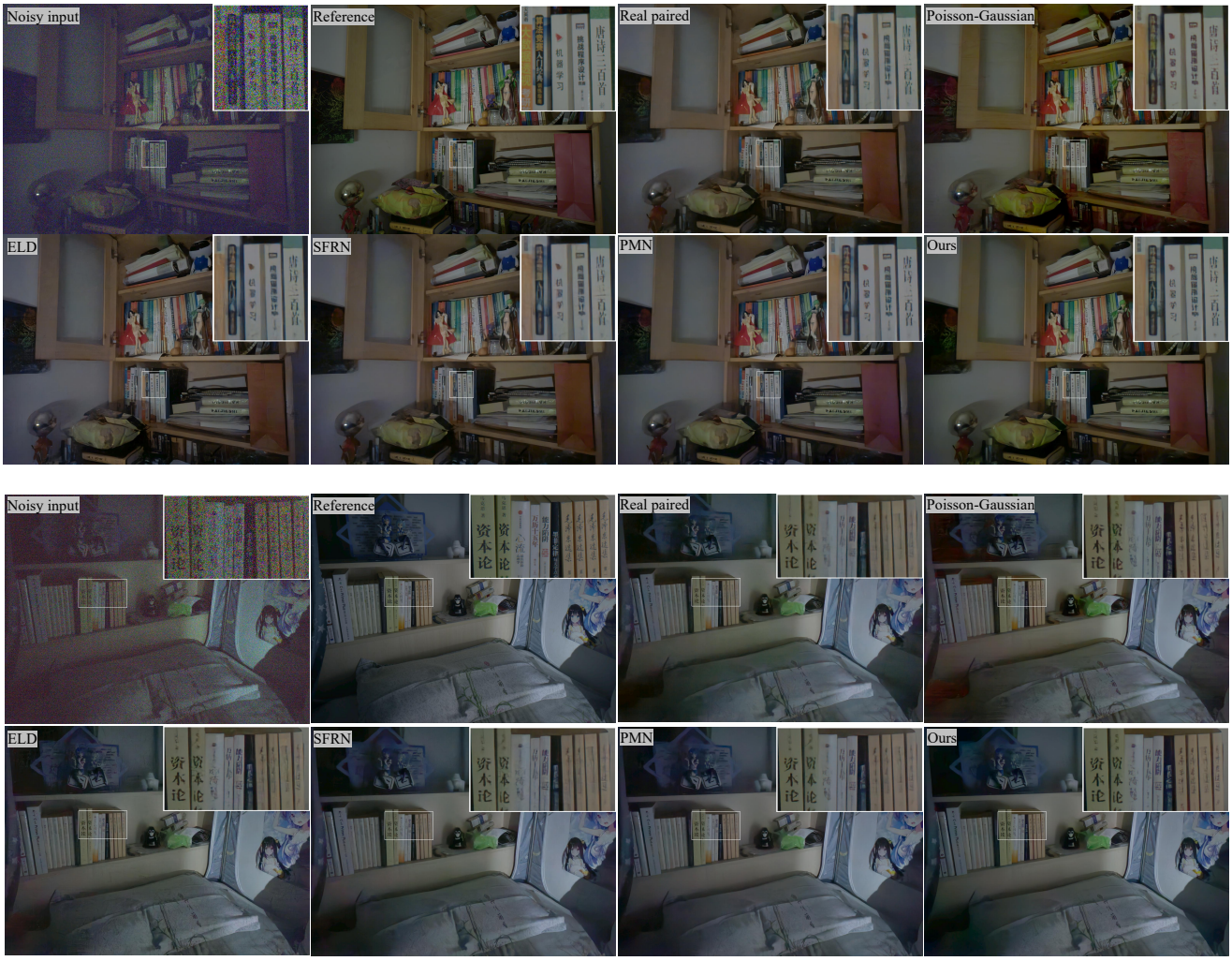

Figure 20. Result comparison of denoisers trained on data synthesized using different methods on the LRID test set. The original images are too dim to clearly show details, so the images presented here were further processed using gamma correction ($\gamma = 1.4$) for improved visibility.

Figure 21. Result comparison of denoisers trained on data synthesized using different methods on the LRID test set. The original images are too dim to clearly show details, so the images presented here were further processed using gamma correction ($\gamma = 1.4$) for improved visibility.