Noise2NoiseFlow: Realistic Camera Noise Modeling without Clean Images —Supplemental Material—

Ali Maleky[1,3,*], Shayan Kousha[1,3,*], Michael S. Brown[3], Marcus A. Brubaker[1,2,3]
[1]York University [2]Vector Institute [3]Samsung AI Center–Toronto

1. Training details

In this section, we give more details about the training procedure. As mentioned in the main paper, we used Adam [3] as optimizer in all of our experiments. We pre-trained the denoiser with N2N loss (Eq. 5 of the main paper) for 2,000 epochs. Also note that the denoiser pre-training step was used only to boost training under different setups, and is not a vital part of the overall training. Training the original Noise2NoiseFlow model from scratch will also produce almost the same results (NLL: -3.498, D_{KL}: 0.0275, PSNR: 52.65).

The supervised DnCNN was trained with MSE using the clean/noisy pairs from SIDD-Medium. Both denoiser pretraining and supervised training used an initial learning rate of 10^{-3}, which was decayed to 10^{-4} at epoch 30, and 5×10^{-5} at epoch 60. We used orthogonal weight initialization [2] for the denoiser architectures and the exact same initial weights for the noise model as used in the Noise Flow paper.

The denoiser was a 9 layer DnCNN and was the same in all experiments except where noted. Noise Flow was re-implemented in PyTorch [4] and carefully tested for consistency against the original implementation. Joint training used a constant learning rate of 10^{-4} for 2,000 epochs though no improvements were generally observed after $\sim$ 600 epochs.

2. Synthetic Noise Experiment

In order to demonstrate that our framework can retrieve the parameters of a supervised trained noise model, we have conducted a synthetic noise experiment. In this setting, we first trained a heteroscedastic Gaussian noise model, which was implemented as a flow layer in Noise Flow. For simplicity, we only took one camera and one ISO setting—namely, iPhone 7 and 800 as ISO level as we had adequate image data for training and evaluation. Under the mentioned setting, the model only has two trainable parameters—namely, β_1 and β_2. We then use this trained model to synthesize noisy image pairs for training a subsequent Noise2NoiseFlow model from scratch with only a heteroscedastic Gaussian layer as its noise model and DnCNN as its denoiser. The results shown in Figure 1 shows that our model can successfully retrieve the parameters of a trained NLF model.

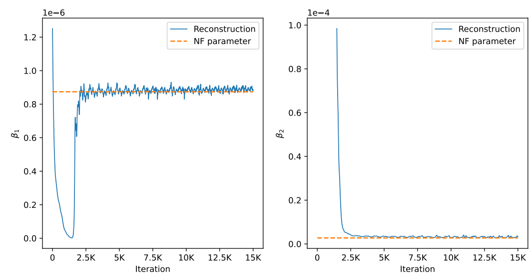

Figure 1. Convergence curve of the two parameters (β_1 and β_2 of the NLF model for a specific camera sensor and ISO level. *NF Parameter* corresponds to the parameters learned by a supervised Noise Flow model and *Reconstruction* Corresponds to the NLF parameters learned by a Noise2NoiseFlow model from synthetic data generated by the supervised Noise Flow model. As evidenced by the figures, the model can successfully retrieve the parameters.

3. Failure Cases

Although no significant unrealistic behaviour was noticed, we visualize 5 noise samples with the worst D_{KL} for Noise2NoiseFlow in Figure 2. While the noise samples are not in the best alignment with the real samples, the generated noise patches do not look very unnatural.

References

[1] A. Abdelhamed, M. A. Brubaker, and M. S. Brown, "Noise flow: Noise modeling with conditional normalizing flows," in *ICCV*, 2019. 2

[2] W. Hu, L. Xiao, and J. Pennington, "Provable benefit of orthogonal initialization in optimizing deep linear networks," *arXiv preprint arXiv:2001.05992*, 2020. 1

*Work performed while interns at the Samsung AI Center–Toronto.

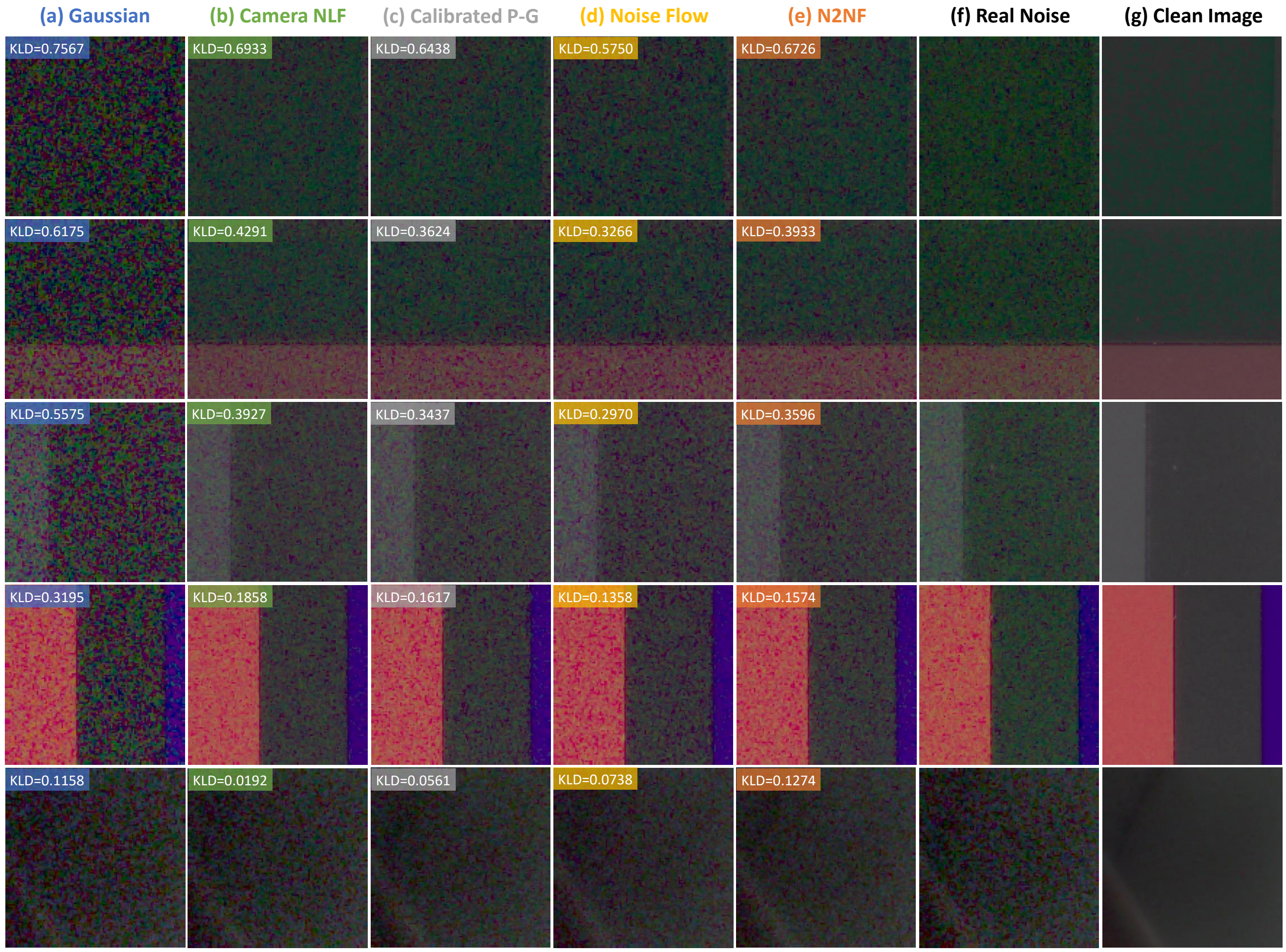

Figure 2. Noise synthesis samples from (a) the AWGN model, (b) Camera NLF, (c) Calibrated P-G [5], (d) Noise Flow [1], and our proposed method, Noise2NoiseFlow, compared to the (f) real noise in SIDD for patches where Noise2NoiseFlow has the worst D_{KL} numbers.

[3] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *ICLR*, 2015. 1

[4] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019. 1

[5] Y. Zhang, H. Qin, X. Wang, and H. Li, "Rethinking noise synthesis and modeling in raw denoising," in *ICCV*, 2021. 2