IDR: Self-Supervised Image Denoising via Iterative Data Refinement Supplementary Material

Yi Zhang[1] Dasong Li[1] Ka Lung Law[2] Xiaogang Wang[1]
Hongwei Qin[2] Hongsheng Li[1]
[1]The Chinese University of Hong Kong [2]SenseTime Research
zhangyi@link.cuhk.edu.hk

Table 1. Denoising performance (PSNR/SSIM) on binomial (**B**) and impulse noise (**I**). The experiment on Kodak and BSDS300 dataset shows consistent results. Best and second best results are **highlighted** and underlined.

Dataset	Noise	N2N [2]	Supervised	Ours
Kodak	**B**	31.48 / 0.939	**31.84 / 0.946**	31.63 / 0.944
	I	**36.04 / 0.975**	35.83 / 0.978	34.92 / 0.977
BSDS300	**B**	31.50 / 0.930	**32.32 / 0.938**	32.18 / 0.936
	I	36.55 / 0.976	**37.48 / 0.980**	37.40 / 0.979

A. Experiments on other synthetic noises

Binomial and impulse noise. We experiment with another two point-wise noise types, binomial noise and impulse noise following the setup of N2N [2]. They can be used to model bad pixels and hot pixels in raw images, respectively. Binomial noise is constructed from the pixel-wise production of a clean image with a random 0-1 mask. It uses a one-channel mask to set some pixels to zeros with probability p and retain other pixels' values. For impulse noise, it randomly sets some pixel channels to 0 or 1 with probability p while keeping the other colors. Some examples are shown in Fig. 2. During training, we uniformly set $p \in [0, 0.95]$ for both binomial noise and impulse noise to train models on a range of noise levels. During inference, we follow the setting of N2N [2] and fix $p = 0.5$. Since they are not zero-mean noises, we have to use different losses for training as recommended in N2N.

The results are shown in Table Fig. 2.Similar to the results on Gaussian noise, our method still outperforms N2N in most cases. N2N produces better results on binomial denoising in the Kodak dataset, which shows 0.2 dB improvement than supervised learning. But it cannot show stable performance when testing on the larger BSDS300 dataset.

Correlated noise. Most previous methods assume that the noise is pixel-wise independent so that the spatially correlated noise is less explored. Our method only requires the noise model no matter whether it is pixel-wise independent or not. We compare our method with the recently refined BM3D [4] on several typical correlated noises. The spatially correlated noise can be created as follows:

$$x = y + v \otimes g, \quad v \sim \mathcal{N}(0, 1), \tag{1}$$

where x and y denote the noisy and clean images, v is the random noise generated from a normal distribution with zero mean and $\sigma = 1$, and g is the convolution kernel for creating the correlated noise. Following the refined BM3D [4], we use 5 types of kernels for comparison.

Table 2. The comparison between our method and the refined BM3D [4] on correlated noise. Best results are **highlighted**. (See the text for more details.)

Dataset	Kernel	BM3D [4]	Ours
Kodak	g_1	31.56 / 0.900	**37.46 / 0.976**
	g_2	29.20 / 0.798	**33.10 / 0.929**
	g_3	31.38 / 0.857	**41.88 / 0.993**
	g_4	27.23 / 0.742	**28.60 / 0.803**
	g_5	26.53 / 0.859	**32.97 / 0.973**
BSDS300	$g1$	26.77 / 0.841	**29.83 / 0.922**
	$g2$	27.84 / 0.787	**31.28 / 0.910**
	$g3$	29.59 / 0.838	**38.22 / 0.982**
	$g4$	26.14 / 0.722	**27.39 / 0.793**
	$g5$	26.31 / 0.870	**33.69 / 0.974**

Some qualitative results are shown in Fig. 4. More results and kernel visualization are provided in the supplementary materials. We also use BSDS300 as our sRGB benchmark and evaluate the default noise level $\sigma = 5$ for all correlated noise. The refined BM3D is noise-aware and our models are trained on $\sigma = 5$. In Table 2, we show the quantitative results on correlated noise. Our methods can deal with spatially correlated noise and outperform the refined BM3D by a large margin.

B. More ablation studies

The generality to Transformer and other CNNs. In the above experiments, we follow N2N [2] and only use U-

Table 3. The Gaussian denoising results (PSNR/SSIM) of our method when using different network architectures. All models are trained with the same training settings.

Models	$\sigma = 25$	$\sigma = 50$	FLOPS(G)
U-Net	31.84 / 0.875	28.71 / 0.787	6.0
SCN [1]	32.23 / 0.879	29.11 / 0.795	54.5
MPRNet [5]	32.59 / 0.887	29.51 / 0.808	300.6
Swin [3]	32.22 / 0.878	29.04 / 0.792	16.0

Table 4. The **training time** (hours) and **PSNR** comparison between the full version and fast version iterative data refinement. "IDR-n" denotes the testing results after the n-th iteration in the full version iterative data refinement (IDR). "Fast" denotes fast IDR.

Model	IDR-1	IDR-2	IDR-3	IDR-4	IDR-5	Fast
PSNR	24.84	28.01	31.35	31.48	31.49	31.50
Time	8.3	16.6	24.9	33.2	41.5	8.6

Net as our backbone. Here, to show the generality of our method, we adopt both Swin Transformer [3] and recent CNN-based architectures (SCN [1] and MPRNet [5]) in our iterative algorithm. To reduce the training time, we train those heavy models for 20k iterations and other settings are the same as our Gaussian denoising experiment. As shown in Table 3, using larger CNN-based denoising models in our IDR can always produce better results. Swin Transformer shows comparable results as the SCN but with much fewer flops. This indicates our method can well adapt to both CNN and transformer architectures, and it can be further improved by advanced architectures.

Comparison between IDR and fast IDR. In Table 4, we show the intermediate testing results and the corresponding training time cost of our full version IDR on Gaussian denoising. After about 5 full iterations, the full version method converges and shows quite similar performances as the fast IDR. But, due to the one-epoch training and the accumulative training strategy of the fast IDR, the training time is quite close to the full version algorithm.

C. More details of SenseNoise-500 dataset

To ensure that the collected pairs have the same brightness levels, the long-exposure and normal-exposure frames are captured with the same target sensitivity value. Specifically, for the noisy images (normal-exposure), all of them use the default exposure time and ISO recommended by the smartphone (Xiaomi MI9). And, the long-exposed frames are captured with the exposure time t_l starting from 1s and up to 2s, and they keep the same target sensitivity value S via a dynamic ISO ISO_l:

$$t_l = \max\left(\min\left(\frac{S}{100}, 2\right), 1\right), \quad \mathrm{ISO}_l = \frac{S}{t_l}, \tag{2}$$

where $S = \mathrm{ISO}_r \times t_r$ is the fixed target sensitivity value, ISO_r and t_r are recommended ISO and exposure time for noisy images, and $\frac{S}{100}$ is the estimated maximum exposure time and it would be clipped to be within $[1s, 2s]$. To demonstrate the quality of our dataset, we show the extreme indoor and outdoor examples in Fig. 1.

Object motion: We carefully avoid capturing dynamic scenes and light changing scenes and remove unsatisfied scenes when we construct the dataset. As for occasional slight object motions (pixel-level), they may cause some abnormal pixel values across the temporal dimension around the motion areas. But those outliers can be removed naturally since we use the median filter along the temporal dimension to create the ground truth.

OIS: We used Xiaomi Mi 9 with the IMX586 sensor to collect the dataset. The lens optical stabilization mode can be disabled via camera2 API, which is particularly helpful to capture long-exposure images.

The noise levels and diversity: We write an App to show the real-time exposure information on the screen. This helps us to cover a wide range of ISOs. We follow the advice to compare the SNR and the standard deviation (STD) of SNR. SenseNoise-500 SNR range: $[-12.94, 5.79]$ STD: 11.54. SIDD SNR range: $[-12.72, 4.57]$ STD: 11.05. This indicates that our dataset covers a wider range and keeps better diversity compared with the SIDD dataset.

GT quality: Since the scene contents and exposure strategy vary greatly for different datasets, we are unable to compare the SNR of their ground truth directly. The GT for most noisy scenes (ISO25600) can be founded in Fig. 1.

D. Performance gain from large-scale unlabelled data

We add an experiment to compare supervised baseline (UNet and following Table 4's settings) using limited data with the unsupervised training and large-scale unlabeled data on Gaussian noise (Tab. 5). All data are from ImageNet. The data size for supervised methods is set to 0.5k since existing denoising datasets generally contain hundreds of images. When using the same training scale (0.5k), our method produces slightly lower performance than the supervised method on Gaussian noise. Scaling up the unlabeled data size (50k) helps to improve the performance by $\sim$0.2dB over supervised training with limited data.

References

[1] Yuchen Fan et al. Scale-wise convolution for image restoration. In *AAAI*, 2020. 2

Table 5. Training with unlabelled data of different scales.

Sigma \ Data scale	Supervised		Unsupervised (Ours)		
	0.3k	0.5k	0.5k	5k	50k
$\sigma = 25$	31.60	31.70	31.65	31.82	31.84
$\sigma = 50$	28.53	28.60	28.61	28.70	28.71

[2] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 2971–2980. PMLR, 2018. 1

[3] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021. 2

[4] Ymir Mäkinen, Lucio Azzari, and Alessandro Foi. Exact transform-domain noise variance for collaborative filtering of stationary correlated noise. In *ICIP*, pages 185–189. IEEE, 2019. 1, 5

[5] Syed Waqas Zamir et al. Multi-stage progressive image restoration. In *CVPR*, 2021. 2

Figure 1. Examples from the SenseNoise-500 dataset. Even in the extreme low-light indoor (ISO 25600) and outdoor (ISO 24379) scenes, our ground truths are still of high-quality.

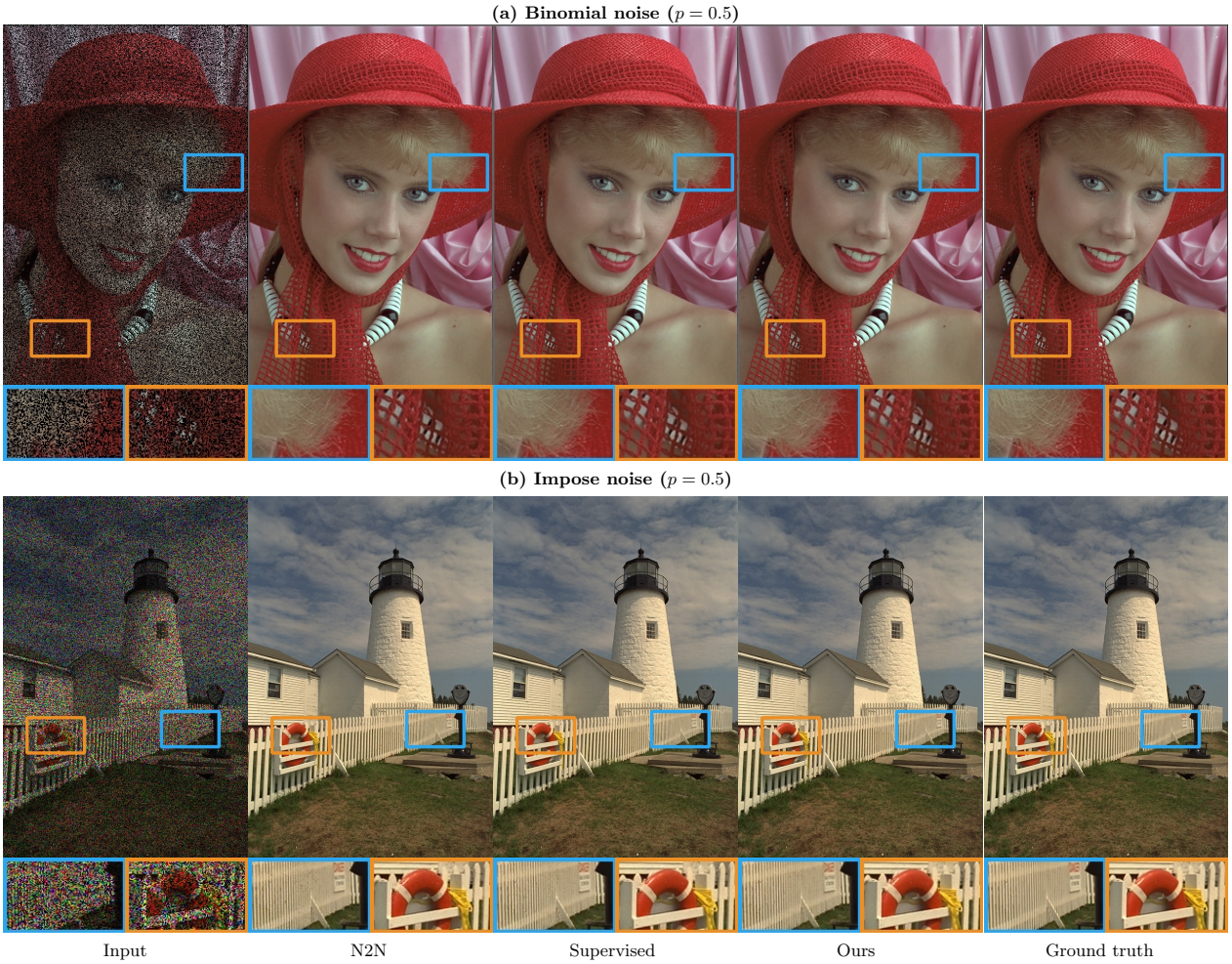

Figure 2. Visualization of removing binomial and impulse noise with the corrupted probability $p = 0.5$. While we only use individual noisy images for training, our qualitative results are very closed to supervised learning.

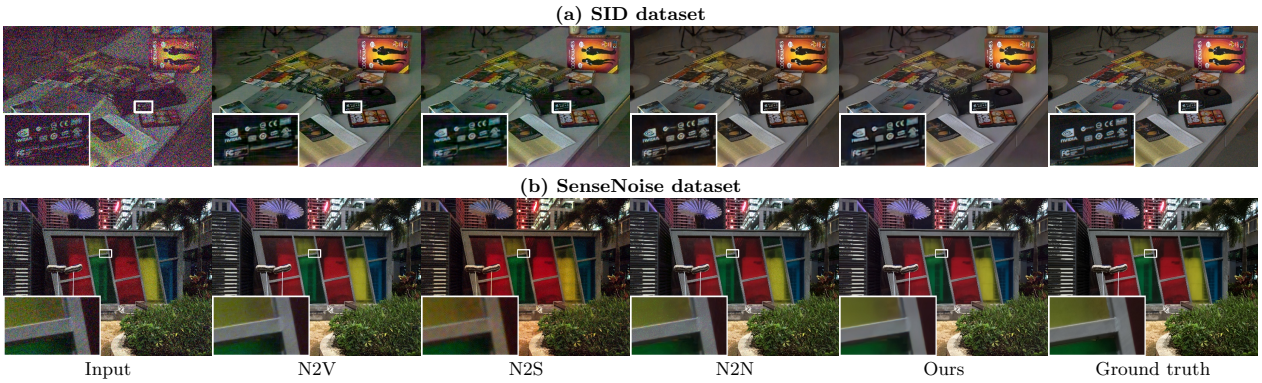

Figure 3. Qualitative results of raw image denoising on SID dataset and our SenseNoise-500 dataset.

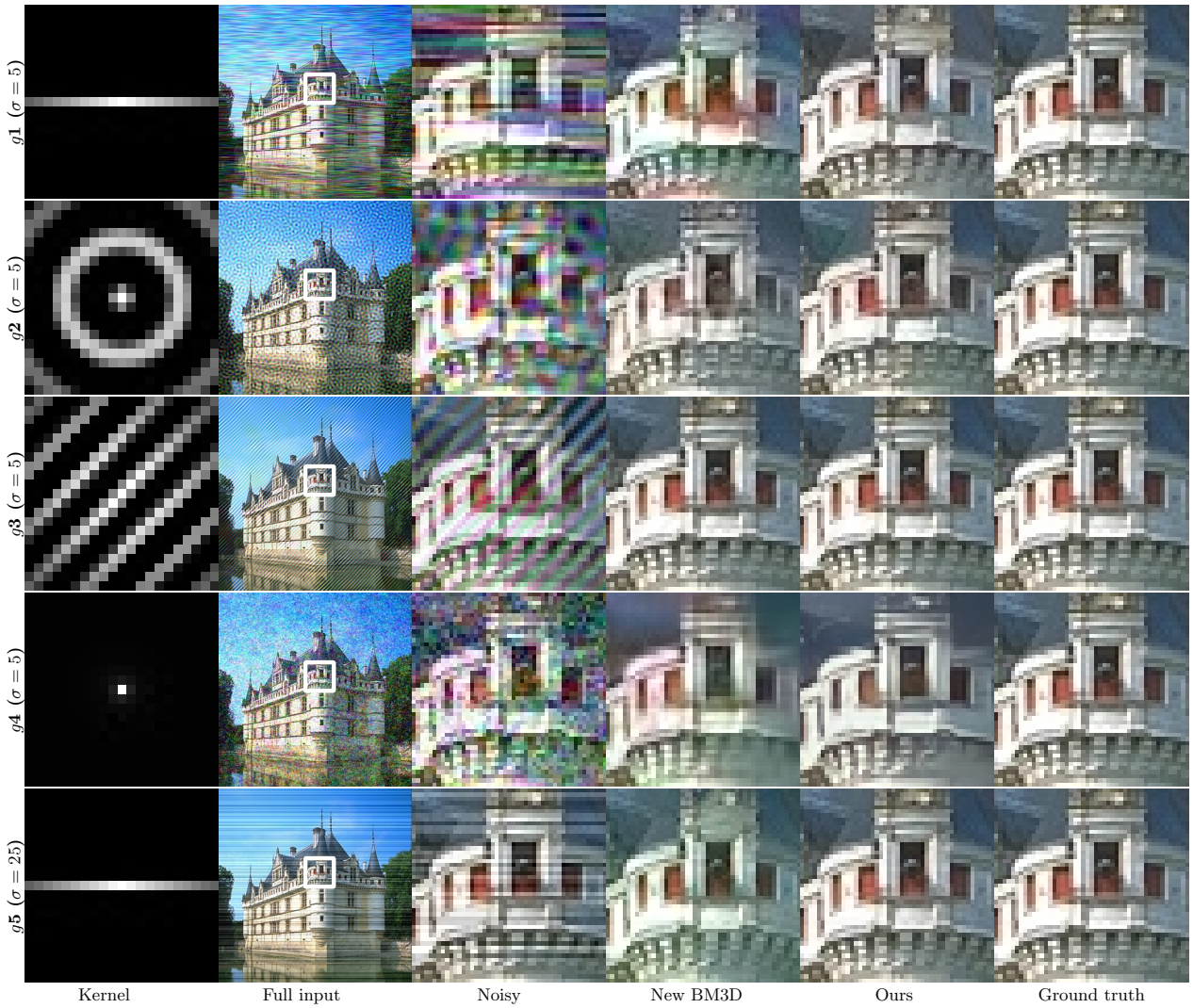

Figure 4. Visualization of five correlated kernels and denoising results on different noise levels. We compare our method with the new BM3D [4] designed for correlated noise.