IDR: Self-Supervised Image Denoising via Iterative Data Refinement

Yi Zhang¹  Dasong Li¹  Ka Lung Law²  Xiaogang Wang¹
Hongwei Qin²  Hongsheng Li¹
¹The Chinese University of Hong Kong  ²SenseTime Research
zhangyi@link.cuhk.edu.hk

Abstract

The lack of large-scale noisy-clean image pairs restricts supervised denoising methods’ deployment in actual applications. While existing unsupervised methods are able to learn image denoising without ground-truth clean images, they either show poor performance or work under impractical settings (e.g., paired noisy images). In this paper, we present a practical unsupervised image denoising method to achieve state-of-the-art denoising performance. Our method only requires single noisy images and a noise model, which is easily accessible in practical raw image denoising. It performs two steps iteratively: (1) Constructing a noisier-noisy dataset with random noise from the noise model; (2) training a model on the noisier-noisy dataset and using the trained model to refine noisy images to obtain the targets used in the next round. We further approximate our full iterative method with a fast algorithm for more efficient training while keeping its original high performance. Experiments on real-world, synthetic, and correlated noise show that our proposed unsupervised denoising approach has superior performances over existing unsupervised methods and competitive performance with supervised methods. In addition, we argue that existing denoising datasets are of low quality and contain only a small number of scenes. To evaluate raw image denoising performance in real-world applications, we build a high-quality raw image dataset SenseNoise-500 that contains 500 real-life scenes. The dataset can serve as a strong benchmark for better evaluating raw image denoising. Code and dataset will be released¹

1. Introduction

In recent years, deep neural networks have made great progress in image denoising. The mainstream of the advances is powered by supervised learning-based approaches [6, 11, 13, 22, 29]. However, large-scale datasets with high-quality ground-truth are still unavailable due to high human and time costs, which limit the learning-based approaches’ applications in real-world scenarios.

A few unsupervised learning-based approaches have been proposed to learn image denoising without ground truth. However, existing unsupervised denoising methods either show poor performance or work under impractical settings. One research direction focuses on unsupervised learning of denoising from single noisy images [5, 14, 16, 28, 31]. Although such training data are convenient to collect, learning from single noisy images shows relatively poor performance since the task is highly ill-posed. Another research direction focuses on learning from paired noisy images [18]. It utilizes two perfectly aligned images of the same scene as the training pairs. Such a method significantly improves the denoising performance and can sometimes even outperform supervised learning-based methods. However, multiple perfectly aligned noisy images of the same scene are sometimes challenging to capture in practice, especially for moving scenes. One recent approach applies noise models to add synthetic noise on the already noisy images to create noiser-noisy image pairs for training a denoising network to partially capture the noise prior [23]. However, such synthetic noiser-noisy training pairs’ distributions cannot well fit that of the actual noisy-
clean image pairs (we call such train-inference distribution mismatch the data bias between the two datasets), which limits the method’s denoising performance on actual noisy images. There are also methods [26,32] that train a separate network for denoising each image separately and show better results. However, such methods inevitably have higher computational costs.

In this paper, we propose an unsupervised denoising method that adopts a practical setting for preparing training data and achieves state-of-the-art performances. It only requires single noisy images and a noise model for training a denoising neural network. Note that the noise model is convenient to obtain in actual raw image denoising (e.g., noise profile in DNG files [15]), and can be both point-wise and correlated types (e.g., raw image noise, stripe noise, multiplicative noise).

To start with, we add noise from the known noise model to actual noisy images to create noisier-noisy dataset, which treats the created noisier images as inputs and actual noisy images as learning targets. Then, we conduct a study on learning denoising from such noisier-noisy datasets to empirically validate two findings: (1) the denoising model trained on the noisier-noisy dataset has the generalization capability of denoising actual noisy images; (2) reducing the mismatch (data bias) between the created noisier-noisy dataset and the ideal noisy-clean dataset results in a better denoising model to process the actual noisy images.

The two findings inspire us to propose the iterative data refinement (IDR) scheme, which iteratively reduces the data bias between the created noisier-noisy dataset and the ideal noisy-clean dataset to train better denoising models. In particular, we first train a denoising model based on the noisier-noisy dataset. According to our finding 1, the network trained on the noisier-noisy dataset can be generalized to denoise the actual noisy images. The denoised images together with the noise model can then be used to create a new intermediate dataset for a new round of training, which regards the denoised images from the previous round as the targets. As the data bias between the noisier-noisy and the ideal noisy-clean dataset is reduced (based on our finding 2), we can train a better denoising model on the newly constructed intermediate dataset in the next round. This newly trained model can denoise the original noisy images again, based on which, better learning targets can be created for the next round. This process can be performed iteratively to gradually train better denoising models. To alleviate the heavy training time cost of this iterative method, we further propose a fast version of our IDR by reducing the training epochs in each data refinement iteration.

To evaluate the denoising models’ performances in real-world raw-image denoising, we collect a low-light dataset SenseNoise-500, which contains 500 diverse scenes with perfectly aligned and high-quality ground truth. Our method outperforms existing unsupervised methods on both real-world and synthetic noise denoising tasks with a wide range of noise types and noise levels. Other than pointwise noises, we also show our algorithm’s denoising capability on typical correlated noises. All results demonstrate the effectiveness and superiority of our proposed method.

Our contributions are summarized as follows:

- A self-supervised image denoising scheme, iterative data refinement (IDR), is proposed. It works under a more practical setting that only requires single noisy images and a noise model.
- We also propose a fast approximation of our full method IDR, which costs the same training time as training an ordinary denoising model while maintaining the similar performance as our full-version model.
- We collect a high-quality raw image denoising dataset for training denoising models and test their performance in real-world scenarios. It contains 500 diverse scenes, with images being well-aligned and the clean images are created with > 60s exposure.
- Our method outperforms existing unsupervised methods on a wide range of noise types and noise levels. In some cases, it even shows comparable performance with fully supervised methods.

2. Related Work

Image denoising has been extensively studied in the literature. In this review, we divide them into three categories: Non-learning denoising methods, learning from clean data, and learning from noisy data. In the end, we also review some popular real-world denoising benchmarks which are widely used in other papers.

Non-learning denoising methods. Representative non-learning methods usually are based on different assumptions. This category of methods include anisotropic diffusion, total variation denoising [27], wavelet domain denoising [25], sparse coding [20], etc. Another important category of methods is driven by image self-similarity. Classical methods such as NLM [7,8], BM3D [12,21] utilize similar patches to remove the noise.

Learning from clean data. Learning from clean data is the mainstream in image denoising. Many works [1,3,11] have explored learning from natural noisy and clean image pairs directly. While the ideal training pairs can be used, the amount of those pairs is quite limited. Another way is to use external clean images and create synthetic noisy images based on the noise model [13,22,35,36]. As a result, there exists a natural domain gap between the real dataset and the synthetic dataset. To reduce the domain gap between synthetic images and natural noisy images, some progress [4,6,33] has been made to process the external data to the target domain so that the external data becomes similar to the testing data. Other works [2,10,13,30,37] focus on building a more accurate noise model for the real-word
noise. Generating more realistic noise for training helps the model generalize better in real applications.

**Learning from noisy data.** One line in unsupervised learning is single-image-based [26,28,32]. They require retraining models for different images, which usually is not practical. As reported in self2self [26], it costs around 1.2 GPU (RTX 2080Ti) hours to retrain and test for a 256 × 256 image. On the other hand, batch-based unsupervised approaches start from Noise2Noise (N2N) [18], which demonstrates how to learn image denoising with noisy image pairs. But noisy image pairs generally are not available in the real world and N2N also requires designing different loss functions according to the noise type. The followed-up works try to learn image denoising by using individual noisy images. Mask-based unsupervised approaches [5,16,17] design different mask schemas to blind pixels in the input noisy images. Then, they train the network to predict the masked pixels according to noisy pixels in the input receptive field. More recent work DBSN [31] uses unpaired clean images to improve unsupervised denoising results through a two-stage scheme. Noiser2noise [23] introduces the noise model into unsupervised learning and trains the network directly for the single noise level denoising. So there still exists the dataset bias.

**Real-world denoising datasets.** Several datasets have been proposed in the literature. Darmstadt Noise Dataset (DND) [24] collects 50 image pairs which can only be used for testing. And most of the images in DND have relatively low levels of noise. Smartphone Image Denoising Dataset (SIDD) [3] was proposed to evaluate denoising on smartphones. It captures multiple times for each scene and contains 10 indoor scenes, but occasionally exhibits obvious misalignments. SID dataset [11] was captured by DSLRs with the under-exposure strategy. It uses a single long-exposure frame as the ground truth.

### 3. Method

In this section, we first introduce the creation of noisier-noisy dataset, based on which, we conduct a study on learning denoising with the noisier-noisy dataset and empirically validate two findings. We then present our self-supervised denoising framework: iterative data refinement (IDR). To accelerate the training process of this iterative method, we further introduce a fast version of IDR to reduce the training time while leading to similar performance as the full version of our IDR.

#### 3.1. Pilot Study on Data Bias of Learning Denoising

For training denoising models with a known noise model, a typical type of approaches [35] is to create synthetic noisy images via adding noise \( n \) onto the ground-truth clean images \( y \), which is denoted as \( y + n \) and used as network input. We name such synthetic training data noisy-clean dataset for learning the mapping \( (y + n) \rightarrow y \). However, the ground-truth clean images \( y \) are generally difficult to obtain. Another plausible way is to add noise \( n \) onto the actual noisy images \( x \) to create the even noisier images \( x + n \), where the later noisier \( x + n \) are treated as inputs and the former noisy images \( x \) are used as learning targets. We call such training data noisier-noisy dataset. Although the later dataset is much easier to obtain, there is inevitable data bias between the above two types of datasets and the denoising network trained on the later noisier-noisy dataset cannot well handle the actual noisy images \( x \). We conduct a pilot study as follows and empirically validate two important findings. Formally, the datasets are denoted as

\[
\text{Noisy-clean Dataset: } \{(y_i + n_i, y_i)\}_{i=1}^N, \\
\text{Noisier-noisy Dataset: } \{(x_i + n_i, x_i)\}_{i=1}^N, \quad (1)
\]

where \( x_i = y_i + n_i \).

Here, \( y_i \) is a clean image (which is very difficult to obtain in practice), \( x_i = y_i + n_i \) is its corresponding noisy image, \( n_i \) is the random noise generated from the noise model, and \( y_i + n_i^2 \) denotes applying the sampled noise \( n_i \) to the actual noisy image \( y_i \) to create a synthetic but even noisier image. For simplicity, we use \( \{y_i + n_i, y_i\} \) and \( \{x_i + n_i, x_i\} \) to denote \( \{(y_i + n_i, y_i)\}_{i=0}^N \) and \( \{(x_i + n_i, x_i)\}_{i=0}^N \), respectively.

To analyze the data bias of training a network with noisier-noisy datasets, we train a U-Net from scratch on both the noisy-clean and noisier-noisy datasets\(^3\) and investigate three representative noise types, raw image, Gaussian, and correlated noise, with various noise levels. They cover common noise types i.e., signal-dependent/signal-independent noises and point-wise/correlated noises. More setups of the noise models can be found in the caption of Fig. 2. For testing, we sample 4 discrete noise levels uniformly from the training noise levels and test the denoising models on actual noisy-clean image pairs, which have real ground truth.

As shown in Fig. 2, training on various noise types and noise levels show consistent results, which lead to the following two findings: (1) The denoising networks trained with noisier-noisy datasets can denoise the actual noisy images (red vs. blue dots in Fig. 2) with the same noise model. (2) With the same noise model, training on the less biased dataset towards the ideal noisy-clean dataset helps the denoising networks achieve better denoising performance (Table 1).

For the first finding, the models trained with noisier-noisy datasets (red dots in Fig. 2) always show better PSNR than the actual noisy images without any denoising (blue dots in Fig. 2) on all noise types, which indicate that the model can denoise actual noisy images \( \{x_i\} \) to some extent.

\(^2\)Note that “s” here represents applying general noise to the images, which is not limited to only additive noise.

\(^3\)We randomly select 70% images of the ImageNet validation dataset for training and the rest for testing.
even with the existence of data bias between the noisier-noisy and noisy-clean datasets. Recently, both single-image-based [32] and batch-based [23] approaches have made similar conclusions on Gaussian noise but they only study a single noise level. Our investigation extends their observations to cover a wider range of noise types and noise levels.

For the second finding, we first remove all data bias in the noisier-noisy dataset, it becomes a noisy-clean dataset (zero data bias). The model trained based on it is considered to have the optimal performances (green dots in Fig. 2) on all three noise types and varying noise levels. We then synthesize different types and strengths of data bias to further verify the 2nd finding. Specifically, we apply Gaussian noise or Gaussian blur of different levels (denoted as $\sigma$) to create biased targets ($y_{gn} \& y_{gb}$) and add three different type of noise to create input noisy images ($y_{gn} + n$ and $y_{gb} + n$). The last six rows show the PSNR drop when models trained with different biased datasets to the ideal noisy-clean dataset (first row).

### Table 1. Denoising performance of models trained with biased datasets with different strengths. We corrupt the clean targets by applying Gaussian noise or Gaussian blur with different strengths ($\sigma$) to create biased targets ($y_{gn} \& y_{gb}$) and add three different type of noise to create input noisy images ($y_{gn} + n$ and $y_{gb} + n$). The last six rows show the PSNR drop when models trained with different biased datasets to the ideal noisy-clean dataset (first row).

<table>
<thead>
<tr>
<th>Training Datasets</th>
<th>$\sigma$</th>
<th>Gaussian Denoising</th>
<th>Raw Denoising</th>
<th>Correlated Denoising</th>
</tr>
</thead>
<tbody>
<tr>
<td>${y + n, y}$</td>
<td>-</td>
<td>30.896</td>
<td>35.263</td>
<td>30.484</td>
</tr>
<tr>
<td>${y_{gn} + n, y_{gn}}$</td>
<td>5</td>
<td>-0.355</td>
<td>-3.100</td>
<td>-0.172</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>-0.093</td>
<td>-1.620</td>
<td>-0.082</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-0.008</td>
<td>-0.310</td>
<td>-0.019</td>
</tr>
<tr>
<td>${y_{gb} + n, y_{gb}}$</td>
<td>5</td>
<td>-0.074</td>
<td>-0.101</td>
<td>-0.944</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>-0.069</td>
<td>-0.097</td>
<td>-0.595</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>-0.067</td>
<td>-0.081</td>
<td>-0.150</td>
</tr>
</tbody>
</table>

performance.

We first train a model $F_0$ based on the initial noisier-noisy dataset $\{x_i + n_i, x_i\}$. According to our finding 1 in Section 3.1, it can denoise the actual noisy images $\{x_i\}$ to some extent. Therefore, we can use it to denoise the noisy images and obtain a new set of refined training dataset:

$$\{F_0(x_i) + n_i, F_0(x_i)\}.$$  \hspace{1cm} (2)

The newly constructed dataset shares the same number of images and scene contents as the original noisier-noisy dataset $\{x_i + n_i, x_i\}$. However, this new dataset has less data bias to the noisy-clean dataset since the $L_2$ distance of their targets have been reduced by our trained denoising model $F_0$ \footnote{The PSNR improvements after denoised by $F_0$ indicates the $L_2$ distance to clean targets has been reduced.} (according to our finding 1 in Section 3.1).

Then, using the less biased dataset construct above, we can train a new denoising model $F_1$ from scratch in the next round:

$$F_1 \leftarrow \{F_0(x_i) + n_i, F_0(x_i)\}.$$  \hspace{1cm} (3)

Since the data bias between the new dataset $\{F_0(x_i) + n_i, F_0(x_i)\}$ and the ideal noisy-clean dataset $\{y_i + n_i, y_i\}$ is mitigated, according to our finding 2 in Section 3.1, the trained model $F_1$ can better generalize to actual noisy images $\{x_i\}$ than $F_0$ trained on the first-round noisier-noisy dataset.

Based on the improved model $F_1$, the above process can be performed iteratively and alternatively by refining the training targets and training better models as shown in Fig. 3. From the model’s perspective, once we have a denoising model $F_m$, a better model $F_{m+1}$ can be trained with the constructed dataset:

$$F_{m+1} \leftarrow \{F_m(x_i) + n_i, F_m(x_i)\}.$$  \hspace{1cm} (4)

The previous model $F_m$ helps to close the gap between the new noisier-noisy dataset $\{F_m(x_i) + n_i, F_m(x_i)\}$ and the

---

**Figure 2.** The denoising performance of denoising networks trained with the noisy-clean dataset and the noisier-noisy datasets of three representative noises and a range of noise levels. For Gaussian noise and correlated noise, we uniformly sample their noise level from [5, 20] and [2, 5] respectively. As for the real-world raw image noise, we use the noise profile of a HUAWEI Mate20, which can be modeled as a Poisson-Gaussian noise model, and sample the ISO from [800, 3200] uniformly.

"Here, the “real-world raw image noise" is under the assumption of the Poisson-Gaussian noise model."
ideal noisy-clean dataset \( \{ y_i + n_i, y_i \} \). In the next round of training, the improvement of the newly created noisier-noisy dataset leads to the advance of our newly trained model \( F_{m+1} \). On the other hand, from the target’s perspective, a series of intermediate refined targets \( \{ F_m(x_i) \} \) (for \( m = 0, 1, \ldots \)) are produced by the intermediate denoising models. As shown in Fig. 1, the noise on training targets is removed and more textures are restored progressively. Moreover, unlike traditional iterative methods that denoise one noisy image several times and lead to losing textures heavily, our method refines the training dataset and denoises the noisy image once during inference.

3.3. Fast Iterative Data Refinement

While the above iterative data refinement does not increase time cost in inference, its training time is proportional to the number of rounds of the proposed self-supervised data refinement. To alleviate the heavy training time cost, we further propose a fast approximated training scheme. The whole algorithm is shown in the Algorithm 1.

We approximate our full method in two aspects. First, we introduce one-epoch training, which performs one round of data refinement after each epoch. Specifically, for each newly constructed dataset, we use it to train our model for only one epoch instead of training it until full convergence as before. In the next epoch, we train the denoising network based on a newly constructed dataset. We sacrifice the time needed for full model optimization but increase the iteration rounds of data refinement. The overhead we introduced is to produce refined targets before each training epoch. It costs less than 5% of the overall training time5. As a result, the total training time is reduced to almost the same as training a denoising model for only one round.

Second, when training on the new dataset at each epoch, our models are initialized by the model from the previous epoch instead of from scratch. This cumulative training strategy helps the denoising network converge faster with the proposed fast data refinement scheme and ensures that the final denoising model has been optimized continuously during the whole training process. The approximation algorithm also converges fast and keeps a similar performance as our full algorithm.

4. SenseNoise-500 dataset

There exists an obvious gap between existing raw image denoising benchmarks and real-world scenarios. For example, SID [11] was created with under-exposure photos which are rare in daily photography. The mobile benchmark SIDD [3] captures multiple images for each scene (10 indoor scenes). And, it suffers from global misalignments occasionally (e.g., see image Nos. 101 & 52 in SIDD) due to the different autofocus or OIS. To evaluate the denoising performance in real-world scenarios, we propose a new high-quality dataset under the common low-light conditions using a popular mobile CMOS sensor: IMX586 with 3000 \( \times \) 4000 pixels. The dataset contains 500 scenes and

---

5The forward process of producing refined targets can be performed in parallel. We use batch size 32 for the inference. For training, our batch size is 4.
is larger than the commonly used denoising benchmarks: SIDD (10 scenes) [3], DND dataset (50 scenes) [24] and SID Sony (231 scenes) [11]. It was collected across 3 cities and covers common indoor and outdoor scenes in daily life. When we construct our dataset, there are some important considerations:

**High-quality ground truth.** The ground truths in the existing denoising benchmarks [3, 11] are still more or less noisy, especially at high noise levels. To obtain high-quality ground truths, we capture a sequence of 64 frames for each scene, among which, 4 frames of them are captured in normal exposure with the default camera setting and used as noisy testing images. The other 60 frames are long-exposure for creating the ground truth. For each pixel location, we collect all pixel values at the same location in the 60 long-exposure images and use the median value as the ground truth. The exposure time for each long-exposure frame lies in $[1\text{ s}, 2\text{ s}]$, which means that the total exposure time for creating each ground truth is between 60s and 120s. It guarantees that all ground truths are noise-free even at the highest noise level.

**Alignment and noise calibration.** To obtain well-aligned noisy/clean image pairs, we capture each image sequence on the tripod and control the mobile camera with a wireless shutter. All camera parameters are set and fixed in advance through camera2 API [19]. In this way, both normal-exposure and long-exposure frames are captured successively without any movement to the device. On the other hand, we try to avoid large object motions and dynamic illuminations in both indoor and outdoor scenes. Under this setting, all frames (long-exposure and normal-exposure) captured for one scene are perfectly aligned. While some companies may provide the noise profiles in raw files (e.g., DNG [15]), they are usually not accurate. We re-calibrate the noise parameters of IMX586 under the noise formulation in ELD [30]. However, we choose the Gaussian distribution to model the read noise, since we observe that it is more robust for mobile sensors. More details and examples of SenseNoise-500 can be found in the supplementary materials.

### 5. Experiments

To evaluate our method, we follow previous works to start with synthetic noises on popular datasets. Then, we move to tackle more challenging real-world raw image denoising, including both DSLR and mobile datasets. Moreover, we conduct ablation studies to show the generalization of our method and make further comparisons. More experiments can be found in supplementary materials.

#### 5.1. Implementation details

**Training details.** In all experiments, we use our fast IDR for comparisons. The network architecture and training settings are kept the same as N2N [18]. Specifically, we use a shallow U-Net architecture without batch normalization layers as our backbone. All models are trained from scratch with batch size 4 for 50k iterations. We use the Adam optimizer and $L_1$ loss function. The initial learning rate is fixed to $3 \times 10^{-4}$, and then halves twice at iteration 25k and 40k. Its input and output channels are differently set to adapt to different denoising tasks (e.g., gray-image denoising has 1 input channel).

**Datasets for Synthetic Experiments.** For synthetic noises, we randomly extract patches of size $256 \times 256$ from the ImageNet validation dataset for training. For testing datasets, we select three largest of them for evaluation: Kodak (24 images) and BSDS300 (300 images) for color images, and BSD68 (68 images) for gray-scale images. Fol-
Table 2. The PSNR/SSIM results of Gaussian denoising on both sRGB datasets (Kodak and BSDS300) and the gray dataset (BSD64). One superior supervised method (MPRnet [34]) is also provided for comparison. Best and second best results are highlighted and underlined.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Kodak</td>
<td>25</td>
<td>31.88 / 0.869</td>
<td>31.63 / 0.869</td>
<td>31.96 / 0.869</td>
<td>32.07 / 0.875</td>
<td>32.39 / 0.886</td>
<td>32.39 / 0.885</td>
<td>32.36 / 0.884</td>
<td>32.85 / 0.893</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>28.64 / 0.772</td>
<td>28.57 / 0.776</td>
<td>28.73 / 0.770</td>
<td>28.81 / 0.783</td>
<td>29.23 / 0.803</td>
<td>29.24 / 0.803</td>
<td>29.27 / 0.803</td>
<td>29.78 / 0.817</td>
</tr>
<tr>
<td>BSDS</td>
<td>25</td>
<td>30.47 / 0.863</td>
<td>30.72 / 0.874</td>
<td>29.57 / 0.815</td>
<td>31.12 / 0.881</td>
<td>31.39 / 0.889</td>
<td>31.40 / 0.889</td>
<td>31.48 / 0.890</td>
<td>31.87 / 0.897</td>
</tr>
<tr>
<td></td>
<td>300</td>
<td>27.14 / 0.745</td>
<td>27.60 / 0.775</td>
<td>26.18 / 0.684</td>
<td>27.87 / 0.782</td>
<td>28.17 / 0.799</td>
<td>28.17 / 0.799</td>
<td>28.25 / 0.802</td>
<td>28.65 / 0.814</td>
</tr>
<tr>
<td>BSD64</td>
<td>25</td>
<td>28.55 / 0.782</td>
<td>27.64 / 0.781</td>
<td>NA</td>
<td>28.81 / 0.818</td>
<td>29.15 / 0.831</td>
<td>29.20 / 0.835</td>
<td>29.20 / 0.835</td>
<td>29.40 / 0.840</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>25.59 / 0.670</td>
<td>25.46 / 0.681</td>
<td>NA</td>
<td>25.95 / 0.703</td>
<td>26.23 / 0.725</td>
<td>26.24 / 0.727</td>
<td>26.25 / 0.726</td>
<td>26.48 / 0.736</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>28.71 / 0.784</td>
<td>28.60 / 0.793</td>
<td>29.11 / 0.785</td>
<td>29.11 / 0.807</td>
<td>29.43 / 0.822</td>
<td>29.44 / 0.823</td>
<td>29.47 / 0.823</td>
<td>29.84 / 0.833</td>
</tr>
</tbody>
</table>

Table 3. Denoising performances (PSNR/SSIM) of different compared methods on raw image denoising. For the SID dataset [11], we follow ELD [30] to show the results on ISO 100, 250, and 300. For our SenseNoise-500 dataset, we divide it into normal scenes (ISO ≤ 12800) and extreme scenes (ISO > 12800). Best and second best results are highlighted and underlined.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SID</td>
<td>100</td>
<td>32.92 / 0.758</td>
<td>35.61 / 0.830</td>
<td>35.64 / 0.827</td>
<td>37.83 / 0.898</td>
<td>38.02 / 0.906</td>
<td>38.60 / 0.912</td>
<td>37.67 / 0.884</td>
</tr>
<tr>
<td></td>
<td>250</td>
<td>29.56 / 0.686</td>
<td>31.39 / 0.698</td>
<td>30.58 / 0.655</td>
<td>33.44 / 0.814</td>
<td>33.88 / 0.819</td>
<td>37.08 / 0.886</td>
<td>33.14 / 0.767</td>
</tr>
<tr>
<td></td>
<td>300</td>
<td>28.88 / 0.674</td>
<td>29.78 / 0.649</td>
<td>28.88 / 0.595</td>
<td>31.61 / 0.769</td>
<td>32.14 / 0.779</td>
<td>36.29 / 0.874</td>
<td>31.36 / 0.727</td>
</tr>
<tr>
<td>SenseNoise-500</td>
<td>≤ 12800</td>
<td>41.05 / 0.958</td>
<td>31.05 / 0.867</td>
<td>41.86 / 0.961</td>
<td>41.22 / 0.956</td>
<td>44.01 / 0.978</td>
<td>44.80 / 0.981</td>
<td>43.36 / 0.978</td>
</tr>
<tr>
<td></td>
<td>&gt; 12800</td>
<td>37.54 / 0.940</td>
<td>27.95 / 0.871</td>
<td>38.28 / 0.948</td>
<td>39.53 / 0.965</td>
<td>40.17 / 0.966</td>
<td>40.23 / 0.966</td>
<td>39.92 / 0.965</td>
</tr>
</tbody>
</table>

Due to the space limitation, we only put the results of Gaussian noise and real-world noise in the main text. Please find the results of other noises in supplementary materials.

5.2. Main results

Due to the space limitation, we only put the results of Gaussian noise and real-world noise in the main text. Please find the results of other noises in supplementary materials.

Details of Gaussian noise experiments. We compare our method with six learning-based methods (N2N [18], N2V [16], N2S [5], DBSN [31] and Noisier2noise [23].) and also the representative traditional method BM3D [21]. For BM3D, we use the authors’ refined implementation\(^6\). For learning-based methods, all of them use the same U-Net architecture except for DBSN [31] provided by the authors, which is 10× larger (97.4G FLOPS) than U-Net (8.2G FLOPS). For DBSN, we use the author’s released models. All other methods are retrained using the author’s codes and keep the same training settings as N2N.

Results of Gaussian noise experiments. The PSNR and SSIM results are shown in Table 2. Our method outperforms all unsupervised methods on both color and gray images. And, surprisingly, our method even surpasses the supervised method slightly in most cases. Both our method and N2D introduce new training targets to unsupervised denoising, and show clear gains compared with other methods that only use single noisy images. But, the well-aligned paired noisy images required by N2N are difficult to capture in practice. On the contrary, the noise model used in our method can be easily calibrated and already saved in raw images. They can be easily obtained or recalibrated [30, 37] in real-world raw image denoising. Some qualitative results are shown in Fig. 4. Our method restores more textures and colors while keeping the high-quality denoising results.

Details of real-world raw noise experiments. For BM3D, we use the mixture Gaussian noise model to estimate the noise level. For learning-based methods, we re-train all methods by adopting the authors’ code.

Results of real-world raw noise experiments. The results are shown in Table 3. Some qualitative results are shown in Fig. 5. Our method performs consistently better than other unsupervised methods on both DSLR and smartphones datasets. Both the traditional method BM3D [21] and the unsupervised methods N2V [16] and N2S [5] with single noisy images cannot work well and show more than

\(^6\)https://pypli.org/project/bm3d/
6. Comparison with supervised learning from external clean sRGB images.

Once knowing the noise model, we can synthesize data for supervised training. So, an interesting comparison of our method would be the supervised training with the synthetic data, especially for the real-world raw image denoising. But, the large-scale clean raw image dataset for each sensor is hard and expensive to collect. Following the common practice, we unprocess the external clean sRGB image to obtain clean raw images [6].

We conduct the study on the SID raw dataset. For the supervised method, we collect the clean sRGB images from MIT5K dataset [9] and unprocess them to obtain raw images [6] with proper parameters. For a fair comparison, we keep the same amount of data and the noise model for both methods. The results are shown in Tab. 4. Our methods could still outperform supervised learning with external clean ground-truths due to the dataset content bias. Then, we remove the image content bias by unprocessing the SID sRGB images for the supervised method, its performance improves over the training on MIT5K, but the performance gap with our method still exists as shown in Tab. 4. While the supervised method ("clean image+simulated noise") uses the same noise model as ours, our method outperforms it over 0.5dB (averaged PSNR 34.17dB vs. 34.68dB) on SID raw dataset. The reason is that there still exists a clear gap (image content, raw image characters) between the synthetic and real clean raw images (unprocessed MIT5K/SID sRGB vs. SID raw).

7. Limitations and conclusions

In this paper, we propose a practical and iterative denoising method (IDR) that outperforms the existing unsupervised denoising methods. Our method is based on empirical findings on data bias of learning denoising. While lacking theoretical support, experiments on three synthetic noises, real-world raw image noise, and even spatially-correlated noise demonstrate the consistent robustness and effectiveness of our method. Moreover, we build a high-quality low-light denoising dataset for evaluating the denoising performance in real-world scenarios. The dataset contains 500 diverse scenes, each with high-quality ground truth. We expect the dataset can serve as a real-world benchmark for image denoising.

8. Acknowledgement

This work is supported in part by Centre for Perceptual and Interactive Intelligence Limited, in part by the General Research Fund through the Research Grants Council of Hong Kong under Grants (Nos. 14204021, 14207319, 14203118, 14208619), in part by Research Impact Fund Grant No. R5001-18, in part by CUHK Strategic Fund.
References


[31] Xiaohe Wu, Ming Liu, Yue Cao, Dongwei Ren, and Wangmeng Zuo. Unpaired learning of deep image denoising. In European Conference on Computer Vision (ECCV), 2020. 1, 3, 7


[34] Syed Waqas Zamir et al. Multi-stage progressive image restoration. In CVPR, 2021. 7

