APR-RD: Complemental Two Steps for Self-Supervised Real Image Denoising

Hyunjun Kim[1], **Nam Ik Cho**[1,2*]

[1]Department of ECE, INMC, Seoul National University, Seoul, Korea
[2]IPAI, Seoul National University, Seoul, Korea
hyunjun0.kim@snu.ac.kr, nicho@snu.ac.kr

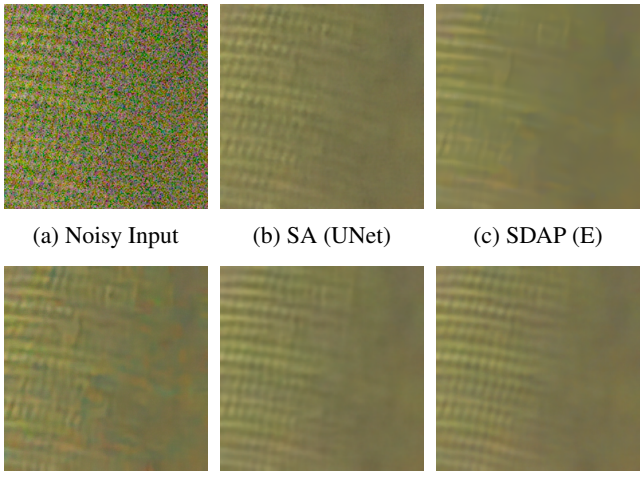
Abstract

Recent advancements in self-supervised denoising have made it possible to train models without needing a large amount of noisy-clean image pairs. A significant development in this area is the use of blind-spot networks (BSNs), which use single noisy images as training pairs by masking some input information to prevent *noise transmission* to the network output. Researchers have shown that BSNs are capable of reconstructing clean pixels from various types of independent pixel-wise degradations, such as synthetic additive white Gaussian noise (AWGN). However, unlike synthetic noise, real noise often contains highly correlated components which can induce noise transmission and reduce the performance of BSNs. To address the spatial correlation of real noise, we propose the Adjacent Pixel Replacer (APR), which decorrelates noise without a downsampling process that is widely adopted in previous research. The dissimilarity in our APR-generated pairs serves as relatively different noise components during training. Hence, it enables the BSN to block noise transmission while utilizing clean information effectively. As a result, BSN can utilize denser information to reconstruct the corresponding center pixel. We also propose Recharged Distillation (RD) to enhance high-frequency textures without additional network modifications. This method selectively refines clean information from recharged noisy pixels during distillation. Extensive experimental results demonstrate that our proposed method outperforms the existing state-of-the-art self-supervised denoising methods in real sRGB space.

Project Page — https://github.com/HYK2017/APRRD

Introduction

Image denoising is an important task in computer vision that has been greatly improved with the development of Convolutional Neural Networks (CNN). Early CNN-based denoising methods (Zhang et al. 2017; Zhang, Zuo, and Zhang 2018; Chen and Pock 2016; Tai et al. 2017) trained with AWGN have demonstrated better performance compared to traditional model-based methods (Dabov et al. 2007; Buades, Coll, and Morel 2005; Gu et al. 2014; Elad and Aharon 2006). However, using a simple noise model is not ideal for accurately representing real-world noise, which involves complex degradations occurring through the in-camera pipeline (Brooks et al. 2019; Wei et al. 2020). This difference between synthetic and real-world scenarios significantly affects the performance of denoisers (Guo et al. 2019; Anwar and Barnes 2019; Kim et al. 2020).



(a) Noisy Input (b) SA (UNet) (c) SDAP (E)

(d) C-BSN (e) AT-BSN (D) (f) APR (RD) (Ours)

Figure 1: Comparison with recent self-supervised denoisers. Our approach outperforms others in noise reduction and detail preservation.

In recent years, researchers have created real-world noise datasets (Nam et al. 2016; Abdelhamed, Lin, and Brown 2018; Brummer and De Vleeschouwer 2019) that include noisy images captured by cameras along with their corresponding clean images. These paired datasets have been valuable for the development and comparison of various image denoising methods. However, creating a real noisy-clean dataset requires a deep understanding of the Image Signal Processor (ISP) responsible for converting data from the raw space to the sRGB space. If the ISP in the capturing device is biased, the denoising results obtained from these noisy images may exhibit color shifts that deviate significantly from the clean ground truth (GT) images. Furthermore, for non-optical imaging systems such as electron microscopy or medical imaging, it is often impractical to obtain real-world clean images, unlike in the case of optical imaging cameras.

*Corresponding author.

Hence, self-supervised training methods (Lehtinen et al. 2018; Batson and Royer 2019; Xu et al. 2020) have gained attention as a GT-free alternative. An important concept in this area is the blind-spot network (BSN) (Krull, Buchholz, and Jug 2019), which uses only surrounding pixels to reconstruct central information with a single noisy image. This indicates that dealing with the correlation of noise elements between pixels is a significant challenge for BSN. Unfortunately, the majority of real noise is highly correlated (Chatterjee et al. 2011; Jin, Facciolo, and Morel 2020), limiting the practical application of BSN (Lee, Son, and Lee 2022).

In an effort to address the challenge of real noise, Neighbor2Neighbor (Huang et al. 2021) introduced a sampling method that generates different noisy pairs from a single noisy image. However, the downscaling-based pair generation was unable to accurately estimate given textures. Furthermore, AT-BSN (Chen et al. 2024) proposed a structurally large blind-spot to handle locally correlated noise. It also introduced a multi-distillation approach to consider the inferred features from trained BSN as multiple targets. Simply aiming for the output of a denoiser disrupts the accurate estimation of the GT distribution.

To overcome the limitations of previous methods, we propose a new pair sampler called the Adjacent Pixel Replacer (APR). This method replaces each pixel in the original image with random adjacent pixels. By using APR, the BSN can more easily break the noise correlation between the input and target of training pairs. In addition, we develop a new distillation technique called Recharged Distillation (RD) that effectively prevents identity mapping toward the noise distribution. Importantly, our proposed approaches outperform existing methods without requiring additional parameters. Visual examples are presented in Figure 1.

Our contributions are summarized as follows:

- We propose a new method called APR for pair sampling. APR enables BSN to effectively remove spatially correlated noise. Since APR does not involve any downscaling process, our sampled pairs are free from aliasing.
- To achieve improved generalization ability, we propose a new approach, RD, which refines additional clean details during the distillation process.
- The combination of APR and RD achieves state-of-the-art (SOTA) performance in self-supervised real sRGB denoising and produces satisfying visual results.

Related Works

Supervised Synthetic Image Denoising

Since the remarkable achievement in AWGN removal reported by DnCNN (Zhang et al. 2017), extensive research has shown that nonlinear operations within CNNs can effectively address complex degradations such as multivariate noise (Foi 2009), mixture noise (Makitalo and Foi 2012; Zhang and Hirakawa 2017), and noise models founded on physics (Zhang et al. 2021; Zou and Fu 2022). However, their predefined assumptions are limited in addressing the more complex characteristics of real-world noise, which is random, signal-dependent, and spatially variant.

Supervised Real Image Denoising

For this reason, researchers have focused on obtaining GT from noisy scenes, rather than finding synthetic noise models. SIDD (Abdelhamed, Lin, and Brown 2018) removes outliers from numerous noisy images taken from the same scene, and then uses weighted least squares to obtain the underlying clean pixels. However, acquiring noise statistics is labor-intensive, so several frameworks have been proposed to enhance the adaptability through limited real-paired datasets. CBDNet (Guo et al. 2019) merges approximated ISP information into the main framework and imposes a penalty in noise-underestimated situations. RIDNet (Anwar and Barnes 2019) is the first to apply feature attention in real image denoising. They utilize global textural information through an activated global pooling vector. Despite these efforts, they remain dependent on GT.

Self-Supervised Image Denoising

The Noise2Noise (N2N) (Lehtinen et al. 2018) demonstrated that CNNs can recover clean pixels from independently corrupted noisy pairs. Following this, research on self-supervised denoising has focused on reducing the noise correlation of training pairs. Noisier2Noise (Moran et al. 2020) and similar approaches consider further corrupted images as decorrelated counterparts. Noise2Void (Krull, Buchholz, and Jug 2019), which can be trained on single noisy images, masks some input pixels and the loss is calculated only on those masked pixels. This method serves as a prototype for BSNs.

Applications of Blind-Spot Network

This section provides an overview of notable BSN-based methods. DBSN (Wu et al. 2020) and Laine-BSN (Laine et al. 2019) are foundational approaches that create a blind-spot in the receptive field using dilated and shifted convolutions, respectively. Based on these methods, AP-BSN (Lee, Son, and Lee 2022) and AT-BSN (Chen et al. 2024) customize their receptive fields by employing pixel downsampling and feature shifting. Furthermore, PUCA (Jang et al. 2024) introduces channel attention to DBSN by patch downsampling. Other researchers consider sampled images from a given noisy image as decorrelated training counterparts. Neighbor2Neighbor (Huang et al. 2021) generates two different noisy images through a random sampler, while SDAP (Pan et al. 2023) uses shuffled sub-images as random training pairs. However, they may suffer from discontinuity caused by downsampling. More recently, distillation has gained attention as a post-processing technique (Chen et al. 2024; Li, Zhang, and Zuo 2024). However, it is often limited to weight-lightening or biased towards pre-trained BSNs. In this paper, we aim to further preserve textural information through scale-invariant sampling and propose a simple but efficient pixel recharging method that surpasses the expressive capability of the trained BSN.

Methodology

In this section, we revisit the two mentioned papers and explain our novel proposal.

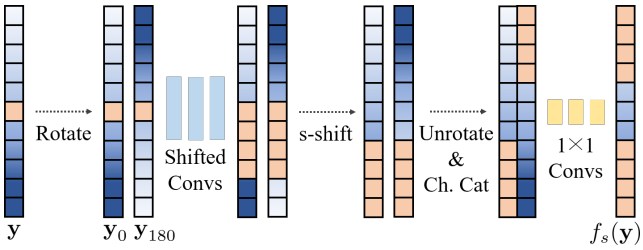


Figure 2: AT-BSN with a simplified 1-D vector. For a 2D array, four directional rotations are used as inputs.

Revisiting Neighbor2Neighbor

NBR2NBR (Huang et al. 2021) introduced neighbor subsampler G to obtain half-sized noisy pairs $(g_1(\mathbf{y}),\ g_2(\mathbf{y}))$ from a single noisy image $\mathbf{y} = \mathbf{x} + \mathbf{n}$ where $\mathbf{x}$ is the GT and $\mathbf{n}$ is the noise component. For the denoiser f_θ, the optimization of the simple reconstruction loss using sampled pairs is expressed as follows:

$$\arg\min_{\theta} \mathbb{E}_{\mathbf{x},\mathbf{y}} |f_\theta(g_1(\mathbf{y})) - g_2(\mathbf{y})|_2^2. \tag{1}$$

During the optimization, the denoiser can avoid the perfect correlation of noise components between sampled pair.

However, since the GTs of $g_1(\mathbf{y})$ and $g_2(\mathbf{y})$ are not identical, the optimized denoiser f^* cannot accurately estimate the original GT $\mathbf{x}$. To correct the undesired learning objective, they impose the following regularization:

$$\begin{aligned} &\arg\min_{\theta} \mathbb{E}_{\mathbf{x},\mathbf{y}} |f_\theta(g_1(\mathbf{y})) - g_2(\mathbf{y})|_2^2, \text{ s.t.} \\ &\mathbb{E}_{\mathbf{x},\mathbf{y}}\{f_\theta(g_1(\mathbf{y})) - g_2(\mathbf{y}) - (g_1(f_\theta(\mathbf{y})) - g_2(f_\theta(\mathbf{y})))\} = 0. \end{aligned} \tag{2}$$

Since G satisfies $\mathbb{E}_{\mathbf{y}|\mathbf{x}}\{g_l(\mathbf{y})\} = g_l(\mathbf{x})$ for $l = \{1, 2\}$, the solution of the regularization term in Equation 2 becomes $f^*(\mathbf{y}) = \mathbf{x}$ and $f^*(g_l(\mathbf{y})) = g_l(\mathbf{x})$. This indicates that the regularization guides the learning objective towards the original GT.

Revisiting AT-BSN

AT-BSN (Chen et al. 2024) proposed a tunable blind-spot for real sRGB denoising. For convenience, we define 'k-shift' as cropping the bottom k rows and zero-padding the top k rows of a feature. Inspired by Laine-BSN (Laine et al. 2019), AT-BSN applies 's-shift' to the output feature of the last shifted convolution to form a blind-spot of size $(2s-1) \times (2s-1)$. The process of AT-BSN is depicted in Figure 2. They use $f_{\theta,s=5}$ during training and $\tilde{f}_{\theta,s=2}$ during inference, where the shift-factor s represents applying the final 's-shift', and '~' indicates the state of the BSN after completing training. That is, the size of the blind-spot for training and inference is 9×9 and 3×3, respectively.

Their additional contribution, multi-teacher distillation D, considers $T = \{\tilde{f}_{\theta,s=0}(\mathbf{y}), \tilde{f}_{\theta,s=1}(\mathbf{y}), .., \tilde{f}_{\theta,s=j-1}(\mathbf{y})\}$ as a 'multi-target' which consists of outputs from different inference blind-spots of the trained AT-BSN:

$$\mathcal{L}_{Distill} = \sum_{i=0}^{j-1} |f_D(\mathbf{y}) - \tilde{f}_{\theta,i}(\mathbf{y})|_1^1. \tag{3}$$

where f_D is a NBSN (i.e., a normal CNN) for distillation.

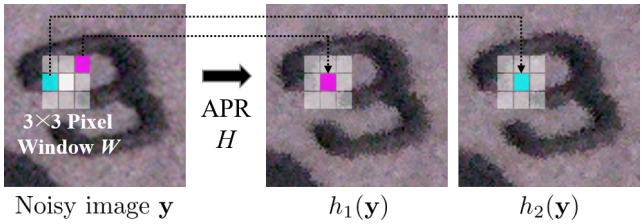


Figure 3: An example of pair sampling using the APR. Two arbitrary pixels are selected within a fixed window which traverses all pixels in a given single noisy image. The selected magenta and cyan pixels are placed at the center position of the window in each pair image.

Scale-Preserving Sampling

We emphasize that the neighbor subsampler G is a method for reducing the size or scale of an image. The aliasing effect caused by G can disrupt the accurate restoration of high-frequency details, thereby limiting the maximum level of performance. Furthermore, pixel aliasing has a greater negative impact on the sRGB color space compared to the raw space because the information from multiple pixels in the raw space gets compressed into a single pixel in the sRGB space during demosaicing.

To alleviate the loss of structural information, we introduce our APR which involves a fixed window that traverses all the pixels within a single image, where the central pixel is replaced by a random adjacent pixel within the window.

Specifically, our APR, which is expressed as $H(\mathbf{y}) = (h_1(\mathbf{y}),\ h_2(\mathbf{y}))$, is detailed as follows:

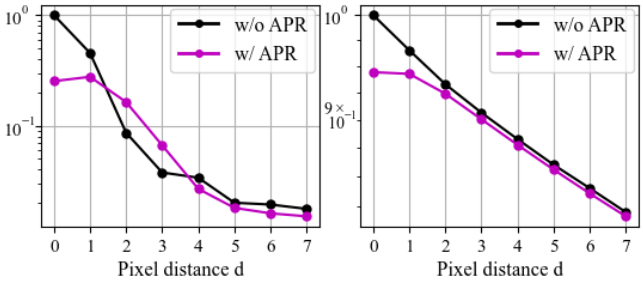
- Define a 3×3 window W, with its center denoted as W_{center}. This window traverses all the pixels in a single noisy image $\mathbf{y}$.
- When W_{center} is positioned on the (i, j)-th pixel of the image, randomly select two different pixels within the window W.
- Each of the selected pixels will then correspond to the (i, j)-th pixel of the APR pairs $h_1(\mathbf{y})$ and $h_2(\mathbf{y})$, respectively.

The APR process is visualized in Figure 3.

Then, we predict the effect of the APR pairs when the network is optimized using a simple reconstruction loss.

$$\arg\min_{\theta} \mathbb{E}_{\mathbf{x},\mathbf{y}} |f_\theta(h_1(\mathbf{y})) - h_2(\mathbf{y})|_2^2. \tag{4}$$

In the typical case of BSN training where $\mathbf{y} = \mathbf{x}+\mathbf{n}$ is used as both the input and target, the cross-pixel correlation between the input and target noise is the same as the inter-pixel correlation of $\mathbf{n}$ itself. This is also true for the GT. However, when the proposed APR is applied, $h_1(\mathbf{y}) = h_1(\mathbf{x}) + h_1(\mathbf{n})$ and $h_2(\mathbf{y}) = h_2(\mathbf{x}) + h_2(\mathbf{n})$ are used as training pairs. Thus, the cross-pixel correlation between the input and target noise becomes the cross-pixel correlation between $h_1(\mathbf{n})$ and $h_2(\mathbf{n})$. Similarly, for the GT, it corresponds to the cross-pixel correlation between $h_1(\mathbf{x})$ and $h_2(\mathbf{x})$. We analyze the differences in these correlations when applying APR. Figure 4 shows the cross-pixel correlation for each situation, based on pixel distance d.



(a) Noise correlation (b) GT correlation

Figure 4: Cross-pixel correlation between the input and target for each component in BSN training with and without the use of APR. This is obtained from a large number of SIDD medium patches. When APR is not applied, the input and target are identical noisy images.

Combination of our APR and AT-BSN

Most real-world noisy images show a slight decrease in GT correlation as the distance between pixels increases, whereas the correlation of noise decreases significantly. Building on this, AT-BSN demonstrates its ability to block correlated noise within a 9×9 blind-spot during training, while utilizing pixel information outside the blind-spot to restore detailed textures. However, if long-range noise correlations exist in the given sample, noise outside the blind-spot may correlate with the target pixel, leading to noise being learned and propagated to the output.

We suggest that APR pairs can serve as a novel tool to further suppress the intrusion of external noise outside the blind-spot during the training of AT-BSN $f_{\theta,s=5}$. According to Figure 4, the noise correlation between the target pixel and its receptive field ($d \geq 5$), which is the external region of the 9×9 blind-spot, is reduced when APR is applied. Thus, AT-BSN can be optimized to reduce noise transmission from the receptive field to the output.

However, Figure 4 also indicates that APR distorts GT correlation, which is an unpleasant trade-off.

Compatibility with Regularization Loss

We recognize that H is another transformation that satisfies $\mathbb{E}_{\mathbf{y}|\mathbf{x}}\{h_m(\mathbf{y})\} = h_m(\mathbf{x})$ for $m = \{1, 2\}$. That is, we can apply the regularization proposed in NBR2NBR to our APR to support GT estimation. According to their equation, it can be easily verified that $f^*(\mathbf{y}) = \mathbf{x}$ and $f^*(h_m(\mathbf{y})) = h_m(\mathbf{x})$ become the solutions:

$$\begin{aligned} &\mathbb{E}_{\mathbf{x},\mathbf{y}}\{f^*(h_1(\mathbf{y})) - h_2(\mathbf{y}) - (h_1(f^*(\mathbf{y})) - h_2(f^*(\mathbf{y})))\} \\ &= \mathbb{E}_{\mathbf{x}}\{\mathbb{E}_{\mathbf{y}|\mathbf{x}}\{h_1(\mathbf{x}) - h_2(\mathbf{y}) - (h_1(\mathbf{x}) - h_2(\mathbf{x}))\}\} \\ &= \mathbb{E}_{\mathbf{x}}\{-\mathbb{E}_{\mathbf{y}|\mathbf{x}}\{h_2(\mathbf{y})\} + h_2(\mathbf{x})\} = 0. \end{aligned} \tag{5}$$

Therefore, our proposed APR can combine each contribution of the two previous studies, and the total training loss for this combination is presented in Equation 6. For better generalization, we adopt the $\mathbf{L}1$ form of the total loss $\mathcal{L}_{APR}$:

$$\begin{aligned} \mathcal{L}_{APR} &= \mathcal{L}_{rec} + \mathcal{L}_{reg} \\ &= |f_{\theta,5}(h_1(\mathbf{y})) - h_2(\mathbf{y})|_1^1 \\ &+ \lambda * |f_{\theta,5}(h_1(\mathbf{y})) - h_2(\mathbf{y}) - (h_1(\hat{f}_{\theta,5}(\mathbf{y})) - h_2(\hat{f}_{\theta,5}(\mathbf{y})))|_1^1. \end{aligned} \tag{6}$$

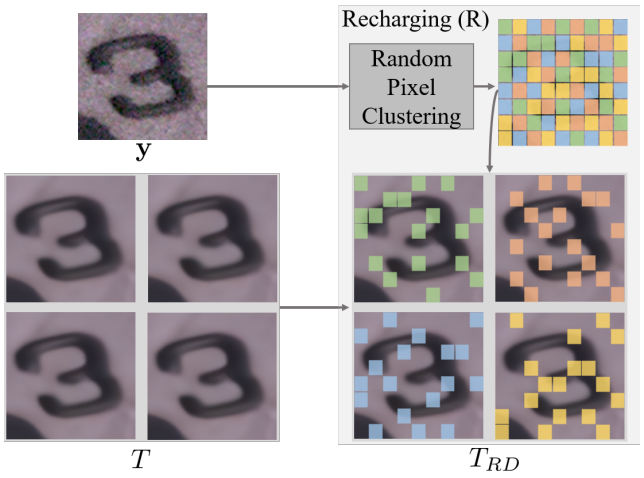


Figure 5: An example of the recharging process in RD when T is a set of 4 different inferred outputs from a trained BSN. The recharged pixels in each element of T_{RD} do not share the same pixel locations.

where $\hat{f}_{\theta,s}(y)$ represents the output without gradients, and the regularization factor λ is set to 4 empirically.

In summary, we define APR training as the integration of the AT-BSN network, our novel sampler APR, and the NBR2NBR training methodology.

Recharged Distillation

We recognize that the previous distillation D can be biased toward trained AT-BSN. Consequently, they trained f_D separately using different settings of T to fit each test dataset. This approach is somewhat impractical for extensive noisy samples with difficult-to-classify noise levels.

To address this, we propose RD, which utilizes T_{RD} as a new multi-target. This new method does not solely rely on the restoration capability of the trained BSN $\tilde{f}_{\theta,s}$, but provides the distillation network f_D with fresh learning guidance towards the original noisy distribution.

The details of our RD are as follows, as visualized in Figure 5. For $T = \{\tilde{f}_{\theta,0}(\mathbf{y}), \tilde{f}_{\theta,1}(\mathbf{y}), .., \tilde{f}_{\theta,j-1}(\mathbf{y})\}$, a set of trained BSN's results,

- The pixels in a noisy $\mathbf{y}$ are randomly clustered into j subsets:
$$\mathbf{y} = \sum_{i=0}^{j-1} \mathbf{y}_{sub(i)} = \sum_{i=0}^{j-1} \mathbf{y} \odot M_i. \tag{7}$$
where M_i for $i = \{0, 1, 2, .., j-1\}$ are random binary masks, each containing an equal number of 1s, and $\odot$ denotes element-wise multiplication.
- Clustered subsets are paired one-to-one with the T elements respectively. Then, non-zero pixels in each subset $\mathbf{y}_{sub(i)}$ are refilled into the same indices of the paired T element $\tilde{f}_{\theta,s=i}(\mathbf{y})$. In other words, the pixels in $\mathbf{y}$ are spread into T elements without index overlapping:
$$R : \tilde{f}_{\theta,i}(\mathbf{y}) \rightarrow \tilde{\mathbf{x}}_i = \tilde{f}_{\theta,i}(\mathbf{y}) \odot (I - M_i) + \mathbf{y} \odot M_i. \tag{8}$$

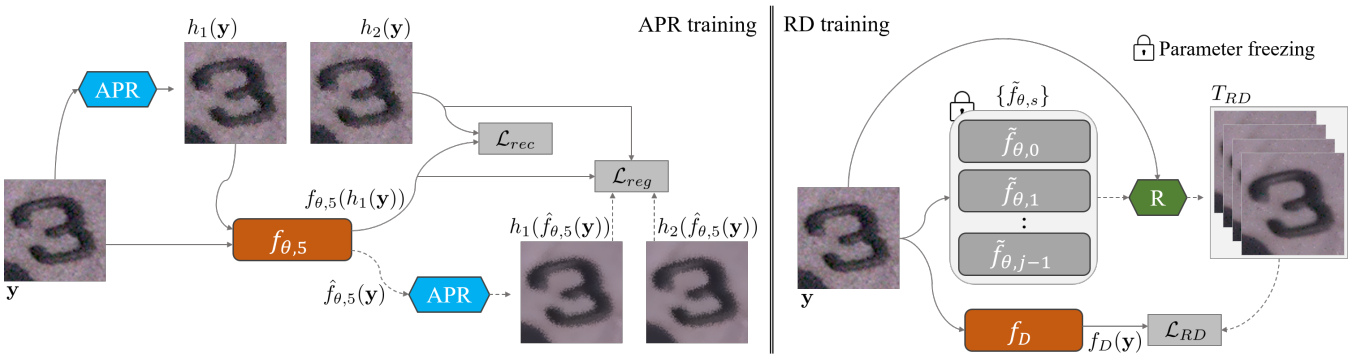


Figure 6: Visualization of APR and RD training. Dashed lines indicate that no gradients are generated for backpropagation. In APR training, the APR pair from the output feature of AT-BSN is not directly computed in the feedback, but is merged into $\mathcal{L}_{reg}$ to control the optimization of $\mathcal{L}_{rec}$, which is optimized with the APR pair from an input noisy image. In RD training, the trained AT-BSN is frozen but provides denoised features to create a new multi-target T_{RD}. The distillation network is optimized with a linear combination of sub-losses computed from each element of the new multi-target.

where I is an all-ones matrix with the same size as $\mathbf{y}$. Therefore, $R(T, \mathbf{y}) = T_{RD} = \{\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, .., \tilde{\mathbf{x}}_{j-1}\}$ becomes a new multi-target.

- The loss of our RD $\mathcal{L}_{RD}$ is the linear sum of sublosses computed from each element of T_{RD}:

$$\mathcal{L}_{RD} = \sum_{i=0}^{j-1} \mathcal{L}_{sub(i)} = \sum_{i=0}^{j-1} |f_D(\mathbf{y}) - \tilde{\mathbf{x}}_i|_1^1. \quad (9)$$

We examine whether our RD can approach the original noisy distribution. The recharged noisy pixels do not occupy the same pixel locations across the elements of T_{RD}. Therefore, for each pixel of the output $f_D(\mathbf{y})$, only one element of T_{RD} provides a noisy pixel as the learning target. Conversely, the remaining elements provide denoised pixels as targets. In other words, each element of T_{RD} mutually regulates noise identity mapping during the distillation. In this situation, f_D can further refine underlying GT in the input y. This is based on the principles of self-supervised denoising.

In summary, our proposed method consists of two phases, 'replacing' and 'recharging', as APR and RD respectively focus on assisting strong denoising through replacing and refining high-frequency details through recharging. The training processes of APR and RD are visualized in Figure 6. In the inference phase of both APR and RD, test images are directly used as input without any processing.

Experiments

Our experiment was conducted in two phases: (1) Training of APR and (2) Training of RD. Each phase involves training BSN and NBSN, respectively.

Training Details

Training Settings of APR (BSN). In the first phase of training, we use the AT-BSN architecture. It is a U-Net (Ronneberger, Fischer, and Brox 2015)-like BSN which consists of shifted convolutions. To ensure consistent result comparison, we adopt the same training settings as theirs, as follows. We optimize APR using the Adam optimizer, with the values of β_1 and β_2 set to 0.9 and 0.999, respectively. The initial learning rate of 0.0003 decreases to zero over 400,000 iterations using a cosine scheduler.

Training Settings of RD (NBSN). For training RD, we use the student version C described in AT-BSN. This NBSN uses the same optimizer settings as BSN, and the initial learning rate of 0.0003 decreases to zero over 200,000 iterations using a cosine scheduler. The difference in our approach is that it is trained only once using a single setting of T_{RD}.

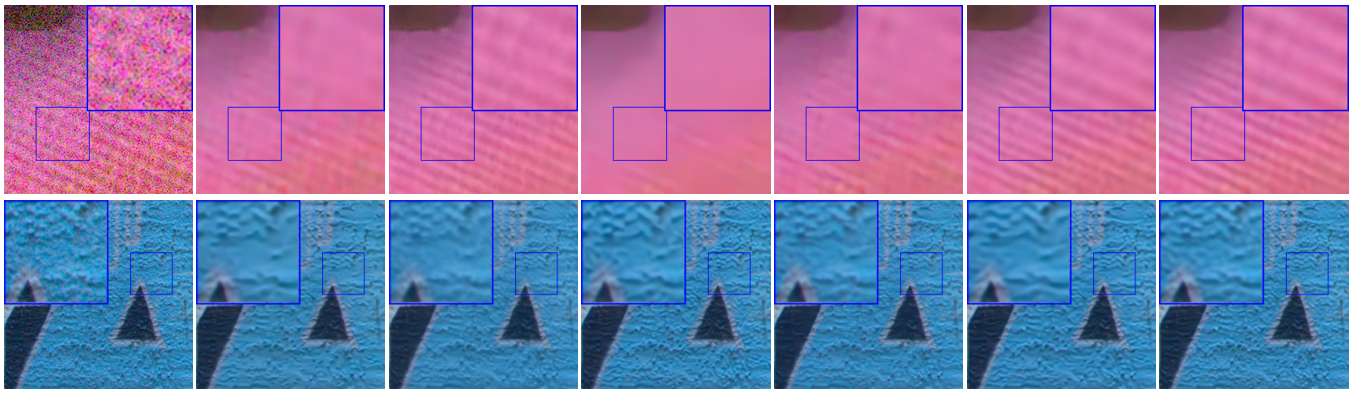
Datasets for Training and Evaluation. We trained and evaluated our method using the SIDD and DND (Plotz and Roth 2017) benchmark, which are real noise datasets obtained from actual camera pipelines. SIDD consists of SIDD medium, validation, and benchmark. We trained the network on the SIDD medium dataset and evaluated it on the other datasets. SIDD validation was evaluated offline using the provided GT. The benchmark results were submitted to their official websites for evaluation.

Quantitative and Qualitative Results. We quantitatively compared our method with various non-learning, supervised (Yue et al. 2019; Zamir et al. 2022), unpaired (Chen et al. 2018; Jang et al. 2021), and self-supervised (Neshatavar et al. 2022; Jang et al. 2023; Li et al. 2023) methods based on PSNR and SSIM metrics. This comparison is presented in Table 1. For qualitative comparison, we include result images of each self-supervised method. These are shown in Figure 7. Our method demonstrates superior performance compared to the baseline AT-BSN. It also shows stable visual quality compared to the latest self-supervised methods in terms of noise removal and high-frequency detail restoration.

Compared to the baseline AT-BSN, we presented the results of APR derived from $\tilde{f}_{\theta,s=1}$, and APR (RD) achieves improved results across all evaluation sets with a single setting of T_{RD}. These outcomes suggest alleviated condition in BSN training and enhanced robustness in distillation. A detailed analysis of these improvements is presented in the subsequent subsection.

Supervision	Methods	SIDD Validation		SIDD Benchmark		DND Benchmark	
		PSNR↑ (dB)	SSIM↑	PSNR↑ (dB)	SSIM↑	PSNR↑ (dB)	SSIM↑
Non-Learning	BM3D	25.71	0.576	25.65	0.685	34.51	0.851
	WNNM	26.05	0.592	25.78	0.809	34.67	0.865
Supervised	DnCNN	37.73	0.943	37.61	0.941	37.90	0.943
	CBDNet	33.07	0.863	33.28	0.868	38.05	0.942
	VDN	39.29	0.956	39.26	0.955	39.38	0.953
	Restormer	39.93	0.960	40.02	0.960	40.03	0.956
Unpaired	GCBD	-	-	-	-	35.58	0.922
	C2N	35.36	0.932	35.35	0.937	37.28	0.924
Self-Supervised	CVF-SID	34.17	0.913	35.04$^\diamond$	0.856$^\diamond$	36.50	0.923
	AP-BSN (R^3)	36.74	0.934	37.14$^\diamond$	0.878$^\diamond$	38.18	0.937
	C-BSN	36.22	0.935	37.05$^\diamond$	0.881$^\diamond$	38.45	0.939
	SDAP (E)	37.30	0.939	37.64$^\diamond$	0.882$^\diamond$	37.86	0.937
	SA (UNet)	37.39	0.934	37.85$^\diamond$	0.880$^\diamond$	38.18	0.938
	AT-BSN (D)	37.80	0.944	38.10$^\diamond$	0.891$^\diamond$	38.34	0.941
	APR (Ours)	37.23	0.936	37.62$^\diamond$	0.872$^\diamond$	38.10	0.938
	APR (RD) (Ours)	**38.00**	**0.947**	**38.26**$^\diamond$	**0.895**$^\diamond$	**38.57**	**0.942**
	APR (RD)† (Ours)	-	-	38.23$^\diamond$	0.895$^\diamond$	38.83	0.944

Table 1: Quantitative comparison on SIDD validation, benchmark, and DND benchmark. We use the †symbol to indicate evaluation results where the network is trained in a fully self-supervised way, and the $^\diamond$symbol to indicate results obtained from the Kaggle competition. Because the SIDD benchmark evaluation server is currently unavailable, we have replaced the results of the self-supervised method with those obtained from the Kaggle competition. Since a pre-trained model of AT-BSN (D) was not provided before submission, we used a model trained by ourselves to obtain the results from the Kaggle competition.



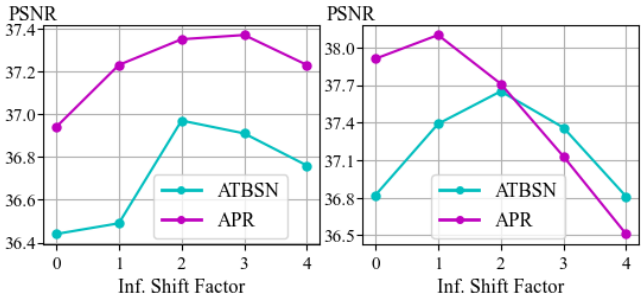
(a) Noisy Input (b) AP-BSN (R3) (c) SA (UNet) (d) SDAP (E) (e) C-BSN (f) AT-BSN (D) (g) Our APR (RD)

Figure 7: Qualitative comparison with other self-supervised denoisers. From top to bottom, each row shows the results for the SIDD validation and the DND benchmark, respectively.

Analysis of APR

Effect of APR Training. We compared the results of two AT-BSN models, one trained with and the other without APR, based on their inference spot size (i.e., shift-factor s on $\tilde{f}_{\theta,s}$). We observe that the model trained with APR can infer the output pixel by utilizing closer pixels. In other words, APR is trained with the same 9×9 blind-spot, but demonstrates desirable performance even with a smaller blind-spot during inference. This comparison is shown in Figure 8.

Empirically, in the case of SIDD, where removing strong noise is more crucial than restoring high-frequency details, the best performance is observed at a relatively larger shift-factor s, while the highest performance gain is seen at $s = 1$. On the other hand, in the case of DND, where restoring fine details is more important, the best performance is observed at $s = 1$. Taking into account that noise is more correlated between nearby pixels, the improved input condition during inference showcases the benefits of APR training, which uses more decorrelated training pairs. It is important to note that the optimal inference spot of APR may vary in each



(a) SIDD validation (b) DND

Figure 8: Results of the SIDD validation and the DND benchmark based on the shift factor for inference. The performance improvement of APR is particularly noticeable with relatively smaller inference factors.

λ	PSNR	SSIM
0	37.091	0.9332
1	37.155	0.9347
2	37.147	0.9326
4	**37.231**	**0.9355**
8	37.142	0.9353
16	37.119	0.9349
32	37.063	0.9351

Table 2: Ablation study of the regularization factor on the SIDD validation.

test dataset. However, our approach ultimately aims to unify them into a single optimal condition through RD.

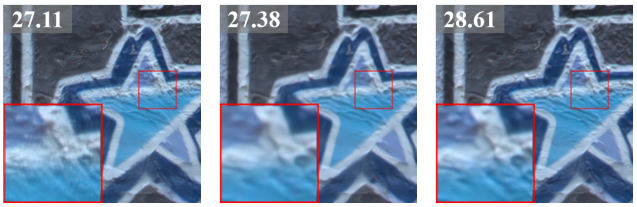
Ablations on Regularization Factor. NBR2NBR has already reported the trade-off between accurate GT estimation and noise contamination depending on the value of λ. However, since we use a different network, sampling method, and task domain compared to theirs, we investigated the optimal value of λ for our setup. Table 2 demonstrates that the optimization of simple reconstruction ($\lambda = 0$) does not yield optimal performance due to GT distortion. It is observed that optimal performance is achieved when $\lambda = 4$. After this point, performance gradually declines due to increased noise interference.

Analysis of RD

Effect of Recharging on Distillation Process. We compared the results of two distillation methods, D and RD, one trained with and the other without the recharging process. According to the principle of self-supervision in denoising, f_D distilled with T_{RD} can access the distribution of clean signals. This results in improved texture and sharpness in the output. As a result, even though APR may exhibit insufficient expressive capability in some samples, our approach avoids simply inheriting it and enhances clean details. These improvements are demonstrated in Table 3 and Figure 9. The higher quantitative results demonstrate that the visual improvement achieved by our recharging method is not merely a hallucination.

Methods	SIDD Validation	DND Benchmark
APR(D)	37.896/0.9450	38.161/0.9347
APR(RD)	**37.999/0.9465**	**38.573/0.9421**

Table 3: Quantitative comparison of each distillation result. D and RD use T and T_{RD} as their multi-targets, respectively.



(a) APR (b) APR (D) (c) APR (RD)

Figure 9: Visual and quantitative examples of APR and distillation results.

T_{RD}	SIDD Validation	DND Benchmark
$\{\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, .., \tilde{\mathbf{x}}_3\}$	37.77/0.944	38.55/**0.943**
$\{\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, .., \tilde{\mathbf{x}}_4\}$	37.93/0.946	38.56/0.942
$\{\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, .., \tilde{\mathbf{x}}_5\}$	**38.00/0.947**	**38.57**/0.942
$\{\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, .., \tilde{\mathbf{x}}_6\}$	37.98/0.946	38.38/0.938
$\{\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, .., \tilde{\mathbf{x}}_7\}$	37.97/0.946	38.35/0.939

Table 4: Ablation study of T_{RD} on the SIDD validation and the DND benchmark.

Ablations on Setting of RD. We conducted a study on the T_{RD} setting used in our RD. We observe another advantage of RD besides its fine restoration capability. RD addresses the impracticality of the previous distillation D, which requires separate training for each evaluation. Table 4 demonstrates that a single setting of $T_{RD} = \{\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, .., \tilde{\mathbf{x}}_5\}$ can achieve uniformly high performance across both evaluations. In other words, RD can mitigate the bias issue in different noise domains by enabling f_D to further estimate the original GT. This indicates that sequential training of our proposed two steps, APR and RD, is a complementary completion that offers better generalization ability.

Conclusion

We have developed the APR, which enhances the network's denoising capabilities, and integrated it with the RD, which focuses on restoring clean details. Although each method has a different primary objective, they complement each other's weaknesses effectively. Our approach utilizes an efficient data augmentation method grounded in the philosophy of self-supervised denoising, requiring no modifications to the network. As a result, it can be compatible with other self-supervised denoising architectures as well.

Acknowledgments

This work was supported in part by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) [NO.RS-2021-II211343, Artificial Intelligence Graduate School Program (Seoul National University)], and in part by [No.RS-2021-II212068, Artificial Intelligence Innovation Hub].

References

Abdelhamed, A.; Lin, S.; and Brown, M. S. 2018. A high-quality denoising dataset for smartphone cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1692–1700.

Anwar, S.; and Barnes, N. 2019. Real image denoising with feature attention. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3155–3164.

Batson, J.; and Royer, L. 2019. Noise2self: Blind denoising by self-supervision. In *International Conference on Machine Learning*, 524–533. PMLR.

Brooks, T.; Mildenhall, B.; Xue, T.; Chen, J.; Sharlet, D.; and Barron, J. T. 2019. Unprocessing images for learned raw denoising. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11036–11045.

Brummer, B.; and De Vleeschouwer, C. 2019. Natural image noise dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 0–0.

Buades, A.; Coll, B.; and Morel, J.-M. 2005. A non-local algorithm for image denoising. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 2, 60–65. Ieee.

Chatterjee, P.; Joshi, N.; Kang, S. B.; and Matsushita, Y. 2011. Noise suppression in low-light images through joint denoising and demosaicing. In *CVPR 2011*, 321–328. IEEE.

Chen, J.; Chen, J.; Chao, H.; and Yang, M. 2018. Image blind denoising with generative adversarial network based noise modeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3155–3164.

Chen, S.; Zhang, J.; Yu, Z.; and Huang, T. 2024. Exploring Efficient Asymmetric Blind-Spots for Self-Supervised Denoising in Real-World Scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2814–2823.

Chen, Y.; and Pock, T. 2016. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 39(6): 1256–1272.

Dabov, K.; Foi, A.; Katkovnik, V.; and Egiazarian, K. 2007. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8): 2080–2095.

Elad, M.; and Aharon, M. 2006. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12): 3736–3745.

Foi, A. 2009. Clipped noisy images: Heteroskedastic modeling and practical denoising. *Signal Processing*, 89(12): 2609–2629.

Gu, S.; Zhang, L.; Zuo, W.; and Feng, X. 2014. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2862–2869.

Guo, S.; Yan, Z.; Zhang, K.; Zuo, W.; and Zhang, L. 2019. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1712–1722.

Huang, T.; Li, S.; Jia, X.; Lu, H.; and Liu, J. 2021. Neighbor2neighbor: Self-supervised denoising from single noisy images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14781–14790.

Jang, G.; Lee, W.; Son, S.; and Lee, K. M. 2021. C2n: Practical generative noise modeling for real-world denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2350–2359.

Jang, H.; Park, J.; Jung, D.; Lew, J.; Bae, H.; and Yoon, S. 2024. PUCA: patch-unshuffle and channel attention for enhanced self-supervised image denoising. *Advances in Neural Information Processing Systems*, 36.

Jang, Y. I.; Lee, K.; Park, G. Y.; Kim, S.; and Cho, N. I. 2023. Self-supervised image denoising with downsampled invariance loss and conditional blind-spot network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 12196–12205.

Jin, Q.; Facciolo, G.; and Morel, J.-M. 2020. A review of an old dilemma: Demosaicking first, or denoising first? In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 514–515.

Kim, Y.; Soh, J. W.; Park, G. Y.; and Cho, N. I. 2020. Transfer learning from synthetic to real-noise denoising with adaptive instance normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3482–3492.

Krull, A.; Buchholz, T.-O.; and Jug, F. 2019. Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2129–2137.

Laine, S.; Karras, T.; Lehtinen, J.; and Aila, T. 2019. High-quality self-supervised deep image denoising. *Advances in Neural Information Processing Systems*, 32.

Lee, W.; Son, S.; and Lee, K. M. 2022. Ap-bsn: Self-supervised denoising for real-world images via asymmetric pd and blind-spot network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17725–17734.

Lehtinen, J.; Munkberg, J.; Hasselgren, J.; Laine, S.; Karras, T.; Aittala, M.; and Aila, T. 2018. Noise2Noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*.

Li, J.; Zhang, Z.; Liu, X.; Feng, C.; Wang, X.; Lei, L.; and Zuo, W. 2023. Spatially adaptive self-supervised learning for real-world image denoising. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9914–9924.

Li, J.; Zhang, Z.; and Zuo, W. 2024. TBSN: Transformer-Based Blind-Spot Network for Self-Supervised Image Denoising. *arXiv preprint arXiv:2404.07846*.

Makitalo, M.; and Foi, A. 2012. Optimal inversion of the generalized Anscombe transformation for Poisson-Gaussian noise. *IEEE transactions on image processing*, 22(1): 91–103.

Moran, N.; Schmidt, D.; Zhong, Y.; and Coady, P. 2020. Noisier2noise: Learning to denoise from unpaired noisy data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12064–12072.

Nam, S.; Hwang, Y.; Matsushita, Y.; and Kim, S. J. 2016. A holistic approach to cross-channel image noise modeling and its application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1683–1691.

Neshatavar, R.; Yavartanoo, M.; Son, S.; and Lee, K. M. 2022. Cvf-sid: Cyclic multi-variate function for self-supervised image denoising by disentangling noise from image. In *Proceedings of the ieee/cvf Conference on Computer Vision and Pattern Recognition*, 17583–17591.

Pan, Y.; Liu, X.; Liao, X.; Cao, Y.; and Ren, C. 2023. Random Sub-Samples Generation for Self-Supervised Real Image Denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 12150–12159.

Plotz, T.; and Roth, S. 2017. Benchmarking denoising algorithms with real photographs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1586–1595.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 234–241. Springer.

Tai, Y.; Yang, J.; Liu, X.; and Xu, C. 2017. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE international conference on computer vision*, 4539–4547.

Wei, K.; Fu, Y.; Yang, J.; and Huang, H. 2020. A physics-based noise formation model for extreme low-light raw denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2758–2767.

Wu, X.; Liu, M.; Cao, Y.; Ren, D.; and Zuo, W. 2020. Unpaired learning of deep image denoising. In *European conference on computer vision*, 352–368. Springer.

Xu, J.; Huang, Y.; Cheng, M.-M.; Liu, L.; Zhu, F.; Xu, Z.; and Shao, L. 2020. Noisy-as-clean: Learning self-supervised denoising from corrupted image. *IEEE Transactions on Image Processing*, 29: 9316–9329.

Yue, Z.; Yong, H.; Zhao, Q.; Meng, D.; and Zhang, L. 2019. Variational denoising network: Toward blind noise modeling and removal. *Advances in neural information processing systems*, 32.

Zamir, S. W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F. S.; and Yang, M.-H. 2022. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5728–5739.

Zhang, J.; and Hirakawa, K. 2017. Improved denoising via Poisson mixture modeling of image sensor noise. *IEEE Transactions on Image Processing*, 26(4): 1565–1578.

Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; and Zhang, L. 2017. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26(7): 3142–3155.

Zhang, K.; Zuo, W.; and Zhang, L. 2018. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Transactions on Image Processing*, 27(9): 4608–4622.

Zhang, Y.; Qin, H.; Wang, X.; and Li, H. 2021. Rethinking noise synthesis and modeling in raw denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4593–4601.

Zou, Y.; and Fu, Y. 2022. Estimating fine-grained noise model via contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12682–12691.