Masked and Shuffled Blind Spot Denoising for Real-World Images

Hamadi Chihaoui
Computer Vision Group
University of Bern, Switzerland
hamadi.chihaoui@unibe.ch

Paolo Favaro
Computer Vision Group
University of Bern, Switzerland
paolo.favaro@unibe.ch

Abstract

We introduce a novel approach to single image denoising based on the Blind Spot Denoising principle, which we call MAsked and SHuffled Blind Spot Denoising (MASH). We focus on the case of correlated noise, which often plagues real images. MASH is the result of a careful analysis to determine the relationships between the level of blindness (masking) of the input and the (unknown) noise correlation. Moreover, we introduce a shuffling technique to weaken the local correlation of noise, which in turn yields an additional denoising performance improvement. We evaluate MASH via extensive experiments on real-world noisy image datasets. We demonstrate state-of-the-art results compared to existing self-supervised denoising methods. Website: https://hamadichihaoui.github.io/mash.

1. Introduction

The removal of noise from real images, *i.e.*, image denoising, is a fundamental and still open problem in image processing despite having a long history of dedicated research (see [9] for an overview of the classic and recent methods). In classic methods, the primary strategies involve manually designing image priors and optimization techniques to enhance both reconstruction accuracy and speed. In contrast, in the context of deep learning methods, neural networks naturally introduce a very powerful prior for images [24] and provide models that could perform denoising efficiently at inference time. These innate capabilities of neural networks opened the doors to a wide range of methods that could not only learn to denoise image from examples of noisy and clean image pairs, but, even more remarkably, directly from single noisy images [12, 15, 25, 26].

In this work, we push the limits of these advanced methods one step further. We focus on the family of methods called Blind Spot Denoising (BSD)[15], since it provides a powerful and general framework. Moreover, we consider the case where only a single image is used for denoising (*i.e.*, we do not rely on a supporting dataset). As also observed by Wang et al [25], training on a dataset may not generalize well on new data, where the noise distribution is unknown. This is particularly true for real images, where noise is often correlated. In these settings, most modern methods find it challenging to handle non-iid data.

In our work, similar to the approach in [21], we explore the more general setting of random masking beyond the single blind spot method introduced in [15]. In our analysis, we uncover valuable connections between the performance of Blind Spot Denoising (BSD) methods trained with various input masking techniques and the degree of noise correlation. Surprisingly, we observe that models trained with a higher masking ratio tend to perform better when dealing with highly correlated noise, whereas models trained with a lower masking ratio excel in denoising tasks with iid noise. This discovery offers two key contributions: 1) it provides a method to estimate the unknown level of noise correlation, and 2) it offers a strategy for achieving enhanced denoising performance. Furthermore, our analysis reveals that noise correlation significantly hampers the denoising capabilities of BSD models. This suggests that a more radical approach would be to directly eliminate the correlation in the input data. An intuitive method to achieve this would involve randomly permuting all pixels that correspond to the same clean-image color intensity. However, this presents a classic chicken and egg dilemma, as we would typically need the clean image to perform the permutation, yet the clean image is precisely what we are trying to restore. To tackle this challenge, we utilize an intermediate denoised image as a pseudo-clean image to define the permutation set. Furthermore, given that adjacent pixels are likely to have similar color intensities, we focus on shuffling only pixels within small neighborhoods. We incorporate these insights into a novel method called MAsked and SHuffled Blind Spot Denoising (MASH), which we elaborate on further in Sec. 3. Our contributions are summarized as follows

- We provide an analysis of BSD, showcasing the impact of various masking ratios on correlated noise and presenting a method for estimating the noise correlation level;
- We introduce MASH, an enhanced version of BSD that

dynamically selects the optimal masking ratio. We also introduce the local pixel shuffling technique to address noise correlation at its source;
- MASH demonstrates marked enhancements over the baseline BSD and attains on par or better results in real-world denoising across multiple datasets.

2. Related Work

In this section, we only focus on work that is closely related to unsupervised image denoising.

Non-learning-based image denoisers. Classic image denoisers [4, 22, 27] manually define image priors of what a clean image is. Some approaches explore sparse representations [2, 7], while others benefit from the patch recurrence prior [32]. BM3D [6], which applies collaborative filtering to similar patches, is one of the best-known non-local methods, due to its high performance across several benchmarks. NLM [3] and WNNM [10] also relate similar patches and use them to reduce noise through a form of implicit averaging.

Unsupervised learning-based image denoisers. The unsupervised learning-based approaches can be classified into two categories based on their training data. The first category is the dataset-based one, which uses a dataset of noisy images to train a denoising model. The second category is the single-image one, which learns a denoiser from a single image at a time.

1) *Dataset-based learning approaches.*

Noise2Noise [17] shows that it is possible to train a denoising neural network without the need for clean images. Indeed, using only pairs of noisy images of the same scene as input/output training data leads to a comparable performance of standard noisy/clean image pair supervised training. Some approaches [12, 15, 25, 26] go one step further and remove the need for noisy image pairs altogether. The main contributions in these approaches are ways to avoid learning the trivial solution of the identity mapping. Recently, [31] proposed the use of pixel-shuffle downsampling (PD) to weaken the spatial correlation of structured noise. In a similar vein, AP-BSN [16] utilized two PDs with different strides for training and testing. CVF-SID [20] attempted to separate the latent image and noise from the noisy input through a cyclic module and self-supervised losses. LUD-VAE [30] established two hidden variables for the noise domain and the latent image domain and optimize for them using unpaired data. [13, 14] explicitly learned a model for real-world noise. These methods rely on noisy or unpaired data, and generalization issues persist.

2) *Test-time training approaches.*

A learning-based image denoising approach without any a-priori dependence on training samples is the least demanding option to denoise a given noisy image. There are a few methods that use a single noisy image to train a deep neural network for denoising. The DIP [24] shows that the inductive bias of convolutional neural networks (CNN) favors the learning of noise-free texture patterns rather than noisy ones, when trying to reconstruct a degraded image. The method trains a CNN to generate a given degraded image from a random input, while early stopping is used as a regularization. DIP is highly sensitive to the choice of the stopping time and thus is not practical as a denoising method. ScoreDVI [5] recently proposes a method for denoising a single image by exploiting score priors embedded in MMSE. [29] addresses the complexity of real noise by mapping the noisy image into a latent space in which the additive white Gaussian noise (AWGN) assumption holds. An encoder-decoder is used for the mapping and an off-the-shelf Gaussian denoiser is used to denoise the encoded image. Finally the image is decoded back to the original space. [21] is a self-supervised approach that use pairs of Bernoulli-sampled instances of the input noisy image to train a denoiser with dropout. The denoiser predicts the masked pixels based on the visible ones. The final denoised image is the average of the predictions generated from multiple instances of the trained model with dropout.

MASH belongs to the category of BSD methods applied to a single noisy image. However, in contrast to the above methods, it uses an optimal masking ratio that is automatically estimated and introduced the technique of local pixel shuffling. To the best of our knowledge, these contributions have not been presented before.

3. Unsupervised Single Image Denoising

In this section, we begin by revisiting the concept of BSD as introduced in a prior work by Krull et al. [15]. Subsequently, we conduct experimental analyses to delve deeper into understanding the effects of various design factors in the BSD method on its performance. For this purpose, we generate a synthetic noisy dataset with noise ranging from independent and identically distributed (iid) to highly correlated, using which we train a base blind spot network with differing levels of ”blindness” (i.e., image masking ratios). These empirical observations lead us to create a novel self-supervised blind-spot framework for unsupervised single image denoising, referred to as MAsked and SHuffled Blind Spot Denoising (MASH), which achieves state-of-the-art performance.

3.1. Revisiting Blind Spot Denoising (BSD)

BSD operates as a self-supervised technique, meaning that its training process does not necessitate pairs of noisy images or pairs of noisy and clean images. Instead, only noisy images are employed. The method applies a masking scheme that hides part of the image at the input and then aims to predict the same hidden part at the output. In this paper, we apply this idea to a single image at a time.

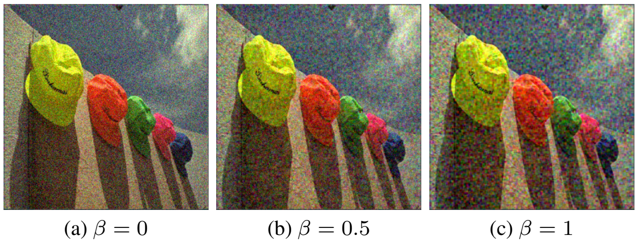

(a) $\beta = 0$ (b) $\beta = 0.5$ (c) $\beta = 1$

Figure 1. Samples of generated noisy images depending on the spatial correlation level. From left to right: noisy image with iid noise, noisy image with moderately correlated noise and noisy image with heavily correlated noise

Formally, considering the noisy observation $\mathbf{y} \in \mathbb{R}^{H\times W\times C}$ corresponding to the clean image $\mathbf{x} \in \mathbb{R}^{H\times W\times C}$, where $H \times W$ represents the image dimensions and C denotes the color channels (usually 1 for grayscale or 3 for RGB), the objective of BSD is to minimize the following empirical risk:

$$\arg\min_{\theta} \sum_{\mathbf{i}\in\Omega} \|f_\theta(\mathbf{y}_{\text{masked}(\mathbf{i})})[\mathbf{i}] - \mathbf{y}[\mathbf{i}]\|_2^2 \tag{1}$$

In Eq. (1), $f_\theta : \mathbb{R}^{H\times W\times C} \rightarrow \mathbb{R}^{H\times W\times C}$ represents the denoising model realized through neural networks with parameters θ, which transforms noisy images into denoised versions. Here, $\mathbf{y}_{\text{masked}(\mathbf{i})}$ denotes the image $\mathbf{y}$ where the pixel $\mathbf{i} \in \Omega$ ($\Omega = [1, \ldots, H] \times [1, \ldots, W]$), has been masked. Additionally, $\mathbf{y}[\mathbf{i}]$ refers to the RGB color at the pixel $\mathbf{i}$. Notice that the above formulation is used for a single image $\mathbf{y}$. It is also possible to extend it to a dataset of images simply by taking the expectation of the above loss with respect to the distribution of $\mathbf{y}$. However, as already mentioned, in this paper we only consider denoising a single image without additional data. In our terminology, we also refer to *signal* when talking about the clean image. BSD assumes that noise at each pixel is statistically independent of its neighbors (thus a noise instance cannot be predicted by nearby noise), while a pixel of the signal (*i.e.*, the clean image) is spatially correlated to its neighboring pixels. These assumptions are key in enabling the separation of noise from signal. More in general, the BSD method can also be seen as an extreme case of masking or sparse image reconstruction/inpainting such as MAE [11], where the masking consists of a single pixel. In practice, the BSD method avoids trivial solutions, such as the identity mapping (simply reconstructing the noisy image), only as long as f_θ does not overfit the data. This is particularly important in the context of single image denoising, where the training dataset is limited in size. Therefore, considering that the degree of masking may have a regularizing effect on f_θ, we investigate the influence of this design choice on single image denoising.

3.2. Diving Deep into Blind Spot Denoising

In this section, our goal is to empirically evaluate the performance of the BSD method by examining the effects of two key factors: 1) the masking ratio used in the BSD method and 2) the characteristics of the noise (specifically, its level of spatial correlation). To conduct this analysis, we generate a denoising dataset synthetically using images from the Kodak dataset [8]. We model the noise using a multivariate normal distribution with a variance-covariance matrix Σ to simulate real-world correlated noise. The correlation $\Sigma[\mathbf{i},\mathbf{j}]$ between pixels $\mathbf{i}$ and $\mathbf{j}$ in the set Ω is defined as follows:

$$\Sigma[\mathbf{i},\mathbf{j}] = \begin{cases} \sigma^2 & \mathbf{i} = \mathbf{j} \\ \beta^{\frac{k-\|\mathbf{i}-\mathbf{j}\|}{k}}\sigma^2 & 0 < \|\mathbf{i}-\mathbf{j}\| \leq \mathrm{k} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $\|\mathbf{i}-\mathbf{j}\|$ is the distance between the pixels $\mathbf{i}$ and $\mathbf{j}$ and $k > 0$ denotes the correlation kernel width. The parameter $\beta > 0$ is used to control the level of spatial correlation in the noise. A higher value of β indicates stronger correlation in the noise. We define three distinct regimes based on the noise correlation: 1) The *iid regime*, where $\beta = 0$, and the noise is independent and identically distributed (iid); 2) The *moderately correlated regime*, where $\beta = 0.5$; 3) The *heavily correlated regime*, where $\beta = 1.0$; We set $k = 3$ and $\sigma = 25$. An extension to this analysis, considering lower noise levels ($\sigma = 15$) and higher noise levels ($\sigma = 40$), is provided in the supplementary material due to space constraints. In this extension, we draw similar conclusions to those obtained for $\sigma = 25$. Fig. 1 illustrates an example of an image with synthetically generated noise for each of the defined regimes.

We define the *blindness mask* $\mathbf{m} : \Omega \mapsto \{0,1\}^{H\times W\times C}$, where Ω represents the set of pixels in our noisy image $\mathbf{y}$.

$$\mathbf{m}[\mathbf{i}] = \begin{cases} 0 & \text{with probability } \tau \text{ ;} \\ 1 & \text{with probability } 1-\tau \text{ .} \end{cases} \tag{3}$$

$\mathbf{m}$ is of the same size as the input images and is determined by a masking ratio parameter τ. The masking ratio τ represents the proportion of zeros in the mask $\mathbf{m}$, calculated as the number of zeros divided by the total number of pixels in the image (HWC). We adopt a similar BSD formulation as [21] with the blindness mask via the following loss

$$L(\theta) = \mathbb{E}_{\mathbf{m}}\left[\|(\mathbf{1}-\mathbf{m}) \odot (f_\theta(\mathbf{y}\odot\mathbf{m}) - \mathbf{y})\|_2^2\right] \tag{4}$$

where $\mathbb{E}_{\mathbf{m}}$ denotes the expectation with respect to the mask $\mathbf{m}$ and $\odot$ denotes the element-wise product.

3.2.1 Impact of the masking ratio on BSD performance

We investigate the impact of the masking ratio τ on the denoising performance depending on the correlation mag-

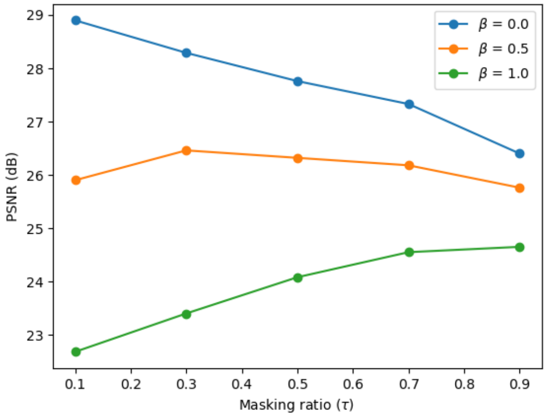


Figure 2. Impact of the masking ratio τ on the generalized BSD denoising performance (PSNR). On the horizontal axis we consider several masking ratios τ and for each we train a BSD model on data with different levels of correlation β. The optimal performance of the trained model shows a strong correlation between the masking ratio an the noise correlation. Low masking benefits the training on data with iid noise and high masking benefits the training on data with highly correlated noise.

nitude controlled by β. We train a denoising network by minimizing the loss (4) with different masking levels in the three noise regimes (*i.e*., with different correlation magnitudes β). The results are shown in Fig. 2. We plot the Peak Signal-to-Noise Ratio (PSNR) values in decibels across the three regimes for various masking configurations (displayed on the x-axis). We notice that the effectiveness of our generalized BSD is greatly influenced by: 1) the correlation of the noise, and 2) the masking ratio. Specifically, in the case of independent and identically distributed (iid) noise, we discovered that a lower masking ratio results in the best performance. Conversely, in scenarios where the noise exhibits high correlation, a higher masking ratio produces superior performance. In situations of moderate correlation, an intermediate masking ratio (ranging from 0.3 to 0.5) is deemed optimal. As observed in the analysis of DIP [24] a neural network model, such as f_θ, tends to first fit clean image patterns and then noise patterns. In scenarios of high spatial noise correlation, the noise tends to exhibit patterns resembling textures present in clean images (refer to the rightmost image in Fig. 1). Consequently, in cases of increased noise correlation, the efficacy of BSD could potentially improve with greater regularization, *i.e*., a higher masking ratio τ.

3.2.2 Handling correlation with local pixel shuffling

By comparing the top performances of BSD under each noise regime in Fig. 2, we can readily observe a significant decrease in performance in the highly correlated noise scenario compared to the iid case, despite both scenarios having the same noise level. This observation prompts the consideration of another strategy to enhance performance in the presence of correlated noise. The idea is to reduce noise correlation without altering the underlying image signal. One intuitive approach is to identify sets of noisy pixels corresponding to identical colors in the clean image and then randomly permute them. By doing so, we can disrupt the spatial correlation of these pixels without affecting the clean image structure. A practical method to implement this concept is to utilize the predicted denoised image as a pseudo-clean image. This pseudo-clean image can be leveraged to identify sets of iso-intensity pixels. Since neighboring pixels are more likely to share similar intensities, the swapping process can be performed locally to maintain coherence in the image. To implement this idea, we first divide the image into two categories of regions: regions with nearly constant intensity and regions with texture. We define the image partition denoted as Ω_{const} as the set of pixels where the local neighborhoods, such as those defined by 4×4 pixel blocks, exhibit similar or identical color intensity values. The remaining partition is categorized with texture (significant variation in intensity patterns).

We define $\mathbf{c}(\mathbf{y})$ as the mapping that assigns a value of 1 to the pixels within the constant intensity regions (Ω_{const}) of $\mathbf{y}$ and a value of 0 to all other pixels within the image domain Ω. $\mathbf{c}(\mathbf{y})$ helps to differentiate between the constant intensity regions and the textured regions within $\mathbf{y}$. The illustration in Fig. 4 provides a visual representation of this partitioning concept. Let $\Gamma(\mathbf{y})$ define the local random permutation of pixels within $s \times s$ (*e.g*., $s = 4$) tiles of $\mathbf{y}$. We

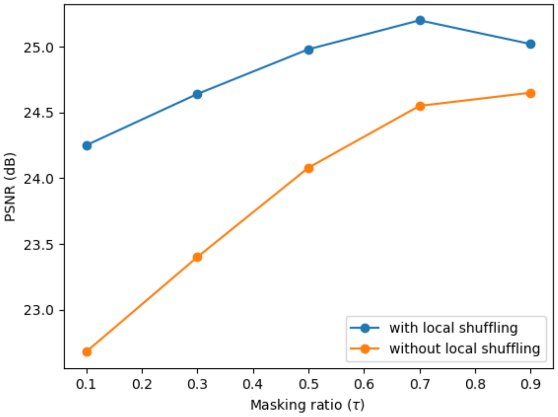


Figure 3. Impact of the local pixel shuffling on the denoising performance (PSNR) when noise is highly correlated. The shuffling of image regions that are approximately constant destroys the noise correlation. This brings a consistent benefit across all masking ratios.

note that the pixel shuffling is performed only within each tile. We now define the shuffled noisy image $\mathbf{y}^{\text{shuffled}}$ as:

$$\mathbf{y}^{\text{shuffled}} = \mathbf{c}(\mathbf{y}) \odot \Gamma(\mathbf{y}) + (\mathbf{1} - \mathbf{c}(\mathbf{y})) \odot \mathbf{y} \tag{5}$$

Then, the shuffled image can serve as the target for the BSD loss in Eq. (4). We call this decorrelation technique Local Pixel Shuffling (LPS). Now we are ready to define the loss of MASH to further reduce the model overfitting caused by highly correlated noise in the BSD approach:

$$L(\theta) = \mathbb{E}_{\mathbf{m}} \left[\|(\mathbf{1} - \mathbf{m}) \odot (f_\theta(\mathbf{y} \odot \mathbf{m}) - \mathbf{y}^{\text{shuffled}})\|_2^2 \right] \tag{6}$$

To determine the region Ω_{const}, we adopt a similar approach to [18]. We utilize the intermediate pseudo-clean image output $\hat{\mathbf{y}}$, which we obtain from f_θ (details in the following sections) after a certain number of training iterations. An intuitive indicator of the similarity of pixels within a tile is their standard deviation $\sigma : \Omega \mapsto [0, \infty)$. Specifically, we calculate the standard deviations within patches of the size of the local tiles independently for each color channel and then average them. Finally, we can derive the partition as defined in Eq. (7), with the parameter $\lambda > 0$ serving as a threshold for the color similarity.

$$\Omega_{\text{const}} = \{\mathbf{i} : \sigma[\mathbf{i}] < \lambda\}, \tag{7}$$

Fig. 3 shows that applying the local pixel shuffling improves the denoising performance when the noise is spatially highly correlated.

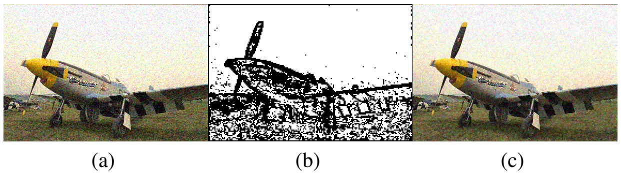


Figure 4. (a) Original noisy image (b) Mask capturing region flatness derived from pseudo-clean prediction (c) Noisy image with local pixel permutation on flat regions.

3.2.3 Automated selection of the BSD masking ratio

To make MASH of practical use, it is essential to have an automated mechanism for determining the noise correlation and, consequently, selecting the optimal masking ratio. Towards this goal, we analyze the estimated noise level predicted by our BSD method using different masking ratios. We denote by $\hat{\sigma}_\tau$ the estimated noise level of the noisy image $\mathbf{y}$ when using a masking scheme with a masking ratio τ:

$$\hat{\sigma}_\tau = \sqrt{\frac{1}{HWC} \|f_\theta(\mathbf{m} \odot \mathbf{y}) - \mathbf{y}\|_2^2}. \tag{8}$$

Fig. 5 illustrates the estimated noise level $\hat{\sigma}_\tau$ during the denoising iterations, which is dependent on the estimated parameters θ, for varying correlation regimes and masking ratios. To ensure a dependable indicator, we focus solely on the noise level estimated at convergence and we define the *noise level estimation gap* ε as the difference in noise levels when utilizing a low masking ratio τ^{low} and a high masking ratio τ^{high}:

$$\varepsilon = |\hat{\sigma}_{\tau^{\text{high}}} - \hat{\sigma}_{\tau^{\text{low}}}| \tag{9}$$

We set $\tau^{\text{low}} = 0.2$ and $\tau^{\text{high}} = 0.8$. We experimentally verify that ε is proportional to the level of correlation in the noise. Therefore, we can use ε as a proxy for assessing the degree of spatial noise correlation present in the input image $\mathbf{y}$.

3.3. MASH

As a conclusion of our prior experimental analysis, we propose to integrate the adaptive masking and local pixel shuffling in the BSD approach. The pseudo-code of our method (MASH) is described in Algorithm 1. We begin by training a model f_θ using two different masking ratios: a high one (τ^{high}) and a low one (τ^{low}). Subsequently, we calculate the noise level estimation gap ε as outlined in Eq. (9). Based on the value of ε, we dynamically determine the masking ratio τ to utilize, as well as whether to activate the local pixel shuffling. Our automated selection of the optimal masking ratio is depicted in Eq. (10).

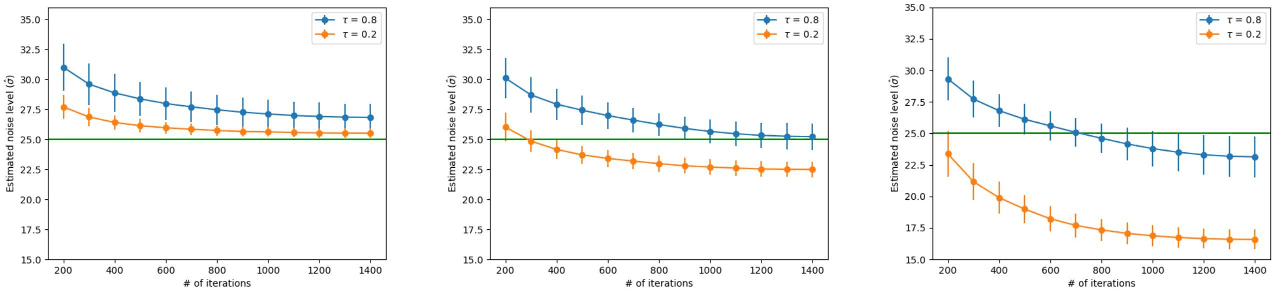
$$\tau^{\text{optimal}} = \begin{cases} \tau^{\text{low}} & \text{if } \varepsilon \leq \varepsilon^{\text{low}}; \\ \tau^{\text{medium}} & \text{if } \varepsilon^{\text{low}} < \varepsilon < \varepsilon^{\text{high}}; \\ \tau^{\text{high}} & \text{if } \varepsilon^{\text{high}} < \varepsilon. \end{cases} \tag{10}$$

If $\varepsilon \leq \varepsilon^{\text{high}}$, we do not apply the local pixel shuffling (indicating low noise correlation) and directly optimize the loss (4). If $\varepsilon^{\text{high}} < \varepsilon$ (implying highly correlated noise), we optimize the loss (4) for N_1 iterations. Subsequently, we determine the partition Ω_{const}. We then apply local random permutation Γ within Ω_{const} and compute $\mathbf{y}^{\text{shuffled}}$. Finally, we resume training using the loss (6). The recovered image $\hat{\mathbf{y}}$ is an ensemble of K predictions. To get each prediction, we sample a random binary mask $\mathbf{m}_\mathbf{p}$ with a masking ratio τ^{optimal} and apply it to the input image. We set $K = 10$.

$$\hat{\mathbf{y}} = \frac{1}{K} \sum_{p=1}^{K} f_\theta(\mathbf{m}_\mathbf{p} \odot \mathbf{y}). \tag{11}$$

4. Experiments

In this section, we will first introduce the experimental settings. We will then present the quantitative and qualitative results of MASH, along with comparisons with other methods.



$\beta = 0.0$ $\beta = 0.5$ $\beta = 1.0$

Figure 5. Estimated noise level based on different correlated noise magnitude and masking ratios.

Algorithm 1 MASH

```
Require: Noisy image $\mathbf{y}$, $f_\theta$,$\tau^{\text{high}}$, $\tau^{\text{medium}}$, $\tau^{\text{low}}$, $\varepsilon^{\text{low}}$, $\varepsilon^{\text{high}}$, $N$ and $N_1 < N$
Ensure: Restored image $\hat{\mathbf{y}}$
Apply the BSD baseline with $\tau^{\text{high}}$ and $\tau^{\text{low}}$ respectively.
Compute $\varepsilon$ from eq. (9)
$\begin{cases} \text{if } \varepsilon < \varepsilon^{\text{low}} & \text{then } \tau^{\text{optimal}} = \tau^{\text{low}} \\ \text{if } \varepsilon^{\text{low}} \leq \varepsilon < \varepsilon^{\text{high}} & \text{then } \tau^{\text{optimal}} = \tau^{\text{medium}} \\ \text{if } \varepsilon^{\text{high}} \leq \varepsilon & \text{then } \tau^{\text{optimal}} = \tau^{\text{high}} \end{cases}$
if $\varepsilon < \varepsilon^{\text{high}}$ then
    for $t : 1 \to N$ do
        Update $f_\theta$ by optimizing eq. (4) using $\tau^{\text{optimal}}$
    end for
else
    for $t : 1 \to N_1$ do
        Update $f_\theta$ by optimizing eq. (4) using $\tau^{\text{optimal}}$
    end for
    compute the partition $\Omega_{\text{const}}$ using eq. (7)
    compute $\mathbf{y}^{\text{shuffled}}$ using eq. (5)
    for $t : N_1 \to N$ do
        Update $f_\theta$ by optimizing eq. (6) using $\tau^{\text{optimal}}$
    end for
end if
Return the restored image $\hat{\mathbf{y}} = \frac{1}{K}\sum_{p=1}^{K} f_\theta(\mathbf{m_p} \odot \mathbf{y})$
```

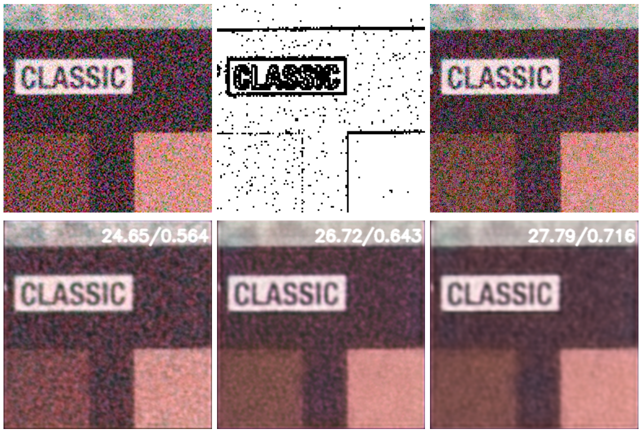


Figure 6. Top: from left to right: original noisy image, Mask capturing region flatness derived from pseudo-clean, Shuffled noisy image (using LPS). Bottom: from left to right: Result using the baseline ($\tau = 0.5$), Result without local pixel shuffling , Ours.

4.1. Experimental settings

Datasets We evaluated our method on four widely-used real-world noise datasets: SIDD (validation and benchmark datasets) [1], FMDD [28], and PolyU [19]. The validation and benchmark datasets of SIDD contain natural sRGB images captured by smartphones, with each dataset consisting of 1280 patches sized $3 \times 256 \times 256$. FMDD contains fluorescence microscopy images with a size of 512×512. The PolyU dataset consists of 100 natural images taken from diverse commercial camera brands, with each image having a size of $3 \times 512 \times 512$.

Implementation details The network architecture for MASH is the same as in Noise2Noise [17]. The denoising network is trained from scratch using the Adam optimizer with cosine annealing. By default, we use the following hyperparameters in our implementation unless otherwise specified: $\tau^{\text{high}} = 0.8$, $\tau^{\text{low}} = 0.2$, $\tau^{\text{medium}} = 0.5$, $\varepsilon^{\text{low}} = 1.5$, $\varepsilon^{\text{high}} = 2.5$, $s = 4$ and $N = 800$. In our supplementary material, we provide a more in-depth discussion on the selection of hyperparameters.

4.2. Evaluation on real-world noise

In our evaluation, we compare our method against several single image-based denoising methods, including Self2Self [21], NN+denoiser [29], DIP [24], BM3D [6], PD-denoising [31], NN+denoiser [29], scoreDVI [5], and APBSN-single [16]. For APBSN-single, we adapt the APBSN method from [16] to directly denoise a single image. The strides of PD in training and testing are 5 and 2, respectively. For NN+denoiser [29], we use its best version, which is NN+BM3D for single image denoising. For the other methods, we either use the authors' code or directly adopt their published results if available. We also include a comparison with a baseline method (denoted as Baseline in Table) which is a blind spot method with the same network architecture as our method but with a fixed

Category	Method	SIDD Validation	SIDD Benchmarck	FMDD	PolyU
Single Image (test-time training)	BM3D [6]	25.65/0.475	25.65/0.685	30.06/0.771	37.40/0.953
	DIP [24]	32.11/0.740	-	32.90/0.854	37.17/0.912
	Self2Self [21]	29.46/0.595	29.51/0.651	30.76/0.695	37.52/0.926
	PD-denoising [31]	33.97/0.820	33.61/0.894	33.01/0.856	37.04/0.940
	NN+denoiser [29]	-	33.18/0.895	32.21/0.831	37.66/0.956
	APBSN-single [16]	30.90/0.818	30.71/0.869	28.43/0.804	29.61/0.897
	ScoreDVI [5]	34.75/**0.856**	34.60/0.920	33.10/0.865	**37.77/0.959**
	Baseline	33.12/0.805	32.67/0.850	32.25/0.824	37.12/0.911
	Ours	**35.06**/0.851	**34.78**/0.900	**33.71/0.882**	37.62/0.932
Noisy/Impaired Dataset	APBSN [16]	-	**36.91/0.931**	31.99/0.836	**37.03/0.951**
	CVF-SID [20]	34.81/0.944	34.71/0.917	**32.73/0.843**	35.86/0.937
	LUD-VAE [30]	**34.91/0.944**	34.82/0.926	-	36.99/0.955
Supervised	DnCNN [23]	37.73 / 0.943	37.61 / 0.941	-	-

Table 1. Quantitative comparisons (PSNR(dB)/SSIM) of our method and other real-world denoising methods including single image-based methods and dataset-based methods on SIDD, FMDD, PolyU datasets. The best results of the unsupervised approaches are marked in **bold**, while the second best ones are underlined.

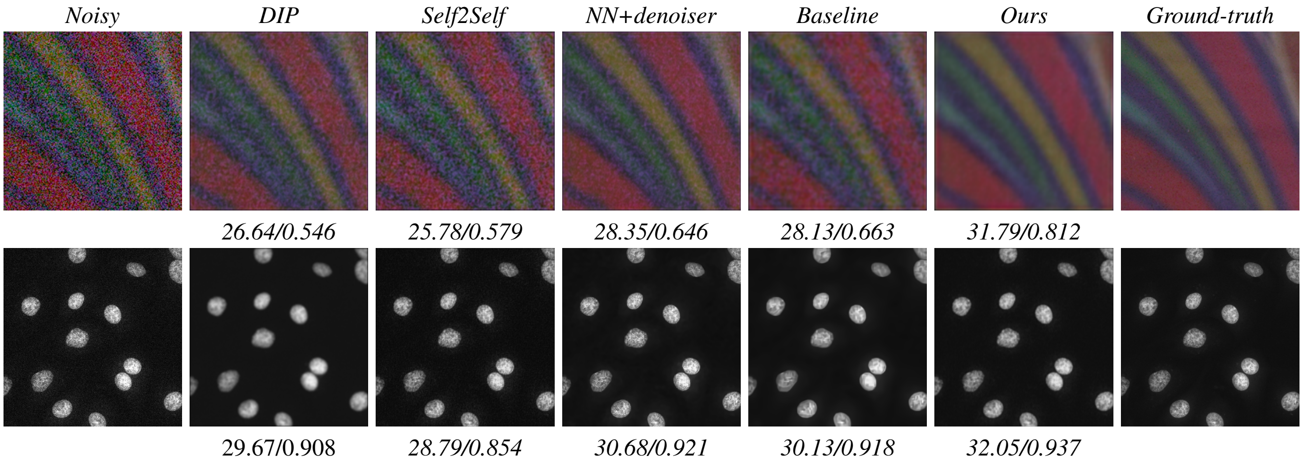


Figure 7. Visual comparison of our method against other single image-based denoising methods in SIDD validation and FMDD datasets. The PSNR/SSIM results are reported under each image.

masking ratio of $\tau = 0.5$. Additionally, we compare against dataset-based methods including CVF-SID [20], LUD-VAE [30], and APBSN [16]. We also provide a reference comparison with the supervised DNCNN [23]. The quantitative comparisons are summarized in Tab. 1. Qualitative comparisons among different single image-based methods on SIDD and FMDD are displayed in Fig. 7. Additional visual comparisons are included in the supplementary material. MASH shows a significant boost over the baseline, with an improvement of about 2 dB for both SIDD datasets and 1.5 dB for the FMDD dataset, highlighting the importance of our adaptive masking scheme and local pixel shuffling. MASH yields competitive results compared to existing single image-based methods. Our method excels in the FMDD dataset by outperforming both the single-image and dataset-based methods, which encompass images with varying noise levels and correlations. Fig. 6 shows the output of MASH on an image from the SIDD validation dataset. Utilizing our adaptive masking scheme and pixel local shuffling results in a significant improvement over the baseline.

	SIDD	FMDD
Adaptive masking accuracy	88.7 %	92.4 %

Table 2. Adaptive masking accuracy on SIDD and FMDD datasets.

Adaptive masking	Local pixel shuffling	SIDD	FMDD
No	No	33.12	32.25
No	Yes	33.86	32.92
Yes	No	34.45	33.56
Yes	Yes	35.06	33.71

Table 3. Ablation of MASH components .

5. Ablations

We perform ablation studies to analyze the impact of each component of our method.

5.1. Influence of Adaptive Masking

We evaluate the effectiveness of our adaptive masking scheme. When the adaptive masking is not applied, we utilize a default masking ratio of $\tau = 0.5$. Tab. 3 demonstrates that adaptive masking yields a notable performance enhancement for both the SIDD and FMDD datasets. We further analyze the effects on each image individually by calculating the success rate of our adaptive masking scheme. To do this, we denoise each image using the baseline with three masking ratios: $\tau^{\text{high}} = 0.8$, $\tau^{\text{medium}} = 0.5$, and $\tau^{\text{low}} = 0.2$, and we store the best result. Then, we denoise each image using MASH. In Tab. 2, we show how many times our algorithm succeeds in selecting the optimal masking ratio that leads to the best performance. Our adaptive masking is considered successful if it corresponds to the masking ratio that results in optimal performance. Tab. 2 indicates that our adaptive masking succeeds in selecting the optimal masking ratio in approximately 90% of cases. Further analysis on the robustness of MASH hyperparameters and showcasing of failure cases can be found in our Supplementary materials.

5.2. Influence of Local Pixel Shuffling

We validate the significance of local pixel shuffling in our method. As shown in Tab. 3, there is an improvement of approximately 0.7 dB for both the SIDD and FMDD datasets when applying local pixel shuffling.

5.3. Influence of the neighborhood size s

Tab. 4 shows an ablation of the neighborhood size s conducted on the SIDD valadition dataset. As s increases, the performance improves then it starts to saturate. We note that s is an hyperparameter related to the local pixels shuffling which is applied only if a high spatial correlated noise is detected.

	$s = 2$	$s = 3$	$s = 4$	$s = 5$	$s = 6$
PSNR	34.65	34.85	35.06	35.15	35.09

Table 4. Ablation of the neighborhood size s on the SIDD validation dataset.

Method	Infer. time (s)	Params (M)	FLOPs (G)
DIP	146.2	13.4	31.06
Self2Self	3546.5	1.0	9.55
NN+denoiser	897.6	13.4	31.06
APBSN-single	121.4	3.66	234.63
ScoreDVI	81.2	13.5	37.87
Baseline	24.6	0.99	11.44
Ours	75.3	0.99	11.44

Table 5. Efficiency comparisons of deep learning-based methods under the input size 256 × 256 × 3.

5.4. Computational Efficiency

We provide an analysis of the inference time, model parameters and floating-point operations per second (FLOPs) of the compared deep learning-based methods. According to the results in Tab. 1 and Tab. 5, our method demonstrates a balance between effectiveness and efficiency. It is worth noting that there is room for speed enhancement with MASH. For example, the initial training phase to determine the optimal masking could be performed in parallel rather than sequentially.

6. Conclusion

We have introduced MASH, a single image denoising method that leverages the blind spot denoising framework. Our approach includes an analysis to detect and alleviate the impact of noise correlation. We demonstrated that the masking ratio plays a critical role in denoising performance, especially in the presence of correlated noise. Building on this analysis, we proposed a method to automatically estimate the level of noise correlation. Additionally, we introduced a technique to directly de-correlate noise in the input image by shuffling pixels with similar denoised color intensities. As a result, our method, MASH, achieves state-of-the-art results compared to existing test-time training approaches across various public benchmarks.

Acknowledgements We acknowledge the support of the SNF project number 200020_200304.

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S. Brown. A high-quality denoising dataset for smartphone cameras. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 6

[2] Jian-Feng Cai Bao, Chenglong and Hui Ji. Fast sparsity-based orthogonal dictionary learning for image restoration. In *Proceedings of the IEEE International Conference on Computer Vision*, 2013. 2

[3] Bartomeu Coll Buades, Antoni and Jean-Michel Morel. Non-local means denoising. In *Image Processing On Line 1*, 2011. 2

[4] Harold C Burger, Christian J Schuler, and Stefan Harmeling. Image denoising: Can plain neural networks compete with bm3d? In *2012 IEEE conference on computer vision and pattern recognition*, pages 2392–2399. IEEE, 2012. 2

[5] Jun Cheng, Tao Liu, and Shan Tan. Score priors guided deep variational inference for unsupervised real-world single image denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12937–12948, 2023. 2, 6, 7

[6] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Bm3d image denoising with shape-adaptive principal component analysis. In *SPARS'09-Signal Processing with Adaptive Sparse Structured Representations*, 2009. 2, 6, 7

[7] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. In *IEEE Transactions on Image processing 15.12*, 2006. 2

[8] Rich Franzen. Kodak lossless true color image suite. *source: http://r0k. us/graphics/kodak*, 4(2):9, 1999. 3

[9] Shuhang Gu and Radu Timofte. A brief review of image denoising algorithms and beyond. *Inpainting and Denoising Challenges*, pages 1–21, 2019. 1

[10] Zhang L. Zuo W. Feng X. Gu, S. Weighted nuclear norm minimization with application to image denoising. In *In Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014. 2

[11] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 3

[12] Tao Huang, Songjiang Li, Xu Jia, Huchuan Lu, and Jianzhuang Liu. Neighbor2neighbor: Self-supervised denoising from single noisy images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14781–14790, 2021. 1, 2

[13] Geonwoon Jang, Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. C2n: Practical generative noise modeling for real-world denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2350–2359, 2021. 2

[14] Shayan Kousha, Ali Maleky, Michael S Brown, and Marcus A Brubaker. Modeling srgb camera noise with normalizing flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17463–17471, 2022. 2

[15] Tim-Oliver Buchholz Krull, Alexander and Florian Jug. Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019. 1, 2

[16] Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. Apbsn: 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*, pages 17725–17734, 2022. 2, 6, 7

[17] Munkberg-J. Hasselgren J. Laine S. Karras T. Aittala M. Aila T. Lehtinen, J. Noise2noise: Learning image restoration without clean data. In *arXiv preprint arXiv:1803.04189*, 2018. 2, 6

[18] Junyi Li, Zhilu Zhang, Xiaoyu Liu, Chaoyu Feng, Xiaotao Wang, Lei Lei, and Wangmeng Zuo. Spatially adaptive self-supervised learning for real-world image denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9914–9924, 2023. 5

[19] Seonghyeon Nam, Youngbae Hwang, Yasuyuki Matsushita, and Seon Joo Kim. 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*, pages 1683–1691, 2016. 6

[20] Reyhaneh Neshatavar, Mohsen Yavartanoo, Sanghyun Son, and Kyoung Mu Lee. 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*, pages 17583–17591, 2022. 2, 7

[21] Yuhui Quan, Mingqin Chen, Tongyao Pang, and Hui Ji. Self2self with dropout: Learning self-supervised denoising from single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1890–1898, 2020. 1, 2, 3, 6, 7

[22] Hossein Talebi and Peyman Milanfar. Global image denoising. *IEEE Transactions on Image Processing*, 23(2):755–768, 2013. 2

[23] Rini Smita Thakur, Ram Narayan Yadav, and Lalita Gupta. State-of-art analysis of image denoising methods using convolutional neural networks. *IET Image Processing*, 13(13): 2367–2380, 2019. 7

[24] Andrea Vedaldi Ulyanov, Dmitry and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018. 1, 2, 4, 6, 7

[25] Zejin Wang, Jiazheng Liu, Guoqing Li, and Hua Han. Blind2unblind: Self-supervised image denoising with visible blind spots. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2027–2036, 2022. 1, 2

[26] Zhengyang Wang Xie, Yaochen and Shuiwang Ji. Noise2same: Optimizing a self-supervised bound for image denoising. In *Advances in Neural Information Processing Systems 33*, 2020. 1, 2

[27] Lei Zhang, Weisheng Dong, David Zhang, and Guangming Shi. Two-stage image denoising by principal component

analysis with local pixel grouping. *Pattern recognition*, 43 (4):1531–1549, 2010. 2

[28] Yide Zhang, Yinhao Zhu, Evan Nichols, Qingfei Wang, Siyuan Zhang, Cody Smith, and Scott Howard. A poisson-gaussian denoising dataset with real fluorescence microscopy images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11710–11718, 2019. 6

[29] Dihan Zheng, Sia Huat Tan, Xiaowen Zhang, Zuoqiang Shi, Kaisheng Ma, and Chenglong Bao. An unsupervised deep learning approach for real-world image denoising. In *International Conference on Learning Representations*, 2020. 2, 6, 7

[30] Dihan Zheng, Xiaowen Zhang, Kaisheng Ma, and Chenglong Bao. Learn from unpaired data for image restoration: A variational bayes approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5889–5903, 2022. 2, 7

[31] Yuqian Zhou, Jianbo Jiao, Haibin Huang, Yang Wang, Jue Wang, Honghui Shi, and Thomas Huang. When awgn-based denoiser meets real noises. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13074–13081, 2020. 2, 6, 7

[32] Maria Zontak, Inbar Mosseri, and Michal Irani. Separating signal from noise using patch recurrence across scales. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1195–1202, 2013. 2