DOI: 10.1111/cgf.14680
Pacific Graphics 2022
N. Umetani, E. Vouga, and C. Wojtan
(Guest Editors)

Volume 41 (2022), Number 7

Learning Multi-Scale Deep Image Prior for High-Quality Unsupervised Image Denoising

Hao Jiang[1], Qing Zhang[*,1], Yongwei Nie[2], Lei Zhu[3], and Wei-Shi Zheng[1]

[1]School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
[2]School of Computer Science and Engineering, South China University of Technology, Guangzhou, China
[3] Thrust of Robotics and Autonomous Systems (ROAS), The Hong Kong University of Science and Technology (Guangzhou), China

Abstract

Recent methods on image denoising have achieved remarkable progress, benefiting mostly from supervised learning on massive noisy/clean image pairs and unsupervised learning on external noisy images. However, due to the domain gap between the training and testing images, these methods typically have limited applicability on unseen images. Although several attempts have been made to avoid the domain gap issue by learning denoising from singe noisy image itself, they are less effective in handling real-world noise because of assuming the noise corruptions are independent and zero mean. In this paper, we go step further beyond prior work by presenting a novel unsupervised image denoising framework trained from single noisy image without making any explicit assumptions on the noise statistics. Our approach is built upon the deep image prior (DIP), which enables diverse image restoration tasks. However, as is, the denoising performance of DIP will significantly deteriorate on non-zero-mean noise and is sensitive to the number of iterations. To overcome this problem, we propose to utilize multi-scale deep image prior by imposing DIP across different image scales under the constraint of a scale consistency. Experiments on synthetic and real datasets demonstrate that our method performs favorably against the state-of-the-art methods for image denoising.

1. Introduction

Image denoising aims to recover a clean image x from an observed noisy image $y = x + n$, where n denotes the corrupted noise. This problem has been widely studied, since the presence of noise would not only significantly degrade the perceptual quality of an image, but also may adversely affect the performance of many fundamental tasks, *e.g.*, object detection [TPL20, CMS*20], tracking [BDGT19, CYZ*21], and image enhancement [ZNZ19, ZNZ*20, WZF*19, ZYX*18, ZNZX15].

Various methods have been proposed to tackle the image denoising problem. Early methods work by exploring sparse and low-rank representation of natural images [BCM05, EA06, DFKE07, GZZF14], while recent methods are mostly deep learning-based. Among them, supervised methods achieve promising performance on images with additive white Gaussian noise (AWGN) by training on noisy/clean image pairs [XXC12, MSY16, Lef17, TYLX17, ZZC*17, LWF*18, GLGT19, GYZ*19, ZAK*20]. However, their performance usually deteriorates on test images that have different image content and noise statistic from the training images (see Figure 1), and a large number of noisy/clean training image pairs are difficult and expensive to collect.

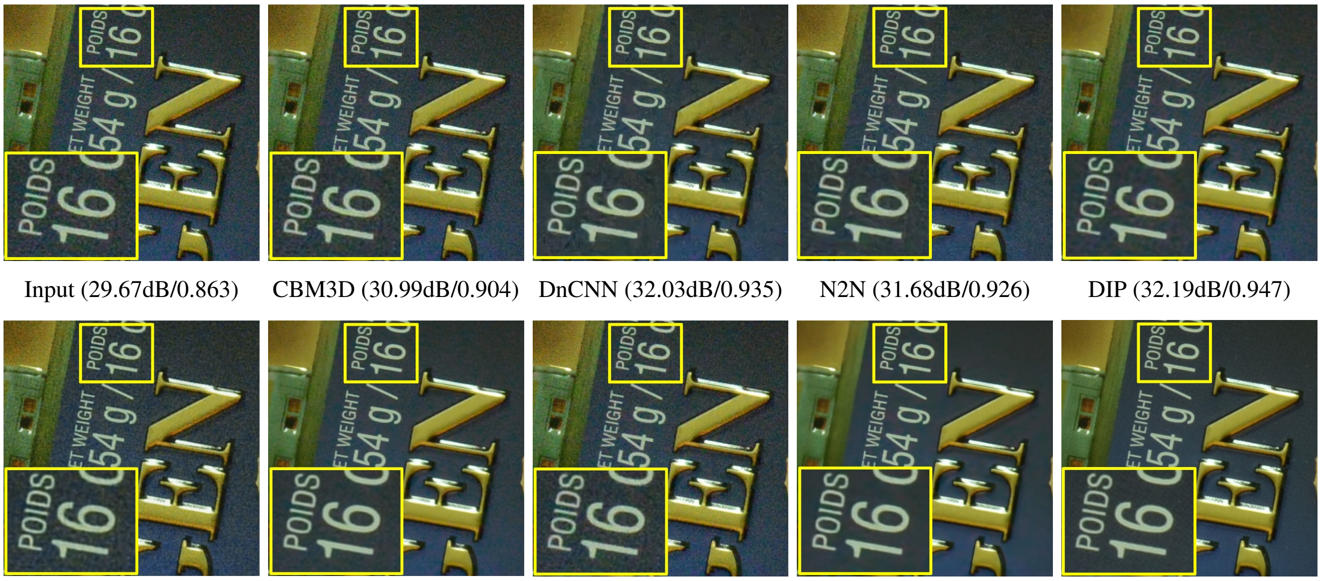
To avoid the dependence on clean training images, some methods proposed to train unsupervised denoising networks from a set of external noisy images [LMH*18, KBJ19, BR19] or single noisy image itself [UVL18, RP19, XHC*20, QCPJ20]. However, these methods still have their respective limitations. For instance, Noise2Noise (N2N) [LMH*18] requires massive paired noisy images with independent noise corruption of the same scene for training, which are difficult to acquire. Deep image prior (DIP) [UVL18] has good performance on zero-mean noise, while real noise is usually not zero-mean [PR17, ALB18] and it is non-trivial to stop its network training at the right moment to achieve the ideal denoising result. Noisy-As-Clean (NAC) [XHC*20] may fail to handle images that break its basic assumption of weak noise. S2S [QCPJ20] requires a prerequisite that the noise corruption is zero-mean and independent between pixels. This method is effective to alleviate the over-fitting arising from training on a single image, but would incur degraded training efficiency.

In this paper, we propose to learn to denoise from a single noisy image, without any explicit modeling or assumption on the noise statistics. We build our network on top of the “Deep Image Prior (DIP)” work by Ulyanov et al. [UVL18], which showed that the structure of a convolutional generation network can capture powerful natural image priors, and can be employed to achieve compelling results for a wide variety tasks (e.g., denoising, super-resolution, in-painting, and layer separation [GSI19]) using only

[*] Corresponding author

© 2022 The Author(s)
Computer Graphics Forum © 2022 The Eurographics Association and John Wiley & Sons Ltd. Published by John Wiley & Sons Ltd.

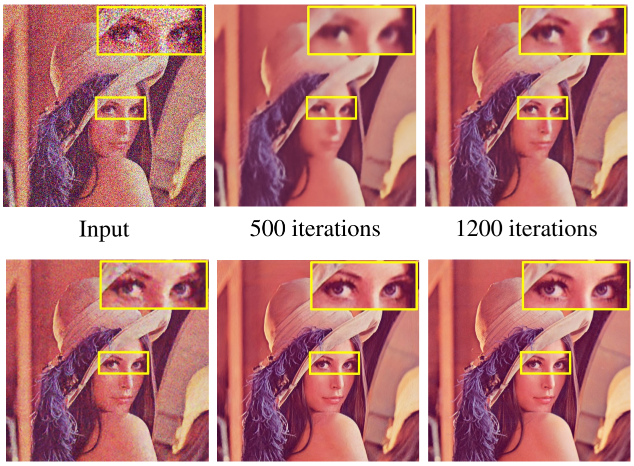
delivered by
EUROGRAPHICS
DIGITAL LIBRARY
www.eg.org
diglib.eg.org



Input (29.67dB/0.863) CBM3D (30.99dB/0.904) DnCNN (32.03dB/0.935) N2N (31.68dB/0.926) DIP (32.19dB/0.947)

NAC (31.40dB/0.7955) S2S (34.41dB/0.963) CycleISP (31.29dB/0.910) **Ours (35.42dB/0.975)** Clean image (GT)

Figure 1: *Comparison with existing image denoising methods on a real noisy image in terms of PSNR(dB)/SSIM. CBM3D [DFKE07] is a traditional denoising method, DnCNN [ZZC* 17] and CycleISP [ZAK* 20] are supervised methods based on noisy/clean image pairs, while N2N [LMH* 18], DIP [UVL18], NAC [XHC* 20] and S2S [QCPJ20] are unsupervised denoising methods trained in absence of clean images.*



Input 500 iterations 1200 iterations

2900 iterations Ours Clean image (GT)

Figure 2: *Denoising results produced by DIP [UVL18] with different numbers of training iterations.*

single training image. However, DIP has two limitations in image denoising. First, it does not work well for non-zero-mean noise. Second, as shown in Figure 2, its performance is sensitive to the moment of stopping its network training. In general, a premature stopping will lead to over-smooth result with degraded image details, while a late stopping may produce a fine-grained reconstruction of the original noisy image.

To address the limitations of DIP and allow more effective image denoising, we in this work present an unsupervised denoising

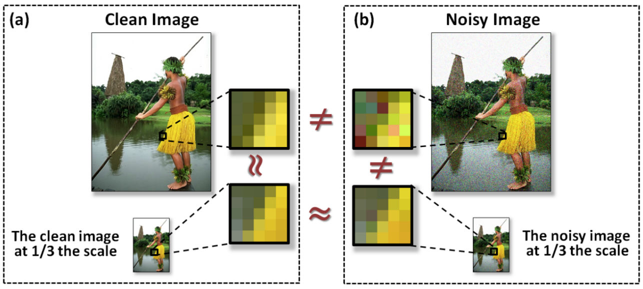

(a)
Clean Image
The clean image at 1/3 the scale
≠
≈
≈
≠
(b)
Noisy Image
The noisy image at 1/3 the scale

Figure 3: *Similarities between a noisy image and its clean counterpart at different image scales. As shown, the corresponding patches across-scale in the clean image share strong similarity, and the clean patch at a 1/3 coarse scale is also very similar to the corresponding noisy patch at the same scale. Image from [ZMI13].*

framework that imposes DIP across different scales of an input noisy image. As shown in Figure 3, our approach is built upon the following observation: *the noise level of an image can be naturally reduced at coarser image scales, making noise corruption that is difficult to handle with DIP at the finest image scale may be easier to handle with DIP at a coarser image scale*. Based on the observation, we develop multi-scale deep image prior (MS-DIP), which is able to robustly generate high-quality denoising results for both synthetic and real noisy images. MS-DIP consists of multiple DIP generator networks, each responsible for learning unsupervised denoising at a certain scale of the given noisy image. Particularly, the denoised image produced by each DIP network at a coarser scale will be used to guide the training of DIP at the next finer scale through a scale consistency loss, such that the output of the finer scale DIP can maintain the noise removal effect learned from the previous scale while recovering previously missing image details. In addition, with the scale consistency loss, we can train each DIP network until convergence to obtain the denoised image, rather than stopping its training in advance as done in the original DIP method [UVL18]. To take full advantage of denoising results from different scales, a multi-scale inference ensemble is developed to average all estimates into a single denoised image. The major contributions of this work are:

- We find that multi-scale deep image prior can be used to enable more effective image denoising.
- We design a novel single-image-based unsupervised denoising framework by coupling deep image priors learned from different image scales.
- Experiments show that our method outperforms previous unsupervised image denoising methods, and can achieve comparable or better results than leading supervised denoising methods.

A preliminary version of this work appeared in [ZNZWS22] for unpublished poster presentation. In this paper, we have improved the paper with the following major changes. First, an additional figure is provided to clearly illustrate the main observation of our approach (see Figure 3). Second, two additional figures are incorporated to show more evaluation of our method on real-world noisy images (see Figures 10 and 11). Last, we provide deeper analysis on the model design and improve the method description.

2. Related Works

This sections reviews previous works on image denoising from the following two aspects, *i.e.*, non-learning-based and learning-based methods, with a focus on recent learning-based methods closely related to our work.

Non-learning-based methods. Prior to the deep learning era, it is a common paradigm to formulate non-learning-based image denoising methods, based on the assumption that the noise corruption and the underlying clean image are of different statistics such that they can be separated by certain observations on natural images. Following this idea, various hand-crafted image priors (*e.g.*, gradient sparsity, patch recurrence, and low rank) were adopted to perform noise removal [ROF92, Cha04, BCM05, EA06, DFKE07, MES07, DLZS11, GZZF14].

Learning-based methods. Recent effort on image denoising is mostly learning-based, since deep neural networks have been proven to be a very powerful tool to infer clean images from their noisy counterparts by learning the statistical difference between the two components. Methods in this category can be further broken down into three groups: (i) methods trained on noisy/clean image pairs; (ii) methods trained on a set of noisy images; (iii) methods trained on a single noisy image.

(i) methods trained on clean/noisy image pairs. Many supervised denoising methods are developed by training on a large amount of noisy/clean image pairs [ZZC*17, ZZZ18, ZZGZ17, GYZ*19, ZTK*20, AB19, YYZ*19, Lef17, BSH12, JLFZ19, CCCY18, Lef18, ZAK*20]. These methods achieve impressive performance on AWGN noise removal, since the paired images employed for supervised learning are typically synthesized according to the AWGN noise model. Due to the domain gap between the synthesized training data and real noisy images, the performance of these methods typically deteriorates on photographs with real noise. Some attempts have been made to alleviate the domain gap by collecting real noisy/clean image pairs for supervised training [ALB18, CCXK18, CCDK19, JZ19, BMX*19, WFYH20]. However, it is difficult to collect a sufficient amount of such image pairs for training a network that generalizes well to unseen images.

(ii) methods trained on a set of noisy images. Since pairs of noisy and clean images are difficult to acquire, several methods proposed to train unsupervised denoising networks from a set of noisy images. N2N [LMH*18] trained a denoising network using paired noisy images of the same scene under the assumption that the noise of paired images is independent. Although this work achieves competitive results, a large number of noisy image pairs are difficult to collect. Instead of using paired noisy images, some recent works proposed to learn unsupervised denoising model from a collection of unorganized noisy images [BR19, KBJ19, LKLA19, KVJ19, WLC*20]. Noise2Void (N2V) [KBJ19] predicted each pixel from its neighboring pixels by learning blind-spot networks. Similar training schemes as the one in [KBJ19] are adopted by later works [BR19, KVJ19, LKLA19] with further performance improvement. More recently, Noiser2Noise [MSZC20] was introduced to generalize N2N [LMH*18] into the setting of a single noisy realization for each image.

(iii) methods trained on single noisy image. Training unsupervised denoising network from a single noisy image has emerged to be a new trend, since it does not suffer from the domain gap problem and is convenient to employ in practice. The first work is originated by DIP [UVL18], which showed that meaningful image patterns are learned more preferentially than random patterns such as noise, when training a randomly initialized convolutional generator network to reconstruct a degraded image. Based on this finding, DIP achieves image denoising by early-stopping a generative network trained for reproducing the original noisy image. Although this method is easy to implement and demonstrates impressive denoising results, its performance is sensitive to the moment choice of stopping the network training, and may not work well for non-zero-mean noise. In order to overcome the over-fitting problem arising from the network training on a single image, Self2Self (S2S) [QCPJ20] proposed to train with dropout on pairs of Bernoulli-sampled instances of the input image. This method produces promising results, but the training scheme significantly degrades the training efficiency. NAC [XHC*20] developed a "Noisy-As-Clean" training strategy for unsupervised image denoising. This strategy has broad applicability, but its effectiveness may deteriorate significantly when the key assumption of weak noise is not met.

3. Our Method

In this section, we describe the proposed unsupervised image denoising framework named as MS-DIP. We first illustrate the motivation of our approach. Next, we introduce the network architecture of MS-DIP, and then elaborate its training, inference, and imple-

© 2022 The Author(s)
Computer Graphics Forum © 2022 The Eurographics Association and John Wiley & Sons Ltd.

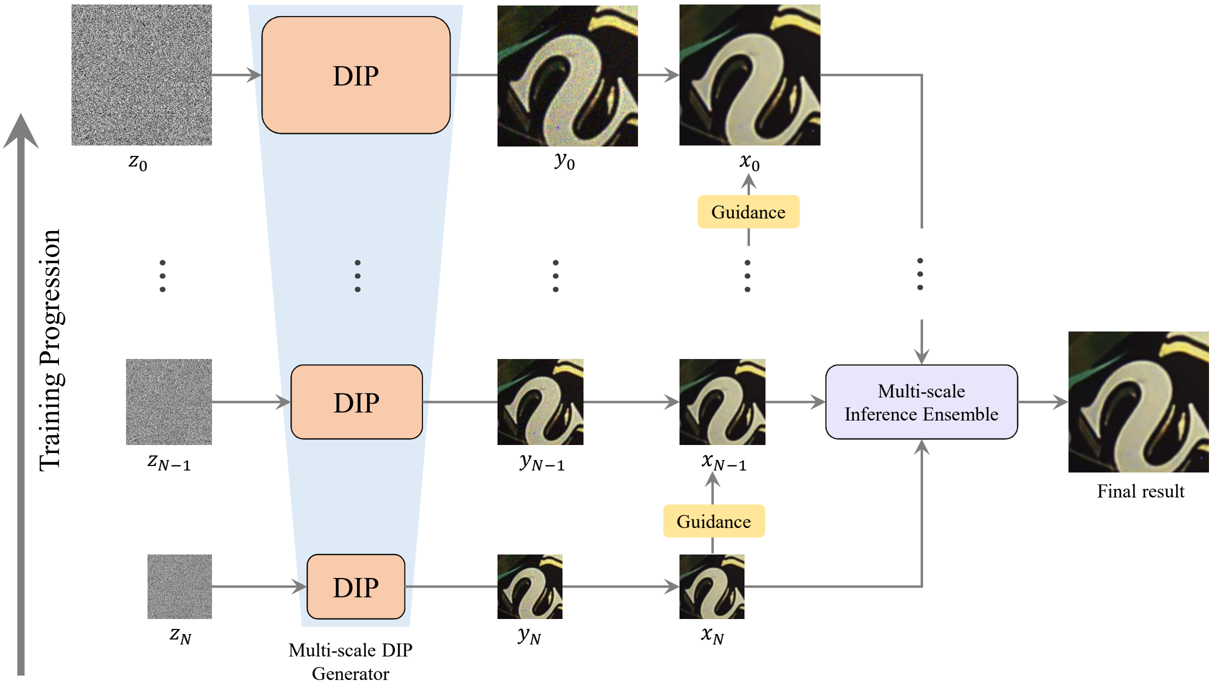


Figure 4: *Overview of the proposed MS-DIP. Our model consists of multiple DIP generator networks, for which both the training and inference are done in a coarse-to-fine fashion. At each scale, a DIP generator is employed to generate a denoised image x_n by reproducing the downsampled noisy image y_n, under the guidance of a denoising output x_{n+1} produced by DIP generator at previous coarser scale (except for the coarsest level). With the denoised images $\{x_N, x_{N-1}, ..., x_0\}$ from all image scales, a multi-scale inference ensemble is performed to produce the final denoising result.*

mentation details. Figure 4 presents the overall denoising workflow of MS-DIP. As shown, MS-DIP employs multiple DIP generators to learn unsupervised denoising of an input noisy image from different image scales, and then averages all the denoising estimates into a single denoised image.

3.1. Motivation: Single DIP vs. Multi-scale DIP

This section describes the motivation of our approach by discussing the necessity of learning multi-scale DIP instead of single-scale DIP for image denoising. We start by giving a brief introduction of how the original single DIP network [UVL18] achieves image denoising. Next, we analyze the limitations of single DIP in image denoising, and illustrate why multi-scale DIP can be employed to allow more effective and robust image denoising.

Denoising by single DIP. Image denoising is achieved in [UVL18] by interpreting a single DIP generator network as a parameterization $x = f_\theta(z)$ of an image x and enforcing the network to reproduce a given noisy image y:

$$\theta^* = \arg\min_\theta \|y - f_\theta(z)\|^2, \quad x^* = f_{\theta^*}(z), \tag{1}$$

where z is a random code vector. θ are initialized random network parameters, while θ^* are parameters learned from optimization. $x^* = f_{\theta^*}(z)$ can be treated as the recovered clean image, since the above parameterization has been shown to present high impedance to image noise.

Limitations of single DIP. Despite single DIP is easy to implement and works well for various image restoration tasks, we found that it has the follow two limitations when applied for image denoising. First, as analyzed in [XHC*20], single DIP is effective to handle zero-mean noise, while it may fail to produce satisfactory results for real-world images with non-zero mean noise. Second, its performance is sensitive to the number of iterations for optimizing the image reconstruction in Eq. (1), which is hard to control.

Why multi-scale DIP works better? Compared with single DIP, multi-scale DIP has the following advantages in image denoising. First of all, as shown in Figure 3, it was observed that the noise level of an image can be naturally reduced at coarser image scales, such that noise difficult to handle with DIP may be easier to handle at a coarser image scale [ZMI13]. Hence, by coupling multiple DIP generator networks across different scales of an image and then combining the denoising results from all image scales into a single denoised image, we are able to obtain higher noise impedance than single DIP, especially for previously challenging signal-dependent real noise. On the other hand, since denoising result produced by DIP networks at coarser image scales can be used to guide the network training of DIP networks at the subsequent finer scales, we can train the entire network (including DIP generator network at each image scale) until convergence to produce the desired denoising results, unlike the original single DIP method [UVL18] which requires manually setting a proper number of training iterations to achieve image denoising.

3.2. Network architecture

The network architecture of our MS-DIP is shown in Figure 4. It consists of a pyramid of DIP generators (same as the one employed in [UVL18]), which are trained over an image pyramid of the given noisy image y: $\{y_0,...,y_N\}$, where y_n is a downsampled version of y with a factor r^n ($r<1$). Each DIP generator aims to produce a denoised image x_n from the downsampled noisy image y_n. This is achieved by reconstructing y_n from random code vector z_n, as illustrated in Eq. (1). The whole network is trained in a coarse-to-fine manner. We start at the coarsest image scale, which has the minimum noise level since it has been observed that noise level drops dramatically at coarser image scales [ZMI13]. Owing to the noise suppression naturally enabled by image downsampling, training DIP network at the coarsest scale allows us to obtain a denoised image with strong noise removal but weak detail preservation. The denoising output of the coarsest scale is then used to guide the training of DIP network at the next finer scale, such that the output of the finer scale DIP can maintain similar noise removal effect while recovering the previously missing details. Subsequent DIP networks at finer image scales are trained similarly. Based on denoising results produced from all image scales, we perform a multi-scale inference ensemble to generate the final denoising result.

3.3. Training

Besides the coarsest image scale y_N whose training is the same as the single DIP introduced in [UVL18], the training loss function for DIP networks at other image scales $n\in[0,N-1]$ is as follows:

$$\mathcal{L}^n_{total} = \mathcal{L}^n_{rec} + \lambda_n \mathcal{L}^n_{sc}, \tag{2}$$

where $\mathcal{L}^n_{rec}$ is a reconstruction loss as in Eq. (1), while $\mathcal{L}^n_{sc}$ is a scale consistency loss that aims to enforce DIP network at the current scale to obtain a denoised image with similar noise removal effect as the denoising output from previous coarser scale. λ_n is a scale-adaptive weight. Below we describe the consistency loss $\mathcal{L}^n_{sc}$ and the weight λ_n in detail.

Scale consistency loss. To avoid bringing back noise from finer scale DIP generators, we design a scale consistency loss to encourage similarity between the training output x_n of the current scale and the known denoised output x_{n+1} from the previous coarser scale. Rather than encouraging the pixels of x_n to exactly match the pixels of x_{n+1}, we follow [JAFF16] to encourage them to have similar feature representations computed by a VGG-16 network pretrained on ImageNet, which is formulated as

$$\mathcal{L}_{sc} = \text{MSE}\left(\phi_i((x_n)\downarrow^r), \phi_i(x_{n+1})\right), \tag{3}$$

where ϕ_i denotes the i-th feature layer of the VGG-16 network. $(x_n)\downarrow^r$ is a downsampled version of x_n by a factor of r, which has the same size as x_{n+1}.

Scale-adaptive weight λ_n. The weight λ_n in Eq. (2) plays an important role in determining the overall denoising performance. Intuitively, a large λ_n tends to make the DIP network to simulate the denoised images from previous coarser scales, and produces a smooth output with degraded image details. On the contrary, a small λ_n may result in noise residual in the denoising output. According to above analysis, setting a proper λ_n for each image scale can help obtain high-quality noise removal results. To this end, we design a scale-adaptive weighting scheme for λ_n, which is expressed as

$$\lambda_n = \sqrt{\sigma_{y_n}(N-n)}, \tag{4}$$

where σ_{y_n} is the noise level of y_n, which is estimated by the method of [CZAH15]. N is the total number of image scales. In general, high noise level of y_n and shallow image scale n correspond to large λ_n. The reason behind this design is twofold. First, when y_n has high noise level, we want to enhance the capability of noise removal by enforcing strong scale consistency to the denoising output from previous scale. Second, as the image scale goes up, the risk of bringing back noise from the finer scale DIP learning becomes high. Hence, we gradually enlarge λ_n to lower the effect of the reconstruction loss to alleviate this problem.

3.4. Inference

Since multiple DIP networks are trained across the image scales, multi-scale denoising results $\{x_0,...,x_N\}$ are thus generated along with the training of MS-DIP. To obtain the final denoising result that gathers all noise removal estimates, a multi-scale inference ensemble is developed.

Multi-scale inference ensemble. The multi-scale denoising outputs $\{x_0,...,x_N\}$ are averaged to obtain the final denoised image x. As $\{x_0,...,x_N\}$ are in different sizes, we choose to average two neighboring results at each time, and then use the obtain result to compute average between the result from the next finer scale. Suppose x_n and x_{n-1} are two results to be averaged, we first upsample x_n to the same resolution as x_{n-1} in an edge-aware fashion by performing joint bilateral upsampling [KCLU07] as

$$\hat{x}^p_n = \frac{1}{\mathcal{Z}_p}\sum_{q_\uparrow\in\Omega_{p_\uparrow}} x^q_n F(\|p-q\|)G(\left\|x^{p_\uparrow}_{n-1} - x^{q_\uparrow}_{n-1}\right\|), \tag{5}$$

where p and q denote coordinates of pixels in x_n, while $p_\uparrow$ and $q_\uparrow$ denote coordinates of pixels in x_{n-1} and the upsampled version $\hat{x}^p_n$ of x_n. F and G are spatial and range filter kernels with standard deviation $\sigma_d=0.5$ and $\sigma_r=0.1$, respectively. Ω denotes a 5×5 window centered at pixel $p_\uparrow$. $\mathcal{Z}_p$ is the normalizing factor that sums the filter weight $F(\cdot)G(\cdot)$. With the upsampled $\hat{x}_n$, we average it with x_{n-1} to update x_{n-1}. The updated x_{n-1} is then used to perform averaging between x_{n-2}, until the finest scale result x_0 is averaged to produce the final denoised image x. Note, the reason we adopt joint bilateral upsampling instead of simple upsampling strategies such as bilinear upsampling and nearest neighbor upsampling is because it is able to produce results with sharper edges and details.

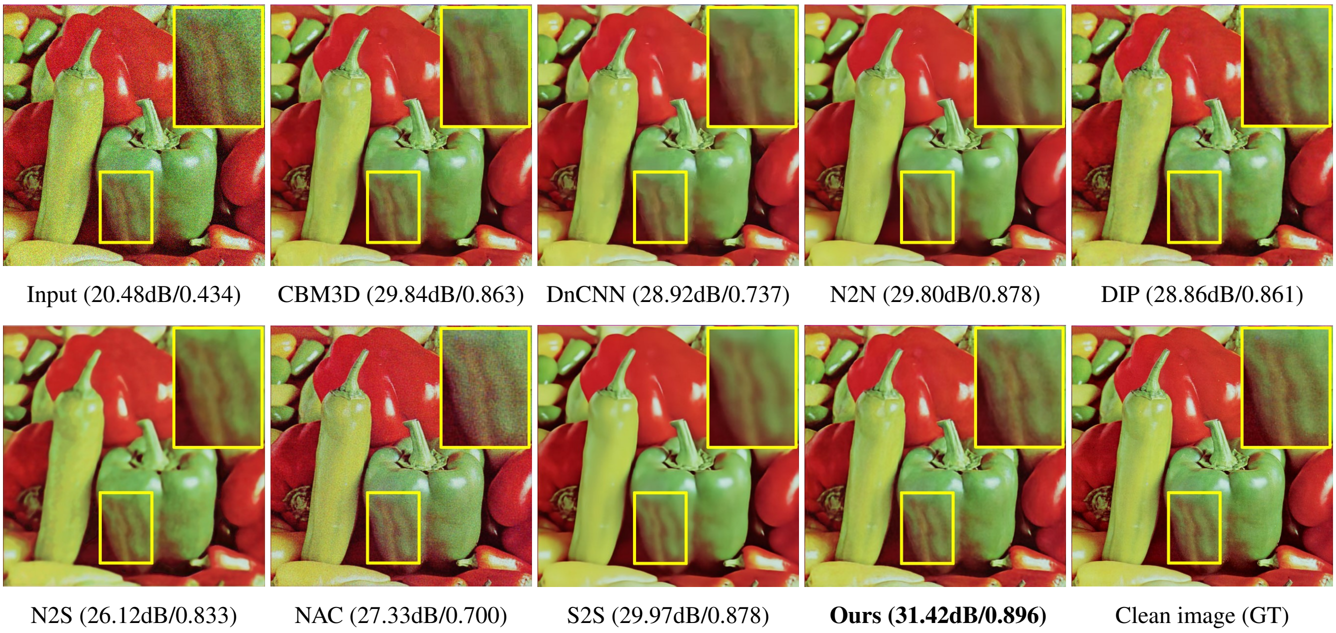
3.5. Implementation Details

Our model is implemented in Pytorch using Adam optimizer with a fixed learning rate of 10^{-3}. The random noise input to DIP generator at each scale is initialized as uniform noise with same size as the downsampled noisy image. To stabilize the network training and achieve more stable results, we follow [UVL18] to perturb the noise code z_n with random Gaussian disturbance at each iteration. In addition, we found that training with a L_1 reconstruction loss at early iterations and then switching to L_2 reconstruction loss can help produce denoising results with clearer structures. The downsampling

© 2022 The Author(s)

Table 1: *Quantitative comparison between our method and state-of-the-art methods on the Set9 and BSD68 datasets in terms of average PSNR(dB)/SSIM. The best numerical results for different AWGN noise levels are shown in* ***boldface****.*

Dataset	Set9								BSD68			
Noise level	σ = 10		σ = 15		σ = 20		σ = 25		σ = 10		σ = 25	
Metric	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
CBM3D [DFKE07]	31.40	0.918	30.87	0.902	30.16	0.899	29.36	0.889	34.43	0.951	30.72	0.910
DnCNN [ZZC*17]	31.06	0.848	30.10	0.824	29.29	0.804	28.58	0.785	33.73	0.945	30.01	0.874
N2N [LMH*18]	30.38	0.914	30.07	0.900	29.70	0.896	29.27	0.891	33.85	0.943	30.37	0.903
DIP [UVL18]	29.68	0.889	29.18	0.880	28.95	0.878	28.09	0.864	33.00	0.932	29.71	0.885
N2S [BR19]	25.10	0.822	25.96	0.825	25.42	0.825	25.37	0.799	29.66	0.922	28.60	0.891
NAC [XHC*20]	29.35	0.759	24.97	0.569	22.50	0.458	20.65	0.383	30.63	0.850	25.56	0.616
S2S [QCPJ20]	30.11	0.900	29.89	0.895	29.59	0.889	29.24	0.884	33.16	0.933	30.45	0.905
Ours	**31.94**	**0.926**	**31.25**	**0.908**	**30.48**	**0.906**	**29.94**	**0.903**	**34.94**	**0.956**	**31.04**	**0.913**



Input (20.48dB/0.434) CBM3D (29.84dB/0.863) DnCNN (28.92dB/0.737) N2N (29.80dB/0.878) DIP (28.86dB/0.861)

N2S (26.12dB/0.833) NAC (27.33dB/0.700) S2S (29.97dB/0.878) **Ours (31.42dB/0.896)** Clean image (GT)

Figure 5: *Visual comparison of blind AWGN denoising on an image from the Set9 dataset with noise level σ = 25.*

factor r is set as 0.8, and the minimum scale is 128 × 128. Note, unlike [UVL18] which requires manually setting a proper number of iterations to achieve image denoising rather than fine-scale image reconstruction, our network can be trained until convergence to generate denoising results, because the multi-scale framework can provide denoising guidance for the training of the DIP generator network at each image scale. In general, our network converges after 800 to 1000 training iterations, depending on the image content.

4. Experiments

In this section, we present experiments to evaluate the proposed MS-DIP on image denoising. We first compare our method with state-of-the-art methods on blind Gaussian denoising and real-world image denoising. Next, we conduct ablation studies to evaluate the model design and discuss the limitations of our method.

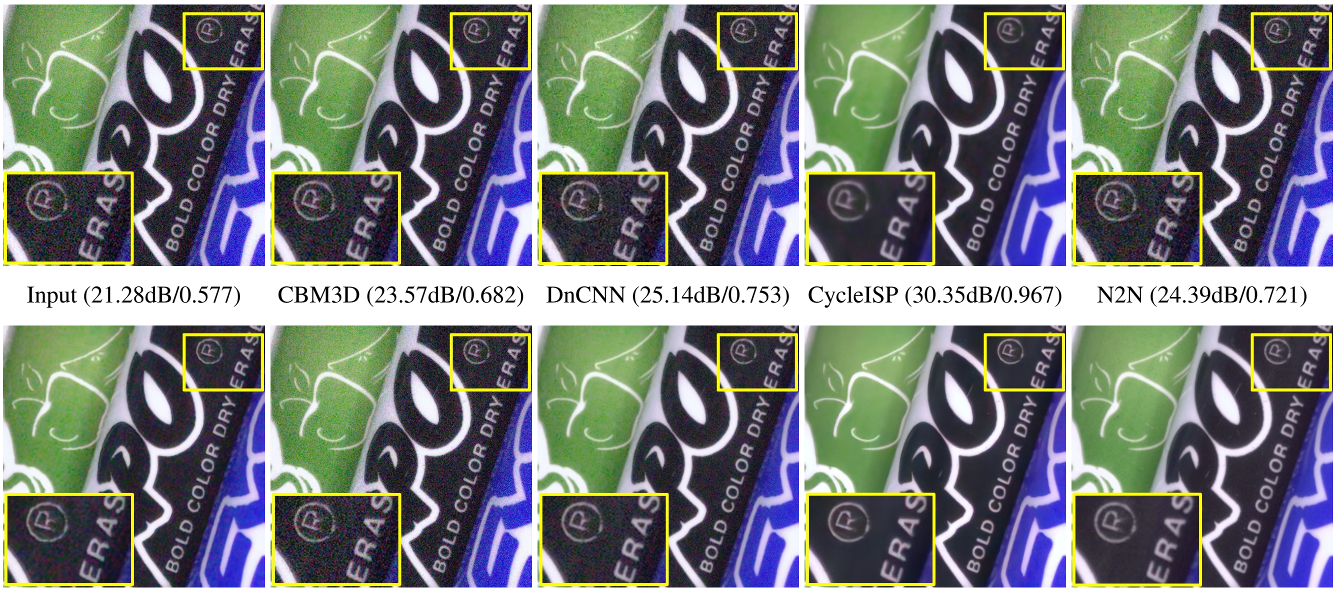
4.1. Blind Gaussian Denoising

Datasets. We evaluate the performance of our method on the benchmark Set9 and BSD68 datasets corrupted by synthetic AWGN noise, which are widely employed by previous works [UVL18, KBJ19, QCPJ20, XHC*20]. The first one contains 9 color images, while the second one has 68 gray-scale images.

Compared methods. We compare our method with various state-of-the-art methods, including: (i) CBM3D [DFKE07], which is a well-performed non-learning-based method; (ii) DnCNN [ZZC*17], a common benchmark for supervised image denoising; (iii) five recent unsupervised denoising methods, i.e., N2N [LMH*18], DIP [UVL18], N2S [BR19], NAC [XHC*20], and S2S [QCPJ20]. Note, N2N and N2S are unsupervised methods trained on a set of noisy images, while DIP, NAC, and S2S are single-image-based unsupervised methods. For fair comparison, we produce results of all the compared methods using publicly-

Table 2: *Quantitative comparison between our method and state-of-the-art methods on the SIDD-Medium and CC datasets.*

Dataset	Metric	Non-learning	Supervised		Unsupervised (datasets)		Unsupervised (single-image)			
		CBM3D	DnCNN	CycleISP	N2N	N2S	DIP	NAC	S2S	Ours
CC	PSNR	35.19	34.65	35.56	35.32	31.86	35.69	36.59	37.29	**37.82**
	SSIM	0.906	0.960	0.962	0.916	0.950	0.926	0.950	0.976	**0.981**
SIDD-Medium	PSNR	35.06	33.40	**36.90**	32.74	33.25	34.05	32.64	35.32	36.76
	SSIM	0.891	0.886	**0.974**	0.870	0.952	0.920	0.769	0.927	0.967



Input (21.28dB/0.577) CBM3D (23.57dB/0.682) DnCNN (25.14dB/0.753) CycleISP (30.35dB/0.967) N2N (24.39dB/0.721)

DIP (27.02dB/0.875) NAC (22.88dB/0.4813) S2S (28.16dB/0.873) **Ours (31.69dB/0.972)** Clean image (GT)

Figure 6: *Visual comparison of real-world image noise removal on an image from the SIDD-Medium dataset.*

available codes or trained models provided by the authors with recommended parameter setting. In addition, since DIP's denoising performance is sensitive to the number of iterations, we thus implemented it multiple times with different number of iterations and adopted the best results for comparison.

Quantitative comparison. To evaluate our method's effectiveness in blind AWGN noise removal, we compare it with the other methods on the Set9 and BSD68 datasets in terms of average PSNR (dB) and SSIM. Table 1 reports the results, where we can see that our method outperforms the others on the two metrics for both benchmark datasets. CBM3D and DnCNN produce very competitive results, since the former is non-blind to the noise level and the latter benefits from supervised training on massive high-quality noisy/clean image pairs. Our method clearly outperforms DIP, manifesting that learning multi-scale deep image prior allows more effective image denoising. N2N and S2S also produce promising results, while their visual results in Figure 5 demonstrates that they tend to generate overly smoothed images with degraded image details.

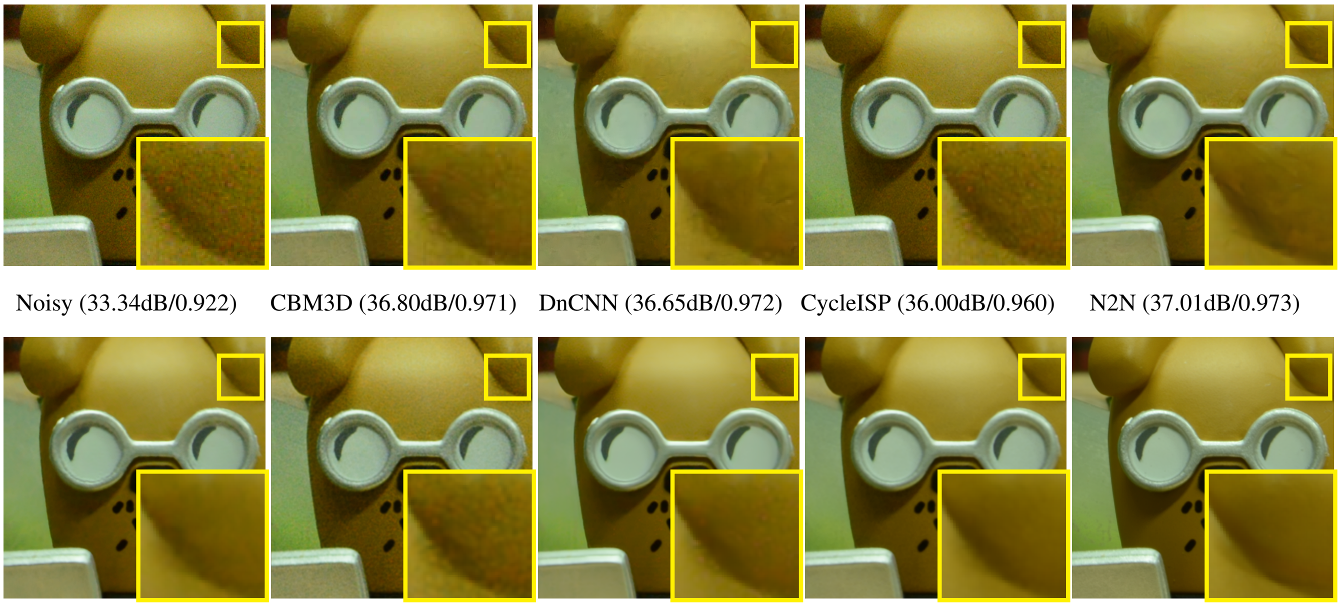
Visual comparison. We further provide visual comparison results in Figure 5. As can be seen, there are obvious noise residuals in results produced by CBM3D, DIP, and NAC, while the results of DnCNN, N2N, N2S, and S2S degrade the image textures and structures. In contrast, our method produces better result, by not only effectively removing the noise, but also faithfully preserving the underlying image details.

4.2. Real-World Noise Removal

Datasets. Two real-world noisy datasets are employed for performance evaluation, which are the SIDD-Medium dataset [ALB18] and the CC dataset [NHMJK16]. The SIDD-Medium dataset contains 160 real noisy images captured by five different smartphone cameras with corresponding ground-truth clean counterparts. The CC dataset consists of images of 11 scenes captured by three cameras, and their corresponding clean images.

Compared methods. Our method is compared with the following eight methods: (i) CBM3D; (ii) DnCNN and CycleISP [ZAK*20]; (iii) N2N, DIP, NAC, and S2S, where CycleISP is a state-of-the-art supervised method. Note, the same method configuration as in Section 4.1 is adopted to achieve fair comparison. Akin to [QCPJ20], we employ [CZAH15] to estimate the noise level for CBM3D.

Quantitative comparison. Table 2 shows the quantitative comparison results. As shown, on both SIDD-Medium and CC, our



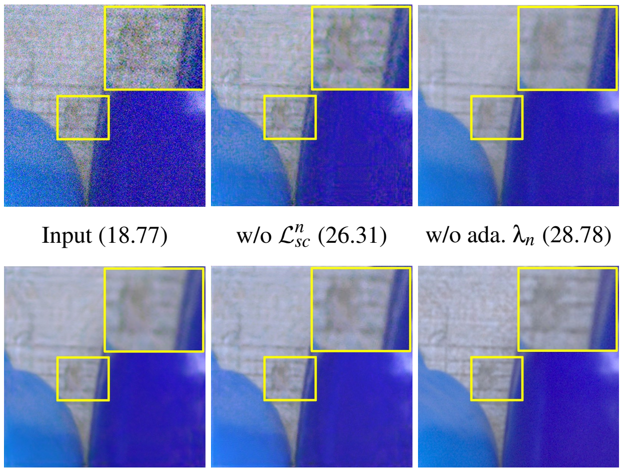
Noisy (33.34dB/0.922) CBM3D (36.80dB/0.971) DnCNN (36.65dB/0.972) CycleISP (36.00dB/0.960) N2N (37.01dB/0.973)

DIP (36.52dB/0.983) NAC (36.20dB/0.900) S2S (38.45dB/0.984) **Ours (40.47dB/0.989)** Clean image (GT)

Figure 7: *Visual comparison of real-world image noise removal on an image from the CC dataset.*

method produces better results than the non-learning-based and unsupervised methods. Benefiting from the dropout-based training strategy, S2S achieves competitive results on CC since noisy images in this dataset are typically corrupted by relatively weak noise, while its performance deteriorates on SIDD-Medium consisting of images with heavy sensor noise. It is worth mentioning that our method also produces comparable or even better results than DnCNN and CycleISP, which are leading supervised methods. Note that, although CycleISP reports best numerical results on SIDD-Medium, our results are very close to that of CycleISP. Furthermore, as shown in Figure 6, we are able to obtain better results than CycleISP on some noisy images from SIDD-Medium with complex textures.

Visual comparison. The visual comparison on SIDD-Medium is shown in Figure 6, where the input image is corrupted by heavy camera sensor noise. As the sensor noise is signal dependent and it is nontrivial to estimate a proper noise level, there are obvious noise residuals in result of CBM3D. Similar issues also appear in results of DnCNN and N2N, mainly due to the domain gap between the training samples and test images. NAC fails to remove noise, because its weak noise assumption is violated by the employed noisy image. DIP, S2S, and CycleISP produce competitive results, while they also induce lightweight noise residuals or degraded image structures. In comparison, our method produces a high-quality result without noticeable noise residual and structure degradation. Figure 7 presents visual comparison on an image from the CC dataset. We can see that our method produces high-quality result, while the compared methods either fail to completely remove the noise, or destroy the underlying texture structure of the input image.



Input (18.77) w/o $\mathcal{L}_{sc}^{n}$ (26.31) w/o ada. λ_n (28.78)

w/o ensem. (30.19) Full method (32.15) Clean image (GT)

Figure 8: *Visual ablation study (with PSNR) on the scale consistency loss $\mathcal{L}_{sc}^{n}$, scale adaptive weight λ_n, and multi-scale inference ensemble. The input image is from the SIDD-Medium dataset.*

4.3. Ablation Study

Besides the visual comparison results shown in Figure 8, we also conducted ablation studies to evaluate the effectiveness of each component in our model. Comparing the numerical results in Table 3, we observe clear performance improvements by adopting the scale consistency loss $\mathcal{L}_{sc}^{n}$, the scale adaptive weight λ_n, and the multi-scale inference ensemble, which convincingly demonstrate their respective effectivenesses. Note, the ablation choice of "w/o

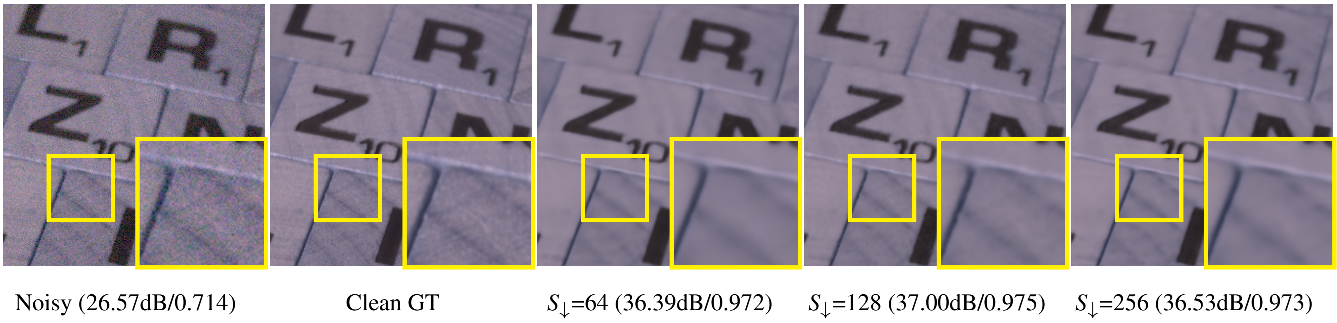

Noisy (26.57dB/0.714)
Clean GT
$S_\downarrow$=64 (36.39dB/0.972)
$S_\downarrow$=128 (37.00dB/0.975)
$S_\downarrow$=256 (36.53dB/0.973)

Figure 9: *Effect of varying minimum image scales $S_\downarrow$ on denoising an image from the SIDD-Medium dataset.*

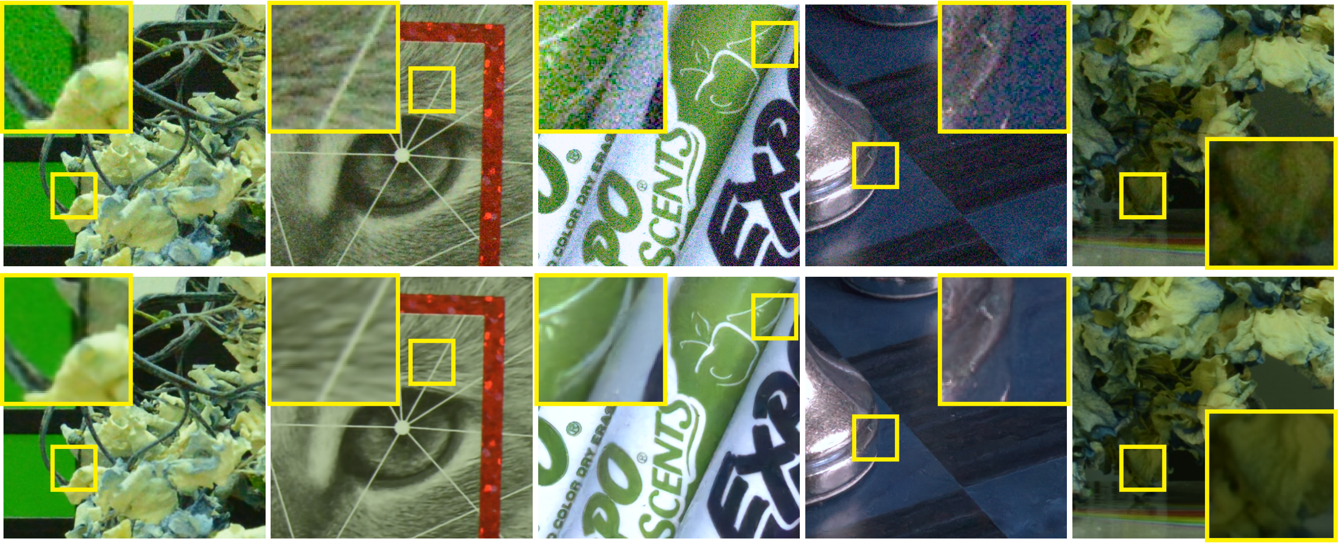


Figure 10: *More denoising results on real-world images produced by our method.*

ensemble" in Table 3 means that the final denoising result is the finest scale output (*i.e.*, x_0 in Figure 4), rather than the combination of the denoising outputs from different image scales. The choice of "w/o adaptive λ_n" indicates that λ_n in Eq. (2) is a fixed value same for all image scales instead of a scale-adaptive value. We also analyzed the effect of varying minimum image scales on the denoising performance, and found that smaller minimum scales may not produce better results, as shown in Figure 9.

Table 3: *Quantitative ablation studies on the scale consistency loss $\mathcal{L}_{sc}^n$, scale adaptive weight λ_n, and multi-scale inference ensemble on the CC and SIDD-Medium datasets (w/o - without).*

Method	CC		SIDD-Medium	
	PSNR	SSIM	PSNR	SSIM
Ours w/o ensemble	37.02	0.963	36.37	0.951
Ours w/o adaptive λ_n	35.83	0.949	35.52	0.934
Ours w/o $\mathcal{L}_{sc}^n$	34.13	0.901	33.27	0.916
Ours (full method)	**37.82**	**0.981**	**36.95**	**0.967**

4.4. More Results on Real-world Noisy Images

Figures 10 and 11 show more results and comparisons on real-world noisy images, where the input images cover a broad range of scenes, subjects, and lighting conditions. As can be seen, for all these cases, our method produces visually compelling results, manifesting its effectiveness in handling real-world noisy images.

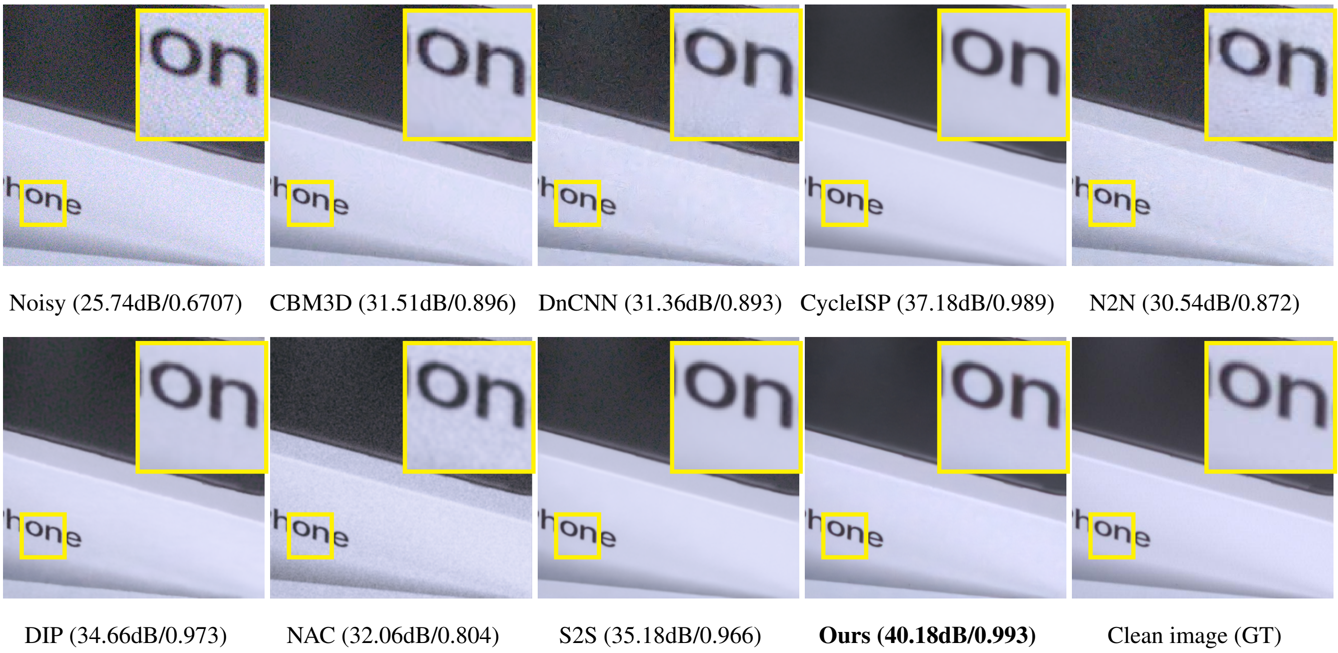
4.5. Effect of Different Number of Iterations

Figure 13 examines the denoising performance of our method with different number of training iterations. As can be seen, unlike the original DIP method whose denoising results are very sensitive to the number of iterations (see Figure 2), we are able to produce high-quality denoising results by simply training our network until convergence.

4.6. Limitations

Although the proposed method provides a simple yet effective exploration to unsupervised image denoising based on single training

© 2022 The Author(s)
Computer Graphics Forum © 2022 The Eurographics Association and John Wiley & Sons Ltd.


Noisy (25.74dB/0.6707) CBM3D (31.51dB/0.896) DnCNN (31.36dB/0.893) CycleISP (37.18dB/0.989) N2N (30.54dB/0.872)

DIP (34.66dB/0.973) NAC (32.06dB/0.804) S2S (35.18dB/0.966) **Ours (40.18dB/0.993)** Clean image (GT)

Figure 11: *More comparison with the state-of-the-art methods on real-world noise removal.*

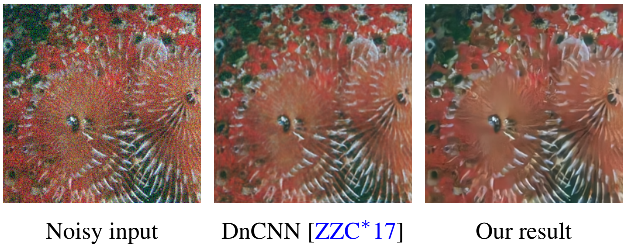

Noisy input DnCNN [ZZC*17] Our result

Figure 12: *Our method fails to completely remove noises from the leftmost noisy image with highly textured background, while faithfully preserving the texture details.*

image, it still has several limitations. First of all, unlike most prior learning-based denoising methods where the training and testing phase are separately, as the testing phase of our method involves the whole time-consuming training procedure, it typically takes relatively high time cost (a few minutes for an image of size 640 × 480) to produce the denoising results. In addition, for real-world noisy images with highly textured background in Figure 12, our method as well as other state-of-the-art methods may fail to completely remove the noise while faithfully preserving the texture details.

5. Conclusion

In this paper, we present MS-DIP, a single-image-based unsupervised framework for high-quality image denoising. It is built upon the observation that the noise level of an image usually drops dramatically at coarser image scales, such that noise removal at coarser scales is more tractable. Based on the observation, we propose to perform image denoising by learning deep image prior across image scales under the guidance of denoising outputs produced by previous coarser scales, and then averaging the denoising outputs from different scales into a single denoised image. Experiments on benchmark synthetic and real-world datasets show that our method outperforms previous unsupervised image denoising methods, and can achieve comparable or even better results than the state-of-the-art supervised image denoising methods.

Acknowledgments We thank the anonymous reviewers for their valuable comments. This work was partially supported by the National Natural Science Foundation of China (No. 61802453, No. 62072191, and No. 61902275), the Science and Technology Project of Guangzhou (No. 202102020574), and the Natural Science Foundation of Guangdong Province project (2019A1515010860).

References

[AB19] Anwar S., Barnes N.: Real image denoising with feature attention. In *ICCV* (2019), pp. 3155–3164. 3

[ALB18] Abdelhamed A., Lin S., Brown M. S.: A high-quality denoising dataset for smartphone cameras. In *CVPR* (2018), pp. 1692–1700. 1, 3, 7

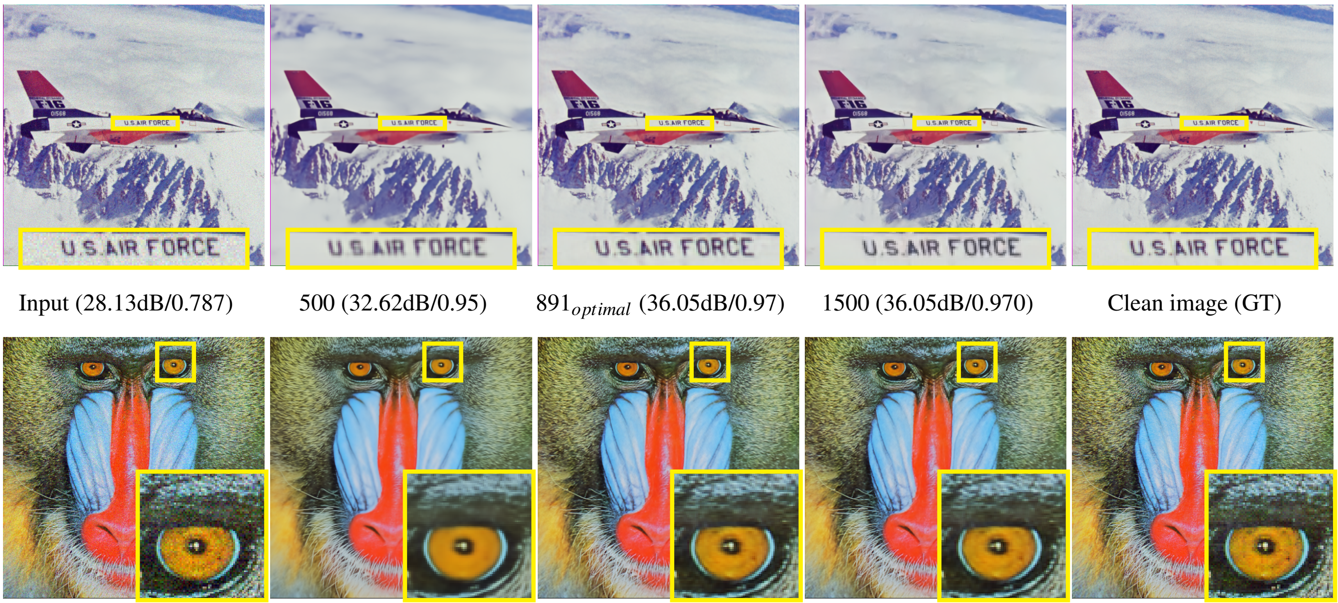
[BCM05] Buades A., Coll B., Morel J.-M.: A non-local algorithm for image denoising. In *CVPR* (2005), pp. 60–65. 1, 3

[BDGT19] Bhat G., Danelljan M., Gool L. V., Timofte R.: Learning discriminative model prediction for tracking. In *ICCV* (2019), pp. 6182–6191. 1

[BMX*19] Brooks T., Mildenhall B., Xue T., Chen J., Sharlet D., Barron J. T.: Unprocessing images for learned raw denoising. In *CVPR* (2019), pp. 11036–11045. 3

[BR19] Batson J., Royer L.: Noise2self: Blind denoising by self-supervision. *arXiv preprint arXiv:1901.11365* (2019). 1, 3, 6

© 2022 The Author(s)



Input (28.13dB/0.787) 500 (32.62dB/0.95) $891_{optimal}$ (36.05dB/0.97) 1500 (36.05dB/0.970) Clean image (GT)

Input (24.67dB/0.814) 500 (24.89dB/0.822) $932_{optimal}$ (26.18dB/0.876) 1500 (26.18dB/0.876) Clean image (GT)

Figure 13: *Effect of different number of iterations on our denoising results. From left to right are the noisy images, three denoising results with different number of iterations (i.e., 500 iterations, optimal number of iterations to reach convergence, and 1500 iterations that beyond convergence), and the ground truths. As shown, different number of iterations before convergence can affect the results, while more iterations after convergence do not change the results.*

[BSH12] Burger H. C., Schuler C. J., Harmeling S.: Image denoising: Can plain neural networks compete with bm3d? In *CVPR* (2012), pp. 2392–2399. 3

[CCCY18] Chen J., Chen J., Chao H., Yang M.: Image blind denoising with generative adversarial network based noise modeling. In *CVPR* (2018), pp. 3155–3164. 3

[CCDK19] Chen C., Chen Q., Do M. N., Koltun V.: Seeing motion in the dark. In *ICCV* (2019), pp. 3185–3194. 3

[CCXK18] Chen C., Chen Q., Xu J., Koltun V.: Learning to see in the dark. In *CVPR* (2018), pp. 3291–3300. 3

[Cha04] Chambolle A.: An algorithm for total variation minimization and applications. *Journal of Mathematical imaging and vision 20*, 1-2 (2004), 89–97. 3

[CMS*20] Carion N., Massa F., Synnaeve G., Usunier N., Kirillov A., Zagoruyko S.: End-to-end object detection with transformers. In *ECCV* (2020), Springer, pp. 213–229. 1

[CYZ*21] Chen X., Yan B., Zhu J., Wang D., Yang X., Lu H.: Transformer tracking. In *CVPR* (2021), pp. 8126–8135. 1

[CZAH15] Chen G., Zhu F., Ann Heng P.: An efficient statistical method for image noise level estimation. In *ICCV* (2015), pp. 477–485. 5, 7

[DFKE07] Dabov K., Foi A., Katkovnik V., Egiazarian K.: Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing 16*, 8 (2007), 2080–2095. 1, 2, 3, 6

[DLZS11] Dong W., Li X., Zhang L., Shi G.: Sparsity-based image denoising via dictionary learning and structural clustering. In *CVPR* (2011), pp. 457–464. 3

[EA06] Elad M., Aharon M.: Image denoising via learned dictionaries and sparse representation. In *CVPR* (2006), pp. 895–900. 1, 3

[GLGT19] Gu S., Li Y., Gool L. V., Timofte R.: Self-guided network for fast image denoising. In *ICCV* (2019), pp. 2511–2520. 1

[GSI19] Gandelsman Y., Shocher A., Irani M.: Double-DIP: Unsupervised image decomposition via coupled deep-image-priors. In *CVPR* (2019), pp. 11018–11027. 1

[GYZ*19] Guo S., Yan Z., Zhang K., Zuo W., Zhang L.: Toward convolutional blind denoising of real photographs. In *CVPR* (2019), pp. 1712–1722. 1, 3

[GZZF14] Gu S., Zhang L., Zuo W., Feng X.: Weighted nuclear norm minimization with application to image denoising. In *CVPR* (2014), pp. 2862–2869. 1, 3

[JAFF16] Johnson J., Alahi A., Fei-Fei L.: Perceptual losses for real-time style transfer and super-resolution. In *ECCV* (2016), pp. 694–711. 5

[JLFZ19] Jia X., Liu S., Feng X., Zhang L.: Focnet: A fractional optimal control network for image denoising. In *CVPR* (2019), pp. 6054–6063. 3

[JZ19] Jiang H., Zheng Y.: Learning to see moving objects in the dark. In *ICCV* (2019), pp. 7324–7333. 3

[KBJ19] Krull A., Buchholz T.-O., Jug F.: Noise2void-learning denoising from single noisy images. In *CVPR* (2019), pp. 2129–2137. 1, 3, 6

[KCLU07] Kopf J., Cohen M. F., Lischinski D., Uyttendaele M.: Joint bilateral upsampling. *ACM Transactions on Graphics (ToG) 26*, 3 (2007), 96–es. 5

[KVJ19] Krull A., Vicar T., Jug F.: Probabilistic noise2void: Unsupervised content-aware denoising. *arXiv preprint arXiv:1906.00651* (2019). 3

[Lef17] Lefkimmiatis S.: Non-local color image denoising with convolutional neural networks. In *CVPR* (2017), pp. 3587–3596. 1, 3

© 2022 The Author(s)
Computer Graphics Forum © 2022 The Eurographics Association and John Wiley & Sons Ltd.

[Lef18] LEFKIMMIATIS S.: Universal denoising networks: a novel cnn architecture for image denoising. In *CVPR* (2018), pp. 3204–3213. 3

[LKLA19] LAINE S., KARRAS T., LEHTINEN J., AILA T.: High-quality self-supervised deep image denoising. In *NeurIPS* (2019), pp. 6970–6980. 3

[LMH*18] LEHTINEN J., MUNKBERG J., HASSELGREN J., LAINE S., KARRAS T., AITTALA M., AILA T.: Noise2noise: Learning image restoration without clean data. In *ICML* (2018), pp. 2965–2974. 1, 2, 3, 6

[LWF*18] LIU D., WEN B., FAN Y., LOY C. C., HUANG T. S.: Non-local recurrent network for image restoration. In *NeurIPS* (2018), pp. 1673–1682. 1

[MES07] MAIRAL J., ELAD M., SAPIRO G.: Sparse representation for color image restoration. *IEEE Transactions on Image Processing 17*, 1 (2007), 53–69. 3

[MSY16] MAO X.-J., SHEN C., YANG Y.-B.: Image restoration using convolutional auto-encoders with symmetric skip connections. *arXiv preprint arXiv:1606.08921* (2016). 1

[MSZC20] MORAN N., SCHMIDT D., ZHONG Y., COADY P.: Noisier2noise: Learning to denoise from unpaired noisy data. In *CVPR* (2020), pp. 12064–12072. 3

[NHMJK16] NAM S., HWANG Y., MATSUSHITA Y., JOO KIM S.: A holistic approach to cross-channel image noise modeling and its application to image denoising. In *CVPR* (2016), pp. 1683–1691. 7

[PR17] PLOTZ T., ROTH S.: Benchmarking denoising algorithms with real photographs. In *CVPR* (2017), pp. 1586–1595. 1

[QCPJ20] QUAN Y., CHEN M., PANG T., JI H.: Self2self with dropout: Learning self-supervised denoising from single image. In *CVPR* (2020), pp. 1890–1898. 1, 2, 3, 6, 7

[ROF92] RUDIN L. I., OSHER S., FATEMI E.: Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena 60*, 1-4 (1992), 259–268. 3

[RP19] REINHARD H., PAUL H.: Deep decoder: Concise image representations from untrained non-convolutional networks. In *ICLR* (2019). 1

[TPL20] TAN M., PANG R., LE Q. V.: Efficientdet: Scalable and efficient object detection. In *CVPR* (2020), pp. 10781–10790. 1

[TYLX17] TAI Y., YANG J., LIU X., XU C.: Memnet: A persistent memory network for image restoration. In *ICCV* (2017), pp. 4539–4547. 1

[UVL18] ULYANOV D., VEDALDI A., LEMPITSKY V.: Deep image prior. In *CVPR* (2018), pp. 9446–9454. 1, 2, 3, 4, 5, 6

[WFYH20] WEI K., FU Y., YANG J., HUANG H.: A physics-based noise formation model for extreme low-light raw denoising. In *CVPR* (2020), pp. 2758–2767. 3

[WLC*20] WU X., LIU M., CAO Y., REN D., ZUO W.: Unpaired learning of deep image denoising. In *ECCV* (2020), pp. 352–368. 3

[WZF*19] WANG R., ZHANG Q., FU C.-W., SHEN X., ZHENG W.-S., JIA J.: Underexposed photo enhancement using deep illumination estimation. In *CVPR* (2019), pp. 6849–6857. 1

[XHC*20] XU J., HUANG Y., CHENG M.-M., LIU L., ZHU F., XU Z., SHAO L.: Noisy-as-clean: Learning self-supervised denoising from corrupted image. *IEEE Transactions on Image Processing* (2020). 1, 2, 3, 4, 6

[XXC12] XIE J., XU L., CHEN E.: Image denoising and inpainting with deep neural networks. In *NeurIPS* (2012), pp. 341–349. 1

[YYZ*19] YUE Z., YONG H., ZHAO Q., MENG D., ZHANG L.: Variational denoising network: Toward blind noise modeling and removal. In *NeurIPS* (2019), pp. 1690–1701. 3

[ZAK*20] ZAMIR S. W., ARORA A., KHAN S., HAYAT M., KHAN F. S., YANG M.-H., SHAO L.: Cycleisp: Real image restoration via improved data synthesis. In *CVPR* (2020), pp. 2696–2705. 1, 2, 3, 7

[ZMI13] ZONTAK M., MOSSERI I., IRANI M.: Separating signal from noise using patch recurrence across scales. In *CVPR* (2013), pp. 1195–1202. 2, 4, 5

[ZNZ19] ZHANG Q., NIE Y., ZHENG W.-S.: Dual illumination estimation for robust exposure correction. In *Computer Graphics Forum* (2019), vol. 38, Wiley Online Library, pp. 243–252. 1

[ZNZ*20] ZHANG Q., NIE Y., ZHU L., XIAO C., ZHENG W.-S.: Enhancing underexposed photos using perceptually bidirectional similarity. *IEEE Transactions on Multimedia 23* (2020), 189–202. 1

[ZNZWS22] ZHANG Q., NIE Y., ZHU L., WEI-SHI Z.: High-quality unsupervised image denoising via multi-scale deep image prior. In *CVM* (2022). 3

[ZNZX15] ZHANG Q., NIE Y., ZHANG L., XIAO C.: Underexposed video enhancement via perception-driven progressive fusion. *IEEE Transactions on Visualization and Computer Graphics 22*, 6 (2015), 1773–1785. 1

[ZTK*20] ZHANG Y., TIAN Y., KONG Y., ZHONG B., FU Y.: Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020). 3

[ZYX*18] ZHANG Q., YUAN G., XIAO C., ZHU L., ZHENG W.-S.: High-quality exposure correction of underexposed photos. In *ACM MM* (2018), pp. 582–590. 1

[ZZC*17] ZHANG K., ZUO W., CHEN Y., MENG D., ZHANG L.: Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing 26*, 7 (2017), 3142–3155. 1, 2, 3, 6, 10

[ZZGZ17] ZHANG K., ZUO W., GU S., ZHANG L.: Learning deep cnn denoiser prior for image restoration. In *CVPR* (2017), pp. 3929–3938. 3

[ZZZ18] ZHANG K., ZUO W., ZHANG L.: FFDNet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing 27*, 9 (2018), 4608–4622. 3