TEST-TIME ADAPTATION FOR REAL-WORLD DENOISING NETWORKS VIA NOISE-AWARE IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Image denoising aims for a challenging task of recovering clean images from unseen noise, which can follow different distributions depending on scenes and camera settings, such as sensors and ISO settings. Previous works have attempted to handle unseen noise by adapting denoising neural networks to each given noisy image. However, a single noisy image can only provide a limited amount of information for training networks. Therefore, we propose to generate noisy images with diverse yet realistic noise that is similar to noise in a given input image. Such noise generation is difficult to achieve given only a single noisy image. To address the challenge, we propose a normalizing flow (NF) framework that can learn the latent representation of noise, conditioned on noisy images. We also employ the Gaussian mixture model to better handle real-world unseen noise by leveraging multiple noise distributions. Using the proposed NF model, our framework can generate multiple synthetic noisy images to facilitate the adaptation of denoising networks to each given image. To further improve the adaptation to unseen noise, we integrate a meta-learning algorithm into our framework. The experimental results demonstrate that our framework substantially improves the performance of several denoising networks on unseen real-world noise across numerous real-world benchmark datasets.

1 INTRODUCTION

Noise is one of the most common and undesired artifacts in the field of signal processing. Therefore, denoising has been studied for many decades across many research areas, among which image denoising is a task to restore a clean image by removing noise from the input image. Recently, there have been dramatic improvements in image denoising through the development of convolutional neural networks (CNN) (Zhang et al., 2017; Chen et al., 2021b; Anwar & Barnes, 2019; Zamir et al., 2020b). Despite their impressive achievements, these supervised learning-based algorithms have critical limitations in that they rely on a simple assumption that added noise follows known simple distributions, such as Gaussian and Poisson. Such assumption hinders the supervised learning-based algorithms from generalizing to real-world noises that do not follow the well-known, simple distributions.

To enable supervised learning-based algorithms to deal with real-world noise, there has been several attempts to collect large datasets consisting of clean-noise image pairs (Plotz & Roth, 2017; Nam et al., 2016; Xu et al., 2018; Anaya & Barbu, 2018; Abdelhamed et al., 2018; 2020). Among them, one of widely used datasets for training is SIDD (Abdelhamed et al., 2018) dataset, which consists of real-world noisy images captured by several smartphone cameras under different ISO, shutter speed, and aperture settings. However, the acquisition of such images requires a lot of time and cost, thereby limiting the volume of real noisy datasets in quantity.

To alleviate these constraints, generative models (Yue et al., 2020; Zamir et al., 2020a; Abdelhamed et al., 2019; Jang et al., 2021; Chang et al., 2020) have been actively studied. Generative approaches adopt generative adversarial network (GAN) or normalizing flow (NF) to learn the distribution of real noise to generate realistic noisy images. Although these previous generative models have shown impressive results, most studies have the limitation that they cannot synthesize real-world noise depending on camera settings, which highly affects the noise distribution (Abdelhamed et al., 2018). Some works (Abdelhamed et al., 2019; Kousha et al., 2022) parameterize the noise types using gain setting and camera model type to generate specific noise, but they only exploit the parameters to synthesize in-distributed noise. Furthermore, considering that only noisy images exist in most realistic scenarios, these approaches have the constraint in a practical view that they require clean images to generate corresponding noisy images. In contrast, some denoising techniques address the data shortage problem in a self-supervised manner (Lehtinen et al., 2018; Batson & Royer, 2019; Krull et al., 2019; Huang et al., 2021). In particular, N2S (Batson & Royer, 2019) and N2V (Krull et al., 2019) can remove noise with unknown distribution by adapting the denoising networks at test-time. However, they require a large number of gradient updates for the test-time adaptation and produce inferior performance compared to supervised learning-based approaches, which can fully utilize distributions of paired datasets, on conventional benchmark datasets.

In this work, combining the advantages of these approaches, we present a new two-stage denoising framework to handle real-world noisy images with unknown noise distributions in sRGB space, which first generates input-noise specific synthetic datasets and then adapts conventional denoising networks with the synthetic datasets. Unlike conventional generative models, ours can fully leverage multiple different noise distributions of train datasets at the generation phase given only a single noisy image, and thus allows us to handle unknown noise distribution. Specifically, we train an NF to transform noisy images with different noise distributions to latent vectors following corresponding Gaussian distributions. With the trained NF and a single noisy input image, we synthesize new noisy images, and then, train denoising networks with the generated noisy images without using ground-truth clean images. Furthermore, we facilitate the test-time adaptation by integrating with a meta-learning scheme, demonstrating the superiority of the proposed algorithms on numerous real-world noise benchmark datasets.

2 Related Work

Deep Image Denoising Many deep learning approaches, including recent transformer-based IPT (Chen et al., 2021a) and SwinIR (Liang et al., 2021), have shown promising results in removing noise with known distributions (*e.g.,* Gaussian). Moreover, there has been several attempts to remove real-world noise whose distribution is not known (Guo et al., 2019; Anwar & Barnes, 2019). However, as it is not easy to acquire pairs of real noisy images and corresponding clean images for training, self-supervised approaches, which do not require clean ground-truth target images, have been studied. For instance, N2N (Lehtinen et al., 2018) is trained with only differently corrupted multiple noisy images under an assumption that the expectation of the added random noise is zero, and N2S (Batson & Royer, 2019) and N2V (Krull et al., 2019) algorithms train the networks with only a single noisy image in a self-supervised manner. AP-BSN (Lee et al., 2022) expands the capability of the N2V to remove real-world sRGB noise using a pixel-shuffle operation. Although these self-supervised approaches can remove noise from unknown distributions, the methodologies entirely depend on a few statistical assumptions and can utilize only a single training image where these simple settings are not enough to handle complex real-world noises.

Noisy Image Generation Due to the lack of real-noise datasets, there are several researches to generate realistic noisy images based on GAN frameworks. CycleISP (Zamir et al., 2020a) presents a CNN-based framework and exploits the sRGB space rather than RAW by considering camera image pipelines. C2N (Jang et al., 2021) is trained with unpaired clean and noisy images and generates synthetic noisy images with adversarial loss. DANet (Yue et al., 2020) learns the joint distribution of the clean and noisy image pair to map the noisy image to the clean one, enabling simultaneous noise generation and elimination. CA-NoiseGAN (Chang et al., 2020) encodes camera-dependent noise information using contrastive learning and generates realistic noise conditioned on that information. However, these generative models also have limitations in being applied in the real-world scenario where only noisy images are accessible but corresponding clean images are not available.

Moreover, there are some generative models based on conditional NF rather than GANs since NF can be trained easily and generate more diverse images. SR Flow (Lugmayr et al., 2020) applies NF to the super-resolution task to learn the distribution of HR images conditioned on LR images, and Noise Flow (Abdelhamed et al., 2019) learns noise distribution of images captured by smartphone cameras conditioned on the camera type and gain setting (*e.g.,* ISO). Notably, NF can also be used as a generative classifier by transforming the input images to corresponding different distributions (Mackowiak et al., 2021; Izmailov et al., 2020; Ardizzone et al., 2020). In our work, we extend this capability of NF to learn multiple different noise distributions in real-world datasets, and we leverage them to handle real noise with unknown distributions.

Test-time Adaptation via Meta-Learning In recent years, interest in the field of meta-learning has increased significantly. Unlike traditional deep learning algorithms, meta-learning can learn from experiences in multiple train episodes. It has many advantages (*e.g.,* data and computing efficiency) and can also improve the deep learning model's generalization and rapid learning ability. Meta-learning has opened up many opportunities in the field of computer vision. Particularly, MAML (Finn et al., 2017) has achieved great success in few-shot classification and has been also employed in various low-level vision tasks. For instance, MAML has been integrated into a single-image super-resolution task to handle unknown degradation kernel (Park et al., 2020) and boost the test-time adaptation speed (Soh et al., 2020). Also, some researches suggest that meta-learning algorithms can be applied to denoising (Lee et al., 2020; Li et al., 2021), and video frame interpolation problems (Choi et al., 2020). These meta-learned networks can be adapted with only a few number of gradient updates to a specific test input and elevate the performance of the conventional networks without changing their original architecture at test-time.

3 PROPOSED METHOD

We propose a two-stage denoising framework that first generates noisy images given noisy input, then adapts the denoising networks to the specific input using the synthetic images at test-time.

3.1 LEARNING NOISE DISTRIBUTION VIA CONDITIONAL NORMALIZING FLOW

We aim to generate realistic noisy images to solve the real-world denoising problem. To generate such synthetic noisy images, it is required to learn distributions of real noisy images as in (Zamir et al., 2020a; Jang et al., 2021; Yue et al., 2020). In this work, we learn the distribution of noisy images in the SIDD dataset (Abdelhamed et al., 2018) since images in the SIDD dataset are captured by various smartphone cameras with different ISO settings which allows us to learn multiple different real-world noise distributions.

To model the noise distribution, we first employ a conditional NF (Rezende & Mohamed, 2015) to map a noisy image to a latent as:

$$z_c = f_\theta(y_c; x) \Longleftrightarrow y_c = f_\theta^{-1}(z_c; x), \tag{1}$$

where y_c denotes a real noisy image captured with camera configuration c. Notably, our camera configuration c consists of smartphone model and ISO setting (*e.g.,* iPhone7 with ISO 1600). The noisy image is mapped to a latent z_c by the invertible conditional NF f with parameter θ, and the condition is given by the corresponding ground truth clean image x. Since noisy images in the SIDD dataset are captured under various camera settings, there exists multiple different noise distributions as introduced in (Abdelhamed et al., 2019). Thus, we learn the noise distributions using camera-configuration specific normal distributions through a single NF as,

$$z_c \sim \mathcal{N}(\mu_c, 1), \tag{2}$$

where μ_c denotes the mean of the specific normal distribution for the configuration c and is also a trainable parameter while assuming unit-variance distribution. Then, we can compute the conditional probability density function of y_c given x as:

$$p_\theta(y_c|x) = \mathcal{N}(z_c; \mu_c, 1) \cdot |\text{det}Df_\theta(y_c; x)|, \tag{3}$$

where $Df_\theta(y_c; x)$ denotes Jacobian of f_θ, and we train the NF by minimizing the negative log-likelihood (NLL) function as follows:

$$\begin{aligned} \mathcal{L}_{NLL}(\theta) &= -\sum_{c=1}^{C} \text{log}p_\theta(y_c|x) \\ &= -\sum_{c=1}^{C} (\text{log}\mathcal{N}(z_c; \mu_c, 1) + \text{log}|\text{det}Df_\theta(y_c; x)|), \end{aligned} \tag{4}$$

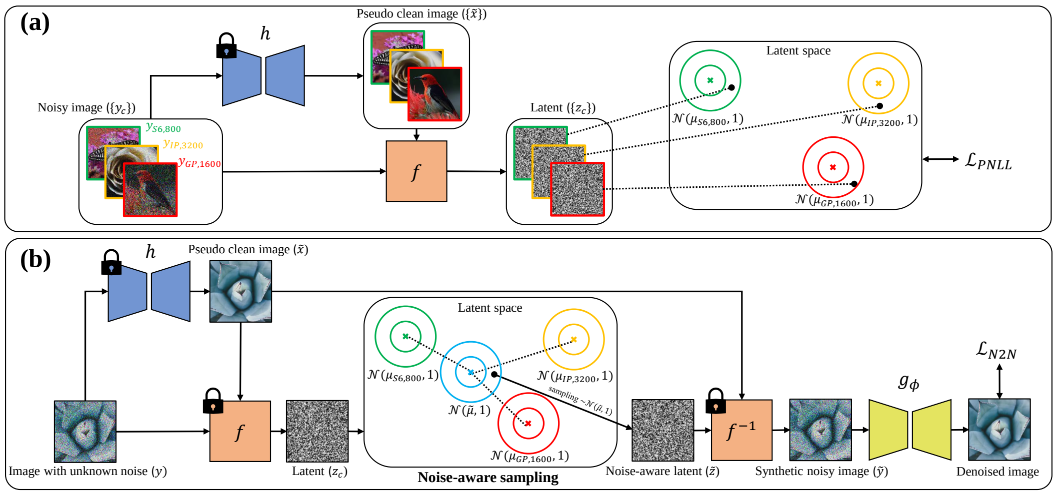

Figure 1: Overview of our proposed framework. (a) The training procedure of our Noise-aware image generation Network (NaN), which uses conditional normalizing flow (NF) network f to map noisy images to Gaussian distributions on the latent space, where each Gaussian distribution corresponds to the camera configuration used for capturing images. NaN is conditioned on input noisy image and pseudo clean image, where pseudo clean image is obtained by a lightweight pseudo clean generator h. (b) The pipeline of test-time adaptation process during inference. We use Gaussian mixture model to obtain more accurate latent representation of given image with unseen noise. From the obtained latent representation, NF generates new synthetic noisy images, which are used to adapt the parameters ϕ of denoising networks g with a standard self-supervised loss function $\mathcal{L}_{\text{N2N}}$.

and C means the number of camera configurations available in the SIDD dataset.

Through the trained conditional NF, we can synthesize noisy images which have similar distribution to y_c by taking random samples from $\mathcal{N}(\mu_c, 1)$ as an input of the f_θ^{-1} given ground truth clean image x as the condition. However, in real scenario, ground truth clean image x corresponding to the noisy input is not available, and thus it is not easy to directly apply the generation process in Equation 1. Therefore, we use a pseudo clean image $\tilde{x}$ for conditioning of NF where $\tilde{x}$ is acquired by feeding the noisy input to a pseudo clean generator h (*i.e.,* $\tilde{x} = h(y_c)$). Thus, we minimize the modified NLL which utilizes $\tilde{x}$ as the condition for our NF as follows:

$$\mathcal{L}_{PNLL}(\theta) = -\sum_{c=1}^{C}(\log\mathcal{N}(z_c; \mu_c, 1) + \log|\det Df_\theta(y_c; \tilde{x})|). \tag{5}$$

Note that conventional denoiser trained on the SIDD dataset can be used for the pseudo clean generator h, and we use DnCNN (Zhang et al., 2017) in this work by considering both performance and speed. In Figure 1(a), a sketch of the proposed NF architecture is illustrated.

3.2 Noise-Aware Image Generation

In this work, we generate noisy images to adapt denoising networks to the given noisy input y with unknown distribution, and thus, we synthesize images whose noise distributions are similar to the distribution of the noise within the input. To do so, we utilize the capability of the NF as a generative classifier (Mackowiak et al., 2021; Izmailov et al., 2020). Specifically, we can obtain a latent z from the noisy input y through the trained NF and measure the fidelity of that latent z for each learned normal distribution (*i.e.,* $\{\mathcal{N}(z; \mu_c, 1)\}$). Then, using these C likelihood values, we can mix the learned C different noise distributions and generate input-noise specific noisy images from the mixture model. We dub this algorithm **NaN** (**N**oise-**a**ware image generation **N**etwork), and the detail of the proposed NaN method is described in Algorithm 1.

Algorithm 1 Noise-Aware Image Generation Network (NaN)

Input: noisy input y

1: $\tilde{x} = h(y)$

2: $z = f_\theta(y; \tilde{x})$

3: Sample $\tilde{z} \sim \mathcal{N}(\tilde{\mu}, 1)$, where $\tilde{\mu} = \sum_c \frac{\mathcal{N}(z;\mu_c,1)}{\sum_c \mathcal{N}(z;\mu_c,1)} \cdot \mu_c$ `// Noise-aware sampling`

4: **return** $\tilde{y} = f_\theta^{-1}(\tilde{z}; \tilde{x})$

Notably, in this study, we train the latent space by using multiple distinct normal distributions to learn many different noise distributions. Then, using the Gaussian Mixture Model (GMM) of these learned distributions in the latent space, we can generate noisy images which follow their mixture distribution where the mixture coefficients are controllable. In our experiments, we demonstrate that the proposed NaN algorithm allows us to handle real-world noisy images even with unseen noise distributions with the aid of our noise-aware generation mechanism.

3.3 Test-time Adaptation for Real-world Denoising

Recent self-supervision-based denoising approaches N2S and N2V have demonstrated that denoising networks can be trained without using the ground-truth clean image. Although these self-supervised methods present novel mechanisms for test-time adaptation and removing real-world noise with unseen distribution, they require statistical assumptions and a large number of gradient updates to adapt the networks parameters to the specific input noise since they use blind-spot methods to train with a single noisy input. To alleviate this problem, we propose a new two-stage test-time adaptation technique for real-world denoising. We first synthesize multiple noisy images with our NaN approach in Algorithm 1 given a single noisy input, and then further update pre-trained denoising networks using the synthetic noisy images at test-time. Specifically, we can adapt the network parameter of the denoiser with only synthetic noisy images (*i.e.,* without ground-truth clean image), and we use the unsupervised loss introduced in N2N (Lehtinen et al., 2018) to train the denoiser, and it is given by,

$$\mathcal{L}_{N2N}(\phi) = E[||g_\phi(\tilde{y}) - y||], \tag{6}$$

where y is an input noisy image given at test time, and $\tilde{y}$ denotes our synthetic noisy image generated by Algorithm 1. Pre-trained denoising network is g and its parameter ϕ can output enhanced denoising result after the adaptation. Algorithm for our **T**est-time **P**arameter **A**daptation (TPA) is described in Algorithm 2. Note that, as we can generate differently corrupted multiple images, we naturally employ the N2N loss. Therefore, we do not need to use blind-spot methods used in N2S and N2V to train with only a single noisy input, which greatly reduce training efficiency.

Algorithm 2 Test-time Parameter Adaptation (TPA)

Input: noisy input y

Require: pre-trained denoiser g_ϕ, adaptation number M, learning rate α

for $i \leftarrow 1$ *to* M **do**

  Generate a noisy image $\tilde{y}$ from Algorithm 1

  $\mathcal{L}_{N2N}(\phi) = ||g_\phi(\tilde{y}) - y||$

  $\phi \leftarrow \phi - \alpha \nabla_\phi \mathcal{L}_{N2N}(\phi)$

end

return ϕ `// return adapted denoising network parameter`

3.4 Fast Test-time Adaptation via Meta-Learning

Owing to the generation of realistic noise images via our proposed NaN, a denoising network can adapt to the given noisy image and improve its denoising performance, as discussed in the experimental section. To better facilitate the adaptation process, we employ model-agnostic meta-learning (MAML) algorithm (Finn et al., 2017), which is known for its capability of adapting to new tasks. MAML achieves such adaptation ability by learning a common initialization, from which the parameters are adapted with a small number of steps to each given task. In this work, we define a task as removing a certain type of noise that corresponds to each camera configuration c. Thus, we learn an initialization of denoiser network parameters using C(=34) different tasks (camera configurations) from the SIDD dataset. From the learned initialization, the denoiser network parameters are updated for the adaptation to each new noisy image.

3.4.1 Meta-train and Fast Test-time Parameter Adpatation

We describe our MAML-based training scheme in Algorithm 3. First, we sample a pair of clean and noisy images (x, y_c) with a specific camera configuration c from the SIDD train dataset. MAML employs a bilevel optimization for training: an inner-loop optimization for adapting parameters to each task and an outer-loop optimization for updating an initialization. The inner-loop optimization is similar to Algorithm 2 in that we use NaN to generate new synthetic noisy images $\tilde{y_c}$ from y_c. In turn, these synthetic images are used to minimize the unsupervised loss in Equation 6 to adapt the parameters of the denoiser network g. After the inner-loop optimization, the adapted denoiser g_{ϕ_c} performs denoising on a synthetic noisy image $\tilde{y_c}$ to obtain an estimated clean image $g_{\phi_c}(\tilde{y_c})$. The outer-loop optimization updates the initialization ϕ to minimize the difference between input noisy and ground-truth clean images x across a batch of camera configurations (tasks). In practice, we observe that using the synthetic noisy image for the outer-loop optimization elevates the denoising performance slightly, we utilize the generated noisy images for the initialization with probability p, and we use p=0.5 in our experiments. Moreover, for the inner-loop optimization, we use the fixed number of update steps (M=5).

Algorithm 3 Meta-train Algorithm

Require: pre-trained denoiser g_ϕ, adaptation number M, learning rate α and β

while *until convergence* **do**
  Sample a pair of clean and noisy images (x, y_c) from the SIDD traindataset
  $\phi_c \leftarrow \phi$
  for $i \leftarrow 1$ *to* M **do**
    Generate $\tilde{y_c}$ from y_c with NaN Algorithm 1
    $\mathcal{L}_{N2N}(\phi_c) = ||g_{\phi_c}(\tilde{y_c}) - y_c||$
    $\phi_c = \phi_c - \alpha \nabla_{\phi_c} \mathcal{L}_{N2N}(\phi_c)$
  end
  Generate $\tilde{y_c}$ from y_c with NaN Algorithm 1
  $r \sim Bernoulli(p)$ `// sample 1 with probability` p
  $Loss(\phi_c) = ||g_{\phi_c}(r \cdot y_c + (1-r) \cdot \tilde{y_c}) - x||$
  $\phi \longleftarrow \phi - \beta \nabla_\phi \Sigma_c Loss(\phi_c)$
end
return ϕ `// return meta-trained denoising network parameter`

During inference, the meta-learned initialization parameters of denoiser are adapted to the given input noisy image using our generated noise images via our **F**ast test-time **P**arameter **A**dpatation (FPA). The adaptation process is the same as inner-loop optimization procedure during meta-training. The adapted denoiser then performs denoising on the input noisy image to obtain a clean image.

4 Experimental Results

Please see our supplementary material for more implementation details and results.

Experimental settings Our NaN is implemented based on the conditional NF framework introduced in SRFlow (Lugmayr et al., 2020). For training NaN, we use the SIDD sRGB train dataset with 34(=C) different camera configurations, and minimize the $\mathcal{L}_{PNLL}$ loss in equation 5 using the Adam optimizer (Kingma & Ba, 2014). Moreover, for the pseudo clean generator h, we use DnCNN (Zhang et al., 2017) fully-trained on the equivalent dataset (SIDD). For evaluation, we measure the performance of the proposed methods on the public datasets, including real noisy images; SIDD (Abdelhamed et al., 2018), SIDD+ (Abdelhamed et al., 2020), Nam (Nam et al., 2016), and PolyU (Xu et al., 2018).

4.1 Quantitative and Qualitative Results

Denoising performance In Table 1, we report the denoising results from our proposed adaptation algorithm with several g networks: DnCNN (Zhang et al., 2017), RIDNet (Anwar & Barnes, 2019),

Denosing Network	Method	SIDD Test		SIDD+		Nam		PolyU		Avg.	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
DnCNN	Fully-pretrained	37.95	0.9440	35.55	0.8883	37.96	0.9502	37.81	0.9539	37.32	0.9341
	+ N2S (M=5)	37.71	0.9410	35.44	0.8843	38.23	0.9536	37.89	0.9554	37.32	0.9336
	+ N2S (M=10)	37.23	0.9340	35.22	0.8752	38.16	0.9523	37.87	0.9551	37.12	0.9292
	+ TPA (M=5)	38.02	**0.9450**	35.68	0.8933	38.58	0.9563	38.17	0.9595	37.61	0.9385
	+ FPA (M=5)	**38.04**	**0.9450**	**35.78**	**0.8966**	**38.67**	**0.9574**	**38.23**	**0.9605**	**37.68**	**0.9339**
RIDNet	Fully-pretrained	**39.05**	0.9530	36.21	0.9058	37.85	0.9556	37.93	0.9606	37.76	0.9438
	+ N2S (M=5)	38.95	0.9530	36.16	0.9029	38.17	0.9558	38.03	0.9601	37.83	0.9430
	+ N2S (M=10)	38.53	0.9510	35.78	0.8929	38.16	0.9538	38.00	0.9590	37.62	0.9392
	+ TPA (M=5)	39.02	**0.9540**	36.48	0.9114	38.49	0.9579	38.21	0.9620	38.05	0.9463
	+ FTA (M=5)	38.98	0.9530	**36.58**	**0.9140**	**38.77**	**0.9610**	**38.32**	**0.9630**	**38.16**	**0.9478**
HINet	Fully-pretrained	**39.87**	**0.9600**	36.39	0.9016	38.04	0.9554	37.95	0.9606	38.06	0.9444
	+ N2S (M=5)	39.72	0.9590	35.62	0.8680	38.23	0.9525	38.10	0.9589	37.92	0.9346
	+ N2S (M=10)	39.04	0.9540	34.09	0.8007	38.00	0.9445	37.85	0.9537	37.25	0.9132
	+ TPA (M=5)	39.84	**0.9600**	36.66	0.9045	38.49	0.9563	38.18	0.9615	38.29	0.9456
	+ FPA (M=5)	**39.87**	**0.9600**	**36.84**	**0.9117**	**38.53**	**0.9575**	**38.19**	**0.9618**	**38.36**	**0.9478**
AP-BSN	Fully-finetuned	36.91	0.9310	35.84	0.9165	38.37	0.9621	38.19	0.9579	37.40	0.9322

Table 1: Adaptation performance on the SIDD, SIDD+, Nam, PolyU dataset is evaluated in terms of PSNR and SSIM. Best and second best denoising results are **highligted** and underlined. Our TPA approach outperforms conventional self-supervised denoising approach N2S, and our FPA approach produces the best denoising/adaptation results.

Method	GT clean	SIDD val.	SIDD+	Nam	PolyU	Avg.
DANet	✓	0.0932	0.2679	0.2670	0.2758	0.2260
GDANet	✓	0.1871	0.8273	**0.1980**	**0.1082**	0.3302
CycleISP	✓	0.3897	0.3065	0.7145	0.8373	0.5620
C2N	✓	0.1638	0.2110	0.4618	0.5904	0.3568
NaN	✗	**0.0389**	**0.0519**	0.2805	0.3094	**0.1702**

Table 2: Accuracy of generated noisy images from DANet, CycleISP, C2N, and our NaN are compared in terms of KLD.

Denoising	SIDD val.	SIDD+	Nam	PoluU	Avg.
RIDNet + DANet	39.04/0.9159	36.25/0.9098	38.00/0.9575	38.01/0.9617	37.83/0.9362
RIDNet + GDANet	39.07/0.9154	36.15/0.9070	38.18/**0.9605**	38.13/**0.9637**	37.88/0.9367
RIDNet + CycleISP	39.05/**0.9160**	36.22/0.9071	37.82/0.9549	37.90/0.9604	37.75/0.9346
RIDNet + C2N	39.06/0.9154	36.22/0.9090	38.11/0.9594	38.02/0.9624	37.85/0.9366
RIDNet + **TPA**	**39.12**/0.9158	**36.48/0.9114**	**38.49**/0.9579	**38.21**/0.9620	**38.08/0.9368**

Table 3: Results of RIDNet finetuned by synthetic images from DANet, GDANet, CycleISP, C2N, and TPA are compared in terms of PSNR/SSIM values.

and HINet (Chen et al., 2021b). These denoising networks are fully pre-trained on the SIDD trainset.[1] We evaluate the capability of the algorithms while handling unseen real-world noise in the SIDD Test, SIDD+, Nam, and PolyU datasets. Moreover, we compare our methods against other self-supervised denoising algorithms: N2S (Batson & Royer, 2019)) and AP-BSN (Lee et al., 2022). Similar to ours, N2S is model-agnostic and thus can be used to finetune pre-trained networks in a similar way to our framework, and the denoising results after 5 gradient updates (M=5) and 10 gradient updates (M=10) are reported. On the other hand, AP-BSN requires the dedicated blind-spot network architecture as the backbone and thus cannot be used across different pre-trained networks g. Therefore, we use AP-BSN pre-trained on the SIDD trainset and then fully finetune on other datasets according to (Lee et al., 2022). On average and almost all datasets and pre-trained network baselines, our algorithm demonstrates substantial performance improvement compared to N2S and AP-BSN. These experimental results demonstrate the applicability of our algorithm across different networks and noise adaptation capability across different datasets. This is particularly noticeable in SIDD+, Nam, and PolyU datasets that contain significantly different noises from those already seen in the train set. Furthermore, our algorithm is shown to benefit more from MAML-based adaptation (**FPA**) than typical finetune adaptation (**TPA**). Thus, MAML is shown to enable our algorithm to better adapt to new unseen noise. Moreover, our algorithm is shown to render sharper edges and tiny details while suppressing the noise in the texture-less regions, as shown with real-world denoising result visualization (Figure 2).

Noise generation performance The denoising performance of our overall framework hinges on the quality of generated noisy images. As such, we quantify the noise generation quality in terms of Kullback-Leibler divergence (KLD) and compare against DANet (Yue et al., 2020), GDANet (Yue et al., 2020), CycleISP (Zamir et al., 2020a),and C2N (Jang et al., 2021) in Table 2. We use officially available pre-trained parameters of DANet, CycleISP, and C2N, which are also trained on the SIDD sRGB trainset. Note that we use official parameters of GDANet, which are obtained after training on the SIDD, PolyU, and Renoir dataset Anaya & Barbu (2018).

Although previous works need ground-truth clean images to generate noisy images, our NaN uses only the given noisy input and stills shows better noise generation performance on average. To

[1]For HINet, we use official parameters of the network trained on the SIDD Dataset.

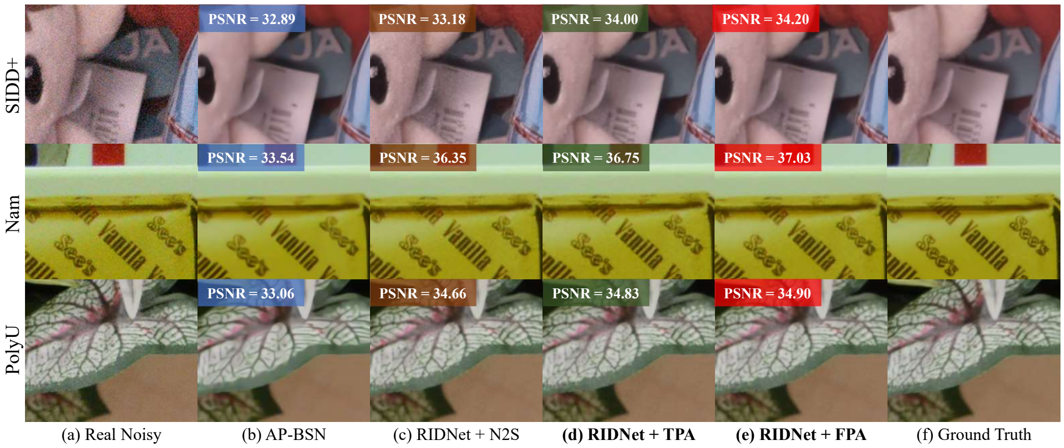

Figure 2: Qualitative comparison results on SIDD+, Nam and PolyU images.

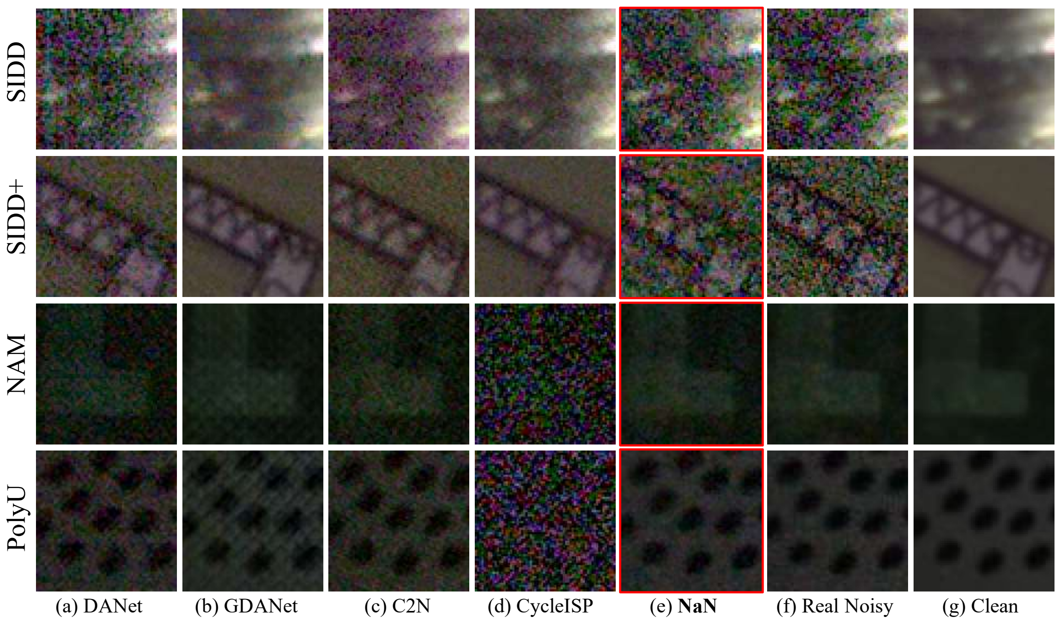

Figure 3: Qualitative results on synthesized noisy images in SIDD, SIDD+, NAM, and PolyU datasets. Our NaN generates more realistic noisy images compared to other generative models.

better assess the performance of noise generation, we visualize noisy images generated by each different generative model in Figure 3. Other generative models seem to generate similar noise across different datasets or generate noise that is different from given real noisy input. On the other hand, our NaN produces realistic noisy images (e) that seem to exhibit noise sampled from the same distribution as each real noisy image (f). This shows our noise-aware generation algorithm successfully approximates unknown noise which (a)-(d) cannot produce.

To further compare the noise generation quality against other noise generation models, we finetune RIDNet (pre-trained on SIDD trainset) with generated noisy images by each model and N2N loss and compare denoising performance on the SIDD validation, SIDD+, Nam, and PolyU datasets, as shown in Table 3. The adaptation with noisy images generated by our NaN exhibits higher performance improvement on average in comparison to other generative models, even though NaN does not use ground-truth clean images while other generative models do. These results suggest that our framework can handle unknown noise even in external datasets (*e.g.,* SIDD+, Nam, PolyU), owing to the capability of our NaN to leverage several different noise distributions.

4.2 Ablation Study

Effect of noise aware sampling Our NaN is trained by assuming C distinct latent distributions, and the latent sampling mechanism presented in Algorithm 1 allows us to generate noise-aware (*i.e.,* input-specific) synthetic images, and we show the efficacy of our mechanism. To do so, we train the conditional NF (f_θ) to have latents that follow a single normal distribution regardless of the camera configuration (*i.e.,* $\mu_c = 0$). In Table 4, we see that noisy images generated from Algorithm 1 when all $\mu_c = 0$ are irrelevant to the specific input noise, and thus the KLD value is much higher than the results by our NaN where $\{\mu_c\}$ are trainable and not zeros.

Latent distribution	SIDD	SIDD+	Nam	PolyU
Single normal distribution (all $\mu_c = 0$)	0.0914	0.1436	0.5946	0.6012
Multiple distinct distributions (trainable $\{\mu_c\}$)	0.0389	0.0519	0.2805	0.3094

Table 4: Efficacy of noise-aware image generation. Two conditional NFs (f_θ) trained under single latent distribution and multiple latent distributions are compared. Noisy images are generated according to Algorithm 1, and we measure the quality of the noisy images in term of KLD on the SIDD, SIDD+, Nam, and PolyU datasets.

Update step We investigate how the performance changes as we vary the number of update steps for test-time adaptation in Figure 4. Across all denoising networks and update steps, our adaptation algorithms bring significantly higher performance improvement than N2S. Notably, our MAML-based FPA with only 5 update steps shows similar performance to our TPA, which takes around 10 update steps to reach a similar performance. Such results demonstrate the efficacy of MAML in facilitating the adaptation process.

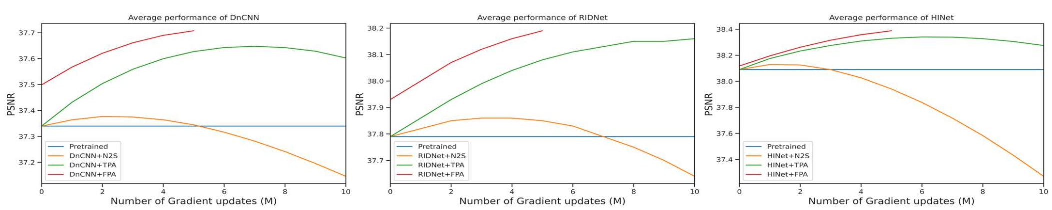

Figure 4: Averaging test-time adaptation results by changing the number of updates (M) on SIDD, SIDD+, Nam, and PolyU datasets. During the 5 gradient updates, our TPA and FPA achieve higher gain in PSNR. Note that we show the results of our FPA with $M = 5$ update steps.

5 Conclusion and Limitation

In this work, we introduce a new test-time adaptation algorithm that enables denoising networks to adapt to unseen and realistic noise without the use of ground-truth clean images. In particular, the proposed adaptation algorithm relies on using synthetic noisy images produced by our noise-aware noisy image generation network, which we coin as NaN. Our NaN employs normalizing flow to synthesize realistic noisy images that exhibit the same noise characteristics as newly given noisy images. Such synthetic noisy images provide rich information of noise present in given noisy images, thereby aiding the adaptation process. Furthermore, we employ a model-agnostic meta-learning algorithm, which further facilitates the adaptation of networks to input images with unseen noise. The experimental results demonstrate the substantial performance improvement brought by our algorithms across several denoising networks, underlining the flexibility and effectiveness of our test-time adaptation algorithm. One may argue that one of the limitations with our framework is the dependence on pseudo clean images and pseudo clean generator. However, as discussed in the supplementary document, our framework demonstrates the robustness to the quality of pseudo clean images to some extent. Regardless, the use of pseudo clean generator itself can be regarded as limitation. Thus, generating diverse noises with only given input noisy image is interesting yet challenging for future research work.

References

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

Abdelrahman Abdelhamed, Marcus A Brubaker, and Michael S Brown. Noise flow: Noise modeling with conditional normalizing flows. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019.

Abdelrahman Abdelhamed, Mahmoud Afifi, Radu Timofte, and Michael S Brown. Ntire 2020 challenge on real image denoising: Dataset, methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Josue Anaya and Adrian Barbu. Renoir–a dataset for real low-light image noise reduction. *Journal of Visual Communication and Image Representation*, 51:144–154, 2018.

Saeed Anwar and Nick Barnes. Real image denoising with feature attention. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019.

Lynton Ardizzone, Radek Mackowiak, Carsten Rother, and Ullrich Köthe. Training normalizing flows with the information bottleneck for competitive generative classification. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Joshua Batson and Loic Royer. Noise2self: Blind denoising by self-supervision. In *International Conference on Machine Learning (ICML)*, 2019.

Ke-Chi Chang, Ren Wang, Hung-Jin Lin, Yu-Lun Liu, Chia-Ping Chen, Yu-Lin Chang, and Hwann-Tzong Chen. Learning camera-aware noise models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021a.

Liangyu Chen, Xin Lu, Jie Zhang, Xiaojie Chu, and Chengpeng Chen. Hinet: Half instance normalization network for image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021b.

Myungsub Choi, Janghoon Choi, Sungyong Baik, Tae Hyun Kim, and Kyoung Mu Lee. Scene-adaptive video frame interpolation via meta-learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, 2017.

Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Tao Huang, Songjiang Li, Xu Jia, Huchuan Lu, and Jianzhuang Liu. Neighbor2neighbor: Self-supervised denoising from single noisy images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Pavel Izmailov, Polina Kirichenko, Marc Finzi, and Andrew Gordon Wilson. Semi-supervised learning with normalizing flows. In *International Conference on Machine Learning (ICML)*, 2020.

Geonwoon Jang, Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. C2n: Practical generative noise modeling for real-world denoising. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Shayan Kousha, Ali Maleky, Michael S Brown, and Marcus A Brubaker. Modeling srgb camera noise with normalizing flows. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Seunghwan Lee, Donghyeon Cho, Jiwon Kim, and Tae Hyun Kim. Self-supervised fast adaptation for denoising via meta-learning. *arXiv preprint arXiv:2001.02899*, 2020.

Wooseok Lee, Sanghyun Son, and Kyoung Mu Lee. Ap-bsn: Self-supervised denoising for real-world images via asymmetric pd and blind-spot network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. In *International Conference on Machine Learning (ICML)*, 2018.

Zheng Li, Jun Li, Chaoyue Wang, Zhiyang Lu, Jun Wang, Hongjian He, and Jun Shi. Meta-learning based interactively connected clique u-net for quantitative susceptibility mapping. *IEEE Transactions on Computational Imaging*, 7:1385–1399, 2021.

Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.

Andreas Lugmayr, Martin Danelljan, Luc Van Gool, and Radu Timofte. Srflow: Learning the super-resolution space with normalizing flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

Radek Mackowiak, Lynton Ardizzone, Ullrich Kothe, and Carsten Rother. Generative classifiers as a basis for trustworthy image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

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 (CVPR)*, 2016.

Seobin Park, Jinsu Yoo, Donghyeon Cho, Jiwon Kim, and Tae Hyun Kim. Fast adaptation to super-resolution networks via meta-learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

Tobias Plotz and Stefan Roth. Benchmarking denoising algorithms with real photographs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning (ICML)*, 2015.

Jae Woong Soh, Sunwoo Cho, and Nam Ik Cho. Meta-transfer learning for zero-shot super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Jun Xu, Hui Li, Zhetong Liang, David Zhang, and Lei Zhang. Real-world noisy image denoising: A new benchmark. *arXiv preprint arXiv:1804.02603*, 2018.

Zongsheng Yue, Qian Zhao, Lei Zhang, and Deyu Meng. Dual adversarial network: Toward real-world noise removal and noise generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Cycleisp: Real image restoration via improved data synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020a.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Learning enriched features for real image restoration and enhancement. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020b.

Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26 (7):3142–3155, 2017.