This CVPR 2020 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Learning Invariant Representation for Unsupervised Image Restoration

Wenchao Du, Hu Chen[†], Hongyu Yang
College of Computer Science, Sichuan University, Chengdu 610065, China
Wenchaodu.scu@gmail.com, huchen@scu.edu.cn, yanghongyu@scu.edu.cn

Abstract

Recently, cross domain transfer has been applied for unsupervised image restoration tasks. However, directly applying existing frameworks would lead to domain-shift problems in translated images due to lack of effective supervision. Instead, we propose an unsupervised learning method that explicitly learns invariant presentation from noisy data and reconstructs clear observations. To do so, we introduce discrete disentangling representation and adversarial domain adaption into general domain transfer framework, aided by extra self-supervised modules including background and semantic consistency constraints, learning robust representation under dual domain constraints, such as feature and image domains. Experiments on synthetic and real noise removal tasks show the proposed method achieves comparable performance with other state-of-the-art supervised and unsupervised methods, while having faster and stable convergence than other domain adaption methods. Code has been released.

1. Introduction

Image restoration (IR) attempts to reconstruct clean signals from their corrupted observations, which is known to be an ill-posed inverse problem. By accommodating different types of corruption distributions, the same mathematical model applies to problems such as image denoising, super-resolution and deblurring. Recently, deep neural networks (*DNNs*) and generative adversarial networks (*GANs*) [10] have shown their superior performance in various low-level vision tasks. Nonetheless, most of these methods need paired training data for specific tasks, which limits their generality, scalability and practicality in real-world multimedia applications. In addition, strong supervision may suffer from the overfitting training and lower generalization to real image corruption types.

More recently, the domain transfer based unsupervised learning methods have attracted lots of attention due to the great progress [9, 18, 20, 21, 40] achieved in style transfer, attribute editing and image translation, *e.g.*, *CycleGAN* [40], *UNIT* [21] and *DRIT* [18]. Although these methods have been expanded to specific restoration tasks, they could not reconstruct the high-quality images due to losing finer details or inconsistency backgrounds, as shown in Fig. 1. Different from *DNNs* based supervised models, which aim at learning a powerful mapping between the noisy and clean images. Directly applying existing domain-transfer methods is unsuitable for generalized image inverse problems due to the following reasons:

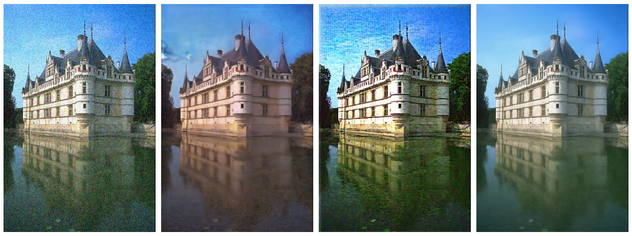

(a) Input (b) *CycleGAN* (c) *UNIT* (d) *Ours*

Figure 1: The typical results for Gaussian Noise. Our method has better ability on noise removal and texture preservation than other domain-transfer methods.

- *Indistinct Domain Boundary.* Image translation aims to learn abstract shared-representations from unpaired data with clear domain characteristics, such as horse-to-zebra, day-to-night, etc. On the contrary, varying noise levels and complicated backgrounds blur domain boundaries between unpaired inputs.
- *Weak Representation.* Unsupervised domain-adaption methods extract high-level representations from unpaired data by shared-weight encoder and explicit target domain discriminator. For slight noisy signals, it is easy to cause domain shift problems in translated images and lead to low-quality reconstruction.
- *Poor Generalization.* Image translation learns a domain mapping from one-to-one image, which hardly captures the generalized semantic and texture representations. This also exacerbates the instability of GAN.

In order to address these problems, inspired by image sparse representation [24] and domain adaption [7, 8], we attempt to learn invariant representation from unpaired samples via domain adaption and reconstruct clean images instead of relying on pure unsupervised domain transfer. Different from general image translation methods [18, 21, 40], our goal is to learn robust intermediate representation free of noise (referred to as *Invariant Representation*) and reconstruct clean observations. Specifically, to achieve this goal, we factorize content and noise representations for corrupted images via disentangled learning; then a representation discriminator is utilized to align features to the expected distribution of clean domain. In addition, the extra self-supervised modules, including background and semantic consistency constraints, are used to supervise representation learning from image domains further.

In short, the main contributions of the paper could be summarized as follows: 1) Propose an unsupervised representation learning method for image restoration based on data-driven, which is easily expanded to other low-level vision tasks, such as super-resolution and deblurring. 2) Disentangle deep representation via dual domain constraints, *i.e.*, feature and image domains. Extra self-supervised modules, including semantic meaning and background consistency modules, further improve the robustness of representations. 3) Build an unsupervised image restoration framework based on cross domain transfer with more effective training and faster convergence speed. To our knowledge, this is the first unsupervised representation learning approach that achieves competing results for processing synthetic and real noise removal with end-to-end training.

2. Related Work

2.1. Single Image Restoraion

Traditional Methods. Classical methods, containing Total Variation [29, 34], *BM3D* [5], Non-local mean [2] and dictionary learning [3, 12], have achieved good performance on general image restoration tasks, such as image denoising, super-resolution and deblurring. In addition, considering that image restoration is in general an ill-posed problem, some methods based on regularization are also proved effective [11, 42].

Deep Neural Networks. Relying on powerful computer sources, data-driven DNN methods have achieved better performance than traditional methods in the past few years. Vincent *et al*. [35] proposed stacked denoising auto-encoder for image restoration. Xie *et al*. [36] combined sparse coding and pre-trained DNN for image denoising and inpainting. Mao *et al*. [26] proposed *RedNet* with symmetric skip connections for noise removal and super-resolution. Zhang *et al*. [39] introduced residual learning for Gaussian noise removal. In general, *DNNs*-based methods could realize superior results on synthetic noise removal via effective supervised training, but it is unsuitable for real-world applications.

2.2. Unsupervised Learning for IR

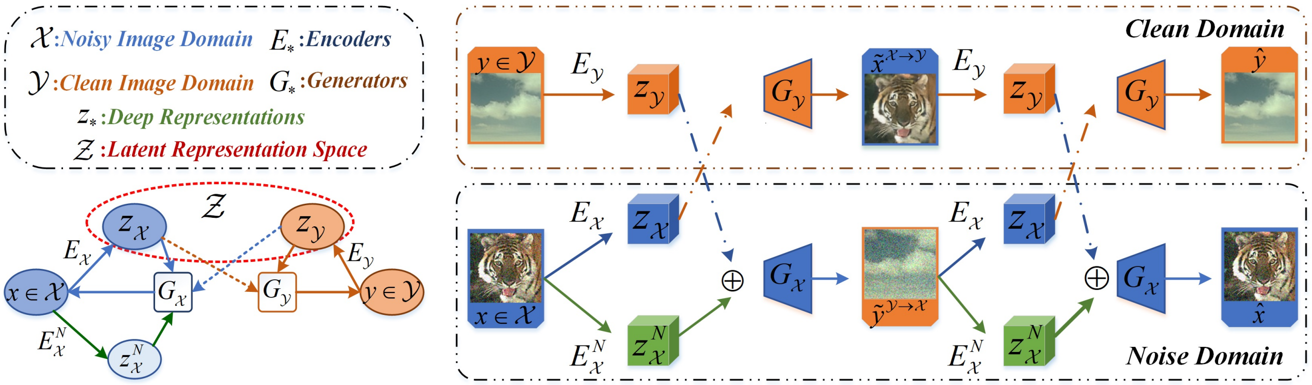
Learning from noisy observations. One interesting direction for unsupervised IR is directly recovering clean signals from noisy observations. Dmitry *et al*. [32] proposed deep image prior (*DIP*) for IR, which requires suitable networks and interrupts its training process based on low-level statistical prior. That is usually unpredictable for different samples. Via zero-mean noise distribution prior, Noise2Noise (*N2N*) [19] directly learns reconstruction between two images with independent noise sampling. That is unsuitable for noise removal in real-world, *e.g.*, medical image denoising. To alleviate this problem, *Noise2Void* [17] predicted a pixel from its surroundings by learning a blind-spot network for corrupted images. Similar to *Noise2Self* [1], this method reduces the training efficiency, but also decreases the denoising performance.

Image Domain Transfer. Another direction solves image restoration by domain transfer, which aims to learn one2one mapping from one domain to another and output image to lie on the manifold of clean image. Previous works, *e.g.*, *CycleGAN* [40], *DualGAN* [37] and *BicycleGAN* [41] have shown great capacity in image translation. Expanding works, containing *CouplesGAN* [22], *UNIT* [21] and *DRIT* [18] learn shared-latent representation for diverse image translation. Along this way, Yuan *et al*. [38] proposed a nested *CycleGAN* to solve the unsupervised image super-resolution. Expanding *DRIT*, Lu *et al*. [23] decoupled image content domain and blur domain to solve image deblurring, referred to as *DRNet*. However, these methods aim to learn stronger domain generators, they require obvious domain boundary and complicated network structure.

3. The Proposed Method

Our goal is to learn abstract intermediate representations from noise inputs and reconstruct clear observations. In a certain way, unsupervised IR could be viewed as a specific domain transfer problem, *i.e.*, from noise domain to clean domain. Therefore, the method is injected into the general domain transfer architecture, as shown in Fig. 2.

In supervised domain transfer, we are given samples (x, y) drawn from a joint distribution $P_{\mathcal{X},\mathcal{Y}}(x, y)$, where $\mathcal{X}$ and $\mathcal{Y}$ are two image domains. For unsupervised domain translation, samples (x, y) are drawn from the marginal distributions $P_{\mathcal{X}}(x)$ and $P_{\mathcal{Y}}(y)$. In order to infer the joint distribution from the marginal samples, a shared-latent space assumption is proposed that there exists a shared latent code z in a shared-latent space $\mathcal{Z}$, so that we can recover both images from this code. Given samples (x, y) from the joint



(a)

(b)

Figure 2: Method Overview. (a) The latent space assumption. Proposed method aims to learn invariant representations from inputs and align them via adversarial domain adaption. (b) Our method is injected into general domain-transfer framework. Extra self-supervised modules are introduced to learn more robust representations.

distribution, this process is presented by

$$z = E_{\mathcal{X}}(x) = E_{\mathcal{Y}}(y) \tag{1}$$

$$x = G_{\mathcal{X}}(z), y = G_{\mathcal{Y}}(z) \tag{2}$$

A key step is how to implement this shared-latent space assumption. To do so, an effective strategy is sharing high-level representation by shared-weight encoder, which samples the features from the unified distribution. However, it is unsuitable for IR that latent representation only contains semantic meanings, which leads to domain shift in recovered images, *e.g*., blurred details and inconsistent backgrounds. Therefore, we attempt to learn more generalized representations containing richer texture and semantic features from inputs, *i.e*., invariant representations. To achieve it, adversarial domain adaption based discrete representation learning and self-supervised constraint modules are introduced into our method. Details are described in the subsections.

3.1. Discrete Representation Learning

Discrete representation aims to compute the latent code z from inputs, where z contains texture and semantic information as much as possible. To do so, we use two auto-encoders to model $\{E_{\mathcal{X}}, G_{\mathcal{X}}\}$ and $\{E_{\mathcal{Y}}, G_{\mathcal{Y}}\}$ separately. Given any unpaired samples (x, y), where $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ separately denote noise and clean sample from different domains, Eq. 1 is reformulated as $z_{\mathcal{X}} = E_{\mathcal{X}}(x)$ and $z_{\mathcal{Y}} = E_{\mathcal{Y}}(y)$. Further, IR could be represented as $F^{\mathcal{X}\to\mathcal{Y}}(x) = G_{\mathcal{Y}}(z_{\mathcal{X}})$. However, considering noise always adheres to high-frequency signals, directly reconstructing clean images is difficult due to varying noise levels and types, which requires powerful domain generator and discriminator. Therefore, we introduce the disentangling representation into our architecture.

Disentangling Representation. For noise sample x, an extra noise encoder $E_{\mathcal{X}}^{N}$ is used to model varying noisy levels and types. The self-reconstruction is formulated by $x = G_{\mathcal{X}}(z_{\mathcal{X}}, z_{\mathcal{X}}^{N})$, where $z_{\mathcal{X}} = E_{\mathcal{X}}(x)$ and $z_{\mathcal{X}}^{N} = E_{\mathcal{X}}^{N}(x)$. Assuming the latent codes $z_{\mathcal{X}}$ and $z_{\mathcal{Y}}$ obey same distribution in shared-space that $\{z_{\mathcal{X}}, z_{\mathcal{Y}}\} \in \mathcal{Z}$, similar to image translation, unsupervised image restoration could be divided into two stages: forward translation and back reconstruction.

Forward Cross Translation. We first extract the representations $\{z_{\mathcal{X}}, z_{\mathcal{Y}}\}$ from (x, y) and extra noise code $z_{\mathcal{X}}^{N}$. Restoration and degradation could be represented by

$$\tilde{x}^{\mathcal{X}\to\mathcal{Y}} = G_{\mathcal{Y}}(z_{\mathcal{X}}) \tag{3}$$

$$\tilde{y}^{\mathcal{Y}\to\mathcal{X}} = G_{\mathcal{X}}(z_{\mathcal{Y}} \oplus z_{\mathcal{X}}^{N}) \tag{4}$$

where $\tilde{x}^{\mathcal{X}\to\mathcal{Y}}$ represents the recovered clean sample, $\tilde{y}^{\mathcal{Y}\to\mathcal{X}}$ denotes the degraded noise sample. $\oplus$ represents channel-wise concatenation operation. $G_{\mathcal{X}}$ and $G_{\mathcal{Y}}$ are viewed as specific domain generators.

Backward Cross Reconstruction. After performing the first translation, reconstruction could be achieved by swapping the inputs $\tilde{x}^{\mathcal{X}\to\mathcal{Y}}$ and $\tilde{y}^{\mathcal{Y}\to\mathcal{X}}$ that:

$$\hat{x} = G_{\mathcal{X}}(E_{\mathcal{Y}}(\tilde{x}^{\mathcal{X}\to\mathcal{Y}}) \oplus E_{\mathcal{X}}^{N}(\tilde{y}^{\mathcal{Y}\to\mathcal{X}})) \tag{5}$$

$$\hat{y} = G_{\mathcal{Y}}(E_{\mathcal{X}}(\tilde{y}^{\mathcal{Y}\to\mathcal{X}})) \tag{6}$$

where $\hat{x}$ and $\hat{y}$ denote reconstructed inputs. To enforce this constraint, we add the cross-cycle consistency loss $\mathcal{L}^{CC}$ for $\mathcal{X}$ and $\mathcal{Y}$ domains:

$$\begin{aligned}\mathcal{L}_{\mathcal{X}}^{CC}(G_{\mathcal{X}}, G_{\mathcal{Y}}, E_{\mathcal{X}}, E_{\mathcal{Y}}, E_{\mathcal{X}}^{N}) = \\ \mathbb{E}_{\mathcal{X}}\left[\|G_{\mathcal{X}}(E_{\mathcal{Y}}(\tilde{x}^{\mathcal{X}\to\mathcal{Y}}) \oplus E_{\mathcal{X}}^{N}(\tilde{y}^{\mathcal{Y}\to\mathcal{X}})) - x\|_1\right]\end{aligned} \tag{7}$$

$$\begin{aligned}\mathcal{L}_{\mathcal{Y}}^{CC}(G_{\mathcal{X}}, G_{\mathcal{Y}}, E_{\mathcal{X}}, E_{\mathcal{Y}}, E_{\mathcal{X}}^{N}) = \\ \mathbb{E}_{\mathcal{Y}}\left[\|G_{\mathcal{Y}}(E_{\mathcal{X}}(\tilde{y}^{\mathcal{Y}\to\mathcal{X}}) - y\|_1\right]\end{aligned} \tag{8}$$

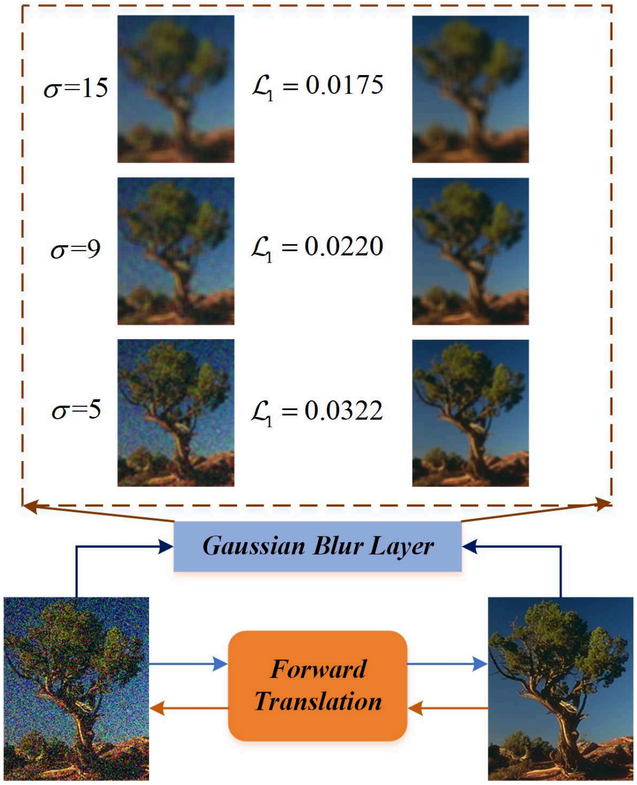


Figure 3: *Background Consistency Module (BCM). BCM* hierarchically uses $\mathcal{L}_1$ loss at different Gaussian-Blur levels to ensure the inputs and outputs have consistency background.

Adversarial Domain Adaption. Another factor is how to embed latent representations $z_{\mathcal{X}}$ and $z_{\mathcal{Y}}$ into shared space. Inspired by unsupervised domain adaption, we implement it by adversarial learning instead of shared-weight encoder. Our goal is to facilitate representations from inputs obeying the similar distribution while preserving richer texture and semantic information of inputs. Therefore, a representation discriminator $D_{\mathcal{R}}$ is utilized in our architecture. We express this feature adversarial loss $\mathcal{L}^{\mathcal{R}}_{adv}$ as

$$\begin{aligned}&\mathcal{L}^{\mathcal{R}}_{adv}(E_{\mathcal{X}}, E_{\mathcal{Y}}, D_{\mathcal{R}}) = \\ &\quad \mathbb{E}_{\mathcal{X}}\left[\frac{1}{2}\log D_{\mathcal{R}}(z_{\mathcal{X}}) + \frac{1}{2}\log(1 - D_{\mathcal{R}}(z_{\mathcal{X}}))\right] + \\ &\qquad \mathbb{E}_{\mathcal{Y}}\left[\frac{1}{2}\log D_{\mathcal{R}}(z_{\mathcal{Y}}) + \frac{1}{2}\log(1 - D_{\mathcal{R}}(z_{\mathcal{Y}}))\right]\end{aligned} \tag{9}$$

3.2. Self-Supervised Constraint

Due to lack of effective supervised signals for translated images, only relying on feature domain discriminant constraints would lead to domain shift problems inevitably in generated images. To speed convergence while learning more robust representations, self-supervised modules including *Background Consistency Module (BCM)* and *Semantic Consistency Module (SCM)* are introduced to provide more reasonable and reliable supervision.

BCM aims to preserve the background consistency between the translated images and inputs. Similar strategies have been applied for self-supervised image reconstruction tasks [14, 28]. These methods use the gradient error to constrain reconstructed images by smoothing the input and output images with blur operators, *e.g.*, Gaussian blur kernel and guided filtering [13]. Different from them, a $\mathcal{L}_1$ loss is directly used for the recovered images instead of gradient error loss in our module, as shown in Fig. 3, which is simple but effective to retain background consistency while recovering finer texture in our experiments. Specifically, a multi-scale Gaussian-Blur operator is used to obtain multi-scale features respectively. Therefore, a background consistency loss $\mathcal{L}_{BC}$ could be formulated as:

$$\mathcal{L}_{BC} = \sum_{\sigma=5,9,15} \lambda_\sigma \|B_\sigma(\chi) - B_\sigma(\tilde{\chi})\|_1 \tag{10}$$

where $B_\sigma(\cdot)$ denotes the Gaussian-Blur operator with blur kernel σ, λ_σ is the hyper-parameter to balance the errors at different Gaussian-Blur levels. χ and $\tilde{\chi}$ denote original input and the translated output, *i.e.*, $\{x, \tilde{x}^{\mathcal{X}\to\mathcal{Y}}\}$ and $\{y, \tilde{y}^{\mathcal{Y}\to\mathcal{X}}\}$. Based on experimental attempts at image denoising, we set λ_σ as $\{0.25, 0.5, 1.0\}$ for $\sigma = \{5, 9, 15\}$ respectively.

In addition, inspired by perception loss [15], the feature from the deeper layers of the pre-trained model contain semantic meanings only, which are noiseless or with little noise. Therefore, different from the general feature loss, which aims to recover finer image texture details via similarities among shallow features, we only extract deeper features as semantic representations from the corrupted and recovered images to keep consistency, referred to as semantic consistency loss $\mathcal{L}_{SC}$. It could be formulated as

$$\mathcal{L}_{SC} = \|\phi_l(\chi) - \phi_l(\tilde{\chi})\|^2_2 \tag{11}$$

where $\phi(\cdot)$ denotes the features from l_{th} layer of the pre-trained model. In our experiments, we use the *conv5-1* layer of VGG-19 [31] pre-trained network on ImageNet.

3.3. Jointly Optimizing

Other than proposed cross-cycle consistency loss, representation adversarial loss and self-supervised loss, we also use other loss functions in our joint optimization.

Target Domain Adversarial Loss. We impose domain adversarial loss $\mathcal{L}^{domain}_{adv}$, where $D_{\mathcal{X}}$ and $D_{\mathcal{Y}}$ attempt to discriminate the realness of generated images from each domain. For the noise domain, we define the adversarial loss

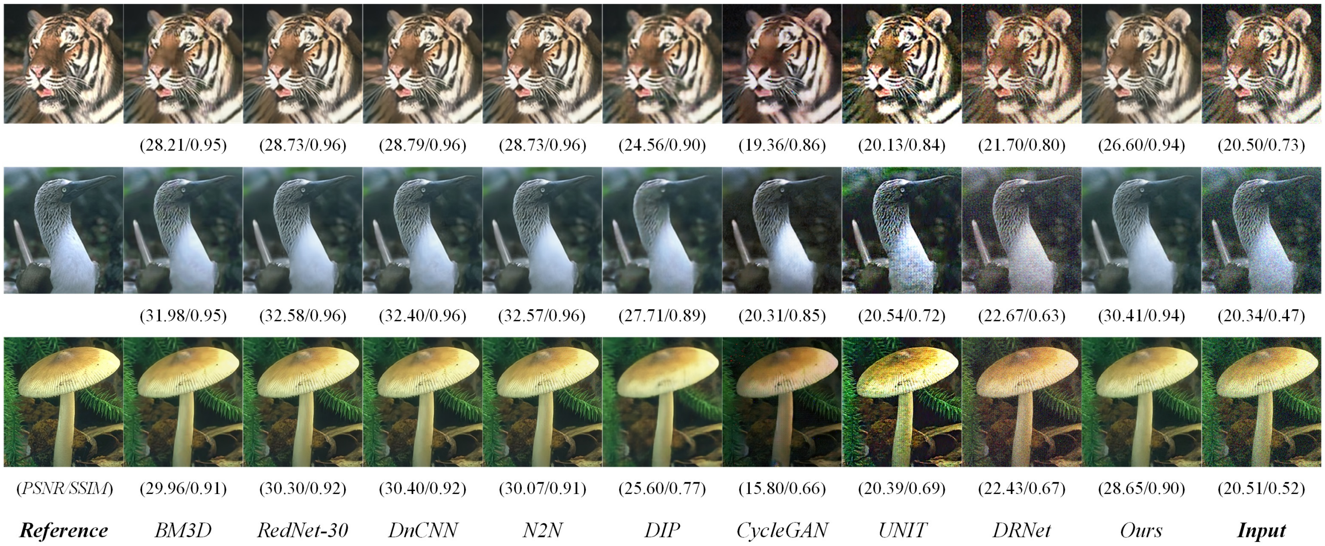


Figure 4: The example results for Gaussian noise on BSD-68. Zooming in for better visualization.

Methods	*BM3D*[5]	*RedNet-30*[26]	*DnCNN*[39]	*N2N*[19]	*DIP*[32]	*CycleGAN*[40]	*UNIT*[21]	*DRNet*[23]	*Ours*
					PSNR$(mean \pm std)$				
$\sigma = 25$	30.18±2.07	30.19±2.07	30.70±2.04	30.21±2.19	26.48±3.14	19.08±2.27	20.21±1.45	21.06±2.23	**29.02±1.93**
$\sigma = 35$	28.09±2.17	28.27±2.30	28.75±2.10	28.28±2.29	26.06±2.78	16.77±1.63	18.96±1.29	19.10±1.70	**27.58±1.98**
$\sigma = 50$	25.87±2.31	25.22±2.84	26.54±2.15	25.85±2.58	24.80±2.25	16.68±2.35	17.10±1.08	16.78±1.22	**24.69±1.59**
					SSIM$(mean \pm std)$				
$\sigma = 25$	0.921±0.03	0.918±0.03	0.931±0.02	0.919±0.03	0.820±0.09	0.808±0.06	0.709±0.08	0.626±0.09	**0.917±0.02**
$\sigma = 35$	0.883±0.04	0.885±0.04	0.901±0.03	0.886±0.04	0.817±0.07	0.731±0.07	0.599±0.10	0.505±0.09	**0.887±0.03**
$\sigma = 50$	0.830±0.06	0.827±0.06	0.857±0.05	0.832±0.06	0.786±0.07	0.696±0.06	0.459±0.11	0.374±0.08	**0.787±0.04**

Table 1: Quantitative results for Gaussian noise reduction on BSD-68 dataset.

$\mathcal{L}^{\mathcal{X}}_{adv}$ as

$$\begin{aligned}\mathcal{L}^{\mathcal{X}}_{adv} = &\mathbb{E}_{x\sim P_{\mathcal{X}}(x)}\left[\log D_{\mathcal{X}}(x)\right] + \\ &\mathbb{E}_{\substack{y\sim P_{\mathcal{Y}}(y)\\ x\sim P_{\mathcal{X}}(x)}}\left[\log(1 - D_{\mathcal{X}}(G_{\mathcal{X}}(E_{\mathcal{Y}}(y), E^{N}_{\mathcal{X}}(x))))\right]\end{aligned} \tag{12}$$

Similarly, we define adversarial loss for clean image domain as

$$\begin{aligned}\mathcal{L}^{\mathcal{Y}}_{adv} =&\mathbb{E}_{y\sim P_{\mathcal{Y}}(y)}\left[\log D_{\mathcal{Y}}(y)\right] + \\ &\mathbb{E}_{x\sim P_{\mathcal{X}}(x)}\left[\log(1 - D_{\mathcal{Y}}(G_{\mathcal{Y}}(E_{\mathcal{X}}(x))))\right]\end{aligned} \tag{13}$$

Self-Reconstruction Loss. In addition to the cross-cycle reconstruction, we also apply a self-reconstruction loss $\mathcal{L}^{Rec}$ to facilitate the training. This process is represented as $\hat{x} = G_{\mathcal{X}}(E_{\mathcal{X}}(x) \oplus E^{N}_{\mathcal{X}}(x))$ and $\hat{y} = G_{\mathcal{Y}}(E_{\mathcal{Y}}(y))$.

KL Loss. In order to model the noise encoder branch, we add a KL divergence loss to regularize the distribution of the noise code $z^{N}_{\mathcal{X}} = E^{N}_{\mathcal{X}}(x)$ to be close to the normal distribution that $p(z^{N}_{\mathcal{X}} \sim N(0,1))$, where $D_{KL} = -\int p(z)\log(\frac{p(z)}{q(z)})dz$.

The full objective function of our method is summarized as follows:

$$\begin{aligned}\min_{E_{\mathcal{X}}, E^{N}_{\mathcal{X}}, E_{\mathcal{Y}}, G_{\mathcal{X}}, G_{\mathcal{Y}}} \max_{D_{\mathcal{X}}, D_{\mathcal{Y}}, D_{\mathcal{R}}} =&\lambda_{\mathcal{R}}\mathcal{L}^{\mathcal{R}}_{adv} + \\ &\lambda_{adv}\mathcal{L}^{domain}_{adv} + \lambda_{CC}\mathcal{L}^{CC} + \lambda_{rec}\mathcal{L}^{Rec} + \\ &\lambda_{bc}\mathcal{L}^{BC} + \lambda_{sc}\mathcal{L}^{SC} + \lambda_{KL}\mathcal{L}^{KL}\end{aligned} \tag{14}$$

where the hyper-parameters λ_{*} control the importance of each term.

Restoration: After learning, we only retain the cross encoder-generator network $\{E_{\mathcal{X}}, G_{\mathcal{Y}}\}$, $E_{\mathcal{X}}$ extracts the domain-invariant representation $z_{\mathcal{X}}$ from corrupted sample x, and $G_{\mathcal{Y}}$ recover the clean image $\tilde{x}^{\mathcal{X}\to\mathcal{Y}}$ from the $z_{\mathcal{X}}$ that $\tilde{x}^{\mathcal{X}\to\mathcal{Y}} = G_{\mathcal{Y}}(E_{\mathcal{X}}(x))$.

4. Experiments

In this section, we first give the implementation details of our method for classical image denoising. Traditional metrics, such as Peak-Signal-Noise-Rate (PSNR) and Structural Similarity (SSIM), are used for evaluation in experiments. Detailed results on synthetic and real noise removal tasks are shown with other state-of-the-art methods. For the synthetic noise removal, we start with general noise distributions including additive white Gaussian noise (AWGN)

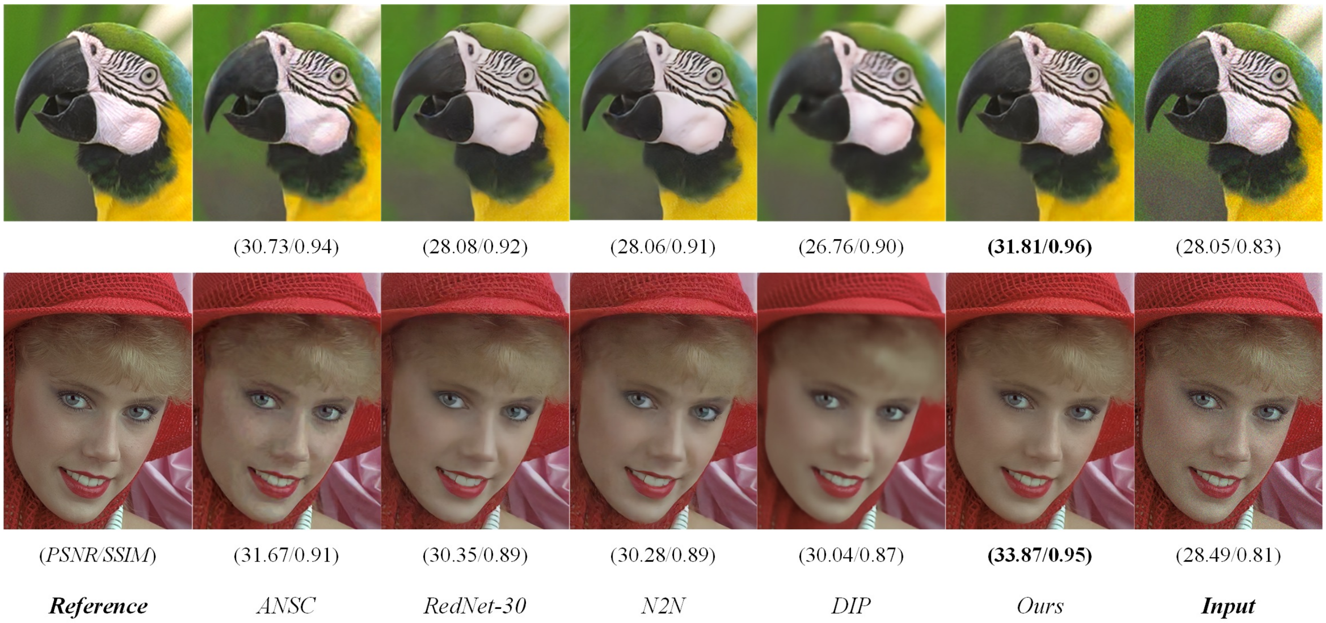


Figure 5: Sample results from Kodak dataset. Best detail visualization by zooming in.

Method	PSNR$(mean \pm std)$	SSIM $(mean \pm std)$
DIP[32]	27.63 ± 2.66	0.838 ± 0.07
N2N[19]	28.39 ± 2.04	0.893 ± 0.03
ANSC[25]	30.68 ± 1.81	0.918 ± 0.02
RedNet-30[26]	28.34 ± 2.07	0.893 ± 0.03
Ours	**32.37** $\pm$ **1.55**	**0.957** $\pm$ **0.01**

Table 2: Quantitative results for Poisson noise.

and Poisson noise. Two well-known datasets BSD68 [27] and Kodak are used to verify the performance of our method in denoising and texture restoration. Furthermore, the real noise images from the medical Low-Dose Computed Tomography (LDCT) dataset are used to evaluate the generalized capacity of the method. Extra ablation study is used to verify the effectiveness of the proposed framework.

4.1. Implementation

We follow the similar network architecture as the one used in [21], the difference is we introduce an extra noise encoder branch and remove the shared-weight encoder. Representation discriminator is a full convolutional network structure, which stacks four convolutional layers with two strides and a global average pooling layer. Proposed framework is implemented with Pytorch [30] and an Nvidia TITAN-V GPU is used in experiments. During the training, we use Adam [16] to perform optimization and momentum is set to 0.9. The learning rate is initially set to 0.0001 and exponential decay over the 10K iterators. In all experiments, we randomly crop 64×64 patches with batch size of 16 for training. Hyper-parameters are set to $\lambda_{\mathcal{R}} = \lambda_{adv}^{domain} = \lambda_{sc} = 1$, $\lambda_{cc} = \lambda_{rec} = 10$, $\lambda_{bc} = 5$ and $\lambda_{KL} = 0.01$.

4.2. Synthetic Noise Removal

We train the model with the images from the Pascal2007 [6] training set. Samples are randomly divided into two parts without coinciding. We add different noise-levels to each sample in part one, which is viewed as corrupted set, and another is clean set. Proposed method needs to estimate the magnitude of noise while removing it (blind image denoising). Some supervised and unsupervised based methods are selected to evaluate.

AWGN Removal. We add the AWGN with zero mean and standard deviation randomly generated with ranges from 5 to 50 for each training example, test on BSD68 with $\sigma = \{25, 35, 50\}$. The representative unsupervised methods, including *DIP* [32], *Noise2Noise* (*N2N*) [19], *CycleGAN* [40], *UNIT* [21] and *DRNet* [23], and supervised methods (*e.g.*, *RedNet-30* [26] and *DnCNN* [39]), are selected to compare the performance on image denoising. Traditional *BM3D* is also included for evaluation. For *CycleGAN*, *UNIT* and *DRNet*, we retrain them with the same training data.

The visualized results from BSD68 dataset are given in Fig. 4. Although all the methods show the ability for noise reduction, domain transfer based unsupervised methods, including *CycleGAN*, *UNIT* and *DRNet*, have obvious domain shift problems, *e.g.*, inconsistent brightness and undesired artifacts, resulting in worse visual perception. *N2N* and *DIP* achieve higher PSNR and SSIM. However, *DIP* loses fine local details and leads to over-smoothness in the generated images. Depending on the zero-mean distribution prior, *N2N* achieves similar results with other supervised methods, such as *RedNet-30* and *DnCNN*. Our approach presents comparable performance on noise removal and texture preserving. Although the PSNR is slightly lower than other supervised methods, our method achieves better visual consistency with natural images. Quantitative results for BSD68 are given in Table. 1. The proposed method shows stronger ability to blind image denoising.

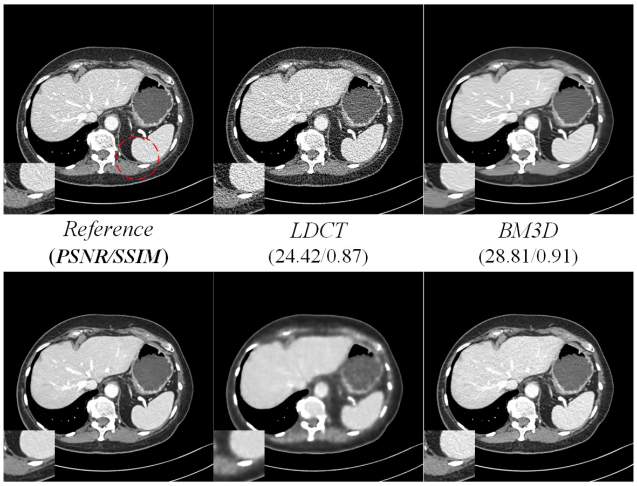
Poisson Noise Removal. For corrupted samples, we randomly generate the noise data from Scikit-image library [33], which generates independent Poisson noise by the number of unique values in the given samples, and test on Kodak[1] dataset. Some representative methods, including *DIP*, *N2N*, *ANSC* [25] and *RedNet-30*, are selected in our evaluations.

Comprehensive results are shown in Fig. 5 and Table. 2. *DIP* tends to generate more blurred results. The traditional *ANSC* method first transforms the Poisson noise into Gaussian (Anscombe transform), then applies the *BM3D* to remove noise, and finally inverts the transform, achieving higher PSNR and SSIM. Considering the different way of generating Poisson noise, the published *RedNet-30* and *N2N* models dont achieve the best results. Our method achieves the highest PSNR and SSIM. In addition, visualized results also show that for slight noise signals, the proposed framework has better generalized capacity to remove noise while restoring finer details.

4.3. Real Noise Removal

X-ray computed tomography (CT) is widely used as important imaging modalities in modern clinical diagnosis. Considering the potential radiation risk to the patient, lowering the radiation dose increases the noise and artifacts in reconstructed images, which can compromise diagnostic information. Typically, noise in x-ray photon measurements can be simply modeled as the combination of Poisson quantum noise and Gaussian electronic noise. However, the noise in reconstructed images is more complicated and does not obey any statistical distribution across the whole image. Therefore, classical image post-processing methods based on noise statistic prior, *e.g.*, *N2N*, are unsuitable for Low-dose CT (LDCT) denoising.

A real clinical dataset authorized by Mayo Clinic for *the 2016 NIH-AAPM-Mayo Clinic Low Dose CT Grand Challenge*[2] is used to evaluate LDCT image reconstruction algorithms, which contains 5936 images in 512×512 resolution from 10 different subjects. We randomly select 4000 images as training set, the remaining is as testing set. *DIP*, *BM3D* and *RedCNN* [4], which is an extended version of *RedNet*, are selected for evaluation in our experiments. The representative results are shown in Fig. 6, *BM3D* introduces waxy artifacts into the reconstructed image. *DIP* fails to generate the fine local structures. *RedCNN* tends to generate smoother images. Our approach achieves the better balance between visual quality and noise removal. Table. 3 gives the quantitative results.



Reference
(***PSNR/SSIM***)

LDCT
(24.42/0.87)

BM3D
(28.81/0.91)

RedCNN
(29.62/0.92)

DIP
(24.90/0.85)

Ours
(28.19/0.92)

Figure 6: LDCT Reconstruction. The display window is $[160, 240]$HU. The red circle denotes ROI area.

Methods	PSNR	SSIM
LDCT	36.3616	0.9423
BM3D[5]	40.6941	0.9755
RedCNN[4]	41.8799	0.9846
DIP[32]	36.2047	0.9500
Ours	40.5857	0.9811

Table 3: Quantitative results on Mayo dataset.

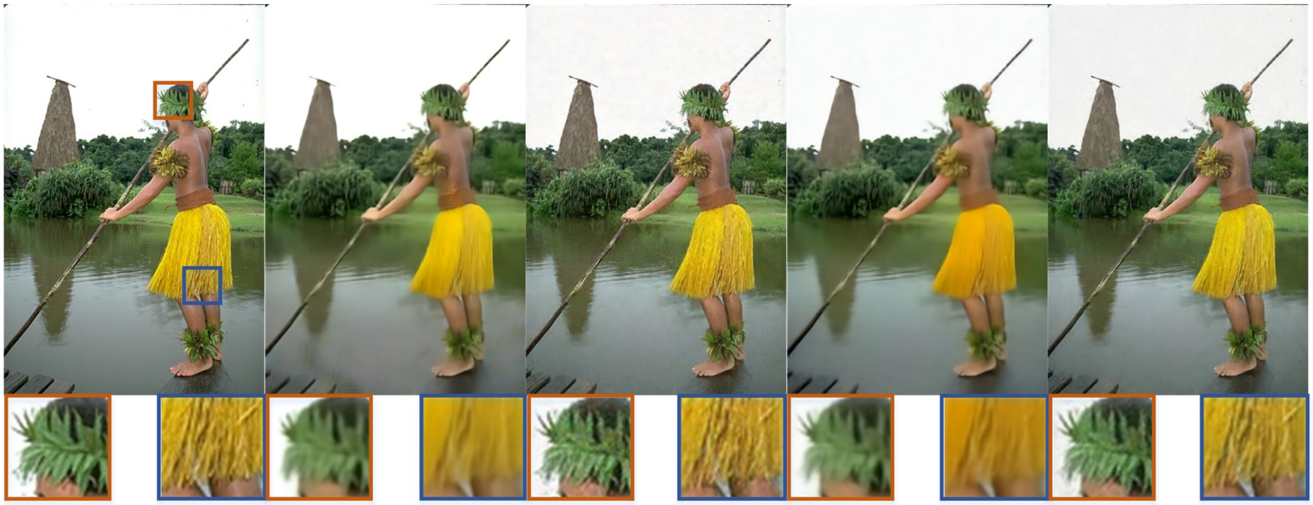
4.4. Ablation Study

In this section, we perform an ablation study to analyze the effects of discrete disentangling representation and self-supervised modules in the proposed framework. Both quantitative and qualitative results on Gaussian noise removal are shown for the following three variants of the proposed method where each component is separately studied: a) Remove the noise encoder branch; b) Remove the representation adversarial network $D^{\mathcal{R}}$, directly learn the representations $z_{\mathcal{X}}$ and $z_{\mathcal{Y}}$ by the target domain constraints only; c) Remove the background consistency constraint from self-supervised modules, only retain the semantic consistency constraints.

The representative results are shown in Fig. 7. Compared with the full model, referred to as (d), directly learning invariant representations from noise images would lead to the generator producing over-smooth results for (a) due to unexpected noise contained in features, which requires

[1] `http://r0k.us/graphics/kodak/`

[2] `http://www.aapm.org/GrandChallenge/LowDoseCT/`



Reference (***PSNR/SSIM***) (a) (27.29/0.8856) (b) (29.54/0.9333) (c) (26.95/0.88) (d) (27.80/0.9348)

Figure 7: The visualized results for each variants. (a) Without $E_{\mathcal{X}}^{N}$. (b) Without $D_{\mathcal{R}}$. (c) Removing *BGM* (d) Full model.

Variants	PSNR($mean \pm std$)	SSIM ($mean \pm std$)
(a)	25.997 ± 1.50	0.828 ± 0.07
(b)	29.452 ± 1.71	0.913 ± 0.02
(c)	25.220 ± 1.62	0.817 ± 0.08
(d)	$\mathbf{29.022 \pm 1.93}$	$\mathbf{0.917 \pm 0.02}$

Table 4: Quantitative results for Gaussian noise with $\sigma = 25$ on BSD-68.

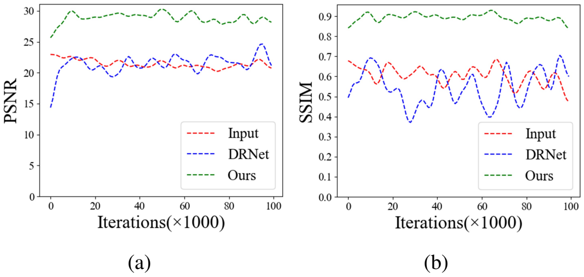


Figure 8: Online training PSNR and SSIM during the 100k iterations.

a powerful domain generator. Although (b) gives the better PSNR and SSIM after removing the feature adversarial module, some undesired artifacts adhere to high-frequency signals. Due to failing to provide the effective self-supervised constraint for the recovered images, although retaining the semantic consistency module, the model (c) also produces domain shift problems in generated images, *e.g.*, inconsistency brightness and blurred details, resulting in worse visual perception. Quantitative results are shown in Table. 4.

In addition, considering *DRNet* [23] has similar architecture with ours, which extends *DRIT* [18] while introducing extra feature loss to solve image deblurring, we select it as a representative domain transfer method to compare the convergence of algorithms on denoising task. Fig. 8 gives the convergence plots for AWAN removal, where we trained two models from scratch on the same training set. Although *DRNet* also uses the similar idea of disentangled representation to solve image restoration, which is different from ours in essence. Varying noise-levels and types lead to unstable learning during training due to lack of clear domain boundary. Aiming to learn invariant representation, our method gives faster and more stable convergence plots.

5. Conclusion

In this paper, we propose an unsupervised learning method for image restoration. Specifically, we aim to learn invariant representations from noise data via disentangling representations and adversarial domain adaption. Aided by effective self-supervised constraints, our method could reconstruct the higher-quality images with finer details and better visual perception. Experiments on synthetic and real image denoising show our method achieves comparable performance with other state-of-the-art methods, and has faster and more stable convergence than other domain adaption methods.

Acknowledge

This work is supported by the National Natural Science Foundation of China under grant 61871277, and in part by the Science and Technology Project of Sichuan Province of China under grant 2019YFH0193.

References

[1] Joshua Batson and Loïc Royer. Noise2self: Blind denoising by self-supervision. In *ICML*, 2019.

[2] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A non-local algorithm for image denoising. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2:60–65 vol. 2, 2005.

[3] Priyam Chatterjee and Peyman Milanfar. Clustering-based denoising with locally learned dictionaries. *IEEE Transactions on Image Processing*, 18:1438–1451, 2009.

[4] Hu Chen, Yi Zhang, Mannudeep Kalra, Feng Lin, Yang Chen, Peixi Liao, Ji liu Zhou, and Ge Wang. Low-dose ct with a residual encoder-decoder convolutional neural network. *IEEE Transactions on Medical Imaging*, 36:2524–2535, 2017.

[5] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen O. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16:2080–2095, 2007.

[6] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, 2009.

[7] Yaroslav Ganin and Victor S. Lempitsky. Unsupervised domain adaptation by backpropagation. *ArXiv*, abs/1409.7495, 2014.

[8] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor S. Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17:59:1–59:35, 2015.

[9] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2414–2423, 2016.

[10] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[11] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2862–2869, 2014.

[12] Shuhang Gu, Wangmeng Zuo, Qi Xie, Deyu Meng, Xiangchu Feng, and Lei Zhang. Convolutional sparse coding for image super-resolution. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1823–1831, 2015.

[13] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:1397–1409, 2013.

[14] Xin Jin, Zhibo Chen, Jianxin Lin, Zhikai Chen, and Wei Zhou. Unsupervised single image deraining with self-supervised constraints. *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2761–2765, 2018.

[15] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[16] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[17] Alexander Krull, Tim-Oliver Buchholz, and Florian Jug. Noise2void - learning denoising from single noisy images. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2124–2132, 2018.

[18] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. *ArXiv*, abs/1808.00948, 2018.

[19] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. *ArXiv*, abs/1803.04189, 2018.

[20] Ming Liu, Yukang Ding, Min Xia, Xiao Liu, Errui Ding, Wangmeng Zuo, and Shilei Wen. Stgan: A unified selective transfer network for arbitrary image attribute editing. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3668–3677, 2019.

[21] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *NIPS*, 2017.

[22] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *NIPS*, 2016.

[23] Boyu Lu, Jun-Cheng Chen, and Rama Chellappa. Unsupervised domain-specific deblurring via disentangled representations. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10217–10226, 2019.

[24] Julien Mairal, Michael Elad, and Guillermo Sapiro. Sparse representation for color image restoration. *IEEE Transactions on Image Processing*, 17:53–69, 2008.

[25] Markku Mäkitalo and Alessandro Foi. Optimal inversion of the anscombe transformation in low-count poisson image denoising. *IEEE Transactions on Image Processing*, 20:99–109, 2011.

[26] Xiao-Jiao Mao, Chunhua Shen, and Yu-Bin Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *NIPS*, 2016.

[27] David R. Martin, Charless C. Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, 2:416–423 vol.2, 2001.

[28] Thekke Madam Nimisha, Sunil Kumar, and A. N. Rajagopalan. Unsupervised class-specific deblurring. In *ECCV*, 2018.

[29] Stanley Osher, Martin Burger, Donald Goldfarb, Jinjun Xu, and Wotao Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4:460–489, 2005.

[30] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary Devito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[32] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Deep image prior. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9446–9454, 2017.

[33] Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, and Tony Yu. scikit-image: image processing in python. In *PeerJ*, 2014.

[34] Luminita A. Vese and Stanley Osher. Image denoising and decomposition with total variation minimization and oscillatory functions. *Journal of Mathematical Imaging and Vision*, 20:7–18, 2004.

[35] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *ICML '08*, 2008.

[36] Junyuan Xie, Linli Xu, and Enhong Chen. Image denoising and inpainting with deep neural networks. In *NIPS*, 2012.

[37] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2868–2876, 2017.

[38] Yuan Yuan, Siyuan Liu, Jiawei Zhang, Yongbing Zhang, Chao Dong, and Liang Lin. Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 814–81409, 2018.

[39] 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*, 26:3142–3155, 2017.

[40] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017.

[41] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A. Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. *ArXiv*, abs/1711.11586, 2017.

[42] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. *2011 International Conference on Computer Vision*, pages 479–486, 2011.