Unsupervised Domain Adaptation for EM Image Denoising With Invertible Networks

Shiyu Deng, Yinda Chen, *Graduate Student Member, IEEE*, Wei Huang, Ruobing Zhang, and Zhiwei Xiong, *Member, IEEE*

***Abstract*—Electron microscopy (EM) image denoising is critical for visualization and subsequent analysis. Despite the remarkable achievements of deep learning-based non-blind denoising methods, their performance drops significantly when domain shifts exist between the training and testing data. To address this issue, unpaired blind denoising methods have been proposed. However, these methods heavily rely on image-to-image translation and neglect the inherent characteristics of EM images, limiting their overall denoising performance. In this paper, we propose the first unsupervised domain adaptive EM image denoising method, which is grounded in the observation that EM images from similar samples share common content characteristics. Specifically, we first disentangle the content representations and the noise components from noisy images and establish a shared domain-agnostic content space via domain alignment to bridge the synthetic images (source domain) and the real images (target domain). To ensure precise domain alignment, we further incorporate domain regularization by enforcing that: the pseudo-noisy images, reconstructed using both content representations and noise components, accurately capture the characteristics of the noisy images from which the noise components originate, all while maintaining semantic consistency with the noisy images from which the content representations originate. To guarantee lossless representation decomposition and image reconstruction, we introduce disentanglement-reconstruction invertible networks. Finally, the reconstructed pseudo-noisy images, paired with their corresponding clean counterparts, serve as valuable training data for the denoising network. Extensive experiments on synthetic and real EM datasets demonstrate the superiority of our method in terms of image restoration quality and downstream neuron segmentation accuracy. Our code is publicly available at https://github.com/sydeng99/DADn.**

***Index Terms*—Unsupervised domain adaptation, image denoising, electron microscopy.**

I. Introduction

ELECTRON microscopy (EM) is a pivotal imaging technique in the field of biomedical image analysis. Its remarkable imaging resolution enables the analysis of biological structures at the nanoscale [1], [2]. However, there exists a fundamental trade-off between image quality and acquisition time. Acquiring high-quality EM images necessitates longer dwell times, resulting in time-consuming processes to obtain clean images with high signal-to-noise ratios. For instance, Zheng et al. [1] spend approximately 16 months to acquire a whole-brain dataset of an adult *Drosophila melanogaster*. Conversely, EM images acquired in shorter dwell times tend to exhibit noise and diminished signal-to-noise characteristics. Therefore, there is an urgent demand for effective denoising algorithms [3] to expedite imaging procedures to acquire clean images with a high signal-to-noise ratio.

Traditional denoising methods [4], [5] primarily rely on non-local and sparse representation techniques, which are time-consuming and computationally expensive. Recently, deep learning-based methods have played dominant roles in image denoising. Initially, a series of supervised denoising methods [6], [7], [8], [9], [10], [11] based on deep convolutional neural networks (CNNs) have demonstrated remarkable performance in non-blind denoising, *e.g.*, removing additive Gaussian white noise (AWGN). However, these methods rely on a large amount of paired noisy-clean images for supervised training, which is time-consuming to collect and align. Furthermore, denoising models trained exclusively on synthetic noisy images often face challenges in effectively generalizing to real-world noisy images, mainly due to domain shifts in noise distributions. As illustrated in Fig. 1, the denoising algorithms DnCNN-G10 and DnCNN-G70 trained on noisy images with AWGN σ of 10 and 70, respectively, cannot generalize well (resulting in residual noise and pseudo-textures) to noisy images with a σ of 25 due to the presence of domain shift in the noise distributions. Instead, DnCNN-G25 trained on noisy images with a σ of 25, without domain shift, achieves the best

Manuscript received 7 May 2024; revised 25 June 2024; accepted 11 July 2024. Date of publication 19 July 2024; date of current version 2 January 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62021001. *(Corresponding authors: Zhiwei Xiong; Ruobing Zhang.)*

Shiyu Deng, Yinda Chen, Wei Huang, and Zhiwei Xiong are with the MoE Key Laboratory of Brain-Inspired Intelligent Perception and Cognition, University of Science and Technology of China, Hefei 230027, China, and also with Anhui Province Key Laboratory of Biomedical Imaging and Intelligent Processing, Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei 230088, China (e-mail: sydeng@mail.ustc.edu.cn; cyd0806@mail.ustc.edu.cn; weih527@mail.ustc.edu.cn; zwxiong@ustc.edu.cn).

Ruobing Zhang is with Jiangsu Key Laboratory of Medical Optics, Suzhou Institute of Biomedical Engineering and Technology, Chinese Academy of Sciences, Suzhou 215163, China, and also with Anhui Province Key Laboratory of Biomedical Imaging and Intelligent Processing, Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei 230088, China (e-mail: zhangrb@sibet.ac.cn).

Digital Object Identifier 10.1109/TMI.2024.3431192

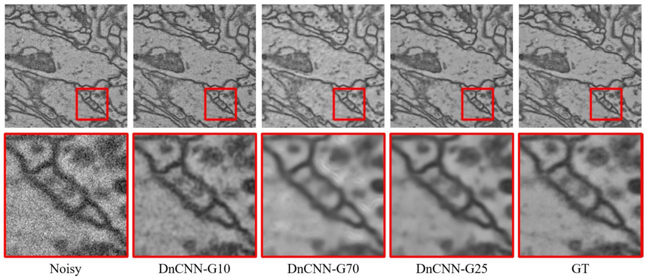


Fig. 1. Visual comparison of denoising results with/without domain shift.

denoising performance. Although self-supervised denoising methods [12], [13], [14], [15] trained solely on noisy images have been developed, these methods assume specific statistical noise distributions. Unfortunately, these assumptions may not consistently hold in practical scenarios, where real-world noise distributions can be complex and intertwined with images, as shown in Fig. 2. Consequently, their denoising effectiveness in real-world cases is limited.

Theoretically, it is quite challenging to perform image denoising based only on noisy images without the knowledge of clean signals and accurate noise distribution. Fortunately, although obtaining a large amount of paired data poses difficulties, we can make use of numerous real noisy images in combination with a limited set of unpaired clean images. This strategy significantly mitigates the cost and labor involved in data acquisition and registration, all while enabling the learning of noise distribution. Along this line, several unpaired denoising methods emerge. Early unpaired denoising methods [16], [17], [18] attempt to generate realistic noisy images using a generative adversarial network (GAN). These generated pseudo-noisy images can be paired with corresponding clean images to train the denoising network. Nevertheless, the complicated nature of real noise distributions poses a learning challenge for a simple adversarial network, leading to limited effectiveness. Due to the success of generative image translation methods [19], recent unpaired denoising methods [20], [21], [22], [23], particularly in the context of biomedical images, have adopted the idea of image-to-image translation to address the inherent challenges. However, it should be noted that these methods heavily rely on image translation, a task known for its inherent complexity. Furthermore, they overlook the intrinsic characteristics of EM images and the potential benefits of synthetic noisy images, limiting their overall denoising performance. Hence, there exists room for further improvement in developing robust and effective unpaired denoising techniques.

To bridge this gap, we propose the first unsupervised domain adaptation method for real-world EM image denoising by using the characteristic of EM images, *i.e.,* biomedical samples exhibit similar textures, such as vesicles and cell membranes. Specifically, our method initiates with domain alignment to establish a shared domain-agnostic content space bridging the synthetic noisy images in the source domain and the real noisy images in the target domain. We achieve this by disentangling content representations and noise components from noisy images in each domain, followed by domain adversarial training [24] between content representations of both domains. To ensure precise domain alignment, we implement domain regularization by reconstructing pseudo-noisy images using both content representations and noise components. The reconstruction is carefully crafted to accurately capture the characteristics of the noisy images from which the noise components originate, all while maintaining semantic consistency with the noisy images from which the content representations originate. To guarantee lossless representation decomposition and image reconstruction, we introduce disentanglement-reconstruction invertible networks. Compared with cycle consistency constraints in previous unpaired denoising algorithms, this design mitigates computational costs and ensures unimpaired bijective transformations. Consequently, we obtain pseudo-noisy images exhibiting a high resemblance to real noisy images, as illustrated in Fig. 2. These generated pseudo-noisy images, along with synthetic noisy images from the source domain, can be paired with the corresponding clean images, which greatly facilitates the training of our denoising network. In this way, our method retains the benefits of existing unpaired methods, while achieving state-of-the-art EM denoising performance on both synthetic and real datasets.

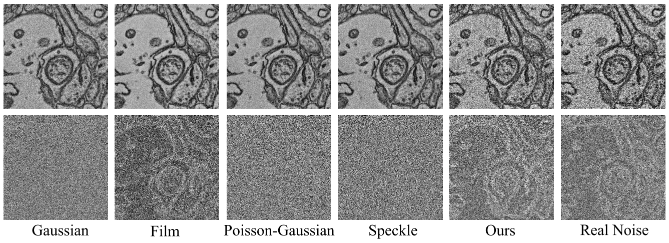


Fig. 2. Top row: An example of common types of synthetic noisy images (*i.e.*, Gaussian, film, Poisson-Gaussian, speckle), a noisy image generated as a byproduct by our proposed method, and a real-world EM noisy image. Bottom row: the corresponding noise maps for each of the noisy images.

The contributions of our work are as follows:

- We present the first method for EM image denoising from the perspective of unsupervised domain adaptation, which exploits the characteristics of EM images and bridges the synthetic source and real target domains.
- We design disentanglement-reconstruction invertible networks for lossless representation decomposition and high-fidelity noisy image reconstruction.
- We construct a well-aligned dataset of real-noisy paired SEM images with varying dwell times, promoting development and evaluation of EM denoising methods.
- We demonstrate state-of-the-art EM denoising performance and improved neuron segmentation accuracy over existing methods through extensive experiments.

II. Related Work

A. Non-Blind Image Denoising

Non-blind denoising typically assumes prior knowledge of the noise distribution (*e.g.*, AWGN) or requires noisy-clean image pairs for training deep learning networks. For instance, DnCNN [6] is a pioneering supervised method for non-blind image denoising, effectively removing AWGN using residual learning and batch normalization. Numerous CNN-based image denoising algorithms have followed DnCNN, such as CBDNet [7], DIDN [8], RIDNet [9], RNAN [11], and InvDN [25], which enhance denoising capabilities by introducing sophisticated network architectures. However, these methods are highly dependent on consistent noise distribution during training and testing, limiting their generalization to real-world scenarios where the noise distribution is unknown or noisy-clean image pairs are not available.

B. Blind Image Denoising

Blind image denoising methods do not require noisy-clean paired training data, including self-/unpaired-supervised ones.

1) Self-Supervised: Self-supervised denoising methods eliminate the need for paired noisy-clean images as training data. Noise2Noise (N2N) [12] trains on noisy image pairs assuming independent identically distributed noise. Noise2Void (N2V) [13] and Noise2Self (N2S) [14] introduce blind-spot networks trained on a set of noisy images without pairing. Follow-up works like Blind2Unblind [15] and AP-BSN [26] further advance N2V. Despite avoiding paired data, these methods rely on specific assumptions about noise statistics, limiting their applicability. DIP [27] explores image priors embedded in CNNs to reconstruct the clean latent image from a single noisy observation. However, DIP requires tedious per-image tuning of early stopping. In summary, existing self-supervised techniques trade off generalizability for removing noisy-clean data pairing requirements.

2) Unpaired Supervised: Unpaired supervised denoising methods provide an alternative solution by utilizing unpaired clean and noisy images. These methods can be broadly classified into two primary categories: noise generation-based and image-to-image translation-based. As a pioneer of noise generation-based unpaired denoising algorithm, Chen et al. propose GCBD [16], which extracts plain noise patches from noisy images and employs a GAN to learn the generation of noise. Similarly, Hong et al. [17] adopt a similar idea, using a conditional GAN (cGAN) to generate pairs of pseudo-noisy and clean images for training the denoising network. Taking it a step further, Jang et al. [18] introduce two branches in the noise generator—one for signal-dependent noise and another for signal-independent noise, respectively. Nevertheless, the complexity of real noise distributions poses a learning challenge for a simple adversarial network, leading to limited performance. On the other hand, image-to-image translation-based unpaired denoising approaches [20], [21], [22], [23] are inspired by CycleGAN [19] and tackle denoising through noisy-to-clean image translation. For example, inspired by [19], Quan et al. propose an asymmetrically cyclic adversarial network [20] to remove artifacts from EM images. Building upon this foundation, Lee et al. introduce self-cooperative learning [22], further improving denoising performance. Additionally, ADN [21] and DRGAN [23] incorporate representation disentanglement. However, these methods heavily rely on image translation and neglect the trait of EM images, limiting their denoising fidelity.

C. Unsupervised Domain Adaptation

Unsupervised domain adaptation (UDA) is a technique that aims to tackle the domain shift issue, improving the performance of a model on a label-free target domain by leveraging knowledge from the source domain with labeled data. Within UDA, feature-level domain adaptation methods [24], [28], [29], [30], [31] play a crucial role. They focus on aligning the domain distributions by adjusting the discriminative feature space. As a pioneer of feature-level UDA, DANN [24] learns domain-invariant features, through adversarial training. Specifically, the feature representations of the source and target data are encouraged to be similar by minimizing domain discrepancy. DAN [29] aims to minimize the maximum mean discrepancy between the source and target domains in feature space. Volpi et al. [30] use GANs to perform data augmentation in the feature space and generate features conditioned to the desired domain. Kang et al. [31] introduce a discrepancy metric CCD to minimize the intra-class discrepancy and maximize the inter-class margin. Another way to address UDA is pixel-level domain adaptation [32], [33], [34], where images are typically generated to contain the content of the source domain and the style of the target domain, mainly achieved by adversarial learning. Taigman et al. [32] propose DTN to translate a source image to a target one under f-consistency constraint. CoGAN [33] learns a joint distribution of multi-domain images and generates a couple of images following different distributions. UNIT [34] introduces image-to-image translation networks that learn a shared latent space through GANs and variational auto-encoders. UDA algorithms facilitate the flourishing of both high-level [35], [36], [37], [38] and low-level [39], [40], [41] visual tasks. However, to the best of our knowledge, few works have studied UDA for EM image denoising. Our work contributes to filling this research gap.

III. Method

We aim to learn an effective denoising model for real noisy images without paired supervision. This process is visually depicted in Fig. 3. To achieve this goal, we utilize two distinct sets of unpaired images: 1) We start with a set of real clean EM images denoted as $\mathcal{Y} = \{y_i\}_{i=1,\ldots,N}$, which we use to artificially generate noisy images $\mathcal{X}^S = \{x_i^S\}_{i=1,\ldots,N}$ via noise modeling that forms the source domain $\mathcal{S}$; 2) Additionally, we have another set consisting of real noisy EM images, denoted as $\mathcal{X}^R = \{x_j^R\}_{j=1,\ldots,M}$, which forms the target domain $\mathcal{T}$. Our framework consists of three fundamental components: 1) Domain alignment: Our initial step involves aligning both domains to obtain a domain-agnostic content space; 2) Domain regularization: We subsequently introduce a domain regularization step to ensure precise domain alignment. This step also helps in generating pseudo-noisy images; 3) Denoising network training: Finally, we train our denoising network using pairs of generated pseudo-noisy images and their corresponding clean images.

A. Domain Alignment

Domain alignment is the first step of our framework to bridge the gap between the source and the target domains. Our domain alignment strategy is founded on the trait that, despite varying noise characteristics, EM images exhibit comparable content when captured using the same imaging protocol. We employ domain alignment to discover the shared content space between the source and target domains.

Specifically, we disentangle each noisy image x into content representations F^C and noise components N using dedicated networks $G_s(\cdot)$ and $G_t(\cdot)$ for the source and target domains, respectively. As illustrated in Fig. 3(a), the synthetic noisy image x_i^S can be effectively separated into its constituent content representations F_i^C and noise components N_i^S, and similarly, the real noisy image x_j^R can be disentangled into its inherent content representation F_j^C and noise components N_j^R. The content representations F_i^C and F_j^C are expected to be aligned with each other and be domain-agnostic. To encourage domain-invariant content learning, we apply adversarial training on F_i^C and F_j^C using a content discriminator D_c. By optimizing the adversarial loss, the extracted content representations from both domains can be aligned to be domain-agnostic. To stabilize the adversarial training, we adopt LSGAN [42]. We assign the domain label of the source domain as 0 and the label of the target domain as 1. The adversarial loss functions can be formulated as follows:

$$\mathcal{L}_{GAN}^{C} = \mathbb{E}_{x_i^S \sim \mathcal{S}(x)}[(D_c(G_s(x_i^S)) - 1)^2], \tag{1}$$

$$\begin{aligned}\mathcal{L}_{D}^{C} &= \mathbb{E}_{x_i^S \sim \mathcal{S}(x)}[D_c(G_s(x_i^S))^2] \\ &\quad + \mathbb{E}_{x_j^R \sim \mathcal{T}(x)}[(D_c(G_t(x_j^R)) - 1)^2].\end{aligned} \tag{2}$$

Considering that the input noisy images themselves provide rough guidance for the content representations, we match the resolutions of F_i^C and F_j^C by bilinear downsampling the original noisy images to obtain content guidance $(x_i^S)_{LR}$ and $(x_j^R)_{LR}$. Hence, we also include a content guidance loss:

$$\mathcal{L}_{guide}^{C} = ||(x_i^S)_{LR} - F_i^C||_1 + ||(x_j^R)_{LR} - F_j^C||_1. \tag{3}$$

The learnable common content space can be established by ensuring that the content representations from both domains are domain-agnostic and closely aligned, bridging the gap between synthetic noisy images (source domain) and real noisy images (target domain).

B. Domain Regularization

To ensure precise domain alignment and obtain paired supervision for denoising network training, we propose domain regularization through pseudo-noisy image reconstruction and semantic consistency enforcement.

1) Pseudo-Noisy Image Reconstruction: To obtain paired supervision for denoising network training, we propose to reconstruct pseudo-noisy images $\hat{x}_i^R$ that effectively embody the noise pattern characteristics of the target domain. Specifically, inspired by the domain adaptation methods [29] in high-level vision tasks, we initially take advantage of the learned content space and disentangle the noise components N_i^S and N_j^R from the source and target images, respectively. By recombining noise components N_j^R with content representations F_i^C and passing through $G_t^{-1}(\cdot)$, we enforce the reconstructed pseudo-noisy image $\hat{x}_i^R$ to resemble the noise distribution of the target domain, while retaining image content from x_i^S. Since $\hat{x}_i^R$ share the same content with x_i^S and thus y_i, we can thus obtain pairs of pseudo-noisy image $\hat{x}_i^R$ and clean image y_i. To ensure the resemblance of reconstructed pseudo-noisy images and real noisy images, we introduce a target domain discriminator D_t. The adversarial loss functions of the target domain are as follows:

$$\mathcal{L}_{GAN}^{\mathcal{T}} = \mathbb{E}_{x_i^S \sim \mathcal{S}(x)}[(D_t(G_t^{-1}(F_i^C, N_j^R)) - 1)^2], \tag{4}$$

$$\begin{aligned}\mathcal{L}_{D}^{\mathcal{T}} &= \mathbb{E}_{x_i^S \sim \mathcal{S}(x)}[D_t(G_t^{-1}(F_i^C, N_j^R))^2] \\ &\quad + \mathbb{E}_{x_j^R \sim \mathcal{T}(x)}[(D_t(x_j^R) - 1)^2].\end{aligned} \tag{5}$$

Symmetrically, we also apply adversarial learning to reconstruct pseudo-noisy images that conform to the noise distribution of the source domain. The pseudo-noisy image $\hat{x}_j^S$ reconstructed with F_j^C and N_i^S through $G_s^{-1}(\cdot)$ should resemble the synthetic noisy images in the source domain. This balanced design further benefits domain alignment and image reconstruction. The adversarial loss functions of the source domain are formulated as follows:

$$\mathcal{L}_{GAN}^{\mathcal{S}} = \mathbb{E}_{x_j^R \sim \mathcal{T}(x)}[(D_s(G_s^{-1}(F_j^C, N_i^S)) - 1)^2], \tag{6}$$

$$\begin{aligned}\mathcal{L}_{D}^{\mathcal{S}} &= \mathbb{E}_{x_j^R \sim \mathcal{T}(x)}[D_s(G_s^{-1}(F_j^C, N_i^S))^2] \\ &\quad + \mathbb{E}_{x_i^S \sim \mathcal{S}(x)}[(D_s(x_i^S) - 1^2].\end{aligned} \tag{7}$$

2) Semantic Consistency Enforcement: Although adversarial learning in Sec.III-B.1 effectively forces the reconstructed pseudo-noisy images $\hat{x}_i^R$ and $\hat{x}_j^S$ to mimic the noise distributions of each domain, the semantic consistency of their contents remains uncertain. To obtain high-quality pseudo-noisy-clean image pairs for denoising network training, we must ensure that $\hat{x}_i^R$ preserves semantic information from y_i. Thus, we enforce semantic consistency on the generated images using a pre-trained encoder E. By matching the semantic features between $\hat{x}_i^R$ and y_i, we can regularize $\hat{x}_i^R$ to inherit image content of x_i^S while mimicking target domain noise characteristics. The semantically consistent pseudo-pairs $\{\hat{x}_i^R, y_i\}_{i=1,\dots,N}$ facilitate training an effective denoising network. Symmetrically, we enforce the pseudo-noisy image $\hat{x}_j^S$ to preserve the image content of X_j^R. Perceptual loss in natural image restoration [43] may have a similar role, but it relies on a VGG network [44] pre-trained on natural image classification tasks. However, due to the domain gaps between natural and EM images, networks pre-trained on natural images may not effectively extract semantic information from EM images. Although a segmentation network trained on EM images could be an ideal feature extractor, obtaining a large number of segmentation labels for EM data is challenging. To address this issue, we employ self-supervised contrastive learning [45] to train an effective feature extractor on the publicly available EM dataset FAFB [1]. Utilizing the pre-trained encoder, we freeze it as a semantic feature extractor. Then, we minimize a pixel-wise feature distance on the extracted features to achieve semantic consistency. As shown in Fig. 3 (b), the formulation

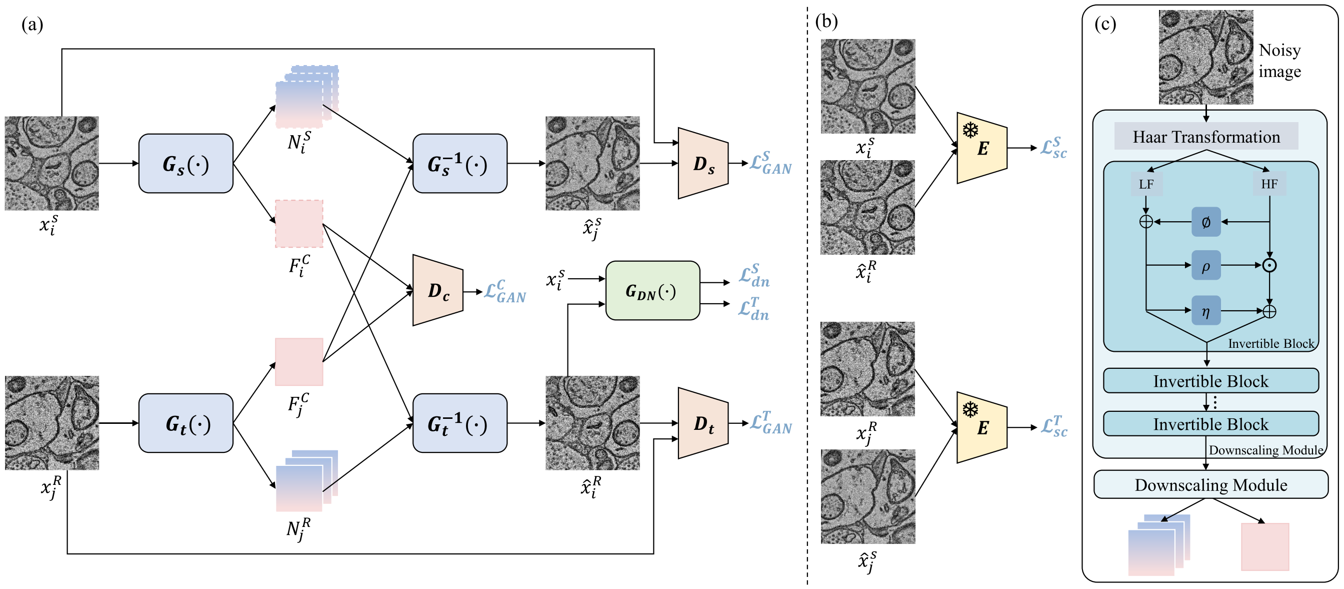


Fig. 3. (a) Overview of our proposed framework. (b) Semantic consistency constraints through a pre-trained encoder E. (c) Specific architecture of disentanglement-reconstruction invertible networks $G_s(\cdot)/G_s^{-1}(\cdot)$ and $G_t(\cdot)/G_t^{-1}(\cdot)$.

of semantic consistency loss can be formulated as follows:

$$\begin{aligned}\mathcal{L}_{sc} &= \mathcal{L}_{sc}^{\mathcal{S}} + \mathcal{L}_{sc}^{\mathcal{T}} \\ &= ||E(x_i^s) - E(\hat{x}_i^R)||_1 + ||E(x_j^R) - E(\hat{x}_j^S)||_1. \end{aligned} \quad (8)$$

3) Disentanglement-Reconstruction Invertible Networks: Previous unsupervised image restoration works [21], [23], [46] have commonly used reconstruction accuracy constraints and cycle consistency constraints to reduce information loss during image translation. Reconstruction accuracy constraints impose $G_s^{-1}(F_i^C, N_i^S) \approx x_i^S$ and $G_t^{-1}(F_j^C, N_j^R) \approx x_j^R$. On the other hand, the constraint of cycle consistency requires that the image, after undergoing two rounds of image translation, remains close to its original versions, *e.g.,* $x_i^S \rightarrow \hat{x}_j^S \rightarrow \tilde{x}_i^S \approx x_i^S$ and $x_j^R \rightarrow \hat{x}_i^R \rightarrow \tilde{x}_j^R \approx x_j^R$. These constraints require additional loss functions, resulting in inefficient model training and computational costs.

To achieve lossless representation decomposition and image reconstruction without supplementary loss functions, we embrace recent advanced flow-based models [25], [47]. Specifically, we employ invertible neural networks (INNs) as our disentanglement-reconstruction networks $G_s(\cdot)/G_s^{-1}(\cdot)$ and $G_t(\cdot)/G_t^{-1}(\cdot)$. Following [25], we adopt a multi-scale architecture that consists of n downscaling modules in the invertible networks, as shown in Fig. 3 (c). Each downscaling module contains a Haar transformation and several invertible blocks. As mentioned in Sec. III-A, the disentanglement network should decouple the content representations and noise components of a noisy image. The Haar transformation can decompose the input image into a low-frequency component and three high-frequency components. The primary image content exists in the low-frequency component, while the noise and some image content details are embedded in the high-frequency components. After the transformation of several invertible blocks, content and noise can be further disentangled. An image with a shape of $H \times W \times C$ can be transformed into a feature map with a shape of $\frac{H}{2} \times \frac{W}{2} \times 4C$. We use the first r-th channels as the content representations, and the remaining channels of the feature map represent the noise components. The reconstruction networks $G_s^{-1}(\cdot)$ and $G_t^{-1}(\cdot)$ are reverse versions of $G_s(\cdot)$ and $G_t(\cdot)$, respectively, enabling them to transform features back into images. By adopting invertible networks, the representation disentanglement and image reconstruction remain naturally lossless without any additional loss function.

C. Denoising Network Training

With pseudo-noisy-clean image pairs $\{\hat{x}_i^R, y_i\}_{i=1,\ldots,N}$ obtained from the aforementioned learning process, our next target is to train an effective denoising network for real noisy images. Due to its efficiency, we keep the same network structure with CBDNet [7], adopting a simple U-Net [48] with residual blocks as the denoising network G_{DN}. We use the L_1 distance between the denoised output and the corresponding clean label as the denoising loss function. We formulate the real denoising loss function as follows:

$$\mathcal{L}_{dn}^{\mathcal{T}} = ||G_{DN}(\hat{x}_i^R) - y_i||_1. \quad (9)$$

To improve the denoising performance and fully exploit the available images, we also feed the synthetic noisy-clean image pairs $\{x_i^S, y_i\}_{i=1,\ldots,N}$ from the source domain into the denoising network G_{DN}. Theoretically, when the discrepancy between the source domain and the target domain is marginal, the abundant synthesized image pairs will substantially facilitate the learning process of the denoising network. The denoising loss function for the source domain is as follows:

$$\mathcal{L}_{dn}^{\mathcal{S}} = ||G_{DN}(x_i^S) - y_i||_1. \quad (10)$$

During the training phase, all networks are trained together simultaneously to find the optimal solution. During the testing phase, we only send the real noisy images to the network to get the denoising results.

D. Loss Function

With the aforementioned losses, the invertible networks $G_s(\cdot)/G_s^{-1}(\cdot)$ and $G_t(\cdot)/G_t^{-1}(\cdot)$ and the denoising network G_{DN} can be trained with the following formulation:

$$\begin{aligned}\mathcal{L} = \lambda_{dn}^{\mathcal{T}}\mathcal{L}_{dn}^{\mathcal{T}} + \lambda_{dn}^{\mathcal{S}}\mathcal{L}_{dn}^{\mathcal{S}} + \lambda_{sc}\mathcal{L}_{sc} \\ + \lambda_{GAN}^{\mathcal{S}}\mathcal{L}_{GAN}^{\mathcal{S}} + \lambda_{GAN}^{\mathcal{T}}\mathcal{L}_{GAN}^{\mathcal{T}} \\ + \lambda_{GAN}^{C}\mathcal{L}_{GAN}^{C} + \lambda_{guide}^{C}\mathcal{L}_{guide}^{C},\end{aligned} \tag{11}$$

where $\lambda_{dn}^{\mathcal{T}}$, $\lambda_{dn}^{\mathcal{S}}$, λ_{sc}, $\lambda_{GAN}^{\mathcal{S}}$, $\lambda_{GAN}^{\mathcal{T}}$, λ_{GAN}^{C}, and λ_{guide}^{C} are loss weights. Loss functions of discriminators D_c, D_t, D_s are formulated in Eq.2, Eq.5, and Eq.7, respectively.

IV. Implementation

A. Data Preparation

To evaluate our method, we prepare both synthetic and real noisy EM datasets. The synthetic dataset derives from the CREMI dataset [2], originally proposed for neuron segmentation in the adult *Drosophila* brain. This dataset contains three subsets, each with 125 training and 125 test images. Each image has a resolution of 1250×1250. To simultaneously evaluate the denoising performance and its impact on segmentation accuracy, we swap the training and test data. Our goal is to create synthetic noisy images that closely conform to real EM noise distribution. We employ three noise models: film noise, the mixture of Gaussian and Poisson noise, and Speckle noise, applying them to subsets A, B, and C, respectively. These models better mimic real noise than AWGN, as demonstrated in Fig. 2. Specifically, for film noise, we set the kernel size to 5 and the maximum intensity to 1. For the Poisson-Gaussian noise, we randomly set the noise level σ for the Gaussian noise between 55 and 85 and the scale for the Poisson noise as a random number between 0.6 and 0.8. For speckle noise, we set its mean at 0.2 and its variance at 0.09. As for unpaired learning methods, we use the first 60 images from each 125-image training set to synthesize noisy images, reserving the rest as unpaired clean images.

In addition, due to the lack of publicly available real noisy-clean EM image pairs, we employ an imaging strategy to obtain some noisy-clean image pairs. Specifically, we use MultiSEM [49] to image the same sample with different dwell times. We use images obtained with a dwell time of $0.05\mu s$ as the noisy images and images obtained with a dwell time of $3.2\mu s$ as the clean images. Since they are imaged from the same sample, the clean and noisy images share consistent content. To achieve pixel-wise alignment of image pairs, we apply elastic registration [50] to register the clean and noisy images of the same sample. In addition, we apply intensity inversion and CLAHE (contrast limited adaptive histogram equalization) to enhance the image contrast as image pre-processing. Finally, we obtain a well-aligned dataset (**RETINA**) of noisy-clean SEM image pairs from *mouse retinas*. This dataset has the same resolution of $4nm$ as CREMI. For a convenient evaluation of our denoising algorithm, we divide this dataset into three parts: training, validation, and testing sets, consisting of 1374, 100, and 400 noisy-clean image pairs, respectively, each with a resolution of 512×512. We also include 1374 unpaired images in the training set to implement our unpaired learning.

B. Training Details

Our framework is consist of five trainable networks: G_s, G_t, D_c, D_s, and D_t. The whole framework is trained jointly in an end-to-end manner. We adopt the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$ to train the networks with a learning rate of 0.0001, and fix it during the training phase. Prior to network input, we crop them into 192×192 patches and perform rotation and flip for data augmentation. The training is performed with a batch size of 2 on two *NVIDIA Titan XP* GPUs. Each mini-batch contains randomly selected patches from unpaired clean and noisy images. All experiments are conduct by PyTorch framework. As for the invertible networks, we leverage the general coupling layer proposed in [51] and densely connected convolutional blocks proposed in ESRGAN [52]. We synthesize noisy images to form the source domain using AWGN with a noise level of 55. The loss weights are kept as: $\lambda_{dn}^{\mathcal{T}} = 1$, $\lambda_{dn}^{\mathcal{S}} = 1$, $\lambda_{sc} = 0.05$, $\lambda_{GAN}^{\mathcal{S}} = 0.2$, $\lambda_{GAN}^{\mathcal{T}} = 0.2$, $\lambda_{GAN}^{C} = 0.1$, and $\lambda_{guide}^{C} = 0.1$.

V. Experimental Results

A. Compared Methods

To thoroughly assess our proposed method, we conduct a comprehensive comparison with a range of advanced denoising techniques, categorized into six classes:

1) Non-learning based: This category includes two classical image denoising algorithms, BM3D [4] and WNNM [5].
2) Supervised† (Real noise): In this ideal case, supervised networks are trained on real noisy-clean image pairs, and the training data match the test data distribution. The denoising results obtained through these methods can be considered as an upper bound.
3) Supervised (Board noise): Networks are trained on a broad range of simulated noisy images paired with their corresponding clean labels, and testing is performed on real noisy images. We simulate noisy training images by adding AWGN with a random σ between 10 and 85.
4) Supervised (Specific noise): Networks are trained on synthetic image pairs, and the simulated noise follows a specific distribution. We simulate the training noisy images by adding AWGN with $\sigma = 55$.
5) Self-supervised: These methods are trained only on noisy images. We include four exceptional self-supervised methods: DIP [27], ZSn2n [53], N2V [13], and B2U [15].
6) Unpaired supervised: These methods are trained on unpaired clean and noisy images, which is also our setting. We include CycleGAN [19] and four state-of-the-art unpaired denoising methods: ISCL [22], ADN [21], C2N [18], and DRGAN [23].

All methods except "Supervised† (Real noise)" are trained without the knowledge of noise distribution of testing data. For each class of supervised training, we apply four popular networks: DnCNN [6], CBDNet [7], DIDN [8], and RIDNet [9].

TABLE I
QUANTITATIVE RESULTS OF IMAGE DENOISING ON SYNTHETIC TEST IMAGES FROM THE CREMI DATASET [2], IN TERMS OF IMAGE RESTORATION FIDELITY (PSNR / SSIM) AND NEURON SEGMENTATION ACCURACY (VOI / ARAND). NOTE THAT † STANDS FOR IDEAL SUPERVISION WHICH IS NOT ALWAYS AVAILABLE IN REAL CASES

Dataset		CREMI-A				CREMI-B				CREMI-C			
Methods		PSNR↑	SSIM↑	VOI↓	ARAND↓	PSNR↑	SSIM↑	VOI↓	ARAND↓	PSNR↑	SSIM↑	VOI↓	ARAND↓
Noisy		18.8912	0.3001	0.9469	0.1167	11.9641	0.0731	4.5982	0.7738	14.9917	0.2769	1.8721	0.1357
Non-learning based	BM3D	29.2521	0.8509	0.9152	0.1135	27.5391	0.7312	1.8460	0.1226	20.0120	0.8315	1.8073	0.1334
	WNNM	28.5070	0.8349	0.9418	0.1161	25.0683	0.6489	2.1135	0.1758	19.8286	0.7785	1.8984	0.1397
Supervised† (Real noise)	DnCNN	32.1370	0.8729	0.9436	0.1195	28.3677	0.7596	1.7756	0.1308	31.7101	0.8702	1.8329	0.1365
	CBDNet	32.1481	0.8737	0.9328	0.1172	28.4634	0.7632	1.7224	0.1199	32.1908	0.8744	1.8516	0.1362
	DIDN	32.2106	0.8766	0.9138	0.1136	28.5284	0.7638	1.7318	0.1208	31.6893	0.8721	1.8306	0.1346
	RIDNet	32.2239	0.8761	0.9257	0.1193	28.5052	0.7645	1.7212	0.1165	32.3644	0.8751	1.8185	0.1330
Supervised (Simu. B. noise)	DnCNN	30.1854	0.8504	0.9207	0.1115	25.2321	0.6397	2.0070	0.1333	21.5630	0.7841	1.8849	0.1413
	CBDNet	30.1709	0.8649	0.9083	0.1120	25.7379	0.6659	1.9419	0.1393	20.5770	0.7240	1.8812	0.1347
	DIDN	27.6189	0.7925	0.9343	0.1160	23.9277	0.5360	2.0953	0.1573	19.7232	0.7501	1.8860	0.1355
	RIDNet	29.2904	0.8583	0.9083	0.1120	25.7051	0.6692	1.8530	0.1179	21.3333	0.7771	1.8801	0.1367
Supervised (Simu. S. noise)	DnCNN	29.5940	0.8136	0.9067	0.1133	25.4207	0.6833	1.8802	0.1303	21.1221	0.6055	2.2417	0.1535
	CBDNet	29.1485	0.8390	0.8897	0.1136	25.6212	0.6511	1.9435	0.1353	21.2571	0.8170	1.8731	0.1349
	DIDN	28.4246	0.8224	0.9248	0.1123	23.8874	0.5360	2.1942	0.1831	19.4537	0.7734	1.8697	0.1343
	RIDNet	29.4556	0.8381	0.8866	0.1107	25.7635	0.6615	1.9327	0.1366	20.7765	0.7536	1.9175	0.1419
Self-supervised	DIP	24.9301	0.6770	1.1065	0.1392	24.8945	0.6198	2.0266	0.1381	18.8924	0.6172	2.1397	0.2039
	ZSn2n	25.4501	0.6338	0.9356	0.1112	24.3310	0.5657	2.2003	0.1750	19.8465	0.7208	1.8933	0.1349
	N2V	26.9151	0.7254	0.9276	0.1129	27.3298	0.7307	1.8131	0.1329	20.6787	0.8158	1.8430	0.1345
	B2U	28.0615	0.8309	0.9516	0.1182	27.3305	0.7259	1.9566	0.1448	19.9928	0.8202	1.8472	0.1341
Unpaired Supervised	CycleGAN	30.0754	0.8178	0.9416	0.1168	24.8142	0.6157	1.9871	0.1343	17.5717	0.6784	2.1997	0.1537
	ISCL	27.4992	0.6955	0.9186	0.1106	24.5935	0.5534	1.9401	0.1383	24.9738	0.6536	1.9388	0.1376
	DRGAN	28.3473	0.7950	0.9619	0.1151	25.2071	0.6809	2.1103	0.1617	27.1759	0.7905	1.9788	0.1393
	ADN	29.1944	0.7928	0.9132	0.1130	26.9903	0.6943	1.8928	0.1323	27.5474	0.8042	1.9521	0.1394
	C2N	30.3313	0.8431	0.9264	0.1155	26.5155	0.6876	1.9083	0.1343	28.8376	0.7962	1.9082	0.1361
	Ours	**31.6192**	**0.8655**	**0.8791**	**0.1093**	**28.0709**	**0.7416**	**1.7194**	**0.1138**	**30.1192**	**0.8211**	**1.8183**	**0.1339**

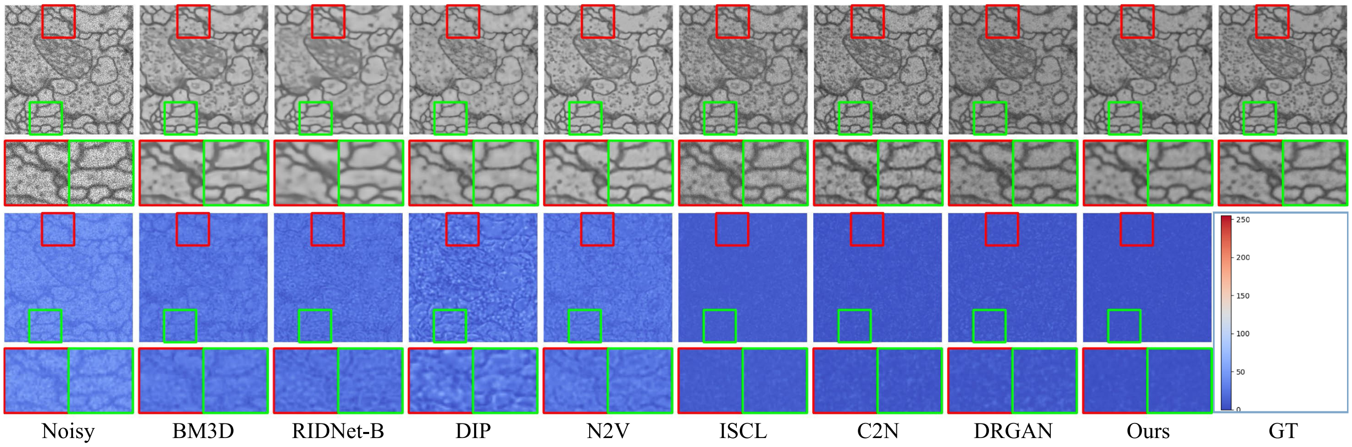


Fig. 4. Visual comparison (denoising results and error maps) on *synthetic* test data.

B. Results on Synthetic Data

The denoising performance is evaluated using the peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) between denoised images and their corresponding groundtruth. As shown in Table I, our method outperforms all methods except the denoising networks trained with ideal supervision, *i.e.*, "Supervised† (Real Noise)", reaching a performance close to it. Self-supervised methods perform poorly due to the lack of knowledge about the distribution of clean images. Supervised methods trained with inconsistent noise distributions, *i.e.*, "Supervised (Simu. B. noise)" and "Supervised (Simu. S. noise)" struggle to obtain pleasing results due to domain shift issues. "Supervised (Simu. B. noise)" performs slightly better, possibly attributed to training on a wider range of noise distributions. On the contrary, unpaired methods exhibit more potential, especially in CREMI-C subsets with significant domain shifts, where only they perform well. Our approach consistently outperforms other unpaired methods with over 1dB improvement in PSNR, showing its superiority.

Fig. 4 presents denoising visual results on the CREMI-C subset. The top two rows display the denoised images, while the bottom two rows show the error maps, reflecting the differences between the denoising results and the groundtruth.

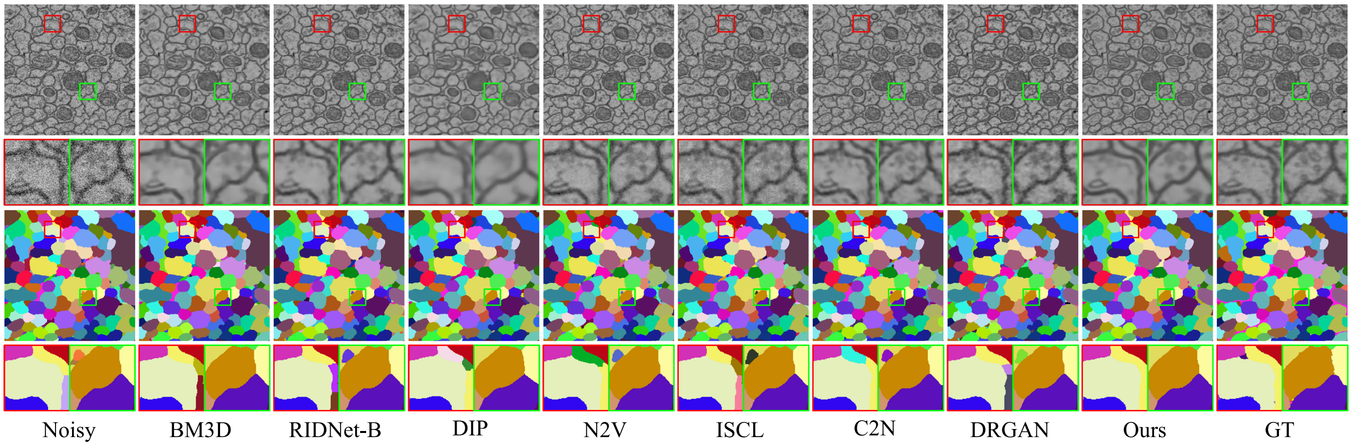


Fig. 5. Exemplar segmentation results on the denoising results of a *synthetic* noisy image. Each pseudo color represents one neuron.

TABLE II
QUANTITATIVE DENOISING RESULTS (PSNR / SSIM) ON REAL IMAGES FROM THE RETINA DATASET

Dataset		Validation		Testset	
Methods		PSNR↑	SSIM↑	PSNR↑	SSIM↑
Noisy		15.6634	0.1934	14.2952	0.1758
Non-learning based	BM3D	22.6492	0.5156	21.2239	0.4719
	WNNM	22.4323	0.4945	21.5921	0.4645
Supervised† (Real noise)	DnCNN	24.0432	0.5365	23.0191	0.5058
	CBDNet	24.1155	0.5124	23.1155	0.5124
	DIDN	24.2939	0.5460	23.2877	0.5172
	RIDNet	24.2366	0.5434	23.2135	0.5144
Supervised (Simu. B. noise)	DnCNN	22.6007	0.5205	20.8463	0.4580
	CBDNet	22.2104	0.5085	20.3308	0.4349
	DIDN	22.6184	0.5208	21.1247	0.4661
	RIDNet	22.7712	0.5329	20.7023	0.4479
Supervised (Simu. S. noise)	DnCNN	21.5513	0.4662	21.2053	0.4810
	CBDNet	20.9893	0.4393	20.7679	0.4650
	DIDN	22.1402	0.4847	21.1448	0.4798
	RIDNet	21.3063	0.4527	21.2326	0.4856
Self-supervised	ZSn2n	17.2194	0.2437	16.0217	0.2267
	N2V	17.6010	0.2572	16.3814	0.2415
	B2U	17.8769	0.2670	16.6687	0.2494
	DIP	21.7172	0.2514	20.5437	0.4345
Unpaired Supervised	CycleGAN	21.5676	0.3955	20.7710	0.3770
	ISCL	22.0068	0.4694	20.8831	0.4369
	ADN	22.2555	0.4491	21.2033	0.4089
	C2N	22.0845	0.4222	21.2548	0.4602
	DRGAN	22.1244	0.4819	21.3647	0.4271
	Ours	**23.5206**	**0.5442**	**22.6452**	**0.5085**

TABLE III
QUANTITATIVE RESULTS (PSNR / SSIM) OF GENERALIZATION ANALYSIS ON CREMI DATASET

Dataset	Methods	CREMI-A	CREMI-B	CREMI-C
CREMI-A	RIDNet	**32.2239/0.8761**	23.5491/0.5634	22.8823/*0.8051*
	ISCL	27.4992/0.6955	24.9920/0.5574	24.3081/0.6679
	ADN	29.1944/0.7928	25.7027/0.6810	26.6956/0.7453
	C2N	30.3313/0.8431	*26.1923/0.6841*	*28.3569*/0.7623
	Ours	*31.6192/0.8655*	**27.6957/0.7412**	**29.7980/0.8197**
CREMI-B	RIDNet	29.1675/*0.8541*	**28.5052/0.7645**	19.6898/*0.7609*
	ISCL	26.2357/0.6661	24.5935/0.5534	24.8054/0.6350
	ADN	28.1880/0.7848	26.9903/0.6943	27.0397/0.7607
	C2N	*30.0387*/0.8325	26.5155/0.6876	*28.0295*/0.7540
	Ours	**31.3431/0.8653**	*28.0709/0.7416*	**29.6123/0.8010**
CREMI-C	RIDNet	22.6140/*0.8377*	20.3121/0.5141	**32.3644/0.8751**
	ISCL	26.0401/0.6686	24.3554/0.5118	24.9738/0.6536
	ADN	*29.2705*/0.8263	25.5250/0.6694	27.5474/0.8042
	C2N	29.2630/0.8048	*26.1296/0.6792*	28.8376/0.7962
	Ours	**31.3671/0.8603**	**27.8762/0.7371**	*30.1192/0.8211*

TABLE IV
QUANTITATIVE RESULTS OF DIFFERENT SOURCE NOISE MODELING STRATEGY ON THE RETINA DATASET

Dataset	Validation		Testset	
Noise Type	PSNR↑	SSIM↑	PSNR↑	SSIM↑
AWGN	23.5206	0.5441	22.6452	0.5085
Film noise	23.5463	0.5158	22.4281	0.4879
Poisson-Gaussian	23.7193	0.5447	22.7369	0.5148
Speckle noise	22.7823	0.5147	21.8688	0.4840

It can be observed that BM3D, RIDNet-B, DIP, and N2V produce over-smoothed denoising results. The error maps show that these methods do not completely remove noise, leaving a considerable amount of noise artifacts. ISCL, C2N, and DRGAN achieve relatively decent denoising results, but some residual noise remains. On the contrary, our approach effectively removes noise while preserving the texture and details of the image, resulting in visual results closest to the groundtruth. Fig. 5 shows the impact of image denoising results from different methods on the downstream neuron segmentation. We utilize the same pre-trained neuron segmentation model [54] in the clean CREMI dataset for evaluation. It is evident that when noise removal is inadequate, segmentation split errors occur. Our method provides the most effective denoising results and achieves the highest accuracy in neuron segmentation, consistent with the variation of information (VOI) and the adapted rand error (ARAND) result in Table I.

C. Results on Real Data

As shown in Table II, our method outperforms other blind denoising methods. Existing self-supervised denoising methods and supervised models trained on synthetic noisy data perform even worse than traditional denoising methods like BM3D and WNNM when faced with complex real-world noise. Unpaired supervised methods generally achieve decent performance on both the validation and test sets but still exhibit some residual noise in the visual denoising results, as shown in Fig. 6. In contrast, our method excels in denoising performance. Compared with other unpaired denoising algorithms, our method improves the performance by more than 1dB. Notably, our method approaches the performance of supervised models trained on real noisy-clean image pairs in terms of the SSIM metric. As shown in Fig. 6, WNNM, RIDNet-B, DIP, B2U, and ISCL retain noticeable noise, especially structural noise related to image content. C2N and DRGAN exhibit relatively promising results. Our method achieves the most visually pleasing results and is closest to clean images.

The denoising results shown in Table II and Fig. 6 also validate the potential of our method in eliminating inherent noise. Generally, the inherent noise in EM images can be classified into several types, including electronic noise, shot noise, and environmental noise. These noises are typically associated with the properties of the electron beam, the characteristics of the sample itself, and the imaging environment. Due to the complexity and variability of imaging processes, image noise generated by different imaging devices on various biological tissues in different environments may exhibit completely different distributions. In order to make denoising method more universally applicable, our algorithm does not explicitly analyze the characteristics of these noises. Instead, it analyzes the commonalities in the potential biological structures of noisy images, implicitly learning the distribution of noise to aid in training the denoising network. Compared to extracting commonalities from complex and diverse noise, we believe that the similarity of biological structures is more significant.

D. Analysis and Discussion

1) Significance of EM Image Denoising: Although there are now some large-scale EM image datasets, such as FAFB and FIB-25, these datasets still require several image processing steps [55], [56], [57], [58] to enhance image quality and clarity from data acquisition to application. Additionally, it is always necessary to obtain different biological sample tissues at different resolutions to analyze the structures of different organisms and explore their functions. Therefore, existing EM image datasets cannot fully meet the needs of biological research, and existing electron microscopy imaging techniques still face a trade-off between imaging speed and imaging quality. Therefore, it remains meaningful to explore image processing algorithms to mitigate this trade-off.

2) Generalization Analysis: We evaluate our method's generalization performance when training with unpaired noisy images and clean images from different brain regions. The results are presented in Table III. For unpaired methods, the first column in the table represents the clean image dataset used for training, and the first row represents the noisy image dataset used for training and testing. In the case of the supervised learning method RIDNet, the first column and first row represent the training and testing datasets, respectively. The results in the table demonstrate the superior generalization ability of our method compared to other unpaired denoising methods. Notably, when there is a distribution difference between the training and testing datasets, our method significantly outperforms the supervised method, *i.e.,* RIDNet.

3) Noise Modeling for Synthetic Data: We also explore how different noise modeling strategies for synthetic noisy images (source domain) impact the denoising performance on real noisy images (target domain). As depicted in Table IV, we compare the effects of additive white Gaussian noise (AWGN), film noise, Poisson-Gaussian noise, and Speckle noise. It can be seen that when Poisson-Gaussian noise is added to the synthetic source images, the denoising model performs the best on real noisy images, reaching a PSNR/SSIM of 23.7193dB/0.5447 on the validation set. This notable improvement can be attributed to that the distribution of Poisson-Gaussian noise is closer to the real noise distribution in the RETINA dataset. By making full use of the valuable source domain data, our approach attains impressive denoising results.

4) Generated Pseudo-Noisy Images: As a byproduct of the unpaired denoising algorithms, we also compare the noise images generated by each method, which should resemble real noisy images. Fig. 7 illustrates the noisy images generated by different algorithms and their enlarged versions. The top two rows depict the noise images generated by each method. CycleGAN and C2N generate noise distributions that are too uniform and do not accurately reflect real noise distributions. ISCL, DRGAN, and ADN produce noisy images that relatively better resemble real noise patterns, but the generated noisy images of DRGAN still lack correlation with the underlying image signal. ISCL and ADN tend to underestimate or overestimate noise levels, respectively. On the contrary, our method generates noisy images that closely resemble real noise distributions, attributed to our domain adaptation strategy.

5) Distribution Analysis: We use t-SNE visualization to analyze the effectiveness of our domain alignment strategy. Fig. 8(a) shows that intermediate representations of the source and target domains obtained through disentanglement networks $G_s(\cdot)$ and $G_t(\cdot)$ can be clustered into three groups. Content representations from both domains, *i.e.,* F_i^C and F_j^C are well-clustered together, while noise components (N_j^R and N_i^S) are separated into two distinct clusters. In Fig. 8(b), the predicted real noise map $\hat{x}_i^R - y_i$ generated by our method closely resembles the distribution of real noise maps, and the predicted synthetic noise map $\hat{x}_j^S - y_j^{\mathcal{T}}$ is close to the synthetic noise distribution in the source domain.

6) Amount of Clean Training Data: In practice, obtaining clean images is much more time-consuming than acquiring noisy images. To evaluate the robustness of denoising methods, we also analyze the impact of reducing the number of clean training images on the performance of each method. As shown in Fig. 9, C2N and ADN show significant performance drops. RIDNet-R† (ideal case) exhibits notable degradation with only 1% clean training images. On the contrary, our method still has relatively stable performance, outperforming RIDNet-R† in SSIM with only 1% clean images, highlighting its practicality.

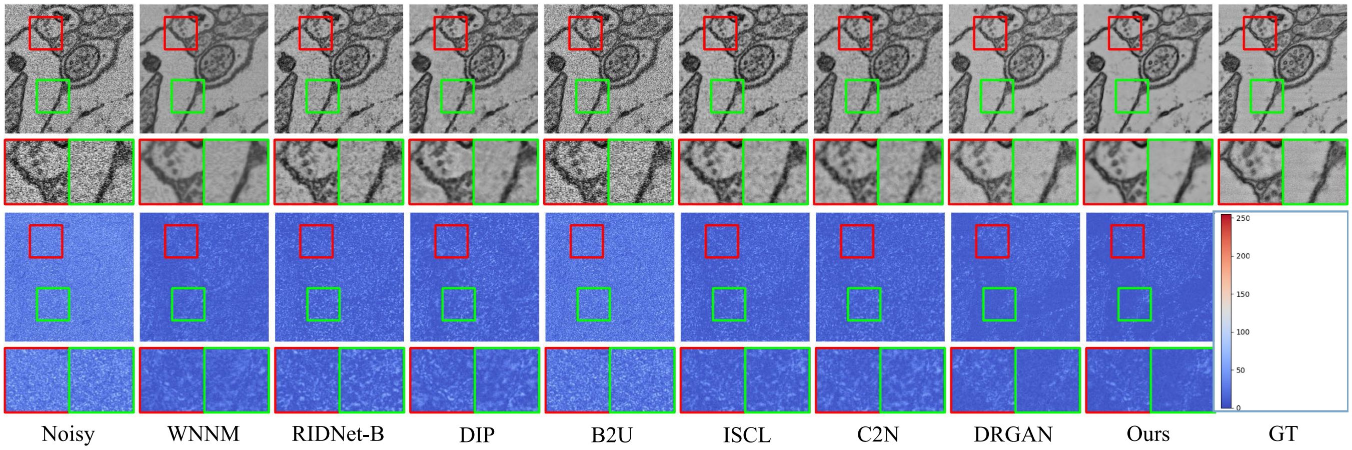


Fig. 6. Visual comparison (denoising results and error maps) on *real* test data.

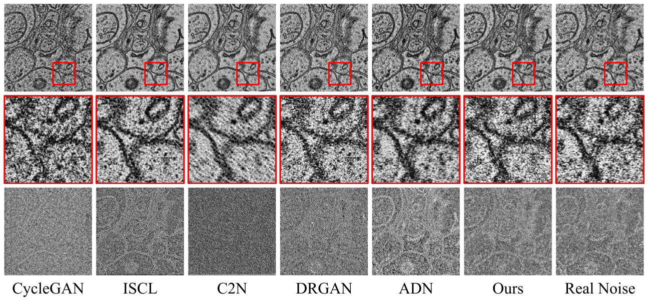
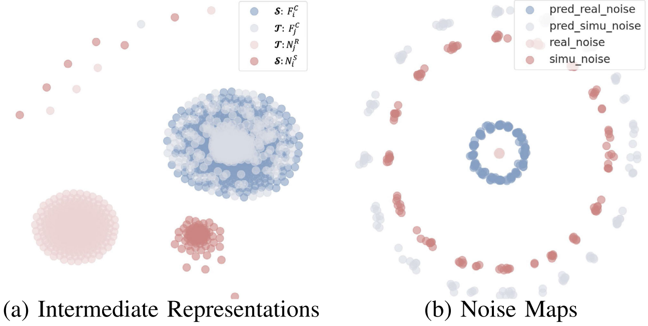
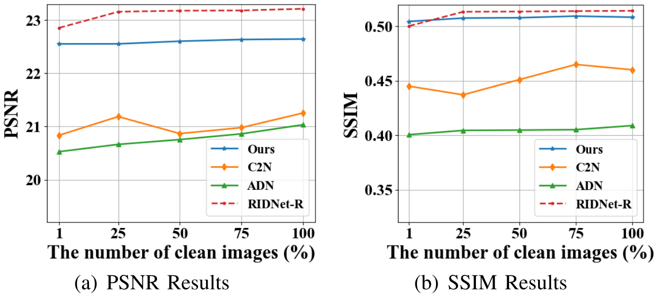


Fig. 7. Visual comparison (generated pseudo-noisy images and the corresponding noise maps) of unpaired denoising methods.



(a) Intermediate Representations (b) Noise Maps

Fig. 8. t-SNE visualization of the distribution of (a) intermediate representations and (b) noise maps.



(a) PSNR Results (b) SSIM Results

Fig. 9. Quantitative results on *real* dataset with fewer clean images.

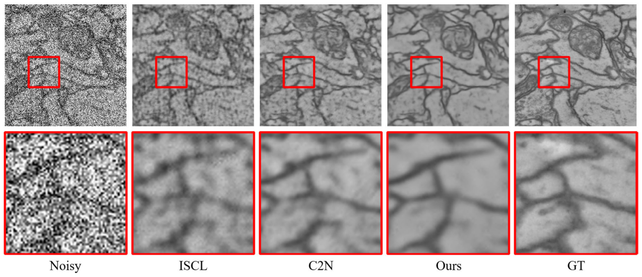


Fig. 10. Visual comparison of denoising results on LR noisy images.

TABLE V
Quantitative Comparison of Denoising Results on Noisy Images With Varying Resolutions

Methods	Resolution	× 2		× 4	
ISCL	LR	23.8545	0.6030	22.0093	0.5214
C2N	LR	24.5874	0.6428	22.0527	0.5020
Ours	LR	25.8486	0.6994	23.1115	0.5697
ISCL	HR	23.6722	0.5713	21.2327	0.4633
C2N	HR	24.3649	0.6189	21.2263	0.4691
Ours	HR	25.3137	0.6578	22.0656	0.5198

TABLE VI
Quantitative Comparison of Segmentation Results on Denoised Images at Different Resolutions

Methods	×2		×4	
	VOI↓	ARAND↓	VOI↓	ARAND↓
ISCL	2.4547	0.1791	5.8877	0.8965
C2N	2.2664	0.1512	5.0096	0.5458
Ours	2.0612	0.1391	4.5808	0.4478

7) Denoising Performance on Noisy Images With Varying Resolution: To further demonstrate the potential of the proposed algorithm for denoising low-resolution images, we conduct experiments on noisy images at different resolutions. Specifically, we first downscaled the original clean CREMI-B images using bicubic operator and then added noise to simulate low-resolution noisy images in practical electron microscopy imaging. Subsequently, we applied different denoising algorithms to these low-resolution noisy images. To obtain high-resolution images, we uniformly upsampled these denoised results using bicubic interpolation, thereby

TABLE VII

Quantitative Comparison of Denoising Results (PSNR / SSIM) with Different Downsampling Algorithms and Downsampling Factors In the Content Guidance Loss

Scale	×2		×4	
Algorithms	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Nearest	23.5153	0.5447	23.4152	0.5300
Bilinear	23.5206	0.5441	23.4323	0.5346
Bicubic	23.7139	0.5469	23.6029	0.5429
Lanczos	23.7394	0.5427	23.6400	0.5328

TABLE VIII

Quantitative Results of Different Loss Weights on Retina Testset ($\lambda_{dn}^T = 1$, $\lambda_{dn}^S = 1$, $\lambda_{sc} = 0.05$)

λ_{GAN}^S	λ_{GAN}^T	λ_{GAN}^C	λ_{guide}^C	PSNR↑	SSIM↑
1	1	1	1	21.9909	0.4973
1	1	1	0.1	22.1699	0.5029
0.2	0.2	0.2	0.1	22.4814	0.5087
0.2	0.2	0.1	0.1	22.6452	0.5085

TABLE IX

Quantitative Comparison of Results With/Without a Pre-Trained Encoder on the CREMI Dataset

Dataset	CREMI-A		CREMI-B		CREMI-C	
Methods	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Ours w/o$\mathcal{L}_{sc}$	31.0331	0.8474	27.4689	0.7324	29.1953	0.8005
Ours	31.6192	0.8655	28.0709	0.7416	30.1192	0.8211
Gain	0.5861	0.0181	0.6020	0.0092	0.9239	0.0206

TABLE X

Parameter Count and Computational Cost Comparison of Denoising Algorithms

Methods	Params.	FLOPs
ISCL	36,693,704	140.56G
ADN	45,907,332	162.32G
C2N	6,958,940	115.97G
DRGAN	64,296,104	214.25G
Ours	4,216,024	98.90G

obtaining clean high-resolution images. Throughout this process, both the clarity and resolution of the images can be improved. Table V demonstrates the denoising results of different denoising algorithms on images with varying resolutions (downsampled by 2 times and 4 times). "LR" represents the results before bicubic interpolation, while "HR" represents the results after bicubic interpolation. It can be observed that our method consistently achieves the optimal performance, showcasing the advantages of our algorithm in enhancing image signal-to-noise ratio and resolution. In Fig. 10, we present a visual comparison of the denoising results obtained by different denoising algorithms on low-resolution (×2) noisy images. To facilitate comparison, both the noisy and denoised images have been upsampled to their original resolution. As shown in the figure, the fine details of biological structures in the low-resolution noisy images are submerged by noise, resulting in a significant degradation of image quality. Although ISCL and C2N exhibit some improvement in image quality, the reconstruction of details remains imperfect. Conversely, our method notably enhances image quality, preserving the integrity of restored image details to a greater extent. After being processed by our algorithm, as shown in "Ours", numerous biological structures in the image are restored and reconstructed. These subtle structures are unknown to denoising algorithms. This example demonstrates that our algorithm has the potential to recover unknown biological structures. To further illustrate the superiority of our denoising algorithm over others at different resolutions, we validated different algorithms on downstream tasks like neuron segmentation. As shown in Table VI, the restoration outcomes of our algorithm consistently outperform others in neuronal segmentation tasks, confirming the accuracy and effectiveness of our method in detail reconstruction compared to other methods.

8) Design of Content Guidance Loss: We employ content guidance loss in domain alignment learning to guide the representation disentanglement. During the disentanglement phase, we utilize downsampling to ensure that the information of the disentangled content representations and noise components matches that of the input noisy image. If downsampling is not employed, the cumulative information of the content representations and the noise components would exceed that of the input image, potentially introducing irrelevant details. The imperfect representation disentanglement would lead to the generation of pseudo-noisy images that incorporate irrelevant information, thereby increasing the difficulty of denoising learning. To facilitate disentanglement learning, we introduce content guidance loss to guide the learning of content representations. The content guidance loss is performed on downscaled noisy images and content representations. As for the implementation of content guidance loss, nearest, bilinear, bicubic, and lanczos are all classic downsampling algorithms. In favor of the low computational cost, we choose the bilinear downsampling. We evaluate the impact of various downsampling algorithms and downsampling scales, presenting the results on RETINA validation set in Table VII.

9) Impact of Loss Weights: We evaluate how different loss weights affect denoising performance on the RETINA test set. The corresponding results are detailed in Table VIII. In our initial experiments, we standardize all loss weights except λ_{sc} to 1. To ensure the comparability of λ_{sc} with other loss terms, we set λ_{sc} to 0.05 and achieve results surpassing all baseline methods (refer to the second row of Table VIII). Subsequently, recognizing the auxiliary role of the content guidance loss in content representation learning, we reduced λ_{guide}^C to 0.1, resulting in a modest performance improvement (refer to the third row of Table VIII). Moreover, to mitigate training instability potentially caused by excessive GAN loss, we decreased the associated weights to 0.2, significantly improving denoising performances (refer to the fourth row of Table VIII). Finally, aiming to alleviate constraints on domain alignment, we set λ_{GAN}^C to 0.1, yielding the optimal results (refer to the fifth row of Table VIII).

10) Analysis of Pre-Trained Encoder: In Sec.III-B.2, we introduce the pre-trained encoder for semantic consistency enforcement. To further analyze the effectiveness of the pre-trained encoder, we investigate the influence of domain gaps between the testing noisy dataset and the training dataset of the pre-trained encoder. The FAFB dataset used for pretraining the semantic encoder and the source synthetic denoising dataset CREMI are both derived from the fruit fly brain, exhibiting certain similarities in distribution. However, the real denoising dataset RETINA, originating from the mouse retina, differs somewhat in distribution from the FAFB dataset. To further illustrate the effectiveness of the pre-trained encoder on noisy datasets with different data distribution variances, we conduct experiments on the CREMI and RETINA datasets, with the results shown in Table IX and Table XI. When comparing the results, it becomes clear that a pre-trained encoder consistently improves denoising performance across datasets with varying degrees of data distribution differences, such as RETINA and CREMI. Moreover, the performance improvement observed with the pre-trained encoder on the RETINA dataset remains notable when compared to that on the CREMI dataset, despite the disparities in data distribution. As shown in Table XI, the results of the validation set using the pre-trained encoder ("Ours": 23.5206/0.5442) are superior to those without using the pre-trained encoder ("Ours w/o $\mathcal{L}_{sc}$": 23.2767/0.5251). Adding the semantic consistency loss provided by the pre-trained encoder still improves denoising performance, demonstrating the generalizability of the pre-trained encoder model. This could be attributed to the fact that the FAFB dataset and the RETINA dataset exhibit some biological semantic consistency, as both datasets extensively contain structures like cell membranes and vesicles. This implies that a pre-trained encoder on datasets with slightly different distributions can still effectively extract semantic information and contribute to the training of denoising networks. In practical applications, even if it is not possible to obtain datasets with completely consistent distributions, other publicly available datasets can be used to train the encoder. Currently, in the field of biomedical imaging, there are many publicly available datasets from electron microscopy, light microscopy, CT, MRI, etc. Thanks to these existing public datasets, the pre-trained encoder will further enhance algorithm performance.

11) Parameter Count and Computational Cost: We conduct a comparison of parameter count and computational cost with other unsupervised denoising algorithms. As shown in Table X, our denoising algorithm has the smallest parameter count and computational complexity among unsupervised denoising algorithms. This is attributed to our elaborate network design, which enhances denoising performance without introducing additional computational overhead.

E. Ablation Study

1) Effectiveness of Each Component: In Table XI, we validate the effectiveness of each component of our method. Specifically, without domain adaptation strategy (Ours w/o DA) and instead utilizing feature disentanglement to achieve noise-to-clean image translation with invertible networks leads to a significant performance drop (about 1dB in PSNR), confirming the effectiveness of our domain adaptation learning strategy. Furthermore, when the semantic consistency constraint is removed (Ours w/o $\mathcal{L}_{sc}$) or the domain alignment constraint is lacking (Ours w/o $\mathcal{L}^{C}_{GAN}$), there is also a considerable performance drop. Additionally, if the domain regularization constraint from the source domain is missing (Ours w/o $\mathcal{L}^{\mathcal{S}}_{GAN}$) or the constraint on denoising the synthetic noisy images is not applied (Ours w/o $\mathcal{L}^{\mathcal{S}}_{dn}$), the best denoising performance cannot be achieved. The performance gap between "Ours w/o $\mathcal{L}^{\mathcal{S}}_{dn}$" and "Ours" is relatively small, indicating that our framework effectively learns target domain information. Therefore, utilizing the target domain denoising loss $\mathcal{L}^{\mathcal{T}}_{dn}$ alone can achieve good denoising performance. However, as mentioned in Sec. III-C, when the differences between the source and target domains are small, incorporating $\mathcal{L}^{\mathcal{S}}_{dn}$ can further improve denoising performance, as it augments the data during the training of the denoising network. These results highlight the importance and utility of each loss function in our approach.

TABLE XI
ABLATION RESULTS (PSNR / SSIM) ON RETINA DATASET

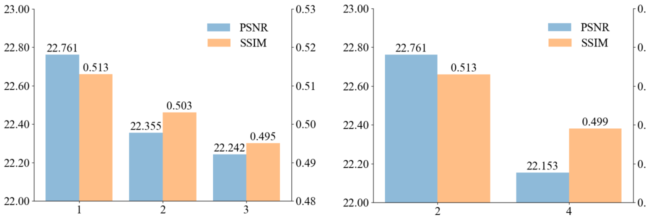
Dataset	Validation		Testset	
Methods	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Ours w/o DA	22.5676	0.5194	21.6248	0.4916
Ours w/o $\mathcal{L}_{sc}$	23.2767	0.5251	22.2354	0.4973
Ours w/o $\mathcal{L}^{C}_{GAN}$	23.3683	0.5351	22.2520	0.5077
Ours w/o $\mathcal{L}^{\mathcal{S}}_{GAN}$	23.4364	0.5382	22.3478	0.5079
Ours w/o $\mathcal{L}^{\mathcal{S}}_{dn}$	23.4997	0.5308	22.4414	0.5047
Ours	**23.5206**	**0.5442**	**22.6452**	**0.5085**

TABLE XII
QUANTITATIVE COMPARISON OF RESULTS WITH/WITHOUT USING INVERTIBLE NETWORKS ON THE RETINA DATASET

Dataset	Validation		Testset	
Methods	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Ours w/ CNN	23.1960	0.5214	22.2794	0.4899
Ours w/ INN	23.5206	0.5441	22.6452	0.5085

2) Necessities of INNs: We incorporate a performance comparison with the conventional CNN employed as the backbone network for disentanglement-reconstruction. To ensure fairness in comparison, we maintained consistency in the training strategy. As indicated by Table XII, the results of "Ours w/ INN" outperform those of "Ours w/ CNN". This is attributed to the superior ability of INN to preserve information throughout the image disentanglement and reconstruction process, consequently producing high-quality pseudo-noisy images. This, in turn, provides better training data for the denoising network, ultimately leading to improved denoising performance.

3) Specific Hyperparameters of INNs: We analyze the hyperparameters' impact in $G_s(\cdot)$ and $G_t(\cdot)$. As shown in Fig. 11,



(a) The value of r (b) The value of n

Fig. 11. Ablation results of hyperparameters of $G_s(\cdot)/G_t(\cdot)$.

the best performance is achieved when the invertible networks $G_s(\cdot)$ and $G_t(\cdot)$ include two downscaling modules ($n = 2$) and use the first channel of the intermediate representations ($r = 1$) to represent the content representations.

VI. Future Work

In this section, we discuss potential research avenues in EM image denoising. These directions aim to refine existing electron microscopy techniques and improve imaging quality. We summarize the following three key areas:

- Joint task of denoising and super-resolution: In most existing EM image restoration works, only a single type of image degradation (such as noise or low resolution) is addressed. Future work should integrate denoising and super-resolution techniques to address noise and low resolution simultaneously, reflecting real-world imaging conditions.
- Utilizing information from 3D EM image sequences: Our current approach focuses on individual noisy images, which ignores adjacent images. In future work, it is possible to leverage the multi-frame information of 3D EM image sequences for denoising. This will result in improved axial continuity and consistency in denoised images.
- Discriminating between biological fine structures and noise: Due to the nanometer-level resolution of existing electron microscopy imaging, noise and small biological structures have certain similarities, which poses a challenge for denoising tasks. In future work, it may be beneficial to utilize prior knowledge of fine biological structures to accurately remove noise while preserving the biological structure in the images.

These future research directions aim to push the boundaries of EM image restoration and contribute to advancements in biological imaging.

VII. Conclusion

In this paper, we present a novel unpaired EM image denoising method from the perspective of domain adaptation. Inspired by the observation that biomedical EM images share content characteristics, our method begins with establishing a domain-agnostic content space between synthetic and real noisy images through domain alignment learning. We further introduce domain regularization to ensure precise domain alignment and generate pseudo-noisy images that conform to the real noise distribution, obtaining image pairs for denoising network training. To achieve lossless representation disentanglement and image reconstruction, we adopt invertible networks in our framework. Extensive experiments on synthetic and real datasets demonstrate the superiority of our method over existing methods quantitatively and qualitatively.

References

[1] Z. Zheng et al., “A complete electron microscopy volume of the brain of adult Drosophila melanogaster,” *Cell*, vol. 174, no. 3, pp. 730–743, Jul. 2018.

[2] *MICCAL Challenge on Circuit Reconstruction From Electron Microscopy Images*, CREMI, 2017. [Online]. Available: https://cremi.org/

[3] J. Roels et al., “An overview of state-of-the-art image restoration in electron microscopy,” *J. Microsc.*, vol. 271, no. 3, pp. 239–254, 2018.

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3-D transform-domain collaborative filtering,” *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[5] S. Gu, L. Zhang, W. Zuo, and X. Feng, “Weighted nuclear norm minimization with application to image denoising,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2862–2869.

[6] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising,” *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.

[7] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, “Toward convolutional blind denoising of real photographs,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1712–1722.

[8] S. Yu, B. Park, and J. Jeong, “Deep iterative down-up CNN for image denoising,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 2095–2103.

[9] A. Saeed and B. Nick, “Real image denoising with feature attention,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 3155–3164.

[10] C. Chen, Z. Xiong, X. Tian, Z.-J. Zha, and F. Wu, “Real-world image denoising with deep boosting,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 12, pp. 3071–3087, Dec. 2020.

[11] Y. Zhang, K. Li, K. Li, B. Zhong, and Y. Fu, “Residual non-local attention networks for image restoration,” 2019, *arXiv:1903.10082*.

[12] J. Lehtinen et al., “Noise2Noise: Learning image restoration without clean data,” 2018, *arXiv:1803.04189*.

[13] A. Krull, T.-O. Buchholz, and F. Jug, “Noise2void-learning denoising from single noisy images,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 2129–2137.

[14] J. Batson and L. Royer, “Noise2Self: Blind denoising by self-supervision,” in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 524–533.

[15] Z. Wang, J. Liu, G. Li, and H. Han, “Blind2unblind: Self-supervised image denoising with visible blind spots,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 2027–2036.

[16] J. Chen, J. Chen, H. Chao, and M. Yang, “Image blind denoising with generative adversarial network based noise modeling,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3155–3164.

[17] Z. Hong, X. Fan, T. Jiang, and J. Feng, “End-to-end unpaired image denoising with conditional adversarial networks,” in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 4, pp. 4140–4149.

[18] G. Jang, W. Lee, S. Son, and K. M. Lee, “C2N: Practical generative noise modeling for real-world denoising,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 2350–2359.

[19] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2223–2232.

[20] T. M. Quan et al., “Removing imaging artifacts in electron microscopy using an asymmetrically cyclic adversarial network without paired training data,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 3804–3813.

[21] H. Liao, W.-A. Lin, S. K. Zhou, and J. Luo, “ADN: Artifact disentanglement network for unsupervised metal artifact reduction,” *IEEE Trans. Med. Imag.*, vol. 39, no. 3, pp. 634–643, Aug. 2019.

[22] K. Lee and W. Jeong, “ISCL: Interdependent self-cooperative learning for unpaired image denoising,” *IEEE Trans. Med. Imag.*, vol. 40, no. 11, pp. 3238–3248, Nov. 2021.

[23] Y. Huang et al., “Noise-powered disentangled representation for unsupervised speckle reduction of optical coherence tomography images,” *IEEE Trans. Med. Imag.*, vol. 40, no. 10, pp. 2600–2614, Oct. 2021.

[24] Y. Ganin and V. Lempitsky, “Unsupervised domain adaptation by backpropagation,” in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1180–1189.

[25] Y. Liu et al., “Invertible denoising network: A light solution for real noise removal,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13365–13374.

[26] W. Lee, S. Son, and K. M. Lee, “AP-BSN: Self-supervised denoising for real-world images via asymmetric PD and blind-spot network,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17725–17734.

[27] V. Lempitsky, A. Vedaldi, and D. Ulyanov, “Deep image prior,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9446–9454.

[28] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, “Deep domain confusion: Maximizing for domain invariance,” 2014, *arXiv:1412.3474*.

[29] M. Long, Y. Cao, J. Wang, and M. Jordan, “Learning transferable features with deep adaptation networks,” in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 97–105.

[30] R. Volpi, P. Morerio, S. Savarese, and V. Murino, “Adversarial feature augmentation for unsupervised domain adaptation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5495–5504.

[31] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, “Contrastive adaptation network for unsupervised domain adaptation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 4893–4902.

[32] Y. Taigman, A. Polyak, and L. Wolf, “Unsupervised cross-domain image generation,” 2016, *arXiv:1611.02200*.

[33] M.-Y. Liu and O. Tuzel, “Coupled generative adversarial networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 1–9.

[34] M.-Y. Liu, T. Breuel, and J. Kautz, “Unsupervised image-to-image translation networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–9.

[35] Y. Ganin et al., “Domain-adversarial training of neural networks,” *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, May 2016.

[36] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, “Adversarial discriminative domain adaptation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7167–7176.

[37] Y. Chen, W. Li, X. Chen, and L. Van Gool, “Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1841–1850.

[38] Y. Zhang et al., “From whole slide imaging to microscopy: Deep microscopy adaptation network for histopathology cancer image classification,” in *Medical Image Computing and Computer Assisted Intervention*. Springer, 2019, pp. 360–368.

[39] W. Wang, H. Zhang, Z. Yuan, and C. Wang, “Unsupervised real-world super-resolution: A domain adaptation perspective,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 4318–4327.

[40] Y. Wei, S. Gu, Y. Li, R. Timofte, L. Jin, and H. Song, “Unsupervised real-world image super resolution via domain-distance aware training,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13385–13394.

[41] Y. Shao, L. Li, W. Ren, C. Gao, and N. Sang, “Domain adaptation for image dehazing,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2808–2817.

[42] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, “Least squares generative adversarial networks,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2794–2802.

[43] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 694–711.

[44] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, *arXiv:1409.1556*.

[45] Y. Chen, W. Huang, X. Liu, S. Deng, Q. Chen, and Z. Xiong, “Learning multiscale consistency for self-supervised electron microscopy instance segmentation,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2024, pp. 1566–1570.

[46] B. Lu, J.-C. Chen, and R. Chellappa, “UID-GAN: Unsupervised image deblurring via disentangled representations,” *IEEE Trans. Biometrics, Behav., Identity Sci.*, vol. 2, no. 1, pp. 26–39, Jan. 2020.

[47] J.-J. Huang and P. L. Dragotti, “WINNet: Wavelet-inspired invertible network for image denoising,” *IEEE Trans. Image Process.*, vol. 31, pp. 4377–4392, 2022.

[48] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in *Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[49] A. Eberle, S. Mikula, R. Schalek, J. Lichtman, M. K. Tate, and D. Zeidler, “High-resolution, high-throughput imaging with a multibeam scanning electron microscope,” *J. Microsc.*, vol. 259, no. 2, pp. 114–120, 2015.

[50] S. Saalfeld, R. Fetter, A. Cardona, and P. Tomancak, “Elastic volume reconstruction from series of ultra-thin microscopy sections,” *Nature Methods*, vol. 9, no. 7, pp. 717–720, 2012.

[51] L. Dinh, D. Krueger, and Y. Bengio, “NICE: Non-linear independent components estimation,” 2014, *arXiv:1410.8516*.

[52] X. Wang et al., “ESRGAN: Enhanced super-resolution generative adversarial networks,” in *Proc. Eur. Conf. Comput. Vis. (ECCV) Workshops*, 2018, pp. 63–79.

[53] Y. Mansour and R. Heckel, “Zero-shot noise2noise: Efficient image denoising without any data,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2023, pp. 14018–14027.

[54] J. Funke et al., “Large scale image segmentation with structured loss based deep learning for connectome reconstruction,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1669–1680, Jul. 2019.

[55] J. Funke, “Automatic neuron reconstruction from anisotropic electron microscopy volumes,” Ph.D. dissertation, Dept. Phys., ETH Zürich, Zürich, Switzerland, 2014.

[56] S. Popovych et al., “Petascale pipeline for precise alignment of images from serial section electron microscopy,” *Nature Commun.*, vol. 15, no. 1, p. 289, Jan. 2024.

[57] T. Macrina et al., “Petascale neural circuit reconstruction: Automated methods,” *bioRxiv*, to be published.

[58] S. Deng, W. Huang, C. Chen, X. Fu, and Z. Xiong, “A unified deep learning framework for ssTEM image restoration,” *IEEE Trans. Med. Imag.*, vol. 41, no. 12, pp. 3734–3746, Dec. 2022.