No-Clean-Reference Image Super-Resolution: Application to Electron Microscopy

Mohammad Khateri, *Student Member, IEEE*, Morteza Ghahremani, *Member, IEEE*, Alejandra Sierra, and Jussi Tohka

***Abstract*—The inability to acquire clean high-resolution (HR) electron microscopy (EM) images over a large brain tissue volume hampers many neuroscience studies. To address this challenge, we propose a deep-learning-based image super-resolution (SR) approach to computationally reconstruct a clean HR 3D-EM image with a large field of view (FoV) from noisy low-resolution (LR) acquisition. Our contributions are I) investigation of training with no-clean references; II) introduction of a novel network architecture, named EMSR, for enhancing the resolution of LR EM images while reducing inherent noise. The EMSR leverages distinctive features in brain EM images–repetitive textural and geometrical patterns amidst less informative backgrounds– via multiscale edge-attention and self-attention mechanisms to emphasize edge features over the background; and, III) comparison of different training strategies including using acquired LR and HR image pairs, i.e., real pairs with no-clean references contaminated with real corruptions, pairs of synthetic LR and acquired HR, as well as acquired LR and denoised HR pairs. Experiments with nine brain datasets showed that training with real pairs can produce high-quality super-resolved results, demonstrating the feasibility of training with nonclean references. Additionally, comparable results were observed, both visually and numerically, when employing denoised and noisy references for training. Moreover, utilizing the network trained with synthetically generated LR images from HR counterparts proved effective in yielding satisfactory SR results, even in certain cases, outperforming training with real pairs. The proposed SR network was compared quantitatively and qualitatively with several established SR techniques, demonstrating either the superiority or competitiveness of the proposed method in recovering fine details while mitigating noise.**

Manuscript received 25 January 2024; revised 7 May 2024 and 19 June 2024; accepted 30 June 2024. Date of publication 10 July 2024; date of current version 31 July 2024. This work was supported in part by the Research Council of Finland under Grant 323385 and Grant 358944, in part by the Flagship of Advanced Mathematics for Sensing Imaging and Modelling, and in part by the Jane and Aatos Erkko Foundation, and in part by the Doctoral Programme in Molecular Medicine at the University of Eastern Finland. The associate editor coordinating the review of this article and approving it for publication was Prof. Chao Zuo. *(Corresponding author: Mohammad Khateri.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Committee of the Provincial Government of Southern Finland under Application No. ESAVI/833/04.10.07/2013, and performed in line with the European Community Council Directives 86/609/EEC.

Mohammad Khateri, Alejandra Sierra, and Jussi Tohka are with the A. I. Virtanen Institute for Molecular Sciences, Faculty of Health Sciences, University of Eastern Finland, 70210 Kuopio, Finland (e-mail: mohammad.khateri@uef.fi; alejandra.sierralopez@uef.fi; jussi.tohka@uef.fi).

Morteza Ghahremani is with the A. I. Virtanen Institute for Molecular Sciences, Faculty of Health Sciences, University of Eastern Finland, 70210 Kuopio, Finland, and also with the Artificial Intelligence in Medical Imaging at the Department of Radiology, Technical University of Munich, 80333 München, Germany (e-mail: morteza.ghahremani@uef.fi).

The code is publicly available at https://github.com/mkhateri/EMSR.

Digital Object Identifier 10.1109/TCI.2024.3426349

***Index Terms*—Deep learning, electron microscopy, neuroscience, no-clean-reference, super-resolution.**

I. Introduction

THREE-DIMENSIONAL electron microscopy (3D-EM) is an essential technique for investigating brain tissue ultrastructures because it allows for 3D visualization at nanometer resolution [1], [2]. Studying brain tissue ultrastructures requires high-resolution (HR) images over a large field of view (FoV) of the brain tissue. However, since imaging at higher resolutions demands denser sampling, it takes more time, proportionally increasing the imaging cost and potential sample damage. Moreover, HR imaging over a large FoV is not feasible under realistic imaging constraints, demanding a trade-off between imaging resolution and FoV. The higher the resolution is, the smaller the FoV [3]. Furthermore, the imperfect components of imaging systems introduce noise into the images [4]. These limitations collectively prevent the acquisition of clean HR EM images over large FoVs of brain tissue, impeding subsequent brain ultrastructure analysis and visualization.

A practical approach to mitigating such limitations in providing clean HR EM images over a large tissue volume includes the following steps: I) low-resolution (LR) imaging of brain samples over a large FoV of interest, II) HR imaging over a small but representative portion of the same samples covered by the LR FoV, and III) utilizing the image super-resolution (SR) technique to computationally reconstruct high-quality HR 3D-EM images from the LR 3D-EM images of brain tissue, which are typically contaminated with noise, artifacts, and distortions.

SR is a low-level vision task that can serve as an integral preprocessing step for many image analyses in neuroscience [5], [6], [7]. It aims to recover the latent clean HR image x from a degraded LR observation y:

$$y = \mathcal{D}_\delta(x), \tag{1}$$

where $\mathcal{D}_\delta(\cdot)$ is the degradation function parameterized by δ, which is noninvertible, making SR an ill-posed inverse problem. $\mathcal{D}_\delta(\cdot)$ includes a convolution operator $\circledast$ with a blur kernel κ, an s-fold undersampling operator $\downarrow_s$, and noise n ($\delta = \{\kappa, \downarrow_s, n\}$) [8]. In practice, δ is unknown and we only have the LR observation.

SR methods can be categorized into two groups: model-based and learning-based methods. Model-based SR methods approximate the degradation function in (1) as a combination

of several operations. Assuming that the blurring kernel and undersampling operator are known and noise is additive:

$$\mathcal{D}_\delta(x) = (x \circledast \kappa)\downarrow_s + n \tag{2}$$

An estimate x^* of an HR image can then be obtained by the maximum a posteriori (MAP) formulation as:

$$x^* = \arg\min_x \{\|y - (x \circledast \kappa)\downarrow_s\|_p^q + \lambda \mathcal{R}(x)\} \tag{3}$$

The first term is the likelihood computed as the ℓ_p-norm distance between the observation y and the degraded latent image x, where $0 < p, q \leq 2$ are determined by the noise distribution [9], [10], [11], [12]. $\mathcal{R}(\cdot)$ is the regularization term, also known as the prior term, which penalizes the unknown latent image x upon our prior knowledge of the data. The parameter λ defines the tradeoff between likelihood and prior terms. To reduce the ill-posed nature of SR problems, many regularization terms have been developed [8], [13], and each has specific pros and cons. Notably, contributions from total variation [14], self-similarity [15], low rank [12], and sparse representation [16] have played a significant role in improving SR performance. Crafted priors enhance SR but have limited performance compared to data-driven methods [8]. Effective SR models involve optimizing multiple priors, which is time- and memory-consuming, and require tuning the tradeoff parameters. Additionally, SR models are specific to certain degradation settings, necessitating separate models for each degradation. Mismatched LR images with different degradations may result in severe artifacts due to domain gaps [17].

Learning-based SR methods learn a mapping between LR and HR image spaces, which is then used to restore the HR image from the given LR input image. Early work, pioneered by [18], restored HR images by capturing the co-occurrence prior between LR and HR image patches. Numerous patch-based methods relying on manifold learning [19], filter learning [20], regression [21], and sparse representation [22] have been introduced. Deep neural network (DNN)-based SR methods have demonstrated remarkable performance [13]. DNNs with end-to-end training avoid the need for explicit design of priors or degradations. Instead, priors and degradations are encapsulated in the training datasets. The commonly used DNN architectures include convolutional neural networks (CNNs) [23], [24], generative adversarial networks (GANs) [25], [26], [27], vision transformers (ViTs) [28], [29], and denoising diffusion probabilistic models (DDPMs) [30], [31]. In this field, many computer vision and biomedical imaging studies have defined a specific degradation function to synthesize LR images from HR counterparts to generate training data [8]. Several studies have also been conducted to incorporate the interpretability of model-based methods into end-to-end learning, e.g., deep unfolding [32], [33], [34], plug-and-play (PnP) [35], [36], [37], [38], and deep equilibrium learning [39], [40]. Although most of these degradation-oriented SR approaches lead to satisfactory results on benchmark datasets, they fail to restore high-quality images in real-world applications [17], such as brain EM images that are the focus of this study.

The computational approaches in super-resolution of EM have been studied in health and material sciences [41], [42], [43], [44]. As a pioneer, [42] proposed a material-specific PnP approach to super-resolve LR EM. Their method was based on the MAP formulation, where the likelihood term was based on a linear degradation model and the prior term was a library-based nonlocal means (LB-NLM) designed on HR EM images acquired within small FoVs. The presence of HR edges and textures corresponding to the LR input image in the designed library yielded super-resolved results with fine details. To reduce computational expenses and improve generalizability, [45] replaced the LB-NLM denoiser with an off-the-shelf Gaussian denoiser, leading to the version of PnP typically used in biomedical applications. However, both methods [42], [45] are essentially model-based, computationally cumbersome, and limited to degradation models. Experiments in both studies were conducted on EM datasets acquired from nanomaterial with simple textural information, which sparsely recurred throughout the image. By leveraging the unique characteristics of such images, authors in [46] devised a patch-based strategy on acquired pairs of LR and HR EM images in the training of LB-NLM, resulting in better performance than the original LB-NLM method but inferior performance compared to DNN-based methods. In [44], the authors introduced a DNN-based SR, named point scanning super-resolution (PSSR), for EM brain images. They proposed a degradation operator, i.e., crappifier, to synthesize LR images from acquired HR counterparts, where the crappifier included additive Gaussian noise followed by a downsampling operator. Using synthetic pairs of LR and HR EM images, they trained a UNet-based residual neural network. The performance of the method was then compared only with that of bilinear interpolation. Although synthetizing pairs of LR and HR EM images can reduce imaging costs, it can increase the domain gap between the input LR EM and the trained SR model.

The DNN-based SR method can implicitly learn EM degradations if trained with acquired and matched pairs of LR and HR images. However, many challenges impede the design of DNN-based frameworks with such data. First, EM images inherently contain noise and artifacts derived from the microscope, sample, and experimental settings. Hence, there is no clean EM image to be used as the reference for training the network. Furthermore, networks pretrained on natural images cannot restore high-quality brain EM images due to the considerable difference in the physics behind photography and EM as well as content dissimilarity between natural and brain EM images. Hence, deploying and designing SR methods for EM images requires specific considerations. In this work, we illustrate and address the mentioned challenges of the SR of EM images. Our key contributions are as follows:

- Investigation of training using no-clean references for ℓ_2 and ℓ_1 loss functions.
- Proposal of a deep learning (DL)-based image SR framework for EM, named EMSR, equipped with edge-attention and self-attention mechanisms for enhanced edge recovery. Sharing the network's modules between the original noisy LR EM image and its noisier version makes it robust to noise.
- Comparison of various training strategies for EM images, including training from pairs of physically acquired LR and HR, synthetically generated LR and HR, as well as LR and denoised HR EM images.

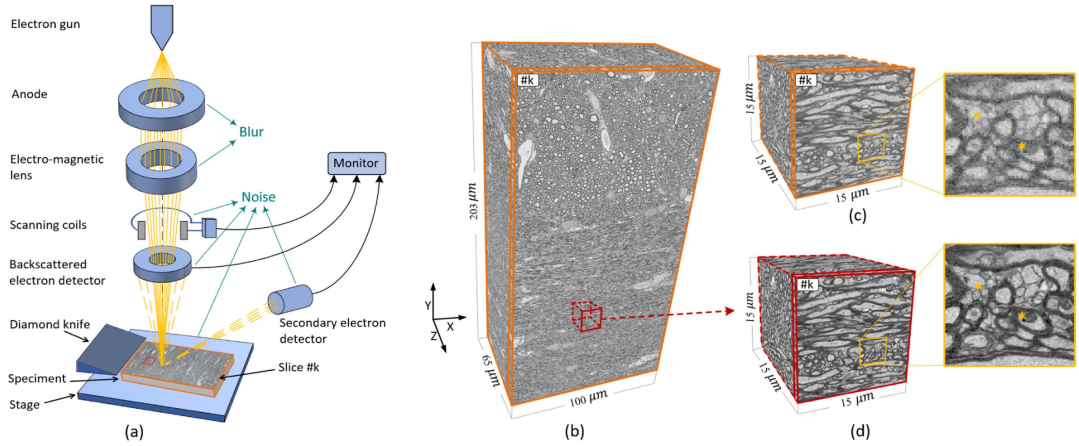

Fig. 1. Schematic diagram of serial block-face scanning electron microscopy and imaging. (a) The electron gun generates streams of electrons that are focused and raster scanned across the sample surface (solid yellow lines). The interaction of these focused electrons with the sample results in the ejection of electron streams (dashed yellow lines), which are collected by detectors to form a 2D image of the k-*th* slice (labeled $\#k$). Note that the region of interest from the sample is imaged at LR with a large FoV (marked in orange), while at HR, the FoV is smaller (marked in red). After imaging a slice, a diamond knife is used to cut the sample to a specific thickness to determine the resolution in the z direction and expose the subsequent block-face for imaging. Imperfections in the imaging device components can introduce blurring and noise in the resultant images (solid green arrows). (b) A stack of 2D image slices constitutes the 3D-EM dataset. (c) LR 3D-EM image corresponding to (d) HR 3D-EM image from a small FoV. The zoomed-in areas in (c) and (d) demonstrate the superior quality of the HR image in terms of contrast and resolution, see asterisks.

The remainder of this article is organized as follows: Section II describes the proposed image super-resolution method, Section III describes experimental results, and finally, Section IV concludes the article.

II. Proposed Method

The supervised training of a network requires numerous pairs of corrupted LR images and corresponding clean reference images. However, brain EM images inevitably include different types of noise, artifacts, and distortions, caused by the imaging system, and experimental settings. Therefore, clean EM images that serve as references are unavailable. Here, we investigate training a neural network for EM SR using physically acquired pairs of LR and HR EM images contaminated with real noise-like corruptions.

A. Electron Microscopy Super-Resolution

In serial block-face scanning electron microscopy (SBEM), a focused high-energy electron beam scans the sample surface, resulting in the acquisition of a 2D image in the xy-plane. The diamond knife subsequently removes the top layer of the sample to a specific thickness in the z direction, revealing the next block-face for imaging. The repetition of this process generates a series of 2D images that are stacked to form a 3D volume image, as illustrated in Fig. 1.

The observed block-face $y \in \mathbb{R}^{m\times m}$ is affected by underlying microscope degradation $\mathcal{D}_{\delta'}(\cdot) : \mathbb{R}^{M\times M} \to \mathbb{R}^{m\times m}$ parameterized by δ', $y = \mathcal{D}_{\delta'}(x)$, where $x \in \mathbb{R}^{M\times M}$ denotes the latent image that we aim to restore, and $M = \tau m$, where τ is the resolution ratio between the HR and LR images, i.e., undersampling ratio. Theoretically, the purpose of the SR process is to recover unknown x via $\mathcal{D}^{-1}_{\delta'}(y)$, which demands finding degradation inversion $\mathcal{D}^{-1}_{\delta'}(\cdot) : \mathbb{R}^{m\times m} \to \mathbb{R}^{M\times M}$. If such a mapping exists, we can obtain HR observations through LR imaging, practically accelerating imaging by a factor τ^2. The microscope degradation parameters, δ', can arise from various sources [4], [47], [48]. These sources include electronic device components such as wires and coils, which produce thermal and electromagnetic interference that is modeled as Gaussian noise. The detector's electron-counting error introduces signal-dependent noise in EM images, which is modeled as Poisson noise. Line-by-line pixel scanning in the SBEM can lead to correlated noise. Imperfect electromagnetic lenses and anodes cause blurred observations due to suboptimal focusing of the electron beam. A high-energy electron beam introduces electron charge and causes absorption-based heating. Cutting the sample with a diamond knife can introduce specific artifacts and distortions. Additionally, mechanical disturbances from the environment and microscope can introduce mechanical noise, further exacerbating image degradation.

Hence, $\mathcal{D}_{\delta'}(\cdot)$ cannot be well parameterized by simplified assumptions such as block-averaging neighbor pixels for the undersampling operator [45]. Implicit modeling of the degradation function can be realized through training a neural network by acquired pairs of LR and HR EM images.

B. Training Without Clean Reference

Training without clean references has been studied in several image restoration tasks, including denoising, magnetic resonance image reconstruction, and text removal [49], [50], [51]. Here, our focus is on investigating such a training approach for commonly used restoration loss functions, i.e., ℓ_2 and ℓ_1, and determining the corruption levels at which this training remains feasible for EM SR.

Supervised training of a network $f_\theta(\cdot)$ for SR requires numerous pairs of degraded LR, y, and clean reference, x. The network's parameters θ are obtained by optimizing the following empirical loss function:

$$\hat{\theta} = \arg\min_{\theta} \mathbb{E}_{(x,y)}[\mathcal{L}(f_\theta(y), x)] \tag{4}$$

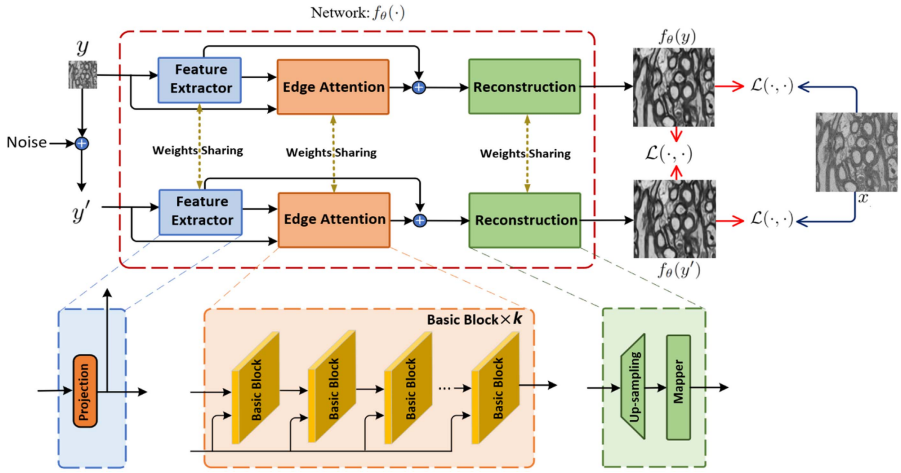

Fig. 2. Overview of the proposed image super-resolution network for training with pairs of corrupted images. The network includes the feature extractor, edge attention, and reconstruction modules, which are shared between the original noisy LR EM image y and its noisier version y'. The network is encouraged to generate two outputs, $f_\theta(y)$ and $f_\theta(y')$, that are consistent with the noisy reference image x. The output from the original image $f_\theta(y)$ serves as the reference for the noisier-noisy input, establishing a noise-robust framework via a self-supervised approach.

By applying the conditional expectation rule for dependent random variables y and x, we can reformulate (4) as follows:

$$\hat{\theta} = \arg\min_{\theta} \mathbb{E}_y \big[\underbrace{\mathbb{E}_{x|y}[\mathcal{L}(f_\theta(y), x)]}_{\text{reference-dependent}} \big] \tag{5}$$

The equation above implies that the network parameters can be optimized separately with respect to y and x over the loss function $\mathcal{L}(\cdot,\cdot)$. Let $\hat{x} = x + n$, where n is an i.i.d. additive noise with mean μ and variance $\sigma_n^2 I$, where $I \in \mathbb{R}^{d\times d}$ is an identity matrix with $d = M^2$.

When the loss function is ℓ_2, we can derive equality that links the solutions of the reference-dependent component in (5) for x and $\hat{x}$ as follows (see Appendix I.A):

$$\begin{aligned}&\mathbb{E}_{\hat{x}|y}[\|f_\theta(y) - \hat{x}\|_2^2] \\ &= \mathbb{E}_{x|y}[\|f_\theta(y) - x\|_2^2] - 2\mu^T \mathbb{E}_{x|y}[f_\theta(y) - x] + d\sigma_n^2 + ||\mu||^2\end{aligned} \tag{6}$$

The equation above states that when μ is close to zero ($\mathbb{E}[n] \approx 0$), the second term on the right-hand side of the equation becomes negligible, i.e., $2\mu^T \mathbb{E}_{x|y}[f_\theta(y) - x] \to 0$. Additionally, the third term σ_n^2, which is noise variance, and the fourth term, which is noise mean, are independent of y and have no effect on the total optimization problem. Therefore, if we substitute the clean image x with a random variable $\hat{x}$ that satisfies $\mathbb{E}[x] \approx \mathbb{E}[\hat{x}]$, the network's parameters will remain close to the optimal. This enables us to replace the clean reference x with its corrupted version $\hat{x}$, provided their expectation values are sufficiently close, which can be accompanied by the practical assumption that noise should not significantly alter the overall variability and structure of the original image, i.e., $\sigma_{\hat{x}}^2 \approx \sigma_x^2$.

In the case of ℓ_1 loss, we can establish the relationship between the solutions of the reference-dependent part in (5) for both x and $\hat{x}$ as below (see Appendix I.B):

$$\begin{aligned}&\Big|\mathbb{E}_{\hat{x}|y}[\|(f_\theta(y) - \hat{x})\|_1] - \mathbb{E}_{x|y}[\|f_\theta(y) - x\|_1]\Big| \\ &\le \frac{|-2\mu^T \mathbb{E}_{x|y}[f_\theta(y) - x] + d\sigma_n^2 + ||\mu||^2|}{g(y, x, \hat{x})},\end{aligned} \tag{7}$$

where $g(y,x,\hat{x}) = \frac{\sqrt{\mathbb{E}_{\hat{x}|y}[\|f_\theta(y)-\hat{x}\|_2^2]}+\sqrt{\mathbb{E}_{x|y}[\|f_\theta(y)-x\|_2^2]}}{\sqrt{d}}$. The inequality above suggests that the difference between the reference-dependent solutions for $\hat{x}$ and x is bounded by a function of μ and σ_n^2. When μ is small, it significantly reduces the dependence on y and tightens the upper bound, which becomes primarily dependent on y through σ_n^2. This implies that weak noise reduces the reliance on y and indicates that it will not significantly alter the overall optimization problem (5). In other words, the network's parameters will remain near optimal even if we replace the clean image x with its noisy version $\hat{x}$, as long as $\mathbb{E}[x] \approx \mathbb{E}[\hat{x}]$, and the overall structure of the clean image is not significantly altered by noise, $\sigma_{\hat{x}}^2 \approx \sigma_x^2$.

These observations indicates that the network can be trained under real-world scenarios where the reference is contaminated with weak noise-like corruptions. Here, we aim to determine the rough acceptable level of these corruptions in brain EM imaging–which was discussed in Section II.B, upon noise statistics μ and σ^2 that overshadow training without clean references. Suppose we can decompose $\hat{x}$ into clean x and noise-like corruption component n, $\hat{x} = x_{clean} + n$. We can then establish the following relationships:

$$\mathbb{E}[\hat{x}] = \mathbb{E}[x_{clean}] + \mathbb{E}[n], \tag{8a}$$

$$\sigma_{\hat{x}}^2 = \sigma_{x_{clean}}^2 + \sigma_n^2 \tag{8b}$$

When the inequalities

$$\mathbb{E}[x_{clean}] \gg \mathbb{E}[n], \tag{9a}$$

$$\sigma_{x_{clean}}^2 \gg \sigma_n^2 \tag{9b}$$

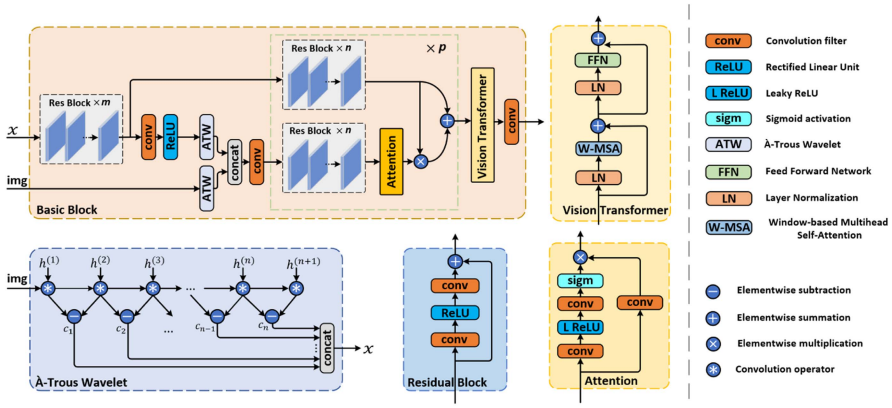

Fig. 3. Modules embedded in the proposed network: Basic Block, Residual Block, Á-Trous Wavelet, Attention Block, and Vision Transformer Block.

hold, $\mathbb{E}[x_{clean}] \approx \mathbb{E}[\hat{x}]$ and $\sigma^2_{x_{clean}} \approx \sigma^2_{\hat{x}}$, which are requirements for training using pairs of corrupted images with ℓ_1 and ℓ_2 loss functions, and guarantee that the content of the underlying image is much stronger than corruptions.

The level of corruption in EM is mostly much lower than that in image content information, satisfying (9), allowing for training network $f_\theta(\cdot)$ from pairs of corrupted images. It is worth mentioning that rare image slices may exhibit levels of corruption inconsistent with constraints stated in (9). These corruptions act as anomalies that the network is unable to learn.

C. Network Architecture

The proposed SR network, which is designed for training using pairs of corrupted LR and HR EM images, is depicted in Fig. 2. It consists of three key modules: feature extractor, edge attention, and reconstruction. These modules are shared between the given LR image and its noisier version.

1) Feature Extractor: The feature extractor ($\mathcal{H}_{FE}$) is employed to extract shallow (X_{SF}) features from the given LR image $\mathbf{y} \in \mathbb{R}^{W\times H\times C}$. It includes a projection, which is a 3×3 convolutional filter. The extraction process is as follows:

$$X_{SF} = \mathcal{H}_{FE}(y) \tag{10}$$

2) Edge Attention: The edge attention module ($\mathcal{H}_{EA}$) takes X_{SF} and y as inputs, extracts deep features and combines them with edge information using multiscale edge attention and self-attention mechanisms, yielding the generation of edge-attention features (X_{EA}). The calculation of the edge-attention module is summarized as:

$$X_{EA} = \mathcal{H}_{EA}(y, X_{SF}) \tag{11}$$

The module consists of k basic blocks, as shown in Fig. 3. In each basic block, the input features pass through m residual blocks with well-studied benefits [52]; then, two parallel paths are taken. In the upper path, the features are fed into residual blocks to produce deep features that are then enhanced using edge information. In the lower path, the features undergo

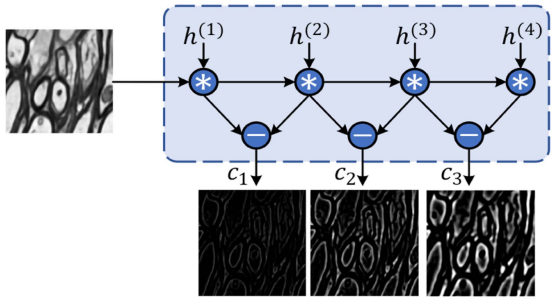

Fig. 4. Multiscale edges extracted from the EM dataset using ATW, where h represents the filter's kernel. The figure illustrates three edge components obtained at different scales, demonstrating the sparsity of edges and underscoring the importance of paying attention to edge details.

convolutional operations and rectified linear unit (ReLU) activation to reconstruct the image in the LR space. The reconstructed image, along with the original LR image, is then fed to an atrous wavelet (ATW) [53], a noise-robust feature extractor, to extract multiscale edges, see Fig. 4. The resulting multiscale edge features are then subjected to concatenation and filtering before being input into the attention block. The attention block generates multiscale attention maps specifically focused on the deep feature edges. Finally, attention maps and deep features are combined through elementwise multiplication. The resulting attention features are then added to the features from the upper path, leading to the generation of multi-scale edge-attention features. Subsequently, these features are passed into a ViT block, which employs a window-based multihead self-attention mechanism to capture both local and global image dependencies within the deep multiscale edge attention features, and finally passes through convolution layers.

Vision Transformer (ViT): ViTs divide a feature map into a sequence of small patches, forming local windows, and utilize self-attention mechanisms to understand the relationships among them. This capacity to comprehend diverse image dependencies is crucial for representation learning performance in low-level vision tasks such as SR techniques. To capture both global and local image dependencies while maintaining computational efficiency, we adopt the window-based multihead

self-attention (W-MSA) method [29]. The attention maps generated by W-MSA are then processed through the feed-forward network (FFN). These W-MSA and FFN components are integrated into a ViT block, as illustrated in Fig. 3, and their computations are outlined as follows:

$$X' = \text{W-MSA}(\text{LN}(X)) + X,$$
$$X'' = \text{FFN}(\text{LN}(X')) + X', \tag{12}$$

where, LN is the layer normalization and X is the input feature map.

In the W-MSA, the input feature map of size $C \times H \times W$ is initially divided into $N = HW/M^2$ nonoverlapping local windows of size $M \times M$, resulting in local feature maps $X \in \mathbb{R}^{M^2 \times C}$. Each of these local feature maps then undergoes the standard self-attention mechanism, with the following calculation:

$$Q = XP_Q, \quad K = XP_K, \quad V = XP_V, \tag{13}$$

where, $P_Q, P_K, P_V \in \mathbb{R}^{C \times d_k}$ represent the query (Q), key (K), and value (V) projection matrices, respectively; d_k is determined as C/k, where k denotes the number of attention heads. The attention matrix is computed using the self-attention mechanism within the k-th head of the local window:

$$\text{Attention}(Q, K, V) = \text{SoftMax}(QK^T/\sqrt{d_k})V, \tag{14}$$

The concatenation of all the attention heads results in the window-based multihead self-attention (W-MSA) output.

An FFN is a multilayer perceptron (MLP) used to introduce additional nonlinearity to the model through two fully connected layers and ReLU activation.

3) Reconstruction: Shallow features predominantly consist of low frequencies, capturing the overall structure, while the deep features encompass high frequencies corresponding to lost fine details. The long skip connection provides the reconstruction module with low frequencies and makes the training more stable. Furthermore, it helps the edge-attention module focus on learning fine details. The elementwise summation of shallow and deep features in the LR space is less expensive computationally than the alternative concatenation approach, as it maintains the original feature map dimensions, avoiding the complexity and resource demands of concatenation. This summation is fed to the reconstruction module ($\mathcal{H}_R$) to generate a super-resolved image x with enhanced resolution:

$$x = \mathcal{H}_R(X_{EA} + X_{SF}) \tag{15}$$

The reconstruction module includes an upsampling process that enlarges the features by pixel shuffling [54]. This upsampling step is followed by a mapper module that includes convolution layers, which yields the super-resolved image.

4) Weight Sharing: The aforementioned modules are shared between the given LR EM image and its noisier version, as illustrated in Fig. 2. Weight sharing encourages the network to produce consistent outputs for both the given LR image and its noisier version, establishing a noise-robust framework for training. This strategy mitigates the absence of a clean reference: The prediction generated from the given LR EM image serves as a reference for the noisier LR EM branch in a self-supervised approach.

The modules used in the proposed network architecture are summarized in Table I.

D. Loss Functions

We employ the ℓ_p-norm loss, $p \in \{1, 2\}$, as a pixelwise distance measure between the network's prediction $\hat{z}$ and ground truth z: $\mathcal{L}_{\ell_p}(z, \hat{z}) = \|z - \hat{z}\|_p^p$ [13], [55]. Our loss function measures the mismatch between the two network outputs and the reference, namely $\mathcal{L}_{\ell_p}(f_\theta(y), x)$ and $\mathcal{L}_{\ell_p}(f_\theta(y'), x)$, as well as the mismatch between two outputs, $\mathcal{L}_{\ell_p}(f_\theta(y), f_\theta(y'))$, see Fig. 2. The total loss is then defined as:

$$\mathcal{L}_T = \lambda_1 \mathcal{L}_{\ell_p}(f_\theta(y), x) + \lambda_2 \mathcal{L}_{\ell_p}(f_\theta(y'), x) + \lambda_3 \mathcal{L}_{\ell_p}(f_\theta(y), f_\theta(y')) \tag{16}$$

where λ_1, λ_2, and λ_3 are hyperparameters that govern the trade-off between components. In the loss (16), the first and second loss components are supervised, utilizing the real HR EM image x as reference. However, the third loss component is self-supervised, as the output from the original LR EM image serves as the reference for the output from the noisier LR EM image.

It should be noted that the developed theory and method are applicable to real EM datasets contaminated with real noises and corruptions without any reliance on specific noise assumptions, as we will validate through experiments using real EM datasets.

III. Experimental Settings, Results, and Discussion

A. Datasets

We conducted experiments using nine LR and HR 3D-EM datasets acquired from the corpus callosum and cingulum regions associated with the white matter of five rat brains [56]. These datasets were acquired both ipsi- and contra-laterally. For four animals, both ipsi- and contra-lateral datasets were available, while for one animal, only ipsi-lateral data was available. Both the LR and HR datasets were acquired simultaneously using the SBEM technique. The LR datasets were obtained from large tissue volumes of $200 \times 100 \times 65 \mu\text{m}^3$, with a voxel size of $50 \times 50 \times 50$ nm^3. The HR datasets were acquired from smaller tissue volumes of $15 \times 15 \times 15 \mu\text{m}^3$, which were covered by the LR FoV, with a voxel size of $15 \pm 2.5 \times 15 \pm 2.5 \times 50$ nm^3. The LR and HR 3D-EM datasets totaled approximately two hundred gigabytes in size. The pairs of LR and HR from small FoVs were utilized in the experiments. In terms of dimensions, the LR and HR 3D-EM pairs had size ranges within $330 \pm 40 \times 330 \pm 40 \times 550 \pm 150$ and $1024 \times 1024 \times 550 \pm 150$ voxels, respectively.

B. Settings

1) Training: The training datasets were augmented by adding random zero-mean white Gaussian noise with a standard deviation of $\sigma \in [0, 5]$, applying random rotation of

TABLE I
SUMMARY OF MODULES USED IN THE PROPOSED NETWORK ARCHITECTURE

Module	Description	Components	Formula
Feature extractor $\mathcal{H}_{FE}(\cdot)$	Takes LR image y and projects it into the shallow feature space X_{SF}, serving a dual purpose: transferring coarse features to the edge-attention module for deep feature space enhancement, and providing coarse features for reconstruction module.	Projection: convolutional layers; skip connection	$X_{SF} = \mathcal{H}_{FE}(y)$
Edge attention $\mathcal{H}_{EA}(\cdot,\cdot)$	Takes LR image and shallow features as inputs, utilizing multi-scale edge-attention and self-attention mechanisms to generate edge-attention features X_{EA}.	Basic blocks: residual blocks, A-Trous wavelet, attention block, vision transformer	$X_{EA} = \mathcal{H}_{EA}(y, X_{SF})$
Reconstruction $\mathcal{H}_R(\cdot)$	Takes the sum of edge-attention and shallow features. Utilizes an upsampling process to enlarge the feature map, followed by a mapper to generate the super-resolved image x.	Upsampling: pixel shuffling; mapper: convolution, ReLu	$x = \mathcal{H}_R(X_{EA} + X_{SF})$
Loss function $\mathcal{L}_T$	Measures the discrepancy between the two network outputs and the reference, i.e., $\mathcal{L}_{\ell_p}(f_\theta(y), x)$ and $\mathcal{L}_{\ell_p}(f_\theta(y'), x)$, as well as the mismatch between two outputs $\mathcal{L}_{\ell_p}(f_\theta(y), f_\theta(y'))$.	$\mathcal{L}_{\ell_p}$: ℓ_p-norm loss, $p \in \{1, 2\}$	$\mathcal{L}_T = \lambda_1 \mathcal{L}_{\ell_p}(f_\theta(y), x) + \lambda_2 \mathcal{L}_{\ell_p}(f_\theta(y'), x) + \lambda_3 \mathcal{L}_{\ell_p}(f_\theta(y), f_\theta(y'))$
Weight sharing	Encourages the network to generate consistent results for original LR EM and its noisier version.		

$\theta \in \{90°, 180°, 270°\}$, and performing horizontal/vertical flipping on the input data. As inequalities (6) and (7) with conditions (9a) and (9b) suggest, noisier references can lead to inferior SR performance. Therefore, noise was introduced solely to the LR images. This strategy aimed to enhance the diversity of training datasets, thereby improving adaptation to unseen noisier inputs. In the noisier branch, the noisier version of the input image was generated by adding random zero-mean Gaussian noise with a standard deviation of $\sigma \in [0, 5]$, to align with the weak noise typically present in EM. It should be mentioned that weight-sharing aims to encourage network invariance with respect to input images and their slightly perturbed counterparts, promoting learning capability. However, if noise is too strong, the network will more learn consistency of very significant features, ignoring learning fine details, which in turn will degrade SR performance. It is important to highlight that we added zero-mean Gaussian noise to the real LR images, which were already contaminated with real noises and corruptions, including non-Gaussian components. As a result, the final noisy images included real noise components. The network was optimized using Adam [57] for 200,000 steps. The initial learning rate was set to 10^{-4} and halved every 50,000 steps. The network was implemented using the PyTorch framework. The hyperparameters were set as follows: $\lambda_1 = 1$, $\lambda_2 = 1$, and $\lambda_3 = 1$. In the attention block, three scales of edges extracted by ATW were used. The edge attention module was configured with three basic blocks ($k = 3$). Each basic block had four residual blocks ($m = 4$), followed by two parallel sets, each with one residual block ($n = 1$). The ViT block was equipped with sixteen attention heads ($k = 16$), a patch size of four ($M = 4$), and a multilayer perceptron ratio of two. The network maintained a constant number of 64 channels ($C = 64$), and utilized a batch size of two during training.

2) Comparisons: In our comparative analysis, we assessed the performance of our method with the $\mathcal{L}_{\ell_1}/\mathcal{L}_{\ell_2}$ loss function alongside several SR techniques, including standard bicubic, DPIR [36], PSSR [44], and SwinIR [29], setting hyperparameters as in the respective papers. As a preprocessing step, we first utilized bicubic interpolation to resize both the LR and HR images to achieve the closest integer resolution ratio between them. Specifically, we resize the LR and HR images to dimensions of $341 \times 341 \times K$ and $1023 \times 1023 \times K$, leading to a resolution ratio of $\tau = 3$, where K is the number of slices. The proposed network was trained using pairs of 2D slices from 3D-EM datasets, with the LR image size of 341×341 and the HR image size of 1023×1023. Additionally, we investigated three training strategies for the proposed method: training using I) real LR and HR image pairs, II) synthetic LR and HR image pairs, and III) LR and denoised HR image pairs. For synthetic training, LR images are generated using two scenarios. First, by bicubically downsampling real HR images–a common practice in computer vision, we refer to it as Synthetic (I). Second, by introducing random isotropic Gaussian kernel ($\kappa \in [0, 3]$) and random zero-mean Gaussian noise ($\sigma \in [20, 40]$) to real HR image, followed by bicubic down-sampling and the addition of random zero-mean Gaussian noise ($\sigma \in [5, 15]$), we refer to it as Synthetic (II).

C. Quality Evaluation Metrics

To quantitatively assess the effectiveness of the proposed method and compare it with others, we have considered three image quality metrics: the structural similarity index (SSIM) [58], the peak signal-to-noise ratio (PSNR) as standard metrics, as well as the Fourier ring correlation (FRC) [59], which is utilized for evaluating EM SR [45].

1) SSIM: The SSIM quantifies the similarity between restored $\hat{x}$ and reference x images in terms of luminance, contrast, and structure. It is calculated by:

$$\text{SSIM}(x, \hat{x}) = \frac{(2\mu_x\mu_{\hat{x}} + c_1)(2\sigma_{x\hat{x}} + c_2)}{(\mu_x^2 + \mu_{\hat{x}}^2 + c_1)(\sigma_x^2 + \sigma_{\hat{x}}^2 + c_2)}, \tag{17}$$

where μ_x and $\mu_{\hat{x}}$ are the average pixel intensities of x and $\hat{x}$ (luminance). σ_x and $\sigma_{\hat{x}}$ are the standard deviations of x and $\hat{x}$ pixel intensities (contrasts), while $\sigma_{x\hat{x}}$ represents the covariance between x and $\hat{x}$ (structural similarity). c_1 and c_2 are small positive constants for division stability, typically set as 0.01 and 0.03 relative to the maximum pixel value, L.

TABLE II
QUANTITATIVE EVALUATION OF SUPER-RESOLUTION METHODS

Metric	Datasets	Bicubic	DPIR	PSSR	SwinIR	EMSR [OURS /ℓ_1]	EMSR [OURS /ℓ_2]
SSIM ↑	BRAIN1 [IPSI]	0.551 ± 0.012	$\underline{0.712} \pm \underline{0.015}$	0.702 ± 0.014	0.697 ± 0.015	$\mathbf{0.720 \pm 0.015}$	0.705 ± 0.015
	BRAIN1 [CONTRA]	0.719 ± 0.013	$\underline{0.840} \pm \underline{0.014}$	$\mathbf{0.849 \pm 0.013}$	0.832 ± 0.015	0.832 ± 0.015	0.815 ± 0.016
	BRAIN2 [IPSI]	0.669 ± 0.014	0.715 ± 0.014	0.713 ± 0.014	0.694 ± 0.014	$\mathbf{0.739 \pm 0.016}$	$\underline{0.737} \pm \underline{0.014}$
	BRAIN2 [CONTRA]	0.640 ± 0.020	0.669 ± 0.024	0.672 ± 0.026	$\mathbf{0.695 \pm 0.025}$	$\underline{0.688} \pm \underline{0.026}$	$\mathbf{0.695 \pm 0.025}$
	BRAIN3 [IPSI]	0.646 ± 0.008	0.673 ± 0.008	0.666 ± 0.010	$\underline{0.715} \pm \underline{0.009}$	$\underline{0.715} \pm \underline{0.009}$	$\mathbf{0.720 \pm 0.009}$
	BRAIN3 [CONTRA]	0.737 ± 0.026	0.753 ± 0.030	0.740 ± 0.031	$\underline{0.787} \pm \underline{0.026}$	0.780 ± 0.028	$\mathbf{0.792 \pm 0.026}$
	BRAIN4 [IPSI]	0.721 ± 0.007	0.794 ± 0.014	$\underline{0.808} \pm \underline{0.012}$	0.790 ± 0.007	$\mathbf{0.809 \pm 0.009}$	0.796 ± 0.008
	BRAIN4 [CONTRA]	0.684 ± 0.014	0.717 ± 0.016	0.728 ± 0.018	$\mathbf{0.745 \pm 0.018}$	0.735 ± 0.020	$\underline{0.738} \pm \underline{0.018}$
	BRAIN5 [IPSI]	0.615 ± 0.019	0.639 ± 0.016	0.662 ± 0.016	0.669 ± 0.018	$\mathbf{0.687 \pm 0.018}$	$\underline{0.681} \pm \underline{0.017}$
	ALL DATASETS	0.665 ± 0.059	0.724 ± 0.064	0.727 ± 0.065	0.736 ± 0.056	$\mathbf{0.745 \pm 0.051}$	$\underline{0.742} \pm \underline{0.048}$
PSNR ↑	BRAIN1 [IPSI]	23.1 ± 0.3	24.8 ± 0.4	$\mathbf{25.0 \pm 0.4}$	24.4 ± 0.4	$\underline{24.9} \pm \underline{0.4}$	$\underline{24.9} \pm \underline{0.4}$
	BRAIN1 [CONTRA]	24.8 ± 2.6	$\underline{25.9} \pm \underline{2.9}$	$\mathbf{26.1 \pm 3.2}$	24.4 ± 2.9	24.2 ± 2.7	23.8 ± 2.6
	BRAIN2 [IPSI]	23.1 ± 1.5	$\mathbf{23.6 \pm 1.7}$	$\underline{23.6} \pm \underline{2.0}$	22.2 ± 1.2	22.9 ± 1.6	23.0 ± 1.6
	BRAIN2 [CONTRA]	22.8 ± 0.9	$\underline{23.3} \pm \underline{1.0}$	$\mathbf{23.3 \pm 0.9}$	22.5 ± 1.6	22.3 ± 1.5	22.3 ± 1.5
	BRAIN3 [IPSI]	21.9 ± 0.7	22.2 ± 0.8	21.8 ± 0.8	$\mathbf{23.0 \pm 0.8}$	22.7 ± 0.7	$\underline{22.8} \pm \underline{0.7}$
	BRAIN3 [CONTRA]	21.8 ± 1.3	22.1 ± 1.3	21.5 ± 1.3	$\underline{23.0} \pm \underline{1.4}$	22.7 ± 1.5	$\mathbf{23.1 \pm 1.4}$
	BRAIN4 [IPSI]	24.3 ± 0.4	$\underline{25.2} \pm \underline{0.5}$	$\underline{25.2} \pm \underline{0.5}$	$\mathbf{25.3 \pm 0.4}$	24.2 ± 0.6	23.3 ± 0.5
	BRAIN4 [CONTRA]	24.0 ± 0.7	24.4 ± 0.8	$\mathbf{24.6 \pm 0.8}$	22.1 ± 0.9	24.0 ± 0.7	$\underline{24.4} \pm \underline{0.6}$
	BRAIN5 [IPSI]	22.6 ± 0.6	$\underline{22.7} \pm \underline{0.7}$	$\mathbf{23.2 \pm 0.7}$	22.5 ± 0.7	21.6 ± 1.2	21.5 ± 1.2
	ALL DATASETS	23.2 ± 1.0	$\mathbf{23.8 \pm 1.4}$	$\underline{23.8} \pm \underline{1.6}$	23.3 ± 1.1	23.3 ± 1.1	23.2 ± 1.0
$\overline{\text{FRC}}$ ↑	BRAIN1 [IPSI]	0.191 ± 0.004	0.216 ± 0.008	0.227 ± 0.007	0.243 ± 0.010	$\mathbf{0.253 \pm 0.011}$	$\underline{0.246} \pm \underline{0.010}$
	BRAIN1 [CONTRA]	0.198 ± 0.008	0.210 ± 0.008	0.244 ± 0.007	$\underline{0.281} \pm \underline{0.014}$	$\mathbf{0.285 \pm 0.013}$	0.280 ± 0.013
	BRAIN2 [IPSI]	0.226 ± 0.007	0.287 ± 0.012	0.287 ± 0.012	0.310 ± 0.013	$\mathbf{0.319 \pm 0.015}$	$\underline{0.317} \pm \underline{0.013}$
	BRAIN2 [CONTRA]	0.213 ± 0.009	0.220 ± 0.010	0.282 ± 0.013	$\mathbf{0.314 \pm 0.016}$	$\underline{0.307} \pm \underline{0.014}$	0.304 ± 0.014
	BRAIN3 [IPSI]	0.245 ± 0.004	0.246 ± 0.006	0.309 ± 0.006	0.345 ± 0.008	$\mathbf{0.352 \pm 0.008}$	$\underline{0.349} \pm \underline{0.008}$
	BRAIN3 [CONTRA]	0.259 ± 0.007	0.263 ± 0.009	0.319 ± 0.010	$\underline{0.350} \pm \underline{0.016}$	0.349 ± 0.015	$\mathbf{0.352 \pm 0.015}$
	BRAIN4 [IPSI]	0.216 ± 0.004	0.220 ± 0.008	0.273 ± 0.009	0.292 ± 0.010	$\mathbf{0.297 \pm 0.009}$	$\underline{0.297} \pm \underline{0.010}$
	BRAIN4 [CONTRA]	0.261 ± 0.005	0.256 ± 0.008	$\underline{0.342} \pm \underline{0.010}$	$\mathbf{0.349 \pm 0.010}$	0.340 ± 0.012	0.337 ± 0.011
	BRAIN5 [IPSI]	0.227 ± 0.007	0.222 ± 0.008	0.290 ± 0.010	0.311 ± 0.010	$\mathbf{0.323 \pm 0.011}$	$\underline{0.320} \pm \underline{0.011}$
	ALL DATASETS	0.226 ± 0.025	0.238 ± 0.026	0.286 ± 0.036	0.311 ± 0.035	$\mathbf{0.314 \pm 0.032}$	$\underline{0.311} \pm \underline{0.034}$

The best and second-best scores are in bold and underlined, respectively. Reported values for the mean and standard deviation ($\mu \pm \sigma$) for each BRAIN were calculated across all slices. The overall evaluation, highlighted in gray, represents the mean and standard deviation across reported mean values for all datasets.

2) PSNR: The PSNR measures the ratio of the maximum pixel value to the mean square error (MSE) between the reconstructed image $\hat{x}$ and the ground truth x as follows:

$$\text{PSNR}(x, \hat{x}) = 10 \log_{10} \left(\frac{L^2}{\text{MSE}(x, \hat{x})} \right) \tag{18}$$

3) FRC: The FRC measures the correlation between reconstructed image $\hat{x}$ and reference x in the frequency domain when spectra $\mathcal{R}$ is subdivided into N concentric rings r_i, i.e., $\mathcal{R} = \{r_i\}_{i=1}^{N}$. The FRC is calculated using the following formula:

$$\text{FRC}(\mathcal{R}) = \frac{\sum_{r_i \in \mathcal{R}} \mathcal{F}_x(r_i) \overline{\mathcal{F}_{\hat{x}}}(r_i)}{\sqrt{\left(\sum_{r_i \in \mathcal{R}} |\mathcal{F}_x(r_i)|^2\right) \left(\sum_{r_i \in \mathcal{R}} |\mathcal{F}_{\hat{x}}(r_i)|^2\right)}}, \tag{19}$$

where $\mathcal{F}_x(r_i)$ and $\mathcal{F}_{\hat{x}}(r_i)$ are Fourier transformations of x and $\hat{x}$ over ring r_i, respectively, and $\text{FRC}(\mathcal{R})$ provides spectral correlation as a function of spatial frequency. The average correlation across the spectra is denoted by $\overline{\text{FRC}}$.

In the numerical evaluations, the denoised HR 3D-EM images, obtained using the denoising method proposed in [60], were utilized as the ground truth references.

D. Results

1) Method Comparisons: The comparative results were obtained through a fivefold cross-validation process, where data from one animal served as test sets, and data from other animals were used as training sets. The quantitative results are summarized in Table II. The reported average SSIM values reveal inferior performance for the Bicubic method (0.665) compared to DL-based methods–DPIR (0.724), PSSR (0.727), SwinIR (0.736), EMSR[ℓ_1] (0.745), and EMSR[ℓ_2] (0.742). Our approach, EMSR, employing the ℓ_1 and ℓ_2 loss functions, yielded the highest and second-highest scores, respectively. Similarly, with those of the competitors, the average reported FRC values demonstrate the superior performance of the EMSRs in terms of the spectral correlation between the restored and ground truth images. The EMSR achieved the highest FRC score of 0.314 with the ℓ_1 loss function and the second-highest score of 0.311 with the ℓ_2 loss function. The FRC scores for the other methods were: Bicubic (0.226), DPIR (0.238), PSSR (0.286), and SwinIR (0.311). In terms of the PSNR, DPIR achieved the highest score of 23.8; although the PSSR matched this score with an average PSNR of 23.8, it was ranked second due to its higher standard deviation. The PSNR scores for the other methods were: Bicubic (23.2), SwinIR (23.3), EMSR[ℓ_1] (23.3), and EMSR[ℓ_2] (23.2). It is crucial to emphasize that the effectiveness of the PSNR as an evaluation metric for SR model performance is limited. This limitation arises from its

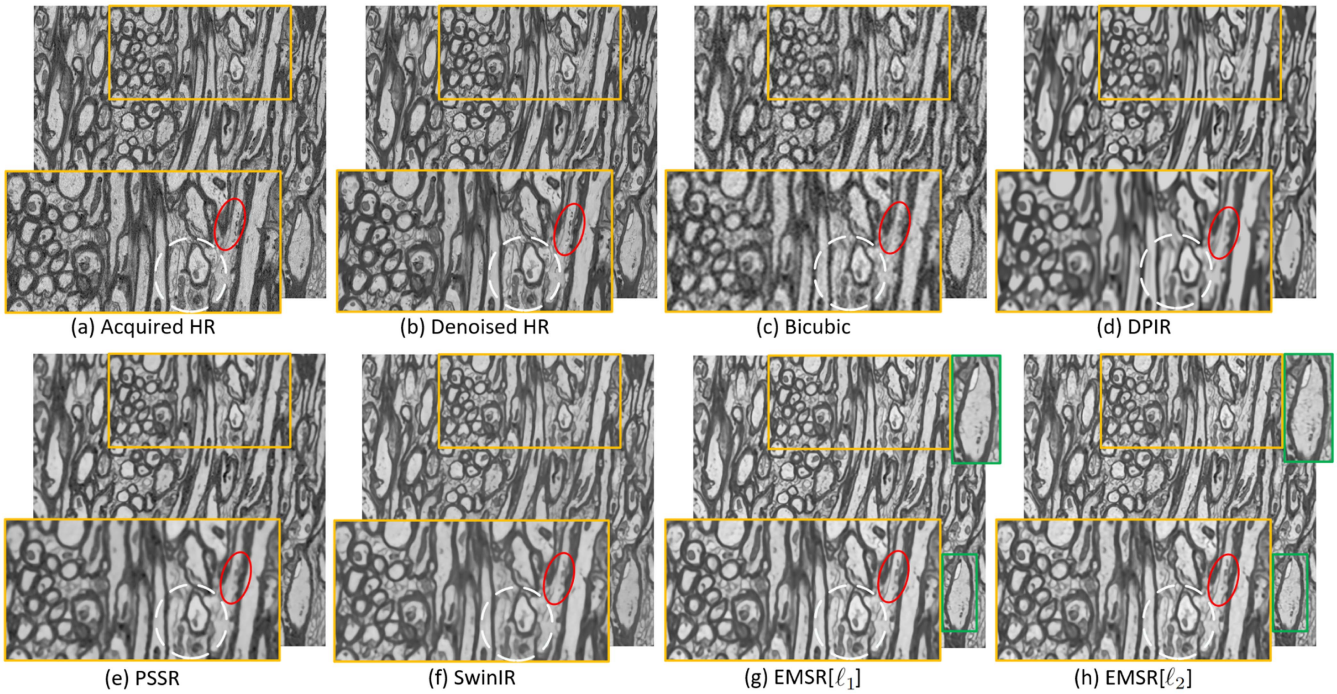

Fig. 5. Visual comparisons of super-resolution methods for BRAIN5[IPSI] are presented, and magnified regions are provided to aid comparison. The dashed circle highlights the superior edge recovery achieved by EMSR[ℓ_1] and EMSR[ℓ_2] compared to other methods, which tend to produce blurred results. The red ellipse indicates the specific area where EMSR[ℓ_1] and EMSR[ℓ_2] successfully restored high-frequency edges that remained unresolved by other techniques. The zoomed-in area marked in green illustrates that EMSR with the ℓ_1 loss exhibits slightly more noise suppression compared to EMSR with the ℓ_2 loss.

pure reliance on pixel values and its inability to capture a direct structural correlation between super-resolved and ground truth images.

To conserve space, we present a curated selection of representative results in Figs. 5 and 7. These figures provide visual insights into scenarios where our proposed method performed the best and where it did not attain the highest quantitative performance.

Fig. 5 shows the BRAIN5[IPSI] results. In this subdataset, our proposed method demonstrated outstanding performance, achieving the best and second-best quantitative results, based on SSIM and FRC, when utilizing ℓ_1 and ℓ_2 loss functions, respectively. Panel (a) shows the bicubically interpolated LR image, which exhibits a lack of visual clarity and maintains noise. Conversely, DL-based SR methods effectively reduce noise, as shown in Fig. 5(d)–(h). Among these methods, DPIR, i.e., the PnP method, produced overly smooth results, particularly when fine details were restored, as shown within regions enclosed by the ellipsoid and dashed circle. The SSIM and FRC scores, which were 0.639 and 0.222, respectively, substantiate the limitation of DPIR in capturing fine details compared to other DL-based methods, such as SwinIR, which achieved an SSIM of 0.669 and an FRC of 0.311. The weakness in recovering details can be attributed to mismatches between priors in the pretrained model and EM images. In contrast, PSSR, SwinIR, and EMSR, which were trained using EM images, exhibited the ability to restore intricate details and nuances characteristic of EM brain images. The PSSR sometimes failed to restore particular intricate edges, as represented by the area confined by an ellipsoid. It also led to smear-out edges, as indicated within the dashed circle. Similarly, SwinIR faced challenges in recovering certain edges, akin to PSSR, showed within the region confined by the ellipsoid. It also introduced blurred output and fuzzy edges within an area marked by the dashed circle. On the other hand, the EMSR with both ℓ_1 and ℓ_2 loss functions successfully super-resolved the LR images by restoring intricate edges with higher contrast while avoiding blurriness. This is quantitatively evident from its superior SSIM and FRC values: EMSR[ℓ_1] had an SSIM of 0.687 and FRC of 0.323, whereas EMSR[ℓ_2] achieved an SSIM of 0.681 and FRC of 0.320. These scores surpass those of PSSR (SSIM of 0.662, FRC of 0.290) and SwinIR (SSIM of 0.669, FRC of 0.311). In the comparison of the EMSR results using the ℓ_1 and ℓ_2 loss functions, ℓ_1 exhibited slightly superior noise suppression(see zoomed-in rectangle marked in green). These results align with the theory that ℓ_1 loss, in contrast to ℓ_2, does not overpenalize large errors, resulting in fewer noise artifacts. The condition checking for training without a clean reference is depicted in Fig. 6.

Fig. 7 shows the BRAIN2[CONTRA] results. In this subset, SwinIR demonstrated superior performance with SSIM and FRC scores of 0.695 and 0.314, respectively, indicating its enhanced structural capabilities. EMSR[ℓ_1] and EMSR[ℓ_2] were the nearest contenders, achieving SSIM scores of 0.688 and 0.695, and FRC scores of 0.307 and 0.304, respectively. While SwinIR did not achieve the highest PSNR, it maintained a satisfactory level of pixel fidelity. Panels (c) and (d) show that bicubic and DPIR generally produced oversmooth details, as denoted by the yellow arrow. Panel (e) revealed that PSSR excelled in enhancing

TABLE III
QUANTITATIVE EVALUATION OF EMSR[ℓ_1] USING DIFFERENT TRAINING STRATEGIES: PAIRS OF REAL LR AND HR, PAIRS OF SYNTHETIC LR AND HR, AND PAIRS OF REAL LR AND DENOISED HR

Metric	Method	BRAIN1		BRAIN2		BRAIN3		BRAIN4		BRAIN5	ALL DATASETS
		IPSI	CONTRA	IPSI	CONTRA	IPSI	CONTRA	IPSI	CONTRA	IPSI	
SSIM ↑	EMSR[Real]	*0.720 ± 0.015*	*0.832 ± 0.015*	*0.739 ± 0.016*	*0.688 ± 0.026*	*0.715 ± 0.009*	*0.780 ± 0.028*	**0.809 ± 0.009**	*0.735 ± 0.020*	**0.687 ± 0.018**	**0.745 ± 0.051**
	EMSR[Denoised]	**0.729 ± 0.016**	**0.839 ± 0.015**	**0.740 ± 0.017**	0.685 ± 0.026	0.712 ± 0.009	0.772 ± 0.028	*0.808 ± 0.009*	0.734 ± 0.020	*0.686 ± 0.018*	*0.745 ± 0.053*
	EMSR[Synthetic (II)]	0.598 ± 0.013	0.780 ± 0.014	0.738 ± 0.014	**0.704 ± 0.021**	**0.724 ± 0.009**	**0.794 ± 0.025**	0.776 ± 0.006	**0.755 ± 0.018**	0.667 ± 0.021	0.726 ± 0.063
	EMSR[Synthetic (I)]	0.519 ± 0.012	0.705 ± 0.015	0.675 ± 0.013	0.645 ± 0.020	0.663 ± 0.008	0.749 ± 0.025	0.718 ± 0.008	0.708 ± 0.015	0.625 ± 0.020	0.667 ± 0.068
PSNR ↑	EMSR[Real]	*24.9 ± 0.4*	24.2 ± 2.7	22.9 ± 1.6	22.3 ± 1.5	*22.7 ± 0.7*	**22.7 ± 1.5**	24.2 ± 0.6	24.0 ± 0.7	21.6 ± 1.2	*23.3 ± 1.1*
	EMSR[Denoised]	**25.0 ± 0.4**	24.5 ± 2.8	*23.4 ± 1.6*	22.6 ± 1.5	*22.7 ± 0.7*	22.1 ± 1.5	22.9 ± 0.7	23.6 ± 0.7	21.5 ± 1.2	23.1 ± 1.1
	EMSR[Synthetic (II)]	23.3 ± 0.3	**25.0 ± 2.6**	**24.0 ± 1.9**	**23.8 ± 1.0**	**22.9 ± 0.9**	*22.4 ± 1.4*	**24.7 ± 0.6**	**24.9 ± 0.9**	**22.8 ± 0.7**	**23.8 ± 1.0**
	EMSR[Synthetic (I)]	22.6 ± 0.3	*24.7 ± 2.5*	23.1 ± 1.6	*22.9 ± 0.9*	22.0 ± 0.8	21.8 ± 1.3	*24.2 ± 0.4*	*24.3 ± 0.7*	*22.7 ± 0.6*	23.1 ± 1.0
$\overline{\text{FRC}}$ ↑	EMSR[Real]	*0.253 ± 0.011*	*0.285 ± 0.013*	*0.319 ± 0.015*	*0.307 ± 0.014*	*0.352 ± 0.008*	*0.349 ± 0.015*	**0.297 ± 0.010**	*0.340 ± 0.012*	**0.323 ± 0.011**	*0.314 ± 0.032*
	EMSR[Denoised]	**0.256 ± 0.011**	**0.288 ± 0.013**	**0.321 ± 0.015**	*0.307 ± 0.014*	0.349 ± 0.009	0.347 ± 0.016	*0.294 ± 0.010*	*0.340 ± 0.012*	*0.320 ± 0.012*	**0.314 ± 0.031**
	EMSR[Synthetic (II)]	0.214 ± 0.008	0.255 ± 0.015	0.311 ± 0.015	**0.311 ± 0.015**	**0.367 ± 0.010**	**0.362 ± 0.018**	0.289 ± 0.012	**0.369 ± 0.013**	0.308 ± 0.015	0.310 ± 0.053
	EMSR[Synthetic (I)]	0.193 ± 0.005	0.205 ± 0.009	0.245 ± 0.010	0.235 ± 0.013	0.285 ± 0.006	0.294 ± 0.011	0.230 ± 0.006	0.313 ± 0.009	0.249 ± 0.010	0.250 ± 0.040

The best and second-best scores are in bold and underlined, respectively. Reported values for mean and standard deviation ($\mu \pm \sigma$) for each brain were calculated across all slices.

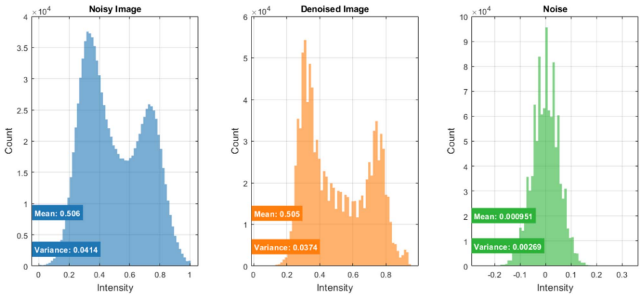

Fig. 6. Conditions for network training using pairs of corrupted images. The histograms of one slice from the BRAIN2[contra] dataset, its denoised version, and noise are illustrated. The mean and variance presented on each plot were examined to investigate the conditions in (9). I) $\mathbb{E}[x_n] = 5.06 \times 10^{-1}$, $\mathbb{E}[x_{clean}] = 5.05 \times 10^{-1}$, and $\mathbb{E}[n] = 9.51 \times 10^{-4}$, satisfying (9a) that mentions $\mathbb{E}[x_{clean}] \gg \mathbb{E}[n]$. II) $\sigma^2_{x_n} = 4.14 \times 10^{-2}, \sigma^2_{x_{clean}} = 3.74 \times 10^{-2}$, and $\sigma^2_n = 2.69 \times 10^{-3}$, meeting the condition in (9b) that $\sigma^2_{x_{clean}} \gg \sigma^2_n$. It should be noted that we considered the denoised reference, obtained from the denoising method in [60], as a clean reference.

details and contrast but faced challenges in recovering fine edges, as indicated by the yellow arrow. SwinIR and EMSR (f)–(h) showed superior resolution enhancement and noise reduction. In particular, SwinIR delivered slightly sharper SR results, highlighted by the yellow arrow, which is in agreement with the compared SSIM and FRC scores. However, the proposed method demonstrated a superior ability to super-resolve two closely situated compartments compared to SwinIR, which struggled to effectively separate them, as indicated by the green arrows; this is likely due to its edge-attention mechanism of the proposed method.

2) Training Strategies: The outcomes of training with different strategies–real pairs featuring corrupted references, real pairs with a denoised reference, and synthetic LR and HR pairs (both Synthetic (I) and (II))–are detailed in Table III. The average quantitative results across all datasets revealed that training with acquired HR images and their denoised versions as references resulted in nearly identical SSIM and FRC scores of 0.745 and 0.314, respectively. However, training with denoised references led to a marginally lower PSNR of 23.1, compared to 23.3 achieved with the original HR images. Additionally, it was noted that training with Synthetic (I) did not attain favorable SR results, with significantly lower scores: SSIM of 0.667, FRC of 0.250, and PSNR of 23.1. In contrast, Synthetic (II) exhibited promising outcomes: it achieved an average SSIM of 0.726 and FRC of 0.310, which, although slightly lower than those achieved with real pairs, resulted in a higher PSNR of 23.8.

Representative results are depicted in Fig. 8, spanning from inferior to superior performance. The results for BRAIN1[IPSI] indicate that the trained network with both the Synthetic (I) and (II) strategies failed to produce satisfactory SR results, as evident from different artifacts. The quantitative results for Synthetic (I) and (II) further demonstrated notably poor performance, with SSIM scores of 0.519 and 0.598 and FRC scores of 0.193 and 0.214, respectively. In contrast, training with real data yielded significantly better results. Training with the original pairs achieved an SSIM of 0.720 and an FRC of 0.253, while the use of denoised references resulted in an SSIM of 0.729 and an FRC of 0.256. These outcomes confirm the shortcomings of training with synthetic data in effectively producing high-quality super-resolved images with fine details. The reason for these shortcomings lies in the inability of the combination of bicubic downsampling and a pool of random Gaussian noise and blurring kernels to effectively match the degradation in the input LR image. Furthermore, training using real pairs demonstrates that training using real pairs with either corrupted or denoised references yielded nearly similar outputs, with only subtle differences, such as slightly more homogeneous areas in the case of training with denoised reference (white arrows).

The results for BRAIN2[CONTRA] indicate encouraging findings. The synthetic (I) training strategy yielded unsatisfactory results as it struggled to match the degradations present in the input LR image, see Fig. 9. However, the Synthetic (II) training strategy, which incorporates a diverse range of degradations, produced superior results compared to training with real pairs. This approach generated sharper edges and enhanced contrast, as highlighted by the dashed green circle. The quantitative results corroborate this improvement, with an SSIM of 0.704, FRC of 0.311, and PSNR of 23.8. These scores are better than those from training with real data, which achieved an SSIM of 0.688, FRC of 0.307, and PSNR of 22.3. The key factor behind these results is the ability of synthetic training, under well-matched degradations, to learn deblurring and denoising while super-resolving the input LR image. The low-level feature fidelity in the synthetic pairs is well-preserved compared to training with acquired LR and HR images, even in the case of Synthetic (I) with bicubic downsampling, evident in black areas marked with asterisks.

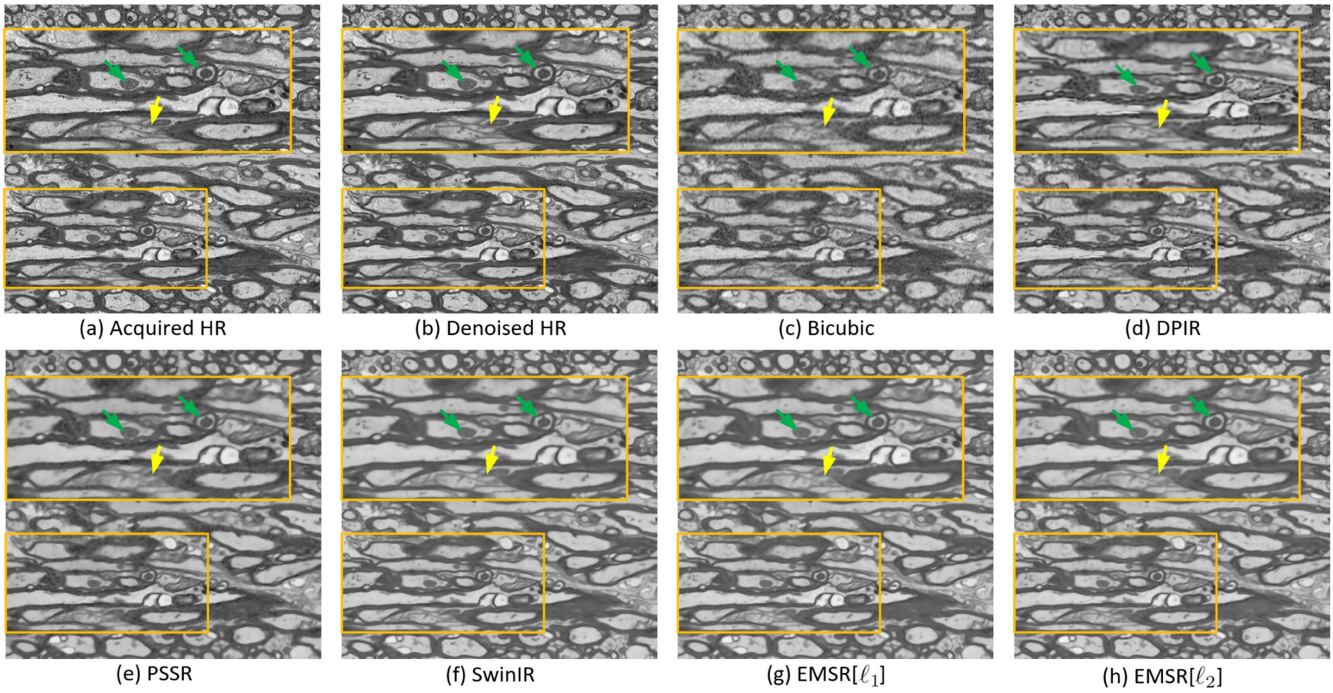

Fig. 7. Visual comparisons of super-resolution methods for BRAIN2[CONTRA] are presented, with magnified regions provided to facilitate comparison. Green arrows indicate that EMSR[ℓ_1] and EMSR[ℓ_2] could successfully super-resolve two closed compartments compared to other methods. The yellow arrow indicates the area where SwinIR outperforms the other methods by producing sharper edges.

This fidelity is reflected in the higher PSNR scores of 22.9 for Synthetic (I) and 23.8 for Synthetic (II), which are higher than that of real pairs with a PSNR of 22.3. From a denoising perspective, training with real pairs may offer better performance, benefiting from the independence of noise-like corruptions in independently acquired LR and HR images, preventing the learning of noise-like patterns with random characteristics. Notably, both noisy and denoised reference training produced similar outputs.

BRAIN3[IPSI] shows additional promising outcomes with the Synthetic (II) strategy, demonstrating superior SR performance in recovering fine details and achieving sharp edges while mitigating noise, as highlighted in areas marked by circles and arrows. In line with the visual observations, the Synthetic (II) training strategy outperformed training with real data across all the metrics, achieving an SSIM of 0.724, FRC of 0.367, and PSNR of 22.9, compared to the real data, which achieved an SSIM of 0.715, FRC of 0.352, and PSNR of 22.7.

When comparing real and synthetic datasets, it is recommended to use real image pairs because they have the potential to enhance the overall quality. The foremost advantage is learning real degradations, which are difficult to simulate; see Fig. 9. Importantly, the separate acquisition of LR and HR images leads to nearly independent noise-like corruption. This independence is beneficial for the network because it prevents the learning of noise-like patterns with random characteristics, learning to denoise while super-resolving LR images. Furthermore, the results indicate that while pairs of acquired synthetic LR, derived from downsampled HR, and HR images are not suitable as training pairs, there is a potential for computationally generated pairs to advance EM super-resolution. Notably, this approach can address mismatches between acquired LR and HR pairs, i.e., coregistration and contrast, reduce imaging time, and lower costs.

3) Super-Resolver as an Enhancer: Applying the trained SR model to HR images with the same resolution enhances image quality. In comparison to a denoiser, it enhances the resolution as well as mitigating noise; see the first row in Fig. 10. However, in situations where there are mismatches between the trained model and the input image, changes in image contrasts may occur, as depicted in the second row of Fig. 10. This observation highlights the potential of SR methods to function as denoisers and enhancers, particularly emphasizing the practical capabilities of a self-supervised SR approach that can address mismatches.

4) Super-Resolution Can Help Distortion Avoidance: EM imaging at HR may result in distortions at the image border in the xy-plane, a phenomenon not observed in LR imaging, as depicted in Fig. 11. However, employing SR techniques enables the generation of an HR image from an LR image, effectively overcoming these distortions.

5) Natural Image Pretrained Networks on Brain EM: Fig. 12 depicts the application of state-of-the-art pretrained networks designed for natural images on brain EM. BSRGAN [25] and Real ESRGAN [27] are two networks designed for the super-resolution of natural images, and were trained on natural and pure synthetic datasets, respectively. When applied to brain EM images, while these methods could restore the overall structure of large tissue compartments, they failed to recover the intricate details and nuances unique to brain EM. In particular, they tended to introduce unrealistic details and cartoonish textures, as visible in the zoomed-in areas.

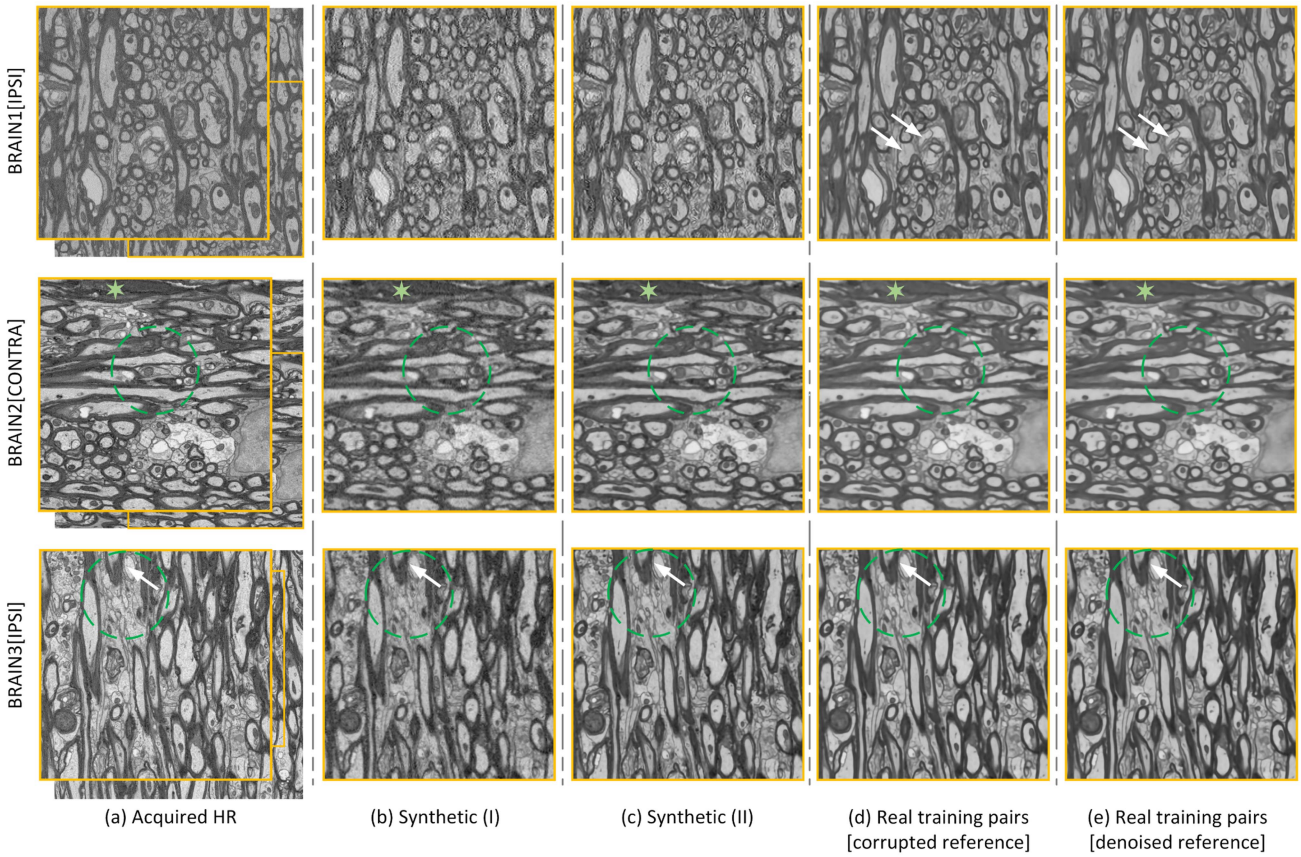

Fig. 8. Illustrative visual comparisons of EMSR[ℓ_1] with different training strategies. Panel (a) displays the acquired HR, while Panels (b) and (c) show Synthetic (I) and Synthetic (II), respectively. Training with real pairs is depicted in both (d) for real noise-like corrupted reference and (e) for denoised reference. The first row displays the results for BRAIN1[IPSI], where Synthetic (I) and (II) yielded unsatisfactory super-resolved images compared to training with real pairs, both corrupted and denoised. The white arrows in this row indicate that training with denoised references led to more homogeneous areas than training with acquired HR references. The second and third rows present the results for BRAIN2[CONTRA] and BRAIN3[IPSI]. While Synthetic (I) failed to produce high-quality super-resolved images, Synthetic (II) delivered superb super-resolution performance. The dashed green circles highlight its ability to generate sharper edges and better contrast than training with real pairs. Asterisks underscore the potential of Synthetic (I) and (II) in maintaining intensity fidelity compared to training with real pairs, while white arrows emphasize their effectiveness in recovering sharp edges and mitigating blurring observed in results from training with real pairs.

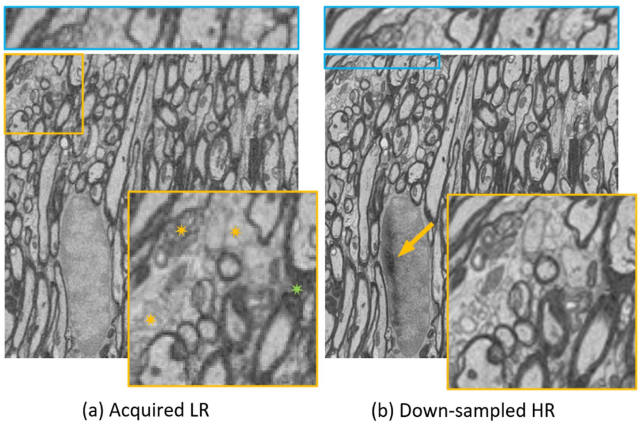

Fig. 9. A comparison between a physically acquired LR EM image and a synthetically generated LR image obtained by downsampling an HR EM image. Notable differences in the fine details and intensity can be observed. The blue rectangle highlights drift distortion at the border of the HR image, which is not present in the LR counterpart. An arrow indicates a charging effect that is only observed in the HR image. The zoomed-in area accentuates the distinct differences in fine details (yellow asterisks), and intensity level (green asterisk).

TABLE IV
NETWORK PARAMETERS AND RUNNING TIME

Method	Bicubic	DPIR	PSSR	SwinIR	EMSR
#Parameters (million)	-	32.7	32.0	11.8	3.2
Time (seconds)	0.017	9.05	0.412	0.309	0.255

E. Method Limitations

While the EMSR method can be used for SR across diverse imaging modalities, it lacks specific adaptations for handling LR and HR image misalignment. Any misregistration present between LR and HR pairs can degrade performance as it propagates through the network. The development of robust networks or loss functions that are invariant to misregistration could offer a solution.

F. Computational Efficiency

A comparison between the number of parameters and running times is presented in Table IV. Bicubic, an interpolation method,

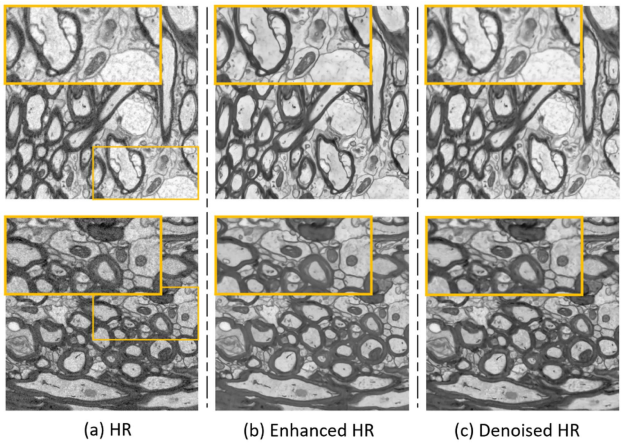

Fig. 10. The trained super-resolution model was applied to HR images with the same resolution as the HR images used for training. (a) Input HR, (b) enhanced using a super-resolver, and (c) denoised HR, which was obtained through the method in [60].

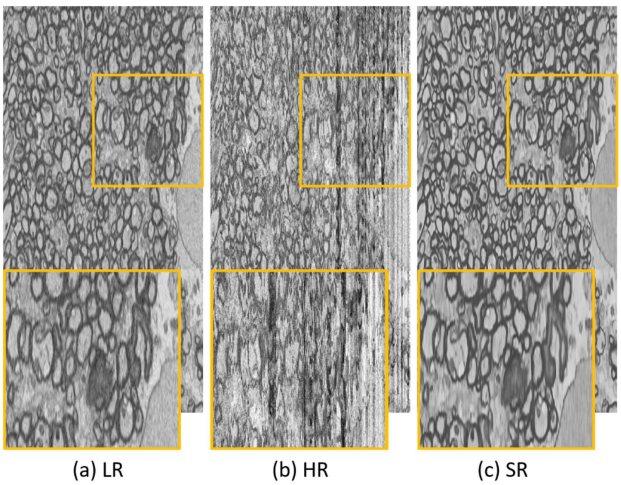

Fig. 11. Distortion in the border of 3D-EM data. The xz-perspective view of (a) bicubically interpolated acquired LR, (b) acquired HR, and (c) super-resolved LR images.

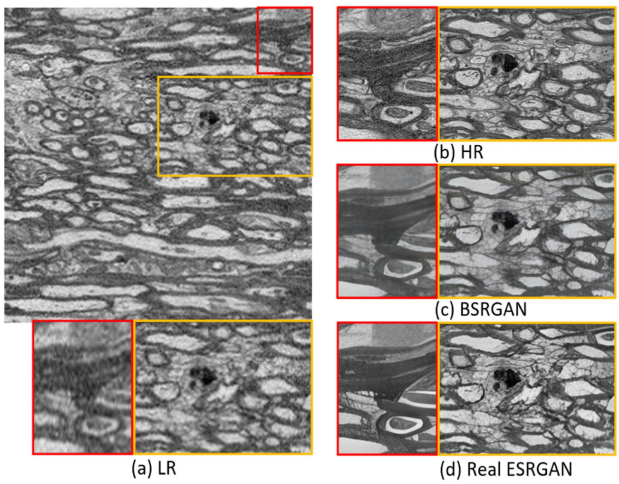

Fig. 12. Super-resolution of EM images using state-of-the-art pretrained networks designed for natural images. (a) LR, (b) HR, (c) BSRGAN, and (d) Real ESRGAN.

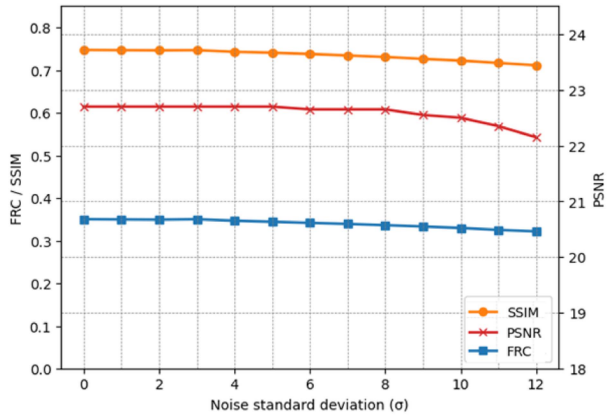

Fig. 13. Robustness of the EMSR[ℓ_1] against noise: The performance of the proposed super-resolution method when both the original noisy LR EM ($\sigma = 0$) and its noisier version ($\sigma > 0$), generated by contaminating the original noisy LR EM with zero-mean Gaussian noise of varying standard deviations, were input to the trained model. Average results, obtained using a training strategy with BRAIN3 datasets as test sets and BRAIN1, BRAIN2, BRAIN4, and BRAIN5 as training sets, are shown.

had no trainable parameters. Among DL-based methods, DPIR and PSSR had 32.7 and 32 million parameters, respectively. For DPIR, this refers to the parameters within the pretrained denoiser in its PnP framework. SwinIR provided a lighter architecture with 11.8 million parameters, while EMSR offered an even lighter architecture with 3.2 million parameters. In terms of computation time, Bicubic required 0.017 seconds to produce a super-resolved image of size 1023×1023. Among DL-based methods, DPIR was notably more time-consuming. As a PnP method that incorporates a Gaussian denoiser within the model-based framework, DPIR required 7.9 seconds to generate a super-resolved image of the same size. All other DL-based methods, i.e., PSSR (0.412 seconds), SwinIR (0.309 seconds), and EMSR (0.255 seconds), had substantially faster running times. Among these methods, the EMSR method was the fastest.

G. Ablation Studies

Robustness to Noise: The SR performance was evaluated on both the original noisy LR EM images and their noisier versions, generated by adding zero-mean Gaussian noise with a standard deviation of $\sigma \in \{1, 2, \ldots, 12\}$ to the original LR EM data. The results are presented in Fig. 13. The overall pattern of declining performance was observed. Here, $\sigma = 0$ represents the original noisy LR EM image. As progressively its noisier versions ($\sigma > 0$) were input into the trained SR model, the stability of the performance metrics was noticeable when $\sigma < 5$. While a decline in performance appeared thereafter, a degree of stability can still be observed, particularly when $\sigma < 8$. This observation confirms that the trained model can consistently generate similar outputs even when inputting noisier LR EM image versions.

Performance with Varying Noisy References: The performance of EMSR[ℓ_1] was assessed when trained on both denoised ($\sigma = 0$) and noisy references ($\sigma > 0$), generated by introducing zero-mean Gaussian noise at various standard deviations

TABLE V
THE PERFORMANCE OF EMSR[ℓ_1] WITH DIFFERENT HYPERPARAMETERS: PATCH SIZES IN VISION TRANSFORMER, NUMBER OF CHANNELS, AND NUMBER OF BASIC BLOCKS WITHIN EDGE-ATTENTION MODULE

Metric	Patch Size			# Channels				# Basic Blocks			
	2×2	4×4	6×6	16	48	64	80	1	2	3	4
SSIM	0.746	0.748	0.748	0.732	0.742	0.748	0.750	0.734	0.746	0.748	0.748
PSNR	22.6	22.7	22.7	22.6	22.6	22.7	23.2	22.3	22.9	22.7	23.3
FRC	0.349	0.351	0.350	0.319	0.344	0.351	0.354	0.334	0.346	0.351	0.352

Evaluated using a training strategy with BRAIN3 datasets as test sets, and BRAIN1, BRAIN2, BRAIN4, and BRAIN5 as training sets.

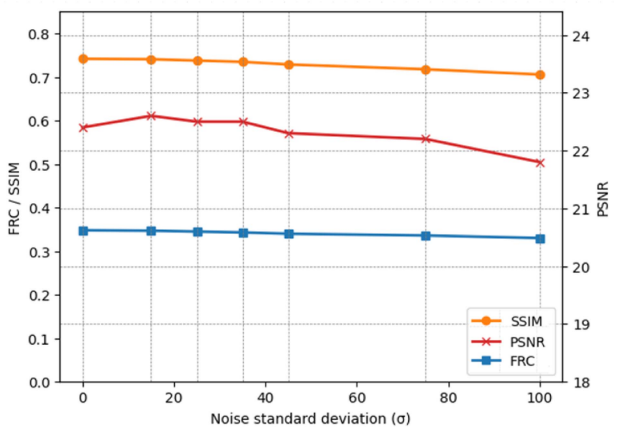

Fig. 14. Training EMSR[ℓ_1] with varying noisy references: This figure illustrates the performance of the EMSR when trained using both denoised ($\sigma = 0$) and noisy HR EM images as references. The noisy references were generated by introducing zero-mean Gaussian noise at various standard deviations ($\sigma > 0$) to denoised HR EM images. Average performance results, obtained using a training strategy with BRAIN3 datasets as test sets and BRAIN1, BRAIN2, BRAIN4, and BRAIN5 as training sets, are shown.

$\sigma \in \{15, 25, 35, 45, 75, 100\}$ to denoised HR EM images. The results revealed a pattern of performance decline, which was exacerbated by higher noise standard deviations (see Fig. 14). This observation aligns with inequality (7) and conditions (9a) and (9b), indicating that stronger noise in the references can lead to inferior outcomes. As the deviation from the clean reference increases, the training moves further away from the optimum, resulting in a notable decrease in SR performance. Notably, when $\sigma < 35$, the performance remained remarkable. However, a gradual decline was observed beyond this value, the acceptable performance was still achievable for $\sigma < 45$. However, for larger values, the performance significantly deteriorated.

Hyperparameter Analysis: The performance of the SR model across three hyperparameters is reported in Table V. The SR performance with varying patch sizes in the vision transformer, taken from the set $\{2 \times 2, 4 \times 4, 6 \times 6\}$, indicates that patch sizes 4×4 and 6×6 outperformed 2×2, with 4×4 achieving a higher FRC than 6×6. Furthermore, an examination of the number of channels within the network, spanning $\{16, 48, 64, 80\}$, illustrates that increasing the number of channels led to improvements across all metrics. Similarly, varying the number of basic blocks, with options ranging from $\{1, 2, 3, 4\}$, demonstrates that a greater number of basic blocks enhanced SR quality, as evidenced by detail-sensitive metrics such as SSIM and FRC. Overall, a larger number of basic blocks tended to improve all the metrics, although variability in the PSNR was observed. Table VI presents the results of assigning specific values to hyperparameters in the loss function (16). This highlights that considering all loss functions can lead to superior performance. For instance, setting ($\lambda_1 = \lambda_2 = 1$, $\lambda_3 = 0.01$) resulted in enhanced performance in detail recovery and intensity fidelity, as evidenced by the SSIM and PSNR, respectively. Additionally, keeping $\lambda_1 = \lambda_2 = 1$ and increasing the weight of the self-supervised loss to $\lambda_3 = 0.25$ effectively enhanced the intensity fidelity, as measured by PSNR, while preserving a high level of detail.

TABLE VI
THE PERFORMANCE OF EMSR[ℓ_1] WITH DIFFERENT HYPERPARAMETERS IN THE LOSS FUNCTION (16)

Metric	$\lambda_1 = 1$ $\lambda_2 = 0$ $\lambda_3 = 0$	$\lambda_1 = 1$ $\lambda_2 = 1$ $\lambda_3 = 0.01$	$\lambda_1 = 1$ $\lambda_2 = 1$ $\lambda_3 = 0.2$	$\lambda_1 = 1$ $\lambda_2 = 1$ $\lambda_3 = 0.25$	$\lambda_1 = 1$ $\lambda_2 = 1$ $\lambda_3 = 1$
SSIM	0.754	0.755	0.754	0.754	0.748
PSNR	22.5	22.6	22.7	22.8	22.7
FRC	0.355	0.355	0.354	0.353	0.351

Evaluated using a training strategy with BRAIN3 datasets as test sets, and BRAIN1, BRAIN2, BRAIN4, and BRAIN5 as training sets.

H. Generative Models

While generative models are popular in image SR, their performance can vary in the context of EM. For example, the GAN-based method in [61] produces visually pleasing HR natural images. However, in [45], this method performed poorly on EM images, introducing significant artifacts even in EM images with simple textures. Furthermore, GAN-based techniques are generally prone to instability and mode collapse [62], which can pose challenges for developing GAN-based SR methods. Among other generative models, diffusion models can produce more realistic SR results, with high perception quality [30], making them popular in various computer vision tasks. However, their application in scientific contexts is sensitive, as the perception-distortion trade-off should not favor perception. Additionally, the high resource demands of diffusion models reduce their practicality, especially for large 3D-EM datasets.

IV. CONCLUSION

We introduced a DL-based SR framework named EMSR to address the challenge of acquiring clean HR 3D-EM images across large tissue volumes. As corruptions are inherent in EM, training neural networks without clean references for ℓ_2 and ℓ_1 loss functions was explored. Following this, we crafted a noise-robust network that integrated both edge-attention and self-attention mechanisms, to focus on enhancing edge features

over less informative backgrounds in brain EM images. Using real LR and HR brain EM image pairs, the network underwent training with LR and HR pairs, along with LR and denoised HR pairs. The experimental results, in line with the discussed theory, confirmed the feasibility of training without clean references for both loss functions. While both losses demonstrated similar SR performance, consistent with the literature, ℓ_1 slightly outperformed ℓ_2. Furthermore, the EMSR method demonstrated superior or competitive results, both quantitatively and qualitatively, compared to established SR methods. In addition to training with real LR and HR pairs, we synthesized LR images from HR images using wide-ranging isotropic Gaussian noise and Gaussian kernels. Experiments with synthetic pairs showed promising results, that were comparable to those of models trained on real pairs. Notably, in some cases, the synthesis produced super-resolved images with sharper edges and improved contrasts, addressing inherent mismatches in LR and HR pairs, e.g., coregistration and contrast. This synthesis could also aid in deblurring while denoising and super-resolving LR EM.

EMSR offers both improved resolution and reduced noise simultaneously, enabling the computational generation of clean HR EM images over large samples from cost-effective LR EM imaging, allowing for use as a neuroimaging preprocessing tool for visualization and analysis.

Appendix A
Training With No-Clean-Reference

Let $\hat{x}$ and x be random variables such that $\hat{x} = x + n$, where n represents i.i.d noise with a mean of μ and a variance of $\sigma_n^2 I$. The reference-dependent solutions in (5), i.e., $\mathbb{E}_{x|y}[\mathcal{L}(f_\theta(y), x)]$ and $\mathbb{E}_{\hat{x}|y}[\mathcal{L}(f_\theta(y), \hat{x})]$, for both the ℓ_2 and ℓ_1 norms are discussed in the following subsections.

A. Solution for the ℓ_2-Norm Loss Function

The proof of (6) is provided below:

$$\begin{aligned}
&\mathbb{E}_{\hat{x}|y}[\|f_\theta(y) - \hat{x}\|_2^2] \\
&= \mathbb{E}_{x,\hat{x}|y}[\|(f_\theta(y) - x - n)\|_2^2] \\
&= \mathbb{E}_{x,\hat{x}|y}[(f_\theta(y) - x - n)^T (f_\theta(y) - x - n)] \\
&= \mathbb{E}_{x,\hat{x}|y}[\|f_\theta(y) - x\|_2^2 - 2n^T(f_\theta(y) - x) + \|n\|_2^2] \\
&= \mathbb{E}_{x|y}[\|f_\theta(y) - x\|_2^2] - 2\mathbb{E}_{x,\hat{x}|y}[n^T(f_\theta(y) - x)] + \mathbb{E}_{x,\hat{x}|y}[\|n\|_2^2] \\
&= \mathbb{E}_{x|y}[\|f_\theta(y) - x\|_2^2] - 2\mathbb{E}_{x,\hat{x}|y}[n^T(f_\theta(y) - x)] + d\sigma_n^2 + ||\mu||^2 \\
&\overset{*}{=} \mathbb{E}_{x|y}[\|f_\theta(y) - x\|_2^2] - 2(\mathbb{E}_{\hat{x}|y}[\hat{x}] - \mathbb{E}_{x|y}[x])^T \mathbb{E}_{x|y}[(f_\theta(y) - x)] \\
&\quad + d\sigma_n^2 + ||\mu||^2 \\
&= \mathbb{E}_{x|y}[\|f_\theta(y) - x\|_2^2] - 2\mu^T \mathbb{E}_{x|y}[f_\theta(y) - x] + d\sigma_n^2 + ||\mu||^2
\end{aligned} \tag{20}$$

* Under the assumption of i.i.d noise, we can establish $\mathbb{E}_{\hat{x}|y}[\hat{x}] - \mathbb{E}_{x|y}[x] = \mathbb{E}_{x,\hat{x}|y}[n] = \mu$.

B. Bounds for the ℓ_1-Norm Loss Function

We derive two upper bounds for the ℓ_1 loss, including (7), by using the following inequality that holds for vectors u and v in the p-norm in $\mathbb{C}^n$:

$$\mathbb{E}[\|u\|_p] - \mathbb{E}[\|v\|_p] \leq \mathbb{E}[\|u - v\|_p] \tag{21}$$

By setting $f_\theta(y) - \hat{x}$ and $f_\theta(y) - x$ as u and v, respectively, we can rewrite the inequality as:

$$\underbrace{\mathbb{E}_{(\hat{x},y)}[\|f_\theta(y) - \hat{x}\|_p] - \mathbb{E}_{(x,y)}[\|f_\theta(y) - x\|_p]}_{0 \leq} \leq \mathbb{E}_{(x,\hat{x})}[\|n\|_p] \tag{22}$$

Without loss of generality, we assume that the training error with a corrupted reference $\hat{x}$ is greater than or equal to the training error with a clean reference x, leading to the nonnegativity of the left-hand side of (22).

Let u be a vector in $\mathbb{C}^n$ with $1 \leq r < p$. Upon a well-known corollary of Hölder's inequality,

$$\|u\|_p \leq \|u\|_r \leq d^{(1/r - 1/p)} \|u\|_p, \tag{23}$$

where d is the dimension of u. By setting $p = 2$ and $r = 1$ in (23), we can establish a connection between the ℓ_1 and ℓ_2 norms as $\|u\|_1 \leq \sqrt{d}\|u\|_2$, which can be transformed by taking the square of each side and applying the expectation rule,

$$\mathbb{E}[\|u\|_1^2] \leq d\mathbb{E}[\|u\|_2^2] \tag{24}$$

Applying Jensen's inequality, which states that $f(\mathbb{E}[x]) \leq \mathbb{E}[f(x)]$ for a convex function $f : \mathbb{R} \to \mathbb{R}$, the inequality above can be lower bounded as follows:

$$\begin{aligned}
(\mathbb{E}[\|u\|_1])^2 &\leq \mathbb{E}[\|u\|_1^2] \\
&\leq d\mathbb{E}[\|u\|_2^2]
\end{aligned} \tag{25}$$

Taking the square root of both sides of (25) yields:

$$\mathbb{E}[\|u\|_1] \leq \sqrt{d}\sqrt{\mathbb{E}[\|u\|_2^2]} \tag{26}$$

Using the above inequality we can establish two upper bounds:

1. *Upper-Bound (I):* Considering (22) with $p = 1$ and (26),

$$\begin{aligned}
0 &\leq \mathbb{E}_{\hat{x}|y}[\|(f_\theta(y) - \hat{x})\|_1] - \mathbb{E}_{x|y}[\|(f_\theta(y) - x)\|_1] \\
&\leq \sqrt{d}\sqrt{\mathbb{E}[\|n\|_2^2]} = \sqrt{d}\sqrt{d\sigma_n^2 + ||\mu||^2}
\end{aligned} \tag{27}$$

2. *Upper-Bound (II):* First, apply inequality (26) to $f_\theta(y) - \hat{x}$ and $f_\theta(y) - x$,

$$\mathbb{E}_{\hat{x}|y}[\|f_\theta(y) - \hat{x}\|_1] \leq \sqrt{d}\sqrt{\mathbb{E}_{\hat{x}|y}[\|f_\theta(y) - \hat{x}\|_2^2]}, \tag{28a}$$

$$\mathbb{E}_{x|y}[\|f_\theta(y) - x\|_1] \leq \sqrt{d}\sqrt{\mathbb{E}_{x|y}[\|f_\theta(y) - x\|_2^2]} \tag{28b}$$

Using (28a) and (28b),

$$\begin{aligned}
0 &\leq \mathbb{E}_{\hat{x}|y}[\|(f_\theta(y) - \hat{x})\|_1] - \mathbb{E}_{x|y}[\|(f_\theta(y) - x)\|_1] \\
&\leq \sqrt{d}\left|\sqrt{\mathbb{E}_{\hat{x}|y}[\|(f_\theta(y) - \hat{x})\|_2^2]} - \sqrt{\mathbb{E}_{x|y}[\|(f_\theta(y) - x)\|_2^2]}\right|
\end{aligned} \tag{29}$$

The inequality above can equivalently be formulated as follows:

$$\begin{aligned} 0 &\le \mathbb{E}_{\hat{x}|y}[\|f_\theta(y)-\hat{x}\|_1] - \mathbb{E}_{x|y}[\|f_\theta(y)-x\|_1] \\ &\le \sqrt{d}\left|\frac{\mathbb{E}_{\hat{x}|y}[\|f_\theta(y)-\hat{x}\|_2^2] - \mathbb{E}_{x|y}[\|f_\theta(y)-x\|_2^2]}{\sqrt{\mathbb{E}_{\hat{x}|y}[\|f_\theta(y)-\hat{x}\|_2^2]}+\sqrt{\mathbb{E}_{x|y}[\|f_\theta(y)-x\|_2^2]}}\right| \\ &\overset{*}{=} \sqrt{d}\left|\frac{-2\mu^T\mathbb{E}_{x|y}[f_\theta(y)-x]+d\sigma_n^2+||\mu||^2}{\sqrt{\mathbb{E}_{\hat{x}|y}[\|f_\theta(y)-\hat{x}\|_2^2]}+\sqrt{\mathbb{E}_{x|y}[\|f_\theta(y)-x\|_2^2]}}\right| \\ &= \frac{|-2\mu^T\mathbb{E}_{x|y}[f_\theta(y)-x]+d\sigma_n^2+||\mu||^2|}{g(y,x,\hat{x})}, \end{aligned} \tag{30}$$

where $g(y,x,\hat{x}) = \frac{\sqrt{\mathbb{E}_{\hat{x}|y}[\|f_\theta(y)-\hat{x}\|_2^2]}+\sqrt{\mathbb{E}_{x|y}[\|f_\theta(y)-x\|_2^2]}}{\sqrt{d}}$. * The difference between solutions for $\hat{x}$ and x when loss function is ℓ_2 norm, see (20). Unlike upper-bound (I), (II) shows dependence on both y and noise statistics.

ACKNOWLEDGMENT

The authors would like to thank the Electron Microscopy Unit at the Institute of Biotechnology, University of Helsinki, Finland, for providing the 3D-EM datasets. The authors would also like to thank the Bioinformatics Center at the University of Eastern Finland, Finland, and the CSC-IT Center for Science, Finland, for providing computational resources.

REFERENCES

[1] D. G. C. Hildebrand et al., "Whole-brain serial-section electron microscopy in larval zebrafish," *Nature*, vol. 545, no. 7654, pp. 345–349, 2017.
[2] Z. Zheng et al., "A complete electron microscopy volume of the brain of adult Drosophila melanogaster," *Cell*, vol. 174, no. 3, pp. 730–743, 2018.
[3] N. Varsano and S. G. Wolf, "Electron microscopy of cellular ultrastructure in three dimensions," *Curr. Opin. Struct. Biol.*, vol. 76, 2022, Art. no. 102444.
[4] 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.
[5] S. Mikula and W. Denk, "High-resolution whole-brain staining for electron microscopic circuit reconstruction," *Nature Methods*, vol. 12, no. 6, pp. 541–546, 2015.
[6] B. Imbrosci, D. Schmitz, and M. Orlando, "Automated detection and localization of synaptic vesicles in electron microscopy images," *eNeuro*, vol. 9, no. 1, 2022, doi: 10.1523/ENEURO.0400-20.2021gg.
[7] 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.
[8] A. Liu, Y. Liu, J. Gu, Y. Qiao, and C. Dong, "Blind image super-resolution: A survey and beyond," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 5461–5480, May 2023.
[9] D. Ren, W. Zuo, D. Zhang, L. Zhang, and M.-H. Yang, "Simultaneous fidelity and regularization learning for image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 284–299, Jan. 2021.
[10] C. A. Bouman, *Foundations of Computational Imaging: A Model-Based Approach*. Philadelphia, PA, USA: SIAM, 2022.
[11] D. Meng and F. D. L. Torre, "Robust matrix factorization with unknown noise," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1337–1344.
[12] X. Cao, Q. Zhao, D. Meng, Y. Chen, and Z. Xu, "Robust low-rank matrix factorization under general mixture noise distributions," *IEEE Trans. Image Process.*, vol. 25, no. 10, pp. 4677–4690, Oct. 2016.
[13] Z. Wang, J. Chen, and S. C. Hoi, "Deep learning for image super-resolution: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3365–3387, Oct. 2021.
[14] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1–4, pp. 259–268, 1992.
[15] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 349–356.
[16] Z. Zha, B. Wen, X. Yuan, J. Zhou, C. Zhu, and A. C. Kot, "Low-rankness guided group sparse representation for image restoration," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 10, pp. 7593–7607, Oct. 2023.
[17] H. Chen et al., "Real-world single image super-resolution: A brief review," *Inf. Fusion*, vol. 79, pp. 124–145, 2022.
[18] W. T. Freeman and E. C. Pasztor, "Markov networks for super-resolution," in *Proc. 34th Annu. Conf. Inf. Sci. Syst.*, 2000.
[19] X. Lu, Y. Yuan, and P. Yan, "Image super-resolution via double sparsity regularized manifold learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 12, pp. 2022–2033, Dec. 2013.
[20] Y. Romano, J. Isidoro, and P. Milanfar, "RAISR: Rapid and accurate image super resolution," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 110–125, Mar. 2017.
[21] J. Yang, Z. Lin, and S. Cohen, "Fast image super-resolution based on in-place example regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 1059–1066.
[22] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.
[23] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. 13th Eur. Conf. Comput. Vis.*, Zurich, Switzerland, 2014, pp. 184–199.
[24] T.-A. Song, S. R. Chowdhury, F. Yang, and J. Dutta, "Super-resolution PET imaging using convolutional neural networks," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 518–528, 2020.
[25] K. Zhang, J. Liang, L. V. Gool, and R. Timofte, "Designing a practical degradation model for deep blind image super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 4791–4800.
[26] Y. Sui, O. Afacan, C. Jaimes, A. Gholipour, and S. K. Warfield, "Scan-specific generative neural network for MRI super-resolution reconstruction," *IEEE Trans. Med. Imag.*, vol. 41, no. 6, pp. 1383–1399, Jun. 2022.
[27] X. Wang, L. Xie, C. Dong, and Y. Shan, "Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 1905–1914.
[28] Z. Lu, J. Li, H. Liu, C. Huang, and T. Zhang, "Transformer for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 457–466.
[29] J. Liang, J. Cao, G. Sun, K. Zhang, L. V. Gool, and R. Timofte, "SwinIR: Image restoration using swin transformer," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 1833–1844.
[30] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 4713–4726, Apr. 2023.
[31] K. Mei, M. Delbracio, H. Talebi, Z. Tu, V. M. Patel, and P. Milanfar, "CoDi: Conditional diffusion distillation for higher-fidelity and faster image generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 9048–9058.
[32] K. Zhang, L. V. Gool, and R. Timofte, "Deep unfolding network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3217–3226.
[33] Q. Ma, J. Jiang, X. Liu, and J. Ma, "Deep unfolding network for spatiospectral image super-resolution," *IEEE Trans. Comput. Imag.*, vol. 8, pp. 28–40, 2022.
[34] W. C. Karl, J. E. Fowler, C. A. Bouman, M. Çetin, B. Wohlberg, and J. C. Ye, "The foundations of computational imaging: A signal processing perspective," *IEEE Signal Process. Mag.*, vol. 40, no. 5, pp. 40–53, Jul. 2023.
[35] S. H. Chan, X. Wang, and O. A. Elgendy, "Plug-and-play ADMM for image restoration: Fixed-point convergence and applications," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 84–98, Mar. 2017.
[36] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. V. Gool, and R. Timofte, "Plug-and-play image restoration with deep denoiser prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 6360–6376, Oct. 2022.
[37] S. Shoushtari, J. Liu, E. P. Chandler, M. S. Asif, and U. S. Kamilov, "Prior mismatch and adaptation in PNP-ADMM with a nonconvex convergence analysis," 2023, *arXiv:2310.00133*.
[38] S. Abu-Hussein, T. Tirer, S. Y. Chun, Y. C. Eldar, and R. Giryes, "Image restoration by deep projected GSURE," in *Proc. Winter Conf. Appl. Comput. Vis.*, pp. 3602–3611, 2022.
[39] Z. Zou, J. Liu, B. Wohlberg, and U. S. Kamilov, "Deep equilibrium learning of explicit regularization functionals for imaging inverse problems," *IEEE Open J. Signal Process.*, vol. 4, pp. 390–398, 2023.

[40] D. Gilton, G. Ongie, and R. Willett, "Deep equilibrium architectures for inverse problems in imaging," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 1123–1133, 2021.
[41] S. Tsiper, O. Dicker, I. Kaizerman, Z. Zohar, M. Segev, and Y. C. Eldar, "Sparsity-based super resolution for SEM images," *Nano Lett.*, vol. 17, no. 9, pp. 5437–5445, 2017.
[42] S. Sreehari, S. Venkatakrishnan, K. L. Bouman, J. P. Simmons, L. F. Drummy, and C. A. Bouman, "Multi-resolution data fusion for super-resolution electron microscopy," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 88–96.
[43] Z. Gao, W. Ma, S. Huang, P. Hua, and C. Lan, "Deep learning for super-resolution in a field emission scanning electron microscope," *AI*, vol. 1, no. 1, pp. 1–10, 2020.
[44] L. Fang et al., "Deep learning-based point-scanning super-resolution imaging," *Nature methods*, vol. 18, no. 4, pp. 406–416, 2021.
[45] E. J. Reid, L. F. Drummy, C. A. Bouman, and G. T. Buzzard, "Multi-resolution data fusion for super resolution imaging," *IEEE Trans. Comput. Imag.*, vol. 8, pp. 81–95, 2022.
[46] Y. Qian, J. Xu, L. F. Drummy, and Y. Ding, "Effective super-resolution methods for paired electron microscopic images," *IEEE Trans. Image Process.*, vol. 29, pp. 7317–7330, 2020.
[47] B. Titze, "Techniques to prevent sample surface charging and reduce beam damage effects for SBEM imaging," Ph.D. dissertation, Dept. Biomed. Opt., Max Planck Inst. Med. Res., Heidelberg, Germany, 2013.
[48] M. G. d. Faria, Y. Haddab, Y. L. Gorrec, and P. Lutz, "Influence of mechanical noise inside a scanning electron microscope," *Rev. Sci. Instrum.*, vol. 86, no. 4, 2015, Art. no. 045105.
[49] J. Lehtinen et al., "Noise2Noise: Learning image restoration without clean data," in *Proc. Int. Conf. Mach. Learn.*, 2018, vol. 80, pp. 2965–2974.
[50] N. Moran et al., "Noisier2noise: Learning to denoise from unpaired noisy data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12064–12072.
[51] A. F. Calvarons, "Improved noise2noise denoising with limited data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 796–805.
[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[53] M. Ghahremani, Y. Liu, and B. Tiddeman, "FFD: Fast feature detector," *IEEE Trans. Image Process.*, vol. 30, pp. 1153–1168, 2021.
[54] W. Shi et al., "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1874–1883.
[55] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 47–57, Mar. 2017.
[56] A. Abdollahzadeh, I. Belevich, E. Jokitalo, A. Sierra, and J. Tohka, "DeepACSON automated segmentation of white matter in 3D electron microscopy," *Commun. Biol.*, vol. 4, no. 1, 2021, Art. no. 179.
[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[58] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[59] N. Banterle, K. H. Bui, E. A. Lemke, and M. Beck, "Fourier ring correlation as a resolution criterion for super-resolution microscopy," *J. Struct. Biol.*, vol. 183, no. 3, pp. 363–367, 2013.
[60] M. Ghahremani, M. Khateri, A. Sierra, and J. Tohka, "Adversarial distortion learning for medical image denoising," 2022, *arXiv:2204.14100*.
[61] K. Zhang, W. Zuo, and L. Zhang, "Deep plug-and-play super-resolution for arbitrary blur kernels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1671–1681.
[62] B. B. Moser, A. S. Shanbhag, F. Raue, S. Frolov, S. Palacio, and A. Dengel, "Diffusion models, image super-resolution and everything: A survey," 2024, *arXiv:2401.00736*.

Mohammad Khateri received the M.Sc. degree in electrical engineering from Tarbiat Modares University, Tehran, Iran, in 2017. He is currently working toward the Ph.D. degree with the A.I. Virtanen Institute for Molecular Sciences, University of Eastern Finland, Kuopio, Finland. His research interests include image processing, pattern recognition, artificial intelligence, and their applications in biomedical imaging.

Morteza Ghahremani received the Ph.D. degree in computer science (AI) in 2021, focusing on 3D reconstruction and point cloud data analysis. Since May 2021, he has been a Postdoctoral Researcher of medical imaging analysis with the University of Eastern Finland, Kuopio, Finland, before joining the Technical University of Munich, Munich, Germany, in November 2022. He has authored or coauthored several papers in top-ranked international journals and conference proceedings, such as CVPR, NeurIPS, ECCV, IEEE TRANSACTIONS ON IMAGE PROCESSING, and others. His research interests include LLM, Generative AI, foundation models, organ/object detection, and multimodal data analysis, with applications in biomedical, and medical imaging.

Alejandra Sierra received the Ph.D. degree in biochemistry from the Autonomous University of Madrid, Madrid, Spain, in 2006. Since then, she has been working with the A.I. Virtanen Institute for Molecular Sciences, University of Eastern Finland, Kuopio, Finland, where she held both Academy of Finland Postdoctoral and Research Fellow positions and is currently the Research Director. Her research interests include the validation and development of magnetic resonance imaging techniques by incorporating microscopic tissue information in the healthy and diseased brain.

Jussi Tohka received the Ph.D. degree in signal processing from the Tampere University of Technology, Tampere, Finland, in 2003. He was a Postdoctoral Fellow with the University of California at Los Angeles, Los Angeles, CA, USA. He held an Academy Research Fellow position with the Department of Signal Processing, Tampere University of Technology. He was a CONEX Professor with the Department of Bioengineering and Aerospace Engineering, Universidad Carlos III de Madrid, Madrid, Spain. He is currently with the A.I. Virtanen Institute for Molecular Sciences, University of Eastern Finland, Kuopio, Finland. His research interests include machine learning, image analysis, and pattern recognition, and their applications to brain imaging.