Small training dataset convolutional neural networks for application-specific super-resolution microscopy

Varun Mannam* **and Scott Howard***
University of Notre Dame, Department of Electrical Engineering, Notre Dame, Indiana, United States

Abstract

Significance: Machine learning (ML) models based on deep convolutional neural networks have been used to significantly increase microscopy resolution, speed [signal-to-noise ratio (SNR)], and data interpretation. The bottleneck in developing effective ML systems is often the need to acquire large datasets to train the neural network. We demonstrate how adding a "dense encoder-decoder" (DenseED) block can be used to effectively train a neural network that produces super-resolution (SR) images from conventional microscopy diffraction-limited (DL) images trained using a small dataset [15 fields of view (FOVs)].

Aim: The ML helps to retrieve SR information from a DL image when trained with a massive training dataset. The aim of this work is to demonstrate a neural network that estimates SR images from DL images using modifications that enable training with a small dataset.

Approach: We employ "DenseED" blocks in existing SR ML network architectures. DenseED blocks use a dense layer that concatenates features from the previous convolutional layer to the next convolutional layer. DenseED blocks in fully convolutional networks (FCNs) estimate the SR images when trained with a small training dataset (15 FOVs) of human cells from the Widefield2SIM dataset and in fluorescent-labeled fixed bovine pulmonary artery endothelial cells samples.

Results: Conventional ML models without DenseED blocks trained on small datasets fail to accurately estimate SR images while models including the DenseED blocks can. The average peak SNR (PSNR) and resolution improvements achieved by networks containing DenseED blocks are ≈3.2 dB and 2×, respectively. We evaluated various configurations of target image generation methods (e.g., experimentally captured a target and computationally generated target) that are used to train FCNs with and without DenseED blocks and showed that including DenseED blocks in simple FCNs outperforms compared to simple FCNs without DenseED blocks.

Conclusions: DenseED blocks in neural networks show accurate extraction of SR images even if the ML model is trained with a small training dataset of 15 FOVs. This approach shows that microscopy applications can use DenseED blocks to train on smaller datasets that are application-specific imaging platforms and there is promise for applying this to other imaging modalities, such as MRI/x-ray, etc.

 [DOI: 10.1117/1.JBO.28.3.036501]

Keywords: small datasets; biomedical imaging; fluorescence microscopy; super-resolution; diffraction-limited; machine learning; fully convolutional networks; convolutional neural networks; generative adversarial networks; dense encoder-decoder; dense layer.

Paper 220201GR received Aug. 26, 2022; accepted for publication Feb. 9, 2023; published online Mar. 14, 2023.

*Adress all correspondence to Varun Mannam, vmannam@nd.edu; Scott Howard, showard@nd.edu

Introduction

Significant technical advances have allowed researchers to break through the fundamental limits in biomedical imaging resolution and speed, subsequently leading to significant improvements in data analysis and interpretation.[1–3] However, many of these approaches require specialized equipment and training, limiting their applicability. For example, the diffraction limit in fluorescence microscopy has been overcome by a wide variety of super-resolution (SR) techniques.[4–7] To make these technical advances more widely available, machine learning (ML) approaches have been used to estimate SR images obtained from those techniques while using conventional and commonly available imaging platforms.[8–10] These ML models are powerful and easily distributable; however, they require significantly large training datasets[9,10] ($\geq$10,000 images) that are often prohibitively expensive and time-consuming to generate. This limitation is especially true for biomedical imaging such as *in vivo* imaging, magnetic resonance imaging (MRI) imaging, and x-ray.[11–14] In addition, the imaging experimental setup for the above-mentioned applications is specific to those applications (denoted as application-specific) with variance across experimental equipment. Without large training datasets, existing ML models are less accurate and not capable of generating SR images from diffraction-limited (DL) images.

In this paper, we develop, demonstrate, and evaluate using a small training dataset (much less than 1000 images) with convolutional neural network (CNN) models by incorporating new dense Encoder-decoder ("DenseED") blocks[15] that can successfully estimate fluorescence microscope images with resolution enhancements. To illustrate this method, we trained a CNN with DenseED blocks using small training datasets, which increased both the resolution by a factor of 2 and the peak signal-to-noise ratio (PSNR) by 3.2 dB. Such performance is not possible using conventional CNNs without DenseED blocks. The results show how ML models can be novel for specific equipment and applications using small datasets acquired by that specific tool.

2 Methods and Dataset Creation

2.1 *Traditional Super-Resolution Methods*

Fluorescence microscopy is a key research tool throughout biology.[16] However, the spatial resolution of an image generated by conventional fluorescence microscopy is limited to a few hundred nanometers defined by the diffraction limit of light.[17] The limited resolution hinders further observation and investigation of objects at a subcellular or molecular scale, such as mitochondria, microtubules, nanopores, and proteins within cells and tissues. Many fluorescence microscopy SR methods can overcome the diffraction limit and achieve better resolutions up to ten times greater than conventional microscopy techniques. Experimental methods, such as stimulated emission depletion (STED),[4] structured illumination microscopy (SIM),[5] and non-linear SIM[18,19] perform SR imaging; typically, they require dedicating imaging platforms. Exploiting the non-linearity of excitation saturation in scanning microscopy enables SR microscopy in conventional microscope platforms.[20–22] Localization and statistical approaches, including stochastic optical reconstruction microscopy (STORM)[6] and photoactivated localization microscopy (PALM)[7] can also enhance the image resolution but require special fluorophores and extensive computation. Computational methods, such as SR radial fluctuation (SRRF),[23] can be used to perform SR imaging. SRRF can generate images with a resolution comparable to localization approaches without requiring complicated hardware setups and special imaging conditions. Even so, it requires numerous DL images to be collected within a single FOV and is computationally expensive. To achieve the benefits of SR techniques on conventional imaging platforms, ML approaches can be used.

2.2 *ML-Based Super-Resolution Methods in Literature*

The ML has gained attention for its fast processing speed and wide applications, such as image classification,[24,25] image denoising,[10] image segmentation,[26] and image compression.[27,28] The

ML models achieve high performance and generalization capacity when trained with a large training dataset.[10] However, obtaining a large training dataset is often prohibitively expensive or difficult.[29] In addition, the variance between the same models of the experimental equipment can be large (due to each application-specific equipment calibration/setup setting being different), making generalizability difficult.[30] Hence, the training dataset size is often limited and application specific. Nonetheless, existing ML models show high performance when trained with a large training dataset. Hence there is a trade-off between application-specific ML model performance vs. training dataset size.[31–35]

In the literature, existing ML-based SR methods can be classified into two categories:[36] fully convolutional networks (FCNs) and generative adversarial networks (GANs). FCNs contain a combination of encoder and decoder blocks[37] as shown in Fig. 1. Some examples of FCN architectures are U-Net,[39] dense nets,[15] residual nets,[40] and AEs.[9] The FCN architecture includes multiple encoder and decoder blocks (convolutional layers), and the output is generated by combining the output of encoder layers from different convolutional layers in encoder and decoder blocks (refer to Fig. 1). To pass the features generated in the encoder blocks to the corresponding decoder blocks, Skip connections are helpful (refer to Fig. 1). GAN architecture is based on simultaneously optimizing two networks (generator and discriminator).[41] Two networks compete to generate the best images similar to target images from input images. In GANs, the generator network is a simple FCN (i.e., the generator consists of encoder and decoder blocks). The discriminator network consists of convolutional layers followed by the fully connected layers that generate the probability that the generator network output (estimated SR image here) looks like the real image (similar to the target image). Because the GAN generator architecture is a simple FCN architecture (to generate SR images), in this paper we show our demonstrated approach using only FCN architecture. Additional details about GANs, including GAN encoder and decoder, can be found in these references.[42–47] More details about the GANs including architecture, loss function, and optimization are provided in our GitHub location (https://github.com/ND-HowardGroup/Application-Specific-Super-resolution.git).

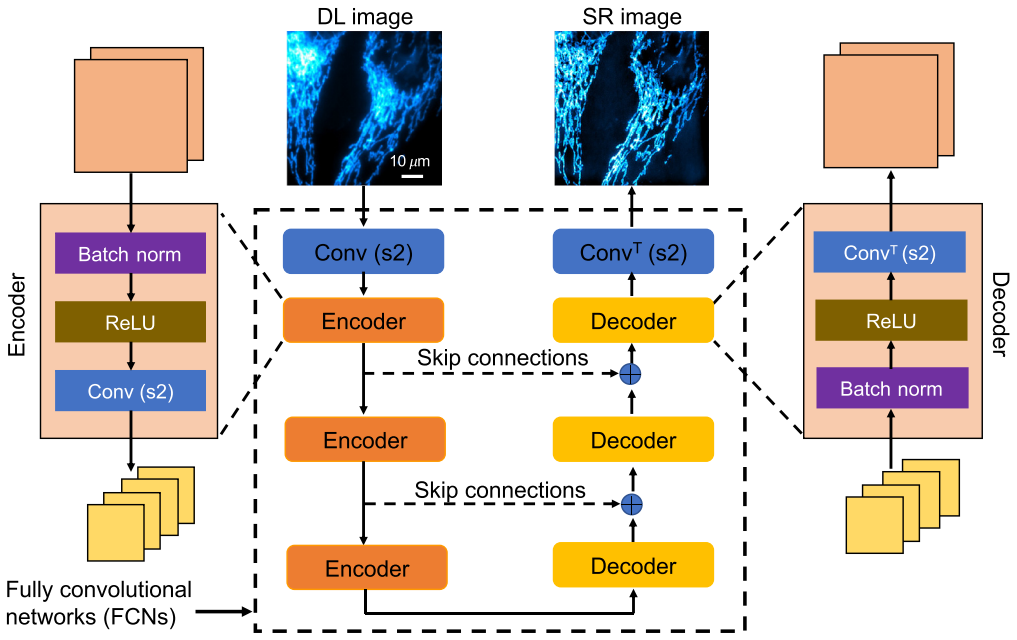

Fig. 1 Block diagram of fully convolutional networks with Skip connections, including the encoder and decoder blocks. Here the network indicates the AE and U-Net architectures without and with Skip connections,[38] respectively. Encoder and decoder blocks consist of batch-norm, ReLU (rectified linear unit), and convolution layers. Conv(s2) and Conv^T(s2) indicate the convolution and convolution transpose layers with a stride of 2, respectively. The symbol $\oplus$ represents the concatenation layer that combines the output from the encoder layer and decoder layer in the number of channels dimension.

In addition, advanced ML models such as zero-shot SR (ZSSR)[48–51] and one-shot SR (OSSR)[52–54] with CNNs have been demonstrated to estimate high-resolution (HR) images from low-resolution ones. In the case of the ZSSR, the ML model is trained with the test image itself (hence, no-training dataset, and it is an unsupervised ML method), and performance is limited due to no training dataset. In the case of the OSSR, an extensive training dataset is used to get the HR features, and ML model weights are stored. After that, a small training dataset is used to retrain the ML model with pretrained model weights. Hence, in the OSSR case, you need two training datasets with similar features in the application-specific imaging. However, these ML models in the literature are trained on color images with datasets such as Set5 dataset,[55] BSD100 images,[56] and DIV2k images[57] but not on application-specific for example, fluorescence microscopy datasets.[29] Wang et al.[51] provided a consolidated summary of SR methods in deep learning. In application-specific SR generation, the existing computational methods that use no training data (self-supervised learning) are computationally expensive (iterative methods like image deconvolution) and lead to poor performance. In contrast, if the training dataset is large, the existing ML-based models provide higher performance, but acquiring a large training dataset (DL and target images) is computationally expensive. Hence, finding a balance between the training dataset size and generated SR images quality is significant, and this paper contributes by showing an ML-based method to mitigate the issue by providing SR images accurately even if the ML model is trained with a small training dataset with input as DL image and target as SR image, respectively. Furthermore, this ML model can be applied to other application-specific SR generation with a small dataset.

In fluorescence microscopy, traditional FCNs have been applied to generate SR images from simulated and experimental data. The trained ML model (FCN) performance is evaluated by comparing the estimated SR images with the target images acquired using SR microscopes. Table 1 shows a few examples of ML models including architecture (either FCN or GAN) and the size of the training dataset used in literature to generate fluorescence microscopy SR images. In Nehme's work,[9] the FCN architecture consists of three encoders and three decoder blocks, respectively, and is trained with 7000 images. In Ayas's work,[58] the FCN architecture includes a 20-layer residual network with blood samples trained with 16,000 images. In Wang's work,[59] the architecture is GAN with the generator network similar to the U-Net[39] architecture, and the discriminator network consists of fully connected layers trained with 2000 bovine pulmonary artery endothelial (BPAE) cells sample images for each fluorophore. Similarly, in Zhang's[60] work the ML model is GAN architecture consisting of a generator network with 16-layer residual connections, and a discriminator network consisting of fully connected layers with 1,080 images of fibroblast in a mouse brain. Finally, in Ouyang's work[61] a GAN architecture with the generator network consists of U-Net with (8,8) encoder and decoder blocks, respectively, and the discriminator network consists of fully connected layers trained with 30,000 PALM images of microtubules. Despite the ability to obtain SR images from DL images, all of the above-mentioned ML-based SR models are data-driven. These trained ML models require a large training dataset (more than 1000 images) to generate SR images in fluorescence microscopy.

Table 1 Summary of existing ML SR methods with fluorescence microscopy data.

Papers	Architecture	Training dataset Size	Sample details
Deep-STORM[9]	FCNs	7000	Microtubules
Residual CNN's[58]	FCNs	16,000	Blood samples
GANs structure[59]	GANs	2000 (each fluorophore)	BPAE samples and nano beads
RFGANs[60]	GANs	1080 (increase in FOV)	Fibroblast in mouse brain
ANNA-PALM[61]	GANs	30,000	Microtubules and nanopores

2.3 *FCN with Dense Encoder-Decoder*

This section explains the DenseED method and how it is derived from the existing FCN architecture to provide SR images when trained with a small training dataset. FCNs[62] are used for pixel-wise prediction, e.g., semantic segmentation,[39] image denoising,[10] SR[36] and low dose computer tomography x-ray reconstruction.[63] Figure 1 shows the FCN architecture with encoding and decoding blocks with Skip connections. The convolutional layer contains the input image convolved with a kernel that extracts particular features from the input images (for example, edges, backgrounds, and objects with different shapes). Here the number of kernels used in the convolutional layer is called the "number of feature maps" and the output of the convolution indicates the "feature map" with its dimension as "feature map size." Typically an encoding block contains the convolutional layer with double feature maps and half of the feature map size. Encoder block is used to extract important features, thereby reducing feature map size to half. In this way, we select only the essential features as the output of the encoder block. The decoder block works exactly opposite to the encoder block; its output reduces the number of feature maps to half and doubles the feature map size. Extracting complex features, such as SR images, from DL images requires more encoder and decoder blocks in the ML model.

However, the feature map reaches a minimum image dimension with more encoding blocks, and SR images cannot be restored using decoder blocks alone without Skip or residual or dense connections, due to vanishing the gradients issue in deep learning[64,65] (see Fig. 1). In other words, coarse features are not passed through the decoder blocks in the case of deep networks. However, this requirement is not necessary when ML models contain only a small number of encoder and decoder blocks. This minimum image dimension of the encoder is called the "latent space." Additionally, as the number of encoder and decoder blocks increases, the number of kernel parameters (i.e., weights of the neural network) increases exponentially, which is parameter inefficient (requiring considerable computation time) for the ML model. As the number of encoder and decoder blocks increases, the feature map size is reduced, and the essential features are lost. Therefore, "Skip connections" are introduced between encoder and decoder blocks to pass finer features (such as mitochondria and microtubules) to the decoder blocks from encoder blocks. This modified FCN architecture, called "U-Net,"[10,39] is shown in Fig. 1 with dashed arrows; $\oplus$ indicates the concatenation of features from the encoder block and the output of the previous decoder block. Another ML model that belongs to the FCN architecture is the "Residual-Net,"[66] which consists of residual layers (or Skip connection from input to output directly) where input is passed through a couple of convolutional layers. Each convolutional layer consists of convolution, nonlinear elements (such as ReLU), and normalization (batch norm) layers. The last convolutional layer output is concatenated with the input. The estimated output image from the convolutional layer is the residual between target and input images (for example, noise: the subtraction of the noise input with a clear target).

To allow for the FCNs with higher performance when trained with a small training dataset, the modified residual connections are helpful. These modified residual connections were originally developed for physical systems and computer vision tasks. DenseED[67] is the state-of-the-art CNN architecture (modified version of residual layers) due to its backbone of dense layers, which passes the extracted features from the previous layer to all next layers in a feed-forward fashion. This paper shows how to utilize these DenseED blocks to build our SR ML model that works with a small dataset. Figure 2(a) shows the demonstrated ML model (DenseED in FCNs) for SR using an ultra-small training dataset. Figure 2(a) is similar to Fig. 1 but with additional DenseED blocks added after the encoder and decoder blocks. Figure 2(b) shows the DenseED block, which consists of multiple dense layers, which is another way of passing features from one layer to the next. Dense layers[15,68] are used to create dense connections between all layers to improve the information (gradient) flow through the complete ML model for better parameter efficiency. Figure 2(c) shows the dense layer connection for i'th dense layer with input feature maps of x_0 (output of the previous layer) and passed through the dense layer with output feature maps of x_1; total feature maps are the concatenation of input and output feature maps $[x_0, x_1]$. In the dense layer, the convolution operation is performed with a stride of 1. Figure 2(b) shows a dense block with three dense layers, where each layer provides two feature maps as output. The dense layer establishes connections from the previous convolutional layer to all subsequent

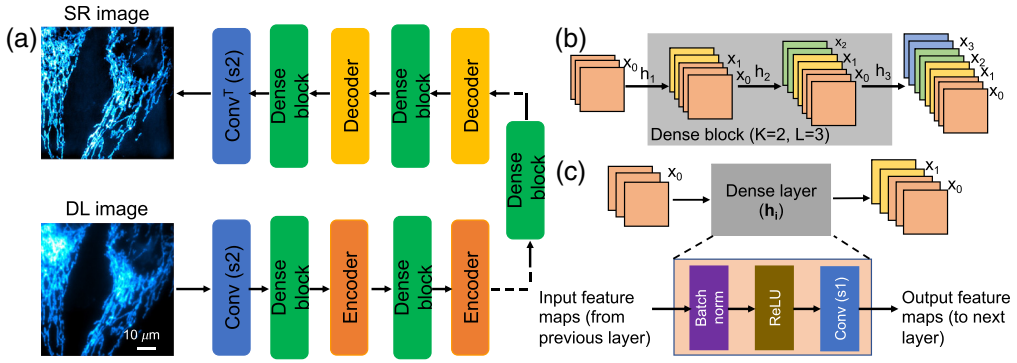

Fig. 2 Block diagram of fully convolutional networks with dense blocks: (a) dense blocks consist of multiple dense layers (b) where each dense layer's input feature maps are concatenated progressively. (c) The dense layer consists of the batch norm, ReLU, and convolution layer with stride 1 in sequence order.

convolutional layers in the dense block. In other words, one layer's input features are concatenated to this layer's output features, which serve as the input features to the next layer. If the input has K_0 feature maps and each layer of the outputs has K feature maps, then the i'th layer would have an input with $K_0 + (i * K)$ feature maps, i.e., the number of feature maps in dense block grows linearly with the depth, and K here is referred to as the growth rate. More dense layers are required for the given feature map size within a dense block to access the complex features. With more dense layers in a dense block, the total output feature maps increase linearly with the growth rate K.

For image enhancements in FCNs, encoding and decoding blocks are required to change feature maps' size, making the concatenation of feature maps unfeasible across different feature map size blocks. Hence particular encoding and decoding blocks are used to solve this problem. A dense block contains multiple dense layers whose input and output feature maps are the same size. Each dense block has two design parameters: the number of layers L and the growth rate K for each layer. We consider the growth rate K a constant value for all the dense blocks in our work. Here the encoding block typically is half the feature map size, whereas the decoding block doubles the feature map size. Both two blocks reduce the number of feature maps to half. Figure 2(a) shows the complete FCNs with the DenseED (SRDenseED) ML model used to generate the SR images using a small training dataset. Dense blocks, encoding blocks, and decoding blocks are marked with different colors as shown in Fig. 2(a). In this work, we set the growth rate to 16, the number of dense blocks to 3, and the number of dense layers in the first, second, and third dense blocks are 3, 6, and 3, respectively.

2.4 *Dataset Creation*

To show the trained ML model's performance, careful selection of the training dataset is essential. In this paper, two different datasets are used to demonstrate our approach. First, the W2S dataset (Widefield2SIM), which includes experimentally captured DL images (using widefield microscopy) and target images (using SIM microscopy).[69] Second, the BPAE dataset, which includes experimentally captured DL images (using custom-built multi-photon fluorescence microscopy[70]) and computationally generated target images (using SRRF technique[23]).

The W2S dataset includes 120 field of view (FOV) widefield DL fluorescence microscopy images (low-resolution: LR) and corresponding 120 FOV SIM images (HR). These experimental images are captured with two different fluorescence microscopy (widefield for LR images and SIM for HR images) and cells are real biological samples, namely, human cells.[69] In each FOV, three different channels (488, 561, and 640 nm) are recorded, and we consider them as individual gray-scale images during the training and inference stages. 400 images of the same FOV are captured and averaged to generate a noise-free DL image. Each image has a size of 512×512 pixels divided into four chunks of 256×256 pixels. Each FOV corresponds to $51.2\ \mu\text{m} \times 51.2\ \mu\text{m}$ (where each pixel size is 100 nm). Before the training process, all the images

in the training dataset are normalized, and normalization is explained in Sec. 2.6. In the case of the W2S dataset, with noise-free (average of 400 images in the same FOV) DL images and noisy (no average of images in the same FOV) DL images as input to the training dataset. In each case, the target image is the experimentally captured SR image (SIM setup).[69]

In the BPAE dataset, the BPAE sample (Invitrogen FluoCells slide #1, F36924 contains Nuclei, F-actin, and Mitochondria) was imaged with our custom-built two-photon fluorescence microscopy system[70] that provides DL images as input of the training dataset. The custom setup consists of an objective lens with 40x magnification (0.8 numerical aperture and 3.5 mm working distance). The two-photon excitation wavelength is 800 nm (for the one-photon system, the excitation wavelength is 400 nm), sample power is six mW, pixel width is 200 nm, pixel dwell-time, 12 μs, and the emission wavelength filter is from 300-700 nm. We used a photomultiplier tube (PMT) to convert the emission photons to current, followed by the transconductance amplifier (TA) to convert them to voltage. A total of 16 FOVs of the BPAE sample were captured, where each FOV consists of 50 DL images, and each image has a size of 256×256 pixels. The images in the 8'th FOV are used as the test dataset, and the remaining 1 to 7 FOVs and 9 to 16 FOVs data are used as the training dataset. Hence the training dataset size is 15 FOVs. We used the SRRF technique[23] to generate SR target images from the DL images. Fifty images of the same FOV are captured and averaged to create a noise-free DL image. Each image has a size of 256×256 pixels divided into four chunks of 128×128 pixels. Before the training, all the images in the training dataset are normalized, and normalization is explained in Sec. 2.6. More details of the SRRF are provided in the results section (please see Sec. 3.2). In addition, this BPAE dataset is provided as open source to validate the performance of the estimated SR images when trained with small datasets. More details about the dataset are provided in the Code and Data Sec. 4.

In this study, we show the effect of the SRDenseED method in FCNs using both W2S and the BPAE datasets.

2.5 *Hyperparameters*

Hyper-parameter search is a critical step in deep learning for quick and accurate results, primarily problem-specific and empirical. Typical hyper-parameters in FCN architecture are batch size, optimizer, and learning rate and are carefully tuned for achieving the best fluorescence microscopy image SR performance. The batch size used in the training stage is set to 3. The "Adam" gradient descent algorithm[71] is used to optimize the loss function between the estimated and target SR images during training. The initial learning rate is set to 3E-3, and weight decay is used to reduce the over-fitting problem to 3E-4. In addition, these parameters are fixed for all ML models: the number of feature maps in the first convolution layer is set to 48, the number of output feature maps is set to 16 (k-value) in every dense block, and the number of epochs is set to 400 such that the loss function reaches a stable point, the number of dense blocks to 3 and the number of dense layers in first, second, and third dense block are 3, 6, and 3, respectively. The training time varies with the training dataset size, and for the small dataset (for 90 FOVs), the training time is less than 4 hrs on a single Nvidia 1080-ti GPU. The number of parameters (kernel weights) for simple FCN (U-Net with three encoders and three decoders) architecture and FCN with three DenseED blocks is 286,704 and 237,204, respectively. More details about the ML model architectures can be found in the Code section.

2.6 *Data Processing*

Typically, biomedical images are too large to fit on a single GPU. Hence images are divided (input and target) into smaller patches when training the ML models. Normalization is applied as part of the pre-processing step to each image before passing it to the ML models (both simple FCNs and FCNs with the SRDenseED ML model). The input to the ML model is an image (I) that is linearly normalized by dividing with the maximum intensity value (here, the maximum value is 255 since images are 8-bit) and subtracting 0.5. Hence, all the pixel values passed through the ML model are always normalized (I_{norm}) and lie between -0.5 and 0.5 ($I_{\text{norm}} = I/255 - 0.5$). In addition, the target SR images are normalized the same as DL images,

and the pixel values lie between −0.5 and 0.5. As part of the post-processing, the output (O_{norm}) from the ML models is post-processed using the de-normalization step using this equation ($O_{\text{denorm}} = (O_{\text{norm}} + 0.5) * 255$). Finally, the estimated SR images are converted to 8-bit images to match the input (DL) and target (SR) image format.

2.7 *Forward Modeling in Super-Resolution Imaging*

In the literature on the computer vision or ML, HR images are taken from a high-quality instrument, which is typically expensive. The high-quality instrument provides minimal artifacts such as better resolution (better point spread function (PSF)) and low noise in the HR images, in this case, low-resolution (LR) images are generated using forward modeling and given in $I_{\text{LR}} = (I_{\text{HR}} * \text{PSF} + n)$, where I_{LR} is the LR image derived from HR image, I_{HR} is the HR image captured using an expensive instrument, PSF is the point spread function to generate LR image from HR image, $*$ is the convolution operation, n additive white gaussian noise with zero mean and σ standard deviation $N(0, \sigma)$. Hence, this generation method of LR images provides a blur due to the convolution of PSF, which is a 2D-Gaussian function. In this case, the ML model works as an inverse problem to detect the HR image from the LR images (which is an alternative to a conventional iterative deconvolution method[72–75]). Other research areas use SR in the context to upscale the low-resolution image from N × N image to MN × MN, where M is the scaling factor, typically, M is either 2, 3, or 4. Hence, the forward modeling is given by $I_{\text{LR}} = (I_{\text{HR}} * \text{PSF})$, where I_{LR} is the LR (down-sampled) image of size N × N, I_{HR} is the HR (upsampled) image of size MN × MN, *PSF* is the Gaussian function to downsample the image, and $*$ is the convolution operation. In this case, the ML model works as an inverse problem to detect the upsampled/up-scaled (HR) image from the down-sampled/down-scaled (LR) images. In contrast, in the case of optical microscopy, the low-resolution images are captured using an instrument that cannot separate close-by cells/samples.[76] Typically, this instrument is low in cost with limited resolution. Hence, the low-resolution images in this field are called "DL images." Also, the HR images are captured using an expensive instrument/technique that provides HR (which can separate the cells), and HR images are called "SR images." Because both the DL and SR are captured using two different instruments, adequate data processing is required to show that both images indicate the same FOV. Hence, in our paper, the DL and SR images are from two instruments with different PSF values. Forward modeling is given as $I_{\text{DL}} = I_{\text{original}} * \text{PSF}_{\text{DL}}$, $I_{\text{SR}} = I_{\text{original}} * \text{PSF}_{\text{SR}}$ where I_{original} is the true object need to image (cells or structure under a microscope); I_{DL} and I_{SR} are DL and SR images, respectively, when the I_{original} is captured with two different systems with PSF values as PSF_{DL} and PSF_{SR}, respectively; and $*$ indicates convolution operation. In this case, the ML model works as an inverse problem to detect the SR images from the DL images. For example, in the W2S dataset, the DL and SR images are captured using wide-field and SIM microscopy systems, and each instrument has a different PSF function. More details about the DL and SR images in the W2S dataset, including image acquisition systems, are provided in the original W2S paper.[69] Finally, in the BPAE dataset, only DL images are captured using our custom-built fluorescence lifetime imaging microscopy (FLIM) system,[70] and corresponding SR images are generated using a computation method called "SRRF".[23] More details about the BPAE datasets are provided in Sec. 3.2.

2.8 *Evaluation Metrics*

Several metrics are used to evaluate the estimated SR images compared with the target SR images. These metrics include structural similarity index measurement (SSIM),[77] PSNR,[58] mean square error (MSE/L2 norm), mean absolute error (MAE/L1 norm), resolution scaled Pearson's correlation coefficient,[78] resolution scaled error,[78] and Fourier ring coefficient (FRC), which measures the close matching (in structures, brightness) of the estimated SR images compared to target SR images.[78] The smaller value of FRC indicates a better SR image matching the target SR image,[78] with the value of 1 perfectly matching the target SR image. The SSIM and PSNR are the most common metrics to quantify the estimation of SR images.[58] To quantitatively evaluate the estimated SR images containing similar image features as the target SR image, we calculate the SSIM between the two. SSIM compares luminance, brightness, and contrast values as a

function of position[77] and measure the similarity between two images on a scale of 0 to 1, with 1 being perfect fidelity. In addition, we evaluate the PSNR of the estimated image relative to a target SR image. PSNR is the measure of MSE between two images normalized to the peak value in an image so that MSE between images with different bit depths or signal levels can be compared. PSNR of a given (X) with reference to ground truth image (Y) in the same FOV is defined as $\mathrm{PSNR}(X,Y) = 10\log(\frac{\max(Y)^2}{\mathrm{MSE}(X,Y)})$, where $\mathrm{MSE}(X,Y) = \frac{1}{N}\sum_{n=1}^{N}(X_n - Y_n)^2$ is the average MSE of X and Y with N pixels. The highest SSIM and PSNR represent the most accurate estimation of the SR image, similar to the target SR image. Hence, this paper evaluates the estimated SR images using SSIM and PSNR metrics.

3 Experimental Results and Discussion

3.1 *SRDenseED with Experimental SR Techniques*

This section shows the training and prediction results (including 30 FOVs) with and without the SRDenseED method in FCN architecture trained using experimentally captured W2S dataset with training dataset size of 5 FOVs, 15 FOVs, 30 FOVs, 45 FOVs, 60 FOVs, 75 FOVs, and 90 FOVs. An estimated SR image from the test dataset validates the trained ML model's accuracy from a DL image during the testing phase. For the comparison purpose, we considered output from the joint denoising and SR (JDSR) results from the original W2S paper[69] that provided the W2S dataset. In this experiment, initially, we choose noise-free diffraction images (see Sec. 2.4) that have high PSNR values as part of the training dataset since the noise in the experimental images degrades the performance of the trained ML models. Later in this section, using noisy diffraction images (see Sec. 2.4) that have low PSNR values as training dataset results are illustrated. In the following experiments, a U-Net architecture[39] with three encoder and decoder layers indicated as simple FCNs. Similarly, in the SRDenseED method, we have selected DenseED(3,6,3) ML model as FCNs with DenseED blocks, where the number of dense layers in the first, second, and third dense blocks are 3, 6, and 3, respectively.

3.1.1 *Training performance using high PSNR W2S dataset*

For the first part, the ML training dataset includes the noise-free (high PSNR) DL images as input and SIM SR images as a target, respectively.

First, we train a simple FCN architecture similar to the U-Net[39] ML model, consisting of three encoders followed by three decoder blocks with the same small dataset. Later, we train the SRDenseED ML models with the same small dataset. The SRDenseED ML model diagram is shown in Fig. 2(a). Different DenseED models' performance can be checked by changing the number of dense blocks and dense layers in each dense block. We start by verifying the ML model's performance with three dense blocks but variable dense layers in each dense block. In this case, the SRDenseED method includes 3, 6, and 3 dense layers in the three dense blocks, respectively. In addition, the non-linear activation layer is set to ReLU, the loss function is the MSE loss between the estimated and target SR images, the learning rate is set to 0.003, and the weight decay that is used to regularize the weights without over-fitting the model is always set to $\frac{1}{10^{\text{th}}}$ of the learning rate. We perform testing on a test dataset (including 30 FOVs) of images that the model never sees during the training step. The hyperparameters are set to the same for simple FCNs and SRDenseED methods. Figure 3 shows the quantitative results of the noise-free DL images as input and SIM images as target images in the training datasets. The SRDenseED model outperforms PSNR compared to conventional FCN networks, and this trend can be seen in the training dataset size. From Fig. 3, especially at the small training dataset size (15 FOVs), there is an average improvement of 1.35 dB in PSNR when using the SRDenseED ML model.

In addition, Fig. 4 shows the quantitative results of PSNR and SSIM over the test dataset (includes 30 FOVs of 3 channels) of estimated SR images from the noise-free DL images. Based on the quantitative results of PSNR and SSIM, the SRDenseED ML models can provide better and more accurate SR images than simple FCN networks when trained using a small training dataset. Even training with a small training dataset (15 FOVs) SRDenseED method can generate

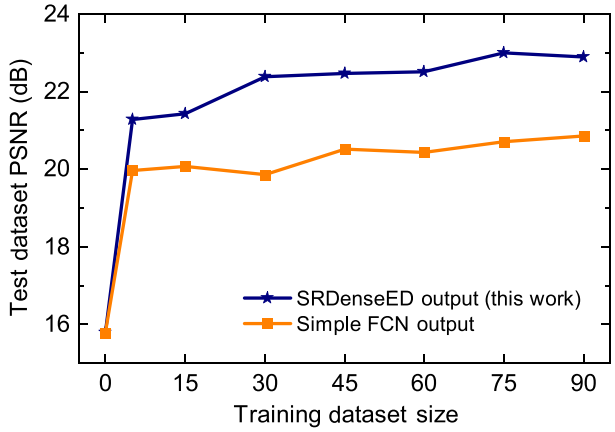

Fig. 3 W2S dataset average PSNR of the test dataset (includes 30 FOVs) versus training dataset size using simple FCNs and SRDenseED networks. Here, the ML models are trained using the high PSNR noise-free DL images.

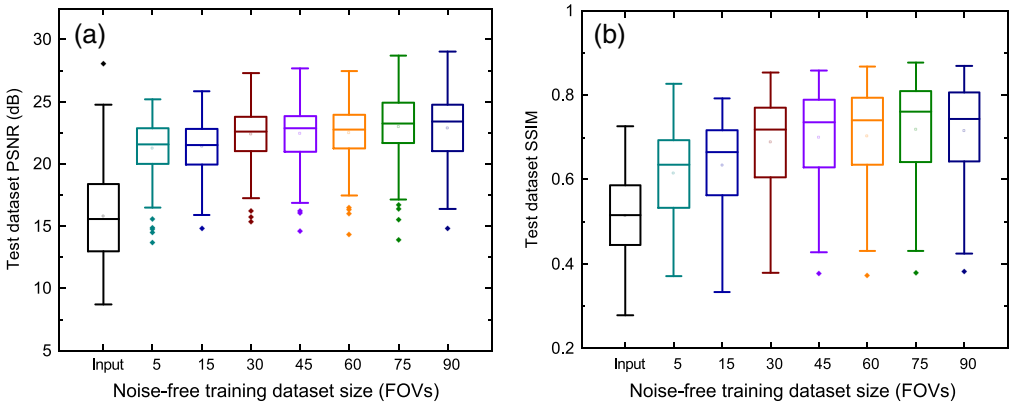

Fig. 4 W2S dataset PSNR: (a) and SSIM (b) versus training dataset size using SRDenseED networks trained using the high PSNR noise-free DL images.

SR images with an average PSNR improvement of 1.35 dB, and this SRDenseED method is helpful in biomedical imaging (x-ray and MRI imaging) to generate SR images. In the SRDenseED method, the PSNR improvement, when trained with a 90 FOVs dataset, is only 0.71 dB more (a difference of 2.02 dB PSNR improvement from 90 FOVs and 1.31 dB from 15 FOVs training data) when compared with simple FCNs. Table 2 shows the estimated SR images' average PSNR when trained with high PSNR noise-free DL images. Here, the SRDenseED method outperformed compared to simple FCNs when trained with a small dataset and confirmed the technique works for application-specific imaging.

Figure 5(a) shows one of the DL noise-free images drawn randomly in the test dataset (10'th FOV, channel 1) as the qualitative representation. Figure 5(b) shows the estimated SR image from the pretrained ML models given in Ref. 69 and is unable to show the clear structures in the estimated SR image. Figure 5(c) shows the estimated SR image within the same FOV when trained with the SRDenseED ML model with a training dataset of 30 FOVs, and this image has better PSNR compared to the raw DL image. Figure 5(d) shows the target SR image captured using the SIM setup and in the same testing FOV. From Fig. 5, the PSNR of the noise-free input image and estimated SR image using the JDSR method[69] and estimated SR image using the SRDenseED method (trained with 15 FOVs) are 19.22 and 17.84, 22.45 dB, respectively. In this case, there is a PSNR improvement of -−1.38 dB, and 3.23 dB of the randomly selected test image using the JDSR method[69] and our SRDenseED methods, respectively. Similarly, the SSIM values of the noise-free input image and estimated SR image using the JDSR method[69] and

Table 2 Quantitative comparison of average PSNR (dB) on test dataset (includes 30 FOVs) of simple FCNs and SRDenseED methods with different training dataset sizes. Here, the ML models are trained using the high PSNR noise-free DL images and ΔPSNR = PSNR from SRDenseED method - PSNR from simple FCNs.

Training dataset size (noise-free)	Simple FCN [PSNR (dB)]	SRDenseED [PSNR (dB)]	ΔPSNR
Input	15.78	15.78	N/A
5	19.96	21.27	1.31
15	20.08	21.43	1.35
30	19.86	22.37	2.51
45	20.52	22.47	1.95
60	20.44	22.51	2.07
75	20.72	22.99	2.28
90	20.86	22.89	2.02

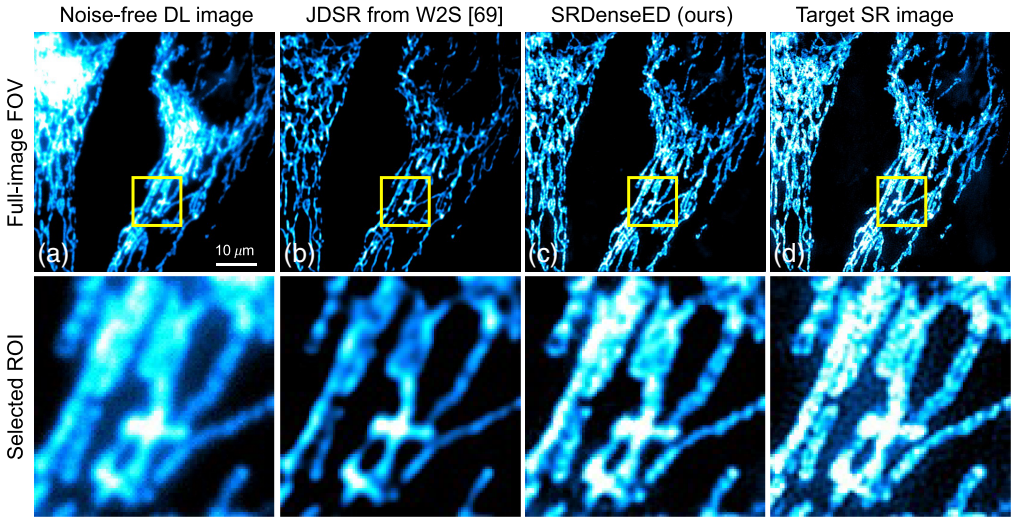

Fig. 5 Sample from the W2S dataset: (a) noise-free DL image, (b) estimated SR image from JDSR method,[69] (c) estimated SR image from the SRDenseED (ours) ML model (test image is taken from 10'th FOV, channel 1), and (d) is the experimentally captured target SR using SIM microscopy. Here the input sample is a DL noise-free image. The top row indicates the full frame (of size 512×512), and the bottom row indicates the region of interest (ROI: marked in the yellow square of size 100×100) from the respective top row images. Scale bar: 10 μm.

estimated SR image using the SRDenseED method (trained with 15 FOVs) are 0.64, 0.63, and 0.82, respectively. In addition, the calculated unscaled FRC value[78] of the noise-free input image and estimated SR image using the JDSR method[69] and estimated SR image using the SRDenseED method (trained with 15 FOVs) are 3.95, 4.15 and 3.77, respectively. From all quantitative metrics, our SRDenseED method provides better SR images than the JDSR method.

3.1.2 *Training performance using low PSNR W2S dataset*

However, obtaining noise-free images in real-time measurements is difficult (when dynamic processes are included) and time-consuming (needing multiple averages with the same

FOV). Hence, the following results show the performance of our demonstrated SRDenseED ML model when trained on DL noisy images.

The response of the trained ML models using a small dataset with simple FCNs and SRDenseED ML models are analyzed with noisy DL images as input. Figure 6 shows the quantitative results of the noisy DL images as input and SIM images as target images in the training datasets. The SRDenseED model outperforms PSNR compared to simple FCNs, and this trend can be seen over the training dataset size (even though the images are noisy and DL). From Fig. 6, especially at the small training dataset size (15 FOVs), there is an average improvement of 0.92 dB in PSNR when using the SRDenseED ML model. In addition, Fig. 7 shows the quantitative results of PSNR and SSIM over the test dataset (includes 30 FOVs of 3 channels). Based on the quantitative results of PSNR and SSIM, the SRDenseED ML models can provide better and more accurate SR images when trained with a small training dataset. Table 3 shows the estimated SR image quantitative metrics when trained with low PSNR noisy DL images. Again, the SRDenseED method outperformed compared to simple FCNs when trained with a small dataset and confirmed the technique works for application-specific imaging. Here the results are not meant to be used as any generalized SR images instead the results are meant for the application-specific imaging modalities/configurations.

Figure 8(a) shows one of the DL noisy images in a test dataset (10'th FOV, channel 1). Figure 8(b) shows the estimated SR image from the pre-trained ML models given in Ref. 69 and is unable to show the clear structures in the estimated SR image. Figure 8(c) shows the

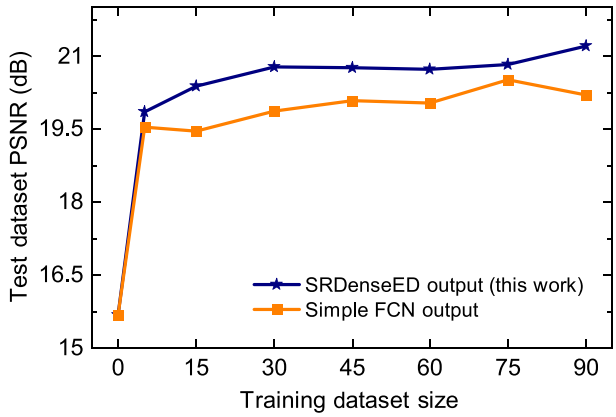

Fig. 6 W2S dataset average PSNR of the test dataset (includes 30 FOVs) versus training dataset size using simple FCNs and SRDenseED networks. Here, the ML models are trained using the low PSNR noisy DL images.

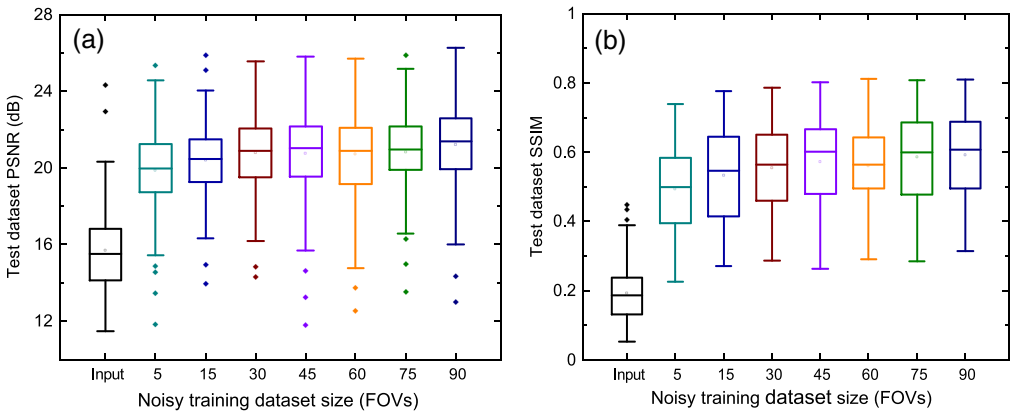

Fig. 7 W2S dataset PSNR (a) and SSIM (b) versus training dataset size using SRDenseED networks trained using the low PSNR noisy DL images.

Table 3 Quantitative comparison of average PSNR (dB) on test dataset (includes 30 FOVs) of simple FCNs and SRDenseED methods trained with different noisy dataset sizes. Here, the ML models are trained using the low PSNR noisy DL images and ΔPSNR = PSNR from SRDenseED method - PSNR from simple FCNs.

Training dataset size (noisy)	Simple FCN [PSNR (dB)]	SRDenseED [PSNR (dB)]	ΔPSNR
Input	15.67	15.67	N/A
5	19.54	19.86	0.31
15	19.45	20.38	0.92
30	19.86	20.78	0.91
45	20.09	20.76	0.68
60	20.04	20.72	0.68
75	20.52	20.84	0.32
90	20.19	21.20	1.01

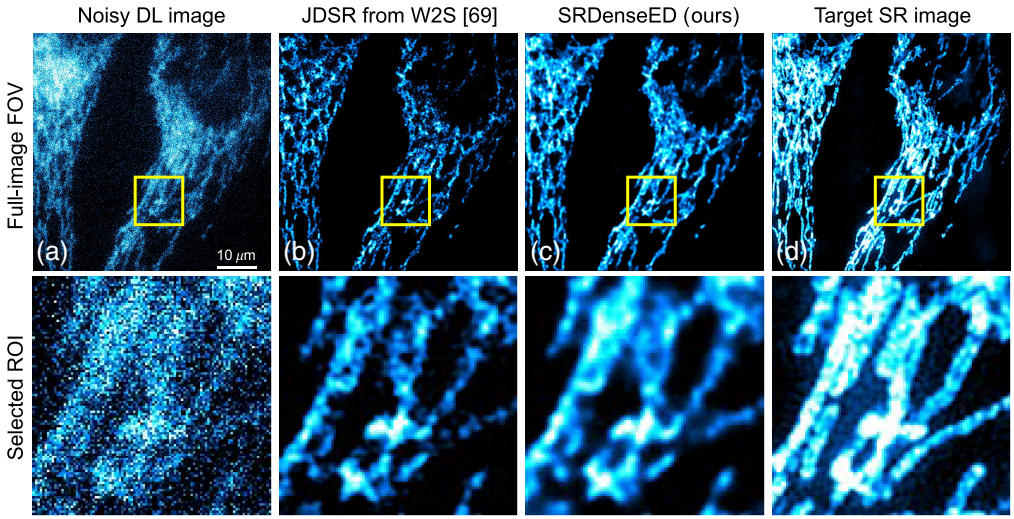

Fig. 8 Sample from the W2S dataset: (a) noisy DL image, (b) estimated SR image from JDSR method,[69] (c) estimated SR image from the SRDenseED (ours) ML model (test image is taken from 10'th FOV, channel 1), and (d) is the experimentally captured target SR using SIM microscopy. Here the input sample is a DL noisy image. The top row indicates the full frame (of size 512×512), and the bottom row indicates the region of interest (ROI: marked in the yellow square of size 100×100) from the respective top row images. Scale bar: 10 μm.

estimated SR image within the same FOV when trained with the SRDenseED ML model, and this image has better PSNR compared to the raw DL image. Figure 8(d) shows the target SR image captured by SIM setup within the same testing FOV. From Fig. 8, the PSNR of the noisy input image and estimated SR image using the JDSR method in W2S paper, and the estimated SR image using the SRDenseED method (trained with 15 FOVs) are 16.72 and 17.41, 20.11 dB, respectively. Hence, in this case, a PSNR improvement of 0.69 and 3.39 dB of the randomly selected test image using the JDSR method and our SRDenseED methods, respectively. Similarly, the SSIM values of the noisy input image and estimated SR image using the JDSR method and estimated SR image using the SRDenseED method (trained with 15 FOVs) are 0.19, 0.59, and 0.69, respectively. As expected, the JDSR method improved PSNR when the input image is noisy compared to noise-free, where the significant contribution is from

the image denoising step. In addition, the calculated unscaled FRC value[78] of the noise-free input image and estimated SR image using the JDSR method[69] and estimated SR image using the SRDenseED method (trained with 15 FOVs) are 5.80, 5.59 and 5.43, respectively. From all quantitative metrics, our SRDenseED method provides better SR images than the JDSR method. We observe that our SRDenseED method (trained with 15 FOVs) provides accurate SR images by providing an average PSNR improvement of 5.65 (21.43-15.78, see Table. 2) dB and 4.71 (20.38 to 15.67, see Table. 3) dB in noise-free DL images as input and noisy DL images as input, respectively. In addition, compared to simple FCN architecture, our SRDenseED method (trained with 15 FOVs) provided an average PSNR improvement of 1.35 and 0.92 dB, in the case of noise-free and noisy DL input images, respectively.

3.2 *SRDenseED with Computational SR Techniques*

Generating SR images requires an additional experimental setup, which is expensive, and the research labs may not have this setup. However, experimental DL image generation is a typical setup, and SR images can be generated using computational methods. For example, SRRF[23] is a computational method to generate SR images within the same FOV from multiple DL images (captured with different time instances). In this section, we captured experimental DL images of BPAE samples (Invitrogen FluoCells slide#1 F36924, mitochondria labeled with MitoTracker Red CMXRos, F-actin labeled with Alexa Fluor 488 phalloidin, nuclei labeled with DAPI) using our custom-built two-photon fluorescence microscopy system.[70] In this step, the captured images include noise. The custom setup consists of an objective lens with 40× magnification (0.8 numerical aperture and 3.5 mm working distance). The two-photon excitation wavelength is 800 nm (for the one-photon system, the excitation wavelength is 400 nm), sample power is six mW, pixel width is 200 nm, pixel dwell-time, 12 μs, and the emission wavelength filter is from 300-700 nm. In our imaging system, all the fluorophores-labeled organelles are excited together using a single excitation wavelength (in this case, 800 nm) and get the collective emission together using a bandpass filter (300-700 nm) that shows all the fluorophores together in the fluorescence intensity image. We used a PMT to convert the emission photons to current, followed by the TA to convert them to voltage. More details about the setup can be found in Ref. 70. A total of 16 different FOVs (small training dataset) of the BPAE sample are captured using our system, where each FOV consists of 50 raw images, and each image has a size of 256×256. The target SR images are generated using the SRRF technique. SRRF method performs two steps,[23] i.e., spatial and temporal steps, to generate SR images. Spatial SRRF estimates and maps the most likely positions of the molecules, followed by temporal SRRF to improve the resolution of the final SR SRRF image using spatial resolution step statistics. The center of the fluorophores is estimated and mapped to a "radiality" map in simple terms. SRRF method provides the SR image in the subpixel range (with a magnification of 5 times by default) and reshapes it (using bilinear interpolation) to the raw image dimension 256×256. Note: SRRF can provide inaccurate target results if the parameters are not set correctly during this target generation stage and more details can be found in Ref. 23.

The experimentally captured DL images (also noisy) and SRRF-generated images are used as the input and target of the small training dataset, respectively. Normalization is applied to each image before passing it to the FCNs with the SRDenseED ML model. The image normalization is conducted by dividing the maximum value in the data type (here, the maximum value is 255) and subtracting 0.5. Hence, all the pixel values passed through the ML model are always normalized and lie between -0.5 and 0.5. The images generated in the 8'th FOV are used as the test dataset, and the remaining 1 to 7 FOVs and 9 to 16 FOVs data are used as the training dataset. The training dataset consists of 15 FOVs, called a "small training dataset". Here the input is a 16-bit grayscale channel.

The quantitative and qualitative results from the test dataset are shown in Fig. 9 after training the ML model using the SRDenseED method. Figure 9(a) shows the experimentally captured (using a custom two-photon FLIM system) nosy DL image of the BPAE sample cell, and Fig. 9(b) indicates a noise-free DL image within the same FOV. Similarly, Fig. 9(a) shows the target SR image generated using the computation SRRF method from multiple DL images. Figure 9(d) shows the estimated SR images from the DenseED (3,6,3) configuration ML model.

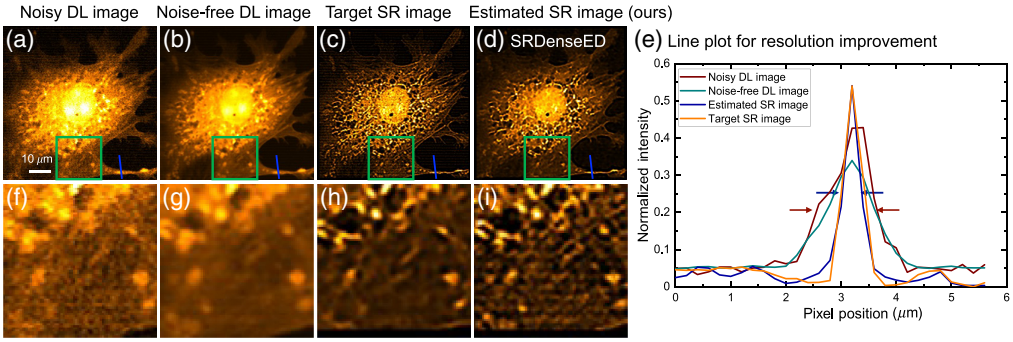

Fig. 9 BPAE sample DL image: (a) acquired with our custom-built two-photon microscope;[70] (b) noise-free image (averaged within the same FOV), and (c) the target SR image generated by SRRF method. (d) Estimated SR image using the trained ML model with dense blocks in FCN. The resolution improvement in panel (e) includes the line plots of images shown in (a, b, c, and d in blue color), respectively, with markers in wine and blue colors indicating the FWHM of DL and estimated SR images. The top row (a, b, c, and d) indicates the full frame (of size 256×256), and the bottom row (f, g, h, and i) indicates the region of interest (ROI: marked in the green square of size 75×75) from the respective top row images. Pixel width, 200 nm; pixel dwell-time, 12 μs; and excitation power, 6 mW.

The estimated SR image accurately estimates submicron features (mitochondria) and is comparable with the target image. Averaging more images within the same FOV improves the PSNR (from 21.24 to 21.89 dB) but is unable to find the sub-micron SR structures [see Fig. 9(b)]. The PSNR values of the noisy DL image, noise-free DL image, and the estimated SRDenseED image are 21.24, 21.89, and 24.73 dB, as shown in Figs. 9(a), 9(b), 9(d) respectively (with respect to target image as shown in Fig. 9(c)). Hence, there is a 3.49 dB improvement in PSNR from the trained SRDenseED method compared to the DL noisy test image. The improvement in the PSNR is due to the identification of small features, and the estimated image closely matches the target image. Hence, the trained ML model with the SRDenseED method can achieve SR from the DL images even though the training dataset size is limited. In addition, Fig. 9(e) provides the qualitative and quantitative metrics on the estimated SR image with a marked region and corresponding line plots of the trained ML model using the DenseED model with three dense blocks and 3,6,3 are the dense layers in each dense block respectively. The full width at half maximum (FWHM) for the DL and estimated SR images are $\approx$1.2 μm and $\approx$0.6 μm, respectively, which shows at least 2$\times$ resolution improvement. The top row in Figs. 9(a), 9(b), 9(c), and 9(d) indicates the full frame (of size 256×256), and the bottom row in Figs. 9(f), 9(g), 9(h), and 9(i) indicates the region of interest (ROI: marked in the green square of size 75×75) from the respective full-FOV images.

Additional qualitative and quantitative results with different DenseED configurations are provided in the GitHub repository (https://github.com/ND-HowardGroup/Application-Specific-Super-resolution.git) on the estimated SR images of the trained ML models. Variations of the estimated SR images PSNR and SSIM are shown, including variation in the learning rate, nonlinear activation function, sample dataset size, and the loss function as the mixed loss of MSE loss and SSIM loss to optimize the MSE loss and SSIM loss simultaneously in FCN architecture. Also, these demonstrated DenseED blocks could be applied to estimate SR images from resolution-limited images with GAN architecture with retraining (more results are shown in the GitHub repository for the W2S dataset and BPAE dataset).

If the test dataset is entirely different from the training dataset, generated SR images might have some artifacts in the output.[79] Also, if the target generation has some artifacts, then the estimated SR using this trained dataset will also have artifacts. Consider the BPAE dataset, where the target image is generated using the SRRF computational method, which can provide SR images with artifacts if the computational parameters are not appropriately set.[23] In this case, the inaccuracy of the ground truth image will affect the performance of the ML model. In addition, the generalization capability of the trained ML model is limited when trained using a small training dataset that might also include artifacts such as hallucination effects, blur, and other cells

to display where the estimated SR image has more details than the ground-truth SR image. Hence it is always recommended to check if the generated SR images have any hallucinations or artifacts using the existing quantitative metrics such as PSNR, SSIM, and FRC, as mentioned in Sec. 2.8. To reduce artifacts, additional steps are required when generating SR images, such as using residual layers.[80]

Finally, the DenseED block in ML model architectures helps to generate SR images when the ML model is trained with a small dataset. The performance improvement depends on optimizing other hyper-parameters and parameters of the network, including learning rate, non-linear activation, loss function, and weight decay, on regularizing the over-fitting. For the SRDenseED method, the number of dense blocks and dense layers are also significant in each dense block. Clearly, from the above experiments, the SRDenseED method provides accurate results compared to simple FCNs.

4 Conclusion

ML models have been previously demonstrated to generate SR from DL images. Such approaches require thousands of training images, which is prohibitively difficult in many biological samples. We showed the FCN architectures with the SRDenseED method, including Dense Encoder-Decoder blocks, to train SR FCNs using a small training dataset. Our results show an accurate estimation of SR images with denseED blocks in conventional ML models [see Figs. 5(b), 8(b), and 9(d)]. We showed the estimated SR image PSNR results and compared them with the target SR images in the case of both experimentally captured SIM setup (as shown in Sec. 3.1) and computationally generated with the SRRF method (as shown in Sec. 3.2), with PSNR improvement of 3.66 dB (in case of noise-free DL images) and 3.49 dB, respectively. Our primary focus was to demonstrate the new ML method (our SRDenseED method) capable of providing application-specific SR (for example, fluorescence microscopy) images when trained using a small training dataset. In addition, we used the SRRF method for the target generation since it is computational and easy to use. Besides, our demonstrated model can work with other SR target generation methods like STED/STORM/PALM/SIM. While we evaluated the technique on SR fluorescence microscopy, this approach shows promise for an extension to other deep-learning-based image enhancements (e.g., image denoising networks,[10,81] image SR,[36,43,82–84] image segmentation networks,[39] and other imaging modalities like x-ray[85–87] and MRI imaging[88]).

Disclosures

The authors declare no conflicts of interest.

Acknowledgments

The authors further acknowledge the Notre Dame Center for Research Computing (CRC) for providing the Nvidia GeForce GTX 1080-Ti GPU resources for training the SR Fluorescence Microscopy dataset in Pytorch. This material is based upon work supported by the National Science Foundation (NSF) (Grant No. CBET-1554516) and the NSF Industry-University Cooperative Research Program (IUCRC) Center for Bioanalytical Metrology (CBM) (Grant No. 1916601).

Code and Data Availability

The results used in this manuscript are open-source and can be accessed via GitHub, provided at https://github.com/ND-HowardGroup/Application-Specific-Super-resolution.git. The code and other resources are provided for public access to obtain the results reported in the manuscript. The SR fluorescence microscopy dataset, including the DL and super-resolution images used to train the DenseED ML models, are provided open-source at https://curate.nd.edu/show/5h73pv66g4s and in Mannam's PhD thesis.[89]

References

1. J. D. Bronzino, *Biomedical Engineering Handbook*, Vol. **2**, Springer Science & Business Media (2000).
2. A. G. Webb, *Introduction to Biomedical Imaging*, John Wiley & Sons (2017).
3. R. Liang, *Optical Design for Biomedical Imaging*, SPIE Press, Bellingham, Washington (2011).
4. S. W. Hell and J. Wichmann, "Breaking the diffraction resolution limit by stimulated emission: stimulated-emission-depletion fluorescence microscopy," *Opt. Lett.* **19**(11), 780–782 (1994).
5. M. G. Gustafsson, "Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy," *J. Microsc.* **198**(2), 82–87 (2000).
6. M. J. Rust, M. Bates, and X. Zhuang, "Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM)," *Nat. Methods* **3**(10), 793 (2006).
7. E. Betzig et al., "Imaging intracellular fluorescent proteins at nanometer resolution," *Science* **313**(5793), 1642–1645 (2006).
8. D. Nie et al., "Medical image synthesis with deep convolutional adversarial networks," *IEEE Trans. Biomed. Eng.* **65**(12), 2720–2730 (2018).
9. E. Nehme et al., "Deep-STORM: super-resolution single-molecule microscopy by deep learning," *Optica* **5**(4), 458–464 (2018).
10. V. Mannam et al., "Real-time image denoising of mixed Poisson–Gaussian noise in fluorescence microscopy images using ImageJ," *Optica* **9**(4), 335–345 (2022).
11. M. D. Robinson et al., "New applications of super-resolution in medical imaging," in *Super-Resolution Imaging*, P. Milanfar, Ed., pp. 384–412, CRC Press (2010).
12. X. Yang et al., "Low-dose X-ray tomography through a deep convolutional neural network," *Sci Rep* **8**(1), 2575 (2018).
13. K. Umehara, J. Ota, and T. Ishida, "Application of super-resolution convolutional neural network for enhancing image resolution in chest CT," *J. Digital Imaging* **31**(4), 441–450 (2018).
14. J. Chun et al., "MRI super-resolution reconstruction for MRI-guided adaptive radiotherapy using cascaded deep learning: in the presence of limited training data and unknown translation model," *Med. Phys.* **46**(9), 4148–4164 (2019).
15. G. Huang et al., "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 4700–4708 (2017).
16. J. W. Lichtman and J.-A. Conchello, "Fluorescence microscopy," *Nat. Methods* **2**(12), 910–919 (2005).
17. E. Abbe, "Contributions to the theory of the microscope and microscopic perception," *Arch. Microsc. Anato.* **9**(1), 413–468 (1873).
18. E. H. Rego et al., "Nonlinear structured-illumination microscopy with a photoswitchable protein reveals cellular structures at 50-nm resolution," *Proc. Natl. Acad. Sci. U. S. A.* **109**(3), E135–E143 (2012).
19. M. G. Gustafsson, "Nonlinear structured-illumination microscopy: wide-field fluorescence imaging with theoretically unlimited resolution," *Proc. Natl. Acad. Sci. U. S. A.* **102**(37), 13081–13086 (2005).
20. K. Fujita et al., "High-resolution confocal microscopy by saturated excitation of fluorescence," *Phys. Rev. Lett.* **99**(22), 228105 (2007).
21. J. Humpolčková, A. Benda, and J. Enderlein, "Optical saturation as a versatile tool to enhance resolution in confocal microscopy," *Biophys. J.* **97**(9), 2623–2629 (2009).
22. Y. Zhang et al., "Generalized stepwise optical saturation enables super-resolution fluorescence lifetime imaging microscopy," *Biomed. Opt. Express* **9**(9), 4077–4093 (2018).
23. N. Gustafsson et al., "Fast live-cell conventional fluorophore nanoscopy with ImageJ through super-resolution radial fluctuations," *Nat. Commun.* **7**, 12471 (2016).
24. I. Goodfellow et al., *Deep Learning*, Vol. **1**, MIT Press, Cambridge (2016).
25. V. Mannam et al., "Machine learning for faster and smarter fluorescence lifetime imaging microscopy," *J. Phys.: Photonics* **2**(4), 042005 (2020).
26. V. Mannam et al., "Convolutional neural network denoising in fluorescence lifetime imaging microscopy (FLIM)," *Proc. SPIE* **11648**, 116481C (2021).

27. V. Mannam et al., "Low dosage 3D volume fluorescence microscopy imaging using compressive sensing," *Proc. SPIE* **11966**, 1196603 (2022).
28. S. Howard et al., "Packet compressed sensing imaging (PCSI): robust image transmission over noisy channels," https://arxiv.org/abs/2009.11455 (2020).
29. Y. Zhang et al., "A poisson-gaussian denoising dataset with real fluorescence microscopy images," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 11710–11718 (2019).
30. J. C. Waters, "Accuracy and precision in quantitative fluorescence microscopy," *J. Cell Biol.* **185**(7), 1135–1148 (2009).
31. Q. Feng et al., "When do gans replicate? On the choice of dataset size," in *Proc. IEEE/CVF Int. Conf. Comput. Vision*, pp. 6701–6710 (2021).
32. M. Hosseinzadeh et al., "Deep learning–assisted prostate cancer detection on bi-parametric MRI: minimum training data size requirements and effect of prior knowledge," *Eur. Radiol.* **32**(4), 2224–2234 (2022).
33. J. Brownlee, "Impact of dataset size on deep learning model skill and performance estimates," *Machine Learning Mastery* (2019), https://machinelearningmastery.com/impact-of-dataset-size-on-deep-learning-model-skill-and-performance-estimates/.
34. R. Baeza-Yates and Z. Liaghat, "Quality-efficiency trade-offs in machine learning for text processing," in *IEEE Int. Conf. BigData*, IEEE, pp. 897–904 (2017).
35. T. Linjordet and K. Balog, "Impact of training dataset size on neural answer selection models," *Lect. Notes Comput. Sci.* **11437**, 828–835 (2019).
36. V. Mannam et al., "Deep learning-based super-resolution fluorescence microscopy on small datasets," *Proc. SPIE* **11650**, 116500O (2021).
37. V. Mannam and A. Kazemi, "Performance analysis of semi-supervised learning in the small-data regime using VAEs," https://arxiv.org/abs/2002.12164 (2020).
38. V. Mannam et al., "Instant image denoising plugin for ImageJ using convolutional neural networks," in *Microsc. Histopathol. and Anal.*, Optical Society of America, p. MW2A–3 (2020).
39. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," *Lect. Notes Comput. Sci.* **9351**, 234–241 (2015).
40. Y. Zhang et al., "Residual dense network for image super-resolution," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 2472–2481 (2018).
41. A. Osokin et al., "GANs for biological image synthesis," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, pp. 2233–2242 (2017).
42. J. Chen et al., "Image blind denoising with generative adversarial network based noise modeling," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 3155–3164 (2018).
43. Y. Li et al., "Dlbi: deep learning guided bayesian inference for structure reconstruction of super-resolution fluorescence microscopy," *Bioinformatics* **34**(13), i284–i294 (2018).
44. K. Agarwal and R. Macháň, "Multiple signal classification algorithm for super-resolution fluorescence microscopy," *Nat. Commun.* **7**, 13752 (2016).
45. J. Liao et al., "Deep-learning-based methods for super-resolution fluorescence microscopy," *J. Innov. Opt. Health Sci.* 2230016 (2022).
46. H. Park et al., "Deep learning enables reference-free isotropic super-resolution for volumetric fluorescence microscopy," *Nat. Commun.* **13**, 3297 (2022).
47. B. Huang, et al., "Enhancing image resolution of confocal fluorescence microscopy with deep learning," *PhotoniX* **4**(1), 1–22 (2023).
48. Z. Cheng et al., "Light field super-resolution with zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 10010–10019 (2021).
49. J. W. Soh, S. Cho, and N. I. Cho, "Meta-transfer learning for zero-shot super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 3516–3525 (2020).
50. A. Shocher, N. Cohen, and M. Irani, ""zero-shot" super-resolution using deep internal learning," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 3118–3126 (2018).
51. Z. Wang, J. Chen, and S. C. Hoi, "Deep learning for image super-resolution: a survey," *IEEE Trans. Pattern Anal. Mach. Intell.* **43**(10), 3365–3387 (2020).

52. R. Tang et al., "Lightweight network with one-shot aggregation for image super-resolution," *J. Real-Time Image Process.* **18**(4), 1275–1284 (2021).
53. J. Cheng et al., ""one-shot" super-resolution via backward style transfer for fast high-resolution style transfer," *IEEE Signal Process. Lett.* **28**, 1485–1489 (2021).
54. W. Wei et al., "Boosting one-shot spectral super-resolution using transfer learning," *IEEE Trans. Comput. Imaging* **6**, 1459–1470 (2020).
55. M. Bevilacqua et al., "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Brit. Mach. Vision Conf.*, pp. 135.1–135.10 (2012).
56. D. Martin et al., "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. Eighth IEEE Int. Conf. Comput. Vision ICCV 2001*, IEEE, vol. **2**, pp. 416–423 (2001).
57. E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: dataset and study," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. Workshops (CVPR)*, pp. 126–135 (2017).
58. S. Ayas and M. Ekinci, "Microscopic image super resolution using deep convolutional neural networks," *Multimedia Tools Appl.* **79**(21), 15397–15415 (2020).
59. H. Wang et al., "Deep learning enables cross-modality super-resolution in fluorescence microscopy," *Nat. Methods* **16**(1), 103–110 (2019).
60. H. Zhang et al., "High-throughput, high-resolution deep learning microscopy based on registration-free generative adversarial network," *Biomed. Opt. Express* **10**(3), 1044–1063 (2019).
61. W. Ouyang et al., "Deep learning massively accelerates super-resolution localization microscopy," *Nat. Biotechnol.* **36**(5), 460–468 (2018).
62. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, pp. 3431–3440 (2015).
63. E. Kang, J. Min, and J. C. Ye, "A deep convolutional neural network using directional wavelets for low-dose X-ray CT reconstruction," *Med. Phys.* **44**(10), e360–e375 (2017).
64. X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. Fourteenth Int. Conf. Artif. Intell. and Stat., JMLR Workshop and Conf. Proc.*, pp. 315–323 (2011).
65. S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Int. Conf. Mach. Learn. (ICML)*, PMLR, pp. 448–456 (2015).
66. K. Zhang et al., "Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.* **26**(7), 3142–3155 (2017).
67. Y. Zhu and N. Zabaras, "Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification," *J. Comput. Phys.* **366**, 415–447 (2018).
68. S. Jégou et al., "The one hundred layers tiramisu: fully convolutional densenets for semantic segmentation," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit. Workshops (CVPR)*, pp. 11–19 (2017).
69. R. Zhou et al., "W2S: microscopy data with joint denoising and super-resolution for widefield to SIM mapping," *Lect. Notes Comput. Sci.* **12535**, 474–491 (2020).
70. Y. Zhang et al., "Instant FLIM enables 4D in vivo lifetime imaging of intact and injured zebrafish and mouse brains," *Optica* **8**, 885–897 (2021).
71. D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," https://arxiv.org/abs/1412.6980 (2014).
72. D. Fish et al., "Blind deconvolution by means of the richardson–Lucy algorithm," *JOSA A* **12**(1), 58–65 (1995).
73. V. Mannam, X. Yuan, and S. Howard, "Deconvolution of fluorescence lifetime imaging microscopy (FLIM)," *Proc. SPIE* **11965**, 1196508 (2022).
74. L.-H. Yeh, *Computational Fluorescence and Phase Super-Resolution Microscopy*, eScholarship, University of California (2019).
75. J. Luo et al., "Super-resolution structured illumination microscopy reconstruction using a least-squares solver," *Front. Phys.* **8**, 118 (2020).
76. K. Prakash et al., "Super-resolution microscopy: a brief history and new avenues," *Philos. Trans. R. Soc. A* **380**(2220), 20210110 (2022).

77. Z. Wang et al., "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.* **13**(4), 600–612 (2004).
78. R. F. Laine et al., "NanoJ: a high-performance open-source super-resolution microscopy toolbox," *J. Phys. D: Appl. Phys.* **52**(16), 163001 (2019).
79. X. Wang et al., "ESRGAN: enhanced super-resolution generative adversarial networks," in *Proc. Eur. Conf. Comput. Vision Workshops (ECCV)* (2018).
80. H. Nasrollahi, K. Farajzadeh, and V. Hoeini, et al., "Deep artifact-free residual network for single-image super-resolution," *Signal, Image Video Process.* **14**(2), 407–415 (2020).
81. M. Weigert et al., "Content-aware image restoration: pushing the limits of fluorescence microscopy," *Nat. Methods* **15**(12), 1090–1097 (2018).
82. W. Zhao et al., "Sparse deconvolution improves the resolution of live-cell super-resolution fluorescence microscopy," *Nat. Biotechnol.* **40**(4), 606–617 (2022).
83. S. S. Kaderuppan et al., "O-Net: a fast and precise deep-learning architecture for computational super-resolved phase-modulated optical microscopy," *Microsc. Microanal.* **28**, 1584–1598 (2022).
84. P. Zelger et al., "Three-dimensional localization microscopy using deep learning," *Opt. Express* **26**(25), 33166–33179 (2018).
85. E. E.-D. Hemdan, M. A. Shouman, and M. E. Karar, "CovidX-net: a framework of deep learning classifiers to diagnose Covid-19 in X-ray images," https://arxiv.org/abs/2003.11055 (2020).
86. R. Jain et al., "Deep learning based detection and analysis of COVID-19 on chest X-ray images," *Appl. Intell.* **51**(3), 1690–1700 (2021).
87. I. M. Baltruschat et al., "Comparison of deep learning approaches for multi-label chest X-ray classification," *Sci. Rep.* **9**, 7381 (2019).
88. J. Liu et al., "Applications of deep learning to MRI images: a survey," *Big Data Mining Anal.* **1**(1), 1–18 (2018).
89. V. Mannam, "Overcoming fundamental limits of three-dimensional in vivo fluorescence imaging using machine learning," University of Notre Dame (2022).

Varun Mannam received his bachelor of technology and master of technology degrees in electronics and communications engineering from the Bapatla Engineering College, Acharya Nagarjuna University, Guntur, Andhra Pradesh, India, and Indian Institute of Technology Kharagpur (IIT Kharagpur), West Bengal, India, in 2010 and 2012, respectively. From 2012 to 2017, he was a staff software engineer to develop wireless connectivity communication protocols (such as Bluetooth and WIFI) at National Instruments R&D Bangalore, India. He came to the University of Notre Dame in 2017 to pursue a doctorate in Electrical Engineering. Currently, he is a PhD student working with Dr. Scott Howard in the biophotonics research group. His research is about the applications of fluorescence microscopy image enhancements that could advance the work of other researchers and medical personnel in various fields. His research interests include deep learning, machine learning, computer vision, FLIM, super-resolution, compressive sensing, and neural circuits.

Scott Howard received his PhD from the Department of Electrical Engineering, Princeton University, Princeton, New Jersey, United States, in 2008. His research at Princeton University focused on high-performance quantum cascade laser design, fabrication, and characterization. He was the 2007 Newport Award winner for excellence in photonics research at Princeton University. He was a postdoctoral research associate at the School of Applied and Engineering Physics, Cornell University, Ithaca, NY, where his research focused on fiber-based nonlinear medical imaging. Currently, he is an associate professor working at the University of Notre Dame. His research focuses on how the interaction of photons and tissue can be used to aid diagnosis and fundamental research in biological fields. The group's work includes developing new techniques to image molecules in real-time, in living tissue in 3D (e.g., fast, full frame, super-resolution FLIM), contrast agent development (e.g., encapsulation of nonlinear optical dyes), and open-source software tools to enhance biomedical research's ability to produce low-noise super-resolution microscopy images using conventional laboratory microscopes.