IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 12, 2022, accepted May 26, 2022, date of publication June 9, 2022, date of current version June 14, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3181177

Restoring Fluorescence Microscopy Images by Transfer Learning From Tailored Data

EZGI DEMIRCAN-TUREYEN[1,2], (Member, IEEE), FATMA PATLAR AKBULUT[1], (Member, IEEE), AND MUSTAFA E. KAMASAK[2], (Member, IEEE)

[1]Department of Computer Engineering, Istanbul Kültür University, 34158 Istanbul, Turkey
[2]Department of Computer Engineering, Istanbul Technical University, 34467 Istanbul, Turkey

Corresponding author: Ezgi Demircan-Tureyen (e.demircan@iku.edu.tr)

ABSTRACT In fluorescence microscopy imaging, noise is a very usual phenomenon. To some extent, it can be suppressed by increasing the amount of the photon exposure; however, it is not preferable since this may not be tolerated by the subjected specimen. Thus, a sophisticated computational method is needed to denoise each acquired micrograph, so that they become more adequate for further feature extraction and image analysis. However, apart from the difficulties of the denoising problem itself, one main challenge is that the absence of the ground-truth images makes the data-driven techniques less applicable. In order to tackle this challenge, we suggest to tailor a dataset by handpicking images from unrelated source datasets. Our tailoring strategy involves exploring some low-level view-based features of the candidate images, and their similarities to those of the fluorescence microscopy images. We pretrain and fine-tune the well-known feed-forward denoising convolutional neural networks (DnCNNs) on our tailored dataset and a very limited amount of fluorescence images, respectively to ensure both the diversity and the content-awareness. The quantitative and visual experimentation show that our approach is able to curate a dataset, which is significantly superior to the arbitrarily chosen source images, and well-approximates to the fluorescence images. Moreover, the combination of the tailored dataset with a few fluorescence data through the use of fine-tuning offers a good balance between the generalization capability and the content-awareness, on the majority of considered scenarios.

INDEX TERMS Bioimaging, convolutional neural networks, fluorescence microscopy, image denoising, mixed Poisson-Gaussian model, transfer learning.

I. INTRODUCTION

In the field of bioimaging, fluorescence microscopy (FM) is a crucial examination tool that visualizes fluorescence emitted by synthetic protein markers (i.e., fluorophores) introduced to the specimen of interest. The images are recorded by the light of certain wavelengths. Due to various instrumental conditions (e.g., resolution limit of microscope, detector's ability to capture the photons), the images obtained by using FM modalities (e.g., confocal, two-photon, wide-field) suffer from noise and blur. Longer excitation time, higher amount of photon exposure, or acquiring multiple images of the same scene could help to reduce the noise and improve imaging quality, however these may deteriorate time resolution of recorded sample dynamics as well as damage to the delicate specimens by altering their chemical structures. One way of obtaining high quality images under low-photon, short exposure, and single-image conditions is postprocessing the image by a denoiser.

One of the main sources of the noise is the imperfections in the imaging system [1]. This type of noise is known as measurement noise and represented by the standard additive white Gaussian noise (AWGN) model, in which the noise values at any pixel are i.i.d. A Gaussian denoiser recovers the underlying clean image x, from the observation y, for the forward corruption model $y = x + \eta$, where $\eta \sim \mathcal{N}(\underline{0}, \sigma^2 I))$ denotes the noise for the standard deviation σ. During the past decades, many techniques have been proposed to suppress AWGN. These techniques are designed by encoding some priors on the underlying clean image, such as spatial contiguity (e.g., total variation [2]), sparsity in a transform domain (e.g., wavelet shrinkage [3]), sparsity over learned

The associate editor coordinating the review of this manuscript and approving it for publication was Haiyong Zheng.

dictionaries (e.g., K-SVD [4]), non-local self-similarity – NSS (e.g., non-local means [5], non-local total variation [6]), or a combination of such priors (e.g., block-matching and 3D filtering (BM3D) [7]). When it comes to today's data-driven techniques, the deep neural networks (DNNs) have gained a lot of popularity in solving image restoration problems, as in solving detection/classification problems, especially in the field of computational biology (e.g., [8] in histopathological image analysis, [9] in computed tomography, [10]–[13] in genome biology, [14]–[16] in microscopy imaging). In contrast to the deep classifiers, the deep restoration networks aim at learning a map from corrupted observations to the latent clean data. As being the simplest restoration task, Gaussian denoising holds the best achievements, which are mostly recorded by the convolutional neural networks (CNNs). The very first paper that uses a five-layer CNN for Gaussian denoising is [17], published in 2009. In [18], the authors adopted the stacked denoising auto-encoders method to solve denoising problem. As a non-CNN example, in [19], a learning framework named as trainable nonlinear reaction diffusion (TNRD) was proposed, and yielded the state-of-the-art results; however, a model trained by TNRD can only handle a certain noise level. Later in [20], a feed-forward network DnCNN that exploits residual learning rather than discriminative one was proposed, which achieved state-of-the-art performance as a Gaussian denoiser. MemNet [21], FFDNet [22], UNet and its non-local version UNLNet [23], and Noise2noise [24] are other notable CNN-based examples.

In the context of FM imaging, another source of noise is weak signaling due to the photon counting problems. The number of photons captured by a detector is rather small, which causes *shot noise*. This noise is stochastically dependent to the intensity of the observed image and follows Poisson distribution. There are various model-driven approaches engineered to handle Poissonian restoration problems (See the paper [25], and the references therein, for a comprehensive review). In fluorescence micrographs, both types of noises present, which is referred to as the mixed Poisson-Gaussian (MPG) noise [26]. In this case, the naive approach is performing the variance stabilization transforms (VSTs such as [27], [28]), which transform the input data to a domain where the noise follows a Gaussian distribution with a fixed variance. These transforms are applied prior to the effective Gaussian denoisers (e.g., VST + NLM [5], VST + BM3D [29]). There are also methods that depend on the preservation of Poisson statistics in Haar wavelet transform domain. PURE-LET [30]–[32] is such an example, which is already in use for the restoration of the fluorescence micrographs. On the other hand, exploiting deep CNNs for Poisson denoising is rare. The recent work [33] suggests a Poisson deblurring framework by exploiting deep learning for regularization purpose to restore microscopy images. They incorporate a neural network only to approximate the regularization function. This paradigm differs from that of the fully convolutional neural networks (FCNNs), which learn an end-to-end mapping between the observed and the desired images (or residuals as in DnCNN). To our best of knowledge, three FCNN methods dominate the fluorescence microscopy denoising literature: (1) Although it is originally a Gaussian denoiser, the DnCNN [20] is frequently involved in the recent experiments reported in the fluorescence microscopy literature as a representative of deep denoisers (See for example [34]–[38]). It reports competitive performance by employing residual learning and batch normalization. (2) The self-supervised Noise2noise [24] and its variants [36], [37], [39] are also preferable in fluorescence microscopy, due to the lack of clean data (See [35], [38], [40]). (3) The content-aware image restoration (CARE) network [41] that uses a series of convolutional layers in a U-Net architecture [42] shows a state-of-the-art performance when trained on a large number of task-specific, well-registered clean/noisy image pairs (See [40], [43], [44]).

The thing that makes well-generalized FM denoising by CNNs challenging is not only the existence of the MPG noise, but also the difficulty of acquiring the ground-truth data that are clean enough and can fairly represent the variety of the microscopic world. The training is mostly established by using potentially large unrelated source datasets (e.g., natural images in [34]), computer-generated images that are synthesized in task-specific manner (e.g., [41], [45]), or the pairs of noisy images without supervising clean data (e.g., [24], [36], [39]). One can effortlessly find an unrelated dataset with a vast number of clean images, however they are usually non-representative. The models trained on such datasets discard the prior knowledge on the target domain, which results in performance drop. On the contrary, task-specific images (synthesized or not) promote content-awareness, however the *content* here is usually very narrow such as a specific biological organism. Although they demonstrate stunning performance, when such a model is applied to an image that it was not trained for, the quality dramatically degrades. Therefore, such approaches are not likely to attain the desired diversity, when the goal is having a well-generalized model to denoise various fluorescence images. On the other side, the self-supervised learning methods entirely eliminate the need for clean data, but they typically provide lower performance [36]–[38]. At this point, it is also worth touching upon the unsupervised learning examples. Deep image prior (DIP) [46] provides a very effective generative model, which was successfully used to improve the quality of the photoacustic microscopy in [47]. As another notable example, cycle generative adversarial network (cycleGAN) [48] was employed to deconvolve microscopy images in [49]. Note that, in this study, we focus solely on the paired scheme, where the data are constituted by the clean and the noisy image pairs. In particular, we concentrate on the data, rather than the method itself, and develop a strategy to curate a training dataset that balances diversity and content-awareness. In order to see how the diversity matters, one could investigate the studies that record better performance by using non-medical (or medical but unrelated) images in medical image classification tasks. In [50], the classification of

abnormalities in CT chest image slices was learnt slightly better when they include STL-10 dataset [51] of natural images than using lung images only. In [52], the authors attempt to classify melanomas in skin lesion images by pretraining VGG network on ImageNet and Kaggle diabetic retinopathy [53] images, and later fine-tuning on a small set of target data. This strategy outperformed the full-training on the skin lesion images. Also in [54], nine different datasets were compared for the classification of polyps in endoscopy images and the texture datasets performed best among others including target (endoscopy) data.

A. CONTRIBUTION

We basically aim at gaining insights into two pertinent questions: (1) can unrelated images with low-level resemblance to the target images approximate to them in training, and (2) can such a substitution even do better by receiving support from a small amount of fluorescence data? In this respect, we tailor a training dataset, which is a hybrid subset of four unrelated public repositories with different contents. The tailored dataset is supposed to bring the data with structural diversity and the data with higher resemblance to the fluorescence images together. We use this dataset to pretrain the well-studied DnCNN and investigate the model's performance. The model is further retrained on the real fluorescence microscopy data by keeping the weights obtained, in order to support content-awareness. Moreover, to make our model applicable for real fluorescence images, we suggest to use it in collaboration with the generalized Anscombe transform (GAT) [55] and the exact unbiased inverse of the GAT [29]. We validate the effectiveness of both the model only, and the entire framework by comparing the results with the baseline methods NLM [5], BM3D [7], K-SVD [56], PURE-LET [30], [31] (for MPG noise only), and the DnCNN [20] models trained on different datasets from different domains.

II. METHOD

From the previous discussion, it is clear that our paper will consider three aspects: (1) tailoring a training set such that it meets the structural similarity expectations, (2) training a DnCNN model on the tailored dataset to transfer some intentionally chosen low-level knowledge along with the additional high-level knowledge from unrelated sources to FM image denoising problem, and (3) incorporating GAT [55] with the exact unbiased inverse of it [29] to the inference framework, so that it can be used for the real world FM images.

A. TAILORING A TRAINING DATASET

The idea of employing unrelated data especially when the data of interest are in short supply is a commonly used strategy in deep learning. This strategy transfers the information from a *source* domain to the *target* domain at hand. Due to lack of available fluorescence data (target dataset), the fluorescence imaging is a convenient target domain to apply this strategy.

When the strongest activations in the shallower layers are shown during the forward pass of a deep model, one can observe the discriminant low-level structures, as in [57]. These are the features of sharp edges, lines, textures, colors, orientations, image symmetry, etc. Distributions of these details play an important role in the perceptual similarity of the images and we may assume that the perceptually similar images can substitute each other. If we can develop a methodology to characterize the fluorescence data as a group of low-level features, many unrelated clean images with similar features may serve as alternative ground-truths. Plus, these images from different sources may even train a better model than the fluorescence images, since by the nature of our methodology, diversity is promoted.

Our goal is gathering a dataset with images whose low-level features are similar to those of the target fluorescence images. We also desire that the images involved in our dataset are with a large variety of contents, shapes, and colors; since we believe that such variations provide additional information to boost the generalization capability of the model. Accordingly, we've selected four publicly available repositories as source datasets, each of which has a different content addressing different categories: Berkeley Segmentation Dataset (BSD500)[1] [58] of 500 natural images, Describable Textures Dataset (DTD)[2] [59] of 5100 textures, PLACES dataset[3] [60] of 1000 scenes, and UC Merced Land Use Dataset (UCMERCED)[4] [61] of 2100 aerial images. We choose these datasets due to their diverse natures. When we visually judge the low-level characteristics of these datasets, we observe that BSD500 and PLACES are rich of structures and colors, whereas DTD is a collection of multifarious textures. UCMERCED, on the other hand, combines flat regions and textures, which, in some sort, might represent FM images well. We believe that the contributions of such distinct datasets intrinsically promote the diversity among the images to be involved in our dataset. On the other hand, for the fluorescence images, we combine 400 images from three different sources: Fluorescence Microscopy Denoising (FMD) dataset[5] [35], Cell Nucleus Segmentation Dataset (NucleusSegData) [62], [63], and a dataset used in cell segmentation of microscopy images [64]. FMD dataset contains images of bovine pulmonary artery endothelial (BPAE) cells (mitochondria, F-actin, nuclei), zebrafish, and mouse brain tissues captured with either a confocal, two-photon, or wide-field microscope, whereas the last two datasets contain images of Hela, fibroblasts, HEPG2, and U2OS cell lines. We refer to the combined dataset as TARGET. We also combined additional 120 fluorescence images from the same repositories for the purpose of fine-tuning. These 120 images are also the images that we use as reference while tailoring our dataset. From now on, this dataset will be referred to as TARGET120.

[1] https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/
[2] https://www.robots.ox.ac.uk/~vgg/data/dtd/
[3] http://places.csail.mit.edu/
[4] http://weegee.vision.ucmerced.edu/datasets/landuse.html
[5] https://github.com/yinhaoz/denoising-fluorescence

All images are preprocessed by cropping the center 180×180 region, subtracting the per-pixel mean of all images, discarding the ones without information, and converting them to grayscale.

Our image selection procedure computes the pairwise similarities between each image in each source dataset, say s, and the images in TARGET120 dataset, say t for each, i.e.,

$$sim(s, t) = D(T(s), T(t)) \tag{1}$$

where T is the feature extractor and D is the similarity measure. In this respect, let us first introduce the features that we use in the selection procedure for the quantitative similarity assessment.

1) COLOR FEATURES

Since the units in the earlier layers are sensitive to color, the images to be used for pretraining can be selected by considering color similarities. We use histogram encodings as it is the simplest and the most usual color feature. It represents the distribution of the intensities in our case, since we're dealing with the scalar-valued images. We uniformly quantized the levels into 256 bins.

2) BASIC IMAGE FEATURES (BIFs)

BIFs [65] provide a way to represent an image by classifying its pixels into seven categories depending on local image structures and symmetry. The method described in [65] first computes the scale-normalized versions of the second order derivative-of-Gaussian (DoG) filter family responses as $c_{qp} = \sigma_B^{p+q} h_{pq}$, where σ_B is the scale of the filter (standard deviation), h_{pq} is one of the DoG filters, p and q are the orders of $\{0, 1, 2\}$. Next, it defines two variables $\gamma_1 := \frac{1}{2}(c_{20} + c_{02})$ and $\gamma_2 := \sqrt{\frac{1}{4}(c_{20} + c_{02})^2 + c_{11}^2}$, and classifies each pixel according to the largest of the set:

$$M = \{\epsilon c_{00}, \sqrt{(c_{10}^2 + c_{01}^2)}, \gamma_1, -\gamma_1, (\gamma_2 + \gamma_1)/\sqrt{2}, \cdots, (\gamma_2 - \gamma_1)/\sqrt{2}, \gamma_2\} \tag{2}$$

where ϵ controls the amount of flatness. If ϵc_{00} is the largest, then the pixel is assigned to be a part of flat region. If $\sqrt{(c_{10}^2 + c_{01}^2)}$ is the largest, there should be a ramp edge around the pixel. γ_1 and $-\gamma_1$ respectively indicate light and dark blobs, while $(\gamma_2 + \gamma_1)/\sqrt{2}$ and $(\gamma_2 - \gamma_1)/\sqrt{2}$ refer to the light and the dark lines. If the winner is γ_2, on the other hand, the pixel is classified as a part of a junction.

3) DIRECTIONAL FEATURES

Directional cues in an image are often exploited by the image recovery methods, e.g., [66] and [67] that employ curvelet transforms respectively for denoising and reconstruction purposes, [68] and [69], [70] that design first-order derivative-based directional priors, and [71] that makes use of the steerable filters. These efforts motivate us to benefit directionality while tailoring our training data. Intuitively, if the images seen by the network have similar directional characteristics with the target data, then they ought to achieve a similar performance.

There are many directional image-analysis tools in the literature. Structure tensor (ST) is such a tool, which is also employed while designing regularizers in handcrafted image restoration methods (e.g., [69], [72]). The *structure tensor* of a scalar-valued image x at any point is a 2×2 symmetric positive semi-definite (PSD) matrix of the form:

$$S_K x = K_{\hat{\sigma}} * (\nabla x \nabla x^T) \tag{3}$$

where $K_{\hat{\sigma}}$ is a Gaussian kernel of standard deviation $\hat{\sigma}$, and ∇ denotes the gradient operator. This semi-local descriptor is capable of summarizing all the gradients within the patch supported by $K_{\hat{\sigma}}$. It characterizes the variations better than the local differential operators.

The local orientation is given by the eigenvector $v^+ \in \mathbb{R}^2$ associated with the greatest eigenvalue λ^+, i.e.,

$$\theta = \frac{1}{\|v^+\|} \begin{bmatrix} v_1^+ \\ v_2^+ \end{bmatrix} \tag{4}$$

As a measure of directionality, one can equip the *coherence* measure $\alpha \in [0, 1]$, defined as:

$$\alpha = \frac{\lambda^+ - \lambda^-}{\lambda^+ + \lambda^-} \tag{5}$$

where the larger values of α is an indicator of the flow-like structures.

To be able to use these direction and coherence fields as a feature descriptor, we express them in histogram form again. We split image into patches of size 30×30. For each patch, we calculate a histogram, where the local orientation determines which bin to be increased, while the local coherence determines how much is added. The number of bins is kept as 18, each of which is corresponding to a 10-degree interval. We perform block normalization on the concatenated histograms over the overlapping patches of size 60×60, by each time shifting a window by 30 pixels. The final feature vector of the entire image is nothing but the concatenation of the normalized histograms (i.e., a $18 \times 4 \times 25 = 1800$ dimensional vector). This idea is very similar to the histogram of oriented gradients (HOG) [73], however by using structure tensor, one assigns more weight to the directions [74].

Another directional feature that we employ is related to the BIFs again. Oriented BIFs (oBIFs) expand BIFs by considering the orientations. The number of categories rise from seven to 23 when eight different orientations of ramp edges, four different orientations of junctions, four orientations of light and four orientations of dark lines are taken into account. In [75], the authors discussed the role of distributional statistics on appearance learning. They used oBIFs to encode the appearance in terms of textons. They computed oBIFs at two different scales (finer and coarser) $\sigma_o = \{1.1, 2.2\}$, which results in 529 possible textons for a pixel within a patch. They consider the histogram of these textons while measuring the similarities. We adopt this system and combine it with the

other aforementioned features with the expectation of better performance. Note that, the MATLAB implementations for the computation of oBIFs were made publicly available in [76].

4) OVERALL FEATURES AND THE SIMILARITY

In brief, we have three histograms encoding the intensities, the orientations (i.e., invariant to the sign of the direction), and the oBIF textons. They are normalized such that they have unit sum and scaled to the same interval. Now, we are interested in how similar a certain image in a subset of TARGET is to the most similar one in a source dataset. In other words, we need to score $sim(s, t)$, for all s and t by using three pairs of histograms at hand. As the similarity measure D in (1), if we had only one pair of histograms (a and b), we would employ Bhattacharya distance subtracted from one, i.e., $D(a, b) = 1 - \frac{1}{\cos(\sqrt{a}\cdot\sqrt{b})}$. Since the number of histogram pairs is three, we take the harmonic mean of histogram similarities into account.

For each image in the TARGET120 dataset, our image selection procedure selects k most similar images that maximizes $sim(s, t)$. Some source images are surely be selected multiple times, since the self-similarity of the TARGET120 dataset is high, thus many target images may vote for exactly the same source images. In our case, we have 8700 candidates coming from four source datasets. 412 source images are selected as the most similar images by setting $k = 23$. This value of k was empirically set such that it yields a value close but more than to 400 (the desired size of the dataset), since the images we suspect to be noisy should also be eliminated. Note that, six of those images are discarded, since they're estimated to be exhibiting Gaussian noise with $\sigma \geq 0.01$. Also note that, our tailored training set (hereinafter referred to as TAILORED) consists only 400 randomly chosen images of the remaining 406 images, for a fair comparison. The final dataset involves 196 DTD, 120 PLACES, 60 BSD500, and 24 UCMERCED images. Fig. 1 shows some sample substitutions from TAILORED dataset. First row shows four images from TARGET120 and in second row, just below each target image, one can see one of the source images selected due to its high resemblance. The TARGET images shown in Fig. 1 are contrast-enhanced for more visibility. Note that, we haven't applied such an enhancement while extracting the features. Moreover, in Fig. 2, an illustration that roughly summarizes the dataset tailoring process is provided. The source datasets are corresponding to BSD500, DTD, PLACES, and UCMERCED, whereas the target dataset is TARGET120. The zoomed inset at the right-hand side sketches the Feature Design & Image Comparison procedure. Each colorful grid displays a portion from a matrix as an image with scaled colors. The entries of those matrices are the similarities between the source and the target descriptors that describe either color, OBIF, or directionality-based features. The figure visually exemplifies color, OBIF, and directional descriptors (histograms) for a pair of target and source image samples.

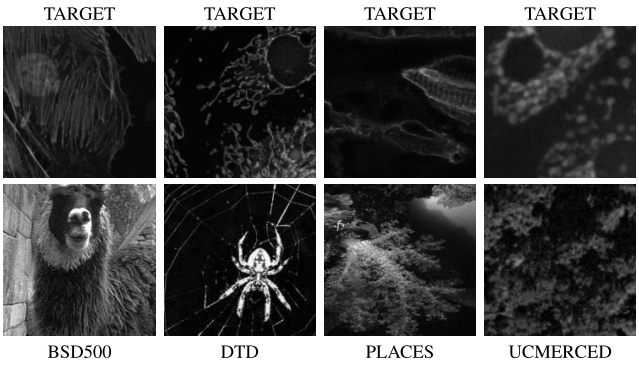


FIGURE 1. Sample substitutions from our TAILORED dataset. The TARGET images are contrast-enhanced here, for more visibility.

We also provide the two-dimensional t-SNE projections in Fig. 3, in order to visualize the distributions of the chosen descriptors within each dataset. The descriptors are projected without using the dataset labels, but shown with labels a posteriori for visualization purposes. The considered low-level features are not expected to set all of the datasets apart, however one can see that they definitely hold some information. The data points from BSD500 and PLACES mostly collapse on each other because of their similar content. Many data points from UCMERCED dataset concentrate on a close neighborhood, while the DTD dataset seems to have a diverse nature. Many DTD data points are coinciding with the TARGET, thus with the TAILORED dataset. Most of the data points from TARGET form clear clusters. This can be interpreted as the chosen low-level features are good at distinguishing the fluorescence images and this disparity may be playing an important role in restoration. Overall, the considered low-level view-based features carry some domain-specific information, especially for fluorescence microscopy domain. Thus, the TAILORED dataset well compromises between the domain invariance and the domain-specificity.

B. TRANSFER LEARNING & FINE-TUNING

As described in the previous section, we transferred the knowledge coming from different source datasets to FM denoising domain. The features are extracted by pretraining a DnCNN model on the tailored dataset, which does not involve any fluorescence images. We train 50 epochs with learning rate decayed exponentially from 1e-1 to 1e-4. For each epoch, we utilized multiple sub-epoch system that reiterates forward and backward by arbitrarily cropping fragments from entire image to train much more patterns. Those mini-batches are of size 128.

As the loss function, (6) is used to learn a residual mapping $\mathcal{R}(y) \approx \eta$ with trainable parameters Θ:

$$\ell(\Theta) = \frac{1}{2N}\sum_{i=1}^{N} \|\mathcal{R}(y_i; \Theta) - (y_i - x_i)\|_F^2 \qquad (6)$$

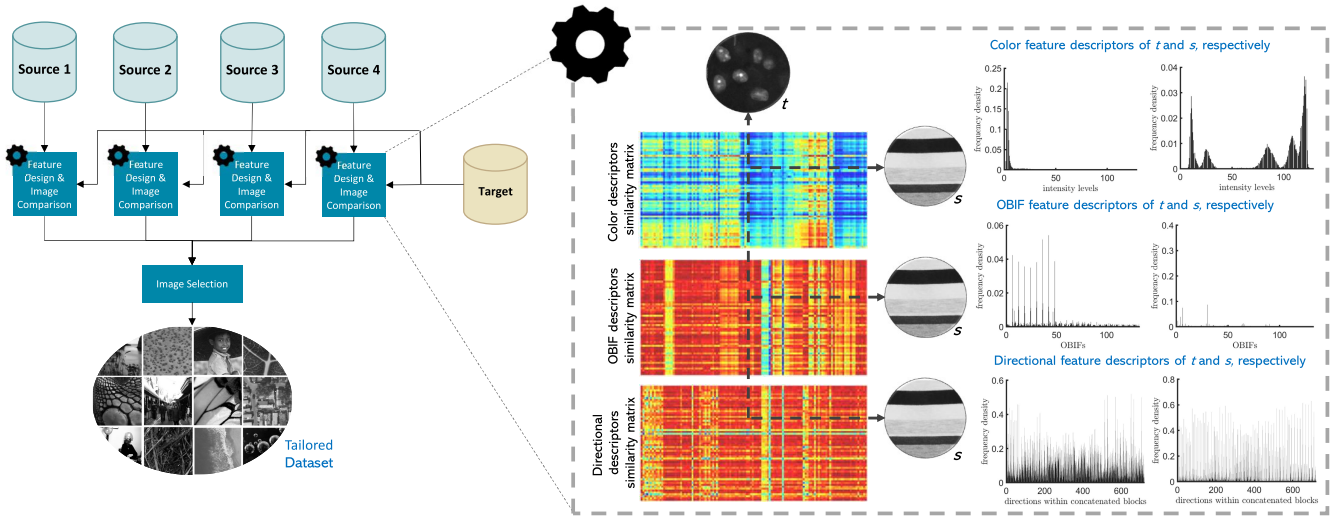
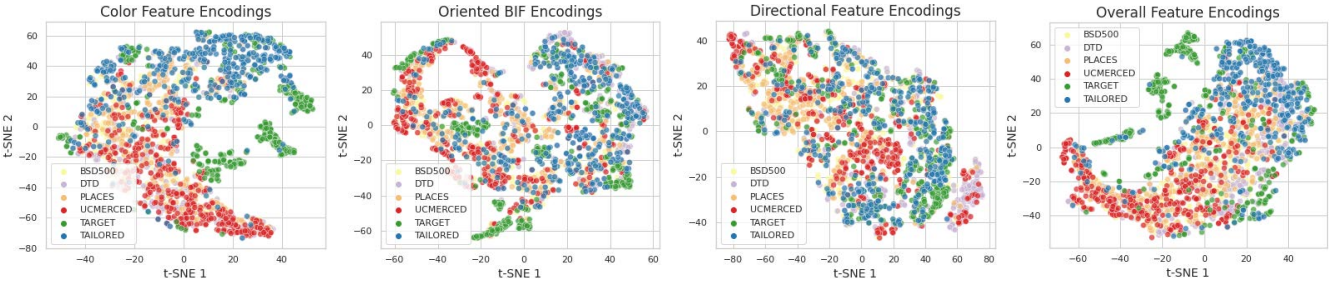


FIGURE 2. A representative illustration on tailoring the pretraining dataset.



(a) (b) (c) (d)

FIGURE 3. t-SNE plots visualizing the distributions of (a) Color-based, (b) Oriented BIF, (c) Directional, and (d) Overall feature descriptors within each dataset.

for N noisy-clean (y_i, x_i) pairs, where $\|\cdot\|_F$ denotes the Frobenius norm.

A frequently used transfer learning strategy is training a network on unrelated source data, then using it to initialize the weights of a network that is further trained on a fair amount of target data. This is what known as fine-tuning. In this respect, we also further retrained all layers on TARGET120, starting with the pretrained weights of TAILORED. Note that, we also tried to retrain only the last layers by freezing the other parameters, however the results were not that satisfactory.

C. DEALING WITH THE MIXED POISSON-GAUSSIAN NOISE

As mentioned earlier, fluorescence images are contaminated by MPG noise. Unlike Gaussian, Poisson noise is signal-dependent. The forward corruption model is

$$y[i] = x[i] + \eta[i] \tag{7}$$

at an image point i, where $\eta[i] = \eta_p(x[i]) + \eta_g[i]$ for p and g subscripts are denoting the Poisson and zero-mean Gaussian components of the noise, respectively. Therefore, for $a > 0$ denoting the scaling coefficient of the detector, $(x[i] + \eta_p(x[i]))/a \sim \mathcal{P}(x[i]/a)$, and $\eta_g \sim \mathcal{N}(0, bI)$, where $b \geq 0$ is the variance of the Gaussian noise. A fluorescence micrograph with a large value of a but a small value of b might be considered as Poisson noise dominated, while the opposite may address to the Gaussian noise dominance.

A naive approach to deal with signal-dependent MPG noise is using variance stabilization techniques (VST). One such VST is generalized Anscombe transform (GAT) [55] of the form:

$$f(y[i]) = \frac{2}{a}\sqrt{\max\left(ay[i] + \frac{3}{8}a^2 + b, 0\right)} \tag{8}$$

such that after transformation, the noise of the new signal approximately behaves like Gaussian with unit variance. Since GAT requires a and b, these parameters need to be fitted before transformation. In this paper, we estimate these parameters by using the method described in [28]. Note that, although the Gaussian noise variance cannot be negative valued, this method could generate negative b values when

the Gaussian noise component is so small. For more details, the reader is referred to [28]. In our work, we just make any negative estimate of b zero.

Once the signal is transformed, the AWGN denoisers become convenient to work with. However, their variance-stabilized outputs also require to be transformed back to the original range. Exact unbiased inverse transformation of the GAT [29] provides an unbiased and accurate way of inversion even at low signal levels, which is usually the case in fluorescence imaging. The closed form approximation of the exact unbiased inverse transform is given as:

$$\tilde{I}(D) = \frac{1}{4}D^2 + \frac{1}{4}\sqrt{\frac{3}{2}}D^{-1} - \frac{11}{8}D^{-2} + \frac{5}{8}\sqrt{\frac{3}{2}}D^{-3} - \frac{1}{8} - b \tag{9}$$

where D is the denoised image (See [29] for details.) Therefore, we come up with the resulting estimate $\tilde{x} = \tilde{I}(D)$.

In our work, we exploit both GAT [55] (with parameter estimation [28]) and its exact unbiased inverse transformation [29] for not only the classical Gaussian denoisers, but also the DnCNN models. In fact, it is not frequent to see deep Gaussian denoisers in combination with a variance stabilization module in the literature. For instance, in the recent benchmarking papers [35], [40] on fluorescence microscopy restoration, GAT (and exact unbiased inverse transformation) was only performed before denoising with the classical Gaussian denoisers (e.g., BM3D, NLM, K-SVD). Although the DnCNN was also among the competing methods, it was kept separate from the variance stabilization. Our framework inspects the parameters a and b, since the estimate values of these parameters may determine the effectiveness of VST (or specifically GAT) usage. When the estimated value of a is negative, our framework switches to non-GAT state, where we do not perform variance stabilization.

After the attachment of the variance stabilization module, the overall inference framework gets into the shape shown in Fig. 4. Note that, we used the source codes of the exact unbiased inverse transform, which were made publicly available by the authors.[6]

III. EXPERIMENTAL RESULTS

A. EXPERIMENTAL SETTINGS

The models were trained on a computer equipped with NVDIA GeForce GTX 1080Ti, while the feature design, image comparison procedures, and testing were all conducted on another computer with Intel Core Processor i7-7500U (2.70-GHz) with 16 GB of memory. All procedures and methods were written in MATLAB. We used the training and testing codes of the MatConvNet [77] implementation[7] of DnCNN. We kept the configuration settings as suggested in [20]. All the models were trained end-to-end by using 400 images of size 180×180.

[6]http://www.cs.tut.fi/~foi/invansc/
[7]https://github.com/cszn/DnCNN

B. COMPARISON AND EVALUATION

We group the experiments into "Gaussian Noise Denoising" and "MPG Noise Denoising". In the first group, we only focus on DnCNN, which is responsible for Gaussian denoising. We evaluate our DnCNN model trained on the tailored dataset. We also assess the effect of fine-tuning on top of pretraining. In the second group of experiments, we consider MPG noise, which is the actual noise type exhibited by FM images. This time, the noise present on the test data is real, rather than synthetic. In both groups, we use an unseen 40-image subset of FMD dataset [35] as the test set. From now on, it is referred to as FMDTest40. Note that the ground-truth images in this set are estimated by averaging 50 captures of the same FoV [35]. Additional to FMDTest40, we use three HeLa cell images acquired with a confocal microscope from [25], six images of actin, mitochondria, membrane, and nuclei acquired with a wide-field fluorescence microscope from [40], and two images of fly wings and planaria acquired with a confocal microscope from [41], and finally four images of human U2OS cells acquired with confocal microscope from [78]. The ground-truth images in these datasets were acquired under combined high photon exposure + long excitation time settings. In order to prepare a test dataset, we crop large images into non-overlapping patches of size 512×512. The smaller ones are all of size 256×256, thus we leave them intact. We discard the patches with no information. At the end, we have a dataset of 55 images (3 HeLa, 22 Actin, 20 Mitochondria, 2 Membrane, 1 Nucleus, 1 Fly Wings, 2 Planaria, and 4 Human U2OS cells). These additional images serve as external validation data used to assess how our model generalizes when the data goes out of the imaging configurations (sample + microscopy type/settings + FoV) in TARGET dataset. A few samples from this external test set are shown in Fig. 5.

For all quantitative comparisons, we use the structural similarity index (SSIM) [79] and the peak signal to noise ratio (PSNR) measures. We also provide denoising results of the competing methods/models for a few sample images in order to assess the proposed approach visually.

1) EXPERIMENT 1- GAUSSIAN NOISE DENOISING

We train seperate DnCNN-S [20] (non-blind, trained for a specific noise level) models by using randomly selected 400-image subsets of each source dataset (BSD500, DTD, PLACES, and UCMERCED), and the TARGET dataset. In order to create clean-noisy image pairs, we degrade the ground-truth images by adding Gaussian noise with four different noise levels, $\sigma = \{15, 25, 35, 50\}$. Thus, we obtain five different DnCNN models for each noise level, which will compete with the two models of ours (i.e., the models before and after fine-tuning). Note, that, as suggested in [20], the training data are cropped into 40×40 patches for the data augmentation and the network depth is set to 17. In addition to the DnCNN models, we also include three state-of-the-art Gaussian denoisers as baselines: NLM [5], BM3D [7], and

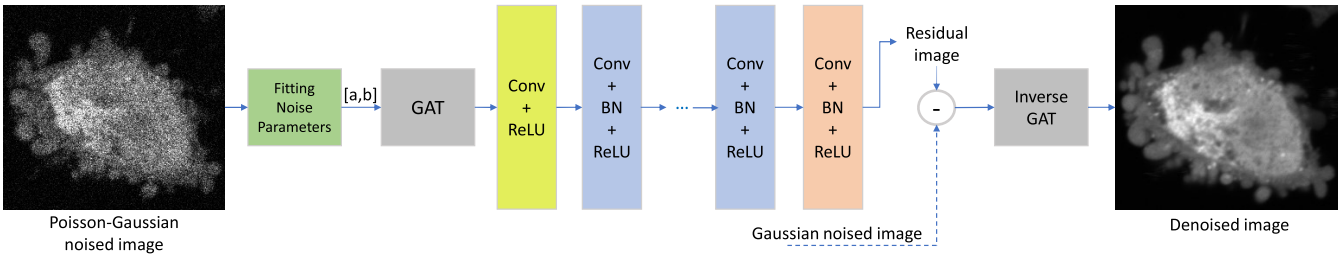


FIGURE 4. The overall framework that we use to restore fluorescence microscopy images with MPG noise.

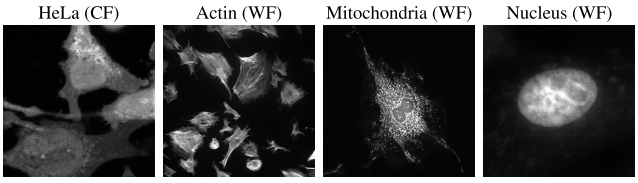


FIGURE 5. Some samples of the external validation images: HeLa cell image is from [25], and the others are from [40]. CF and WF stand for confocal and wide-field microscopy modalities, respectively.

K-SVD [56] to the experimentation. For the publicly available source codes of these methods, the reader may check the websites given in the footnotes: NLM,[8] BM3D,[9] Analysis K-SVD.[10]

The quantitative results on the set FMDTest40 for each noise level are reported in Table 1. PSNR is the former (top) value in each cell, while the latter (bottom) value is SSIM. Each measure is followed by ± the standard error (SE). The best PSNR and SSIM results are highlighted in red. The second bests are also colored with blue in order to underscore the gain. From the reported results, one can observe that the DnCNN models clearly beats the baseline denoisers. When it comes to establishing a comparison between DnCNN models, as one may expect, DnCNN (TARGET) performs consistently better than all the models trained by using four source datasets, both in terms of PSNR and SSIM. Only when $\sigma = 25$, the difference is not so distinguishable; in fact, DnCNN (BSD500) performs at the same level. DnCNN (TARGET) also beats our DnCNN (TAILORED), except for the case $\sigma = 25$ again. The gain of using target data rather than our tailored dataset is significant, when the level of noise is low (i.e., $\sigma = 15$); but the difference is not so prominent, when $\sigma = 35$ and $\sigma = 50$. The success of TARGET dataset on this experiment was expected due to the exact similarity between the training and the test sets. But we can draw the conclusion that our tailored dataset managed to catch a similar (-0.1 dB to $+0.06$ dB) PSNR performance, except for the case $\sigma = 15$, even though it only considers the similarities among low-level features. Its PSNR and SSIM scores are consistently above the other source datasets, even the gain is sometimes small (e.g., 0.08 dB over DnCNN (BSD500) when $\sigma = 25$). When it comes to our fine-tuned model, it achieves the best results for all noise levels. We observe PSNR gains between 0.1 dB and 0.25 dB at higher noise levels, when compared to DnCNN (TARGET), although there is no gain when $\sigma = 15$.

Fig. 6 illustrates some visual results of all competing methods/models, when the noise level $\sigma = 35$. We pick three sample images, which are referred to as Image-1, -2, and -3, from FMDTest40. The very first row shows the ground-truths, whereas the second row shows the noisy observations. Each ground-truth and observed image is followed by a detail patch corresponding to the yellow box superimposed on it. The rest of the rows are illustrating the restored images. Each column corresponds to a different method/model. Just below the output images, the detail images are provided. Under such a high level of noise, the results produced by the baseline techniques are not pleasant at all. NLM and K-SVD tend to over-smooth fine details. BM3D generates artifacts. When it comes to the DnCNN models, it can be seen that the tiny details in Image-1 (mouse brain tissues) are much more visible in the results obtained by DnCNN (TARGET) and DnCNN (TAILORED). By the other models, these details are smoothed out. On the other side, the fibres in Image-2 restored by DnCNN (DTD) and DnCNN (UCMERCED) are not distinguishable at all. They become more visible (but still not satisfying enough), when using BSD500 and PLACES, possibly because fibrous textures present in these diverse and content-rich datasets. Our TAILORED dataset leads to a better performance for Image-2, although it causes a cloudy look around the fibres. DnCNN (TARGET)'s result is smooth in the direction that stresses the fibrous structure, but still not as good as the fine-tuned DnCNN (TAILORED), where the gaps between the fibres are clearer. Image-3 involves curvilinear structures corresponding to the intracellular mitochondrial network of BPAE cells. Such structures are mostly interrupted by the models at hand. In detail images, we see that a line-like structure is almost disappeared by DnCNN (BSD500), DnCNN (DTD), and DnCNN (TARGET) models. The datasets PLACES and UCMERCED do better in this example, when compared to the other source datasets, probably due to their rich content in line-like structures.

[8]https://www.mathworks.com/matlabcentral/fileexchange/13619-toolbox-non-local-means

[9]https://webpages.tuni.fi/foi/GCF-BM3D

[10]https://elad.cs.technion.ac.il/software/

TABLE 1. PSNR / SSIM comparisons on Gaussian noise denoising of the FMDTest40 [35].

σ:	15	25	35	50
Observed Images	25.53 ± 0.019 0.401 ± 0.002	21.59 ± 0.018 0.230 ± 0.002	19.01 ± 0.017 0.148 ± 0.001	16.25 ± 0.014 0.088 ± 0.001
NLM [5]	33.20 ± 0.058 0.859 ± 0.001	31.11 ± 0.065 0.813 ± 0.002	28.64 ± 0.047 0.679 ± 0.001	26.57 ± 0.038 0.557 ± 0.001
BM3D [7]	33.25 ± 0.075 0.872 ± 0.002	32.17 ± 0.066 0.857 ± 0.002	28.10 ± 0.057 0.699 ± 0.002	21.80 ± 0.044 0.376 ± 0.002
K-SVD [4]	33.54 ± 0.145 0.837 ± 0.004	30.80 ± 0.185 0.742 ± 0.005	28.19 ± 0.365 0.625 ± 0.009	25.33 ± 0.470 0.476 ± 0.010
DnCNN (BSD500)	36.70 ± 0.074 0.918 ± 0.001	34.57 ± 0.079 0.884 ± 0.001	33.20 ± 0.081 0.855 ± 0.002	31.58 ± 0.076 0.814 ± 0.002
DnCNN (DTD)	36.64 ± 0.075 0.916 ± 0.001	34.54 ± 0.078 0.881 ± 0.002	33.10 ± 0.078 0.850 ± 0.002	31.77 ± 0.082 0.819 ± 0.002
DnCNN (PLACES)	36.62 ± 0.074 0.917 ± 0.001	34.40 ± 0.076 0.879 ± 0.001	33.04 ± 0.078 0.851 ± 0.002	31.60 ± 0.078 0.817 ± 0.002
DnCNN (UCMERCED)	36.29 ± 0.066 0.903 ± 0.001	34.50 ± 0.080 0.880 ± 0.002	33.08 ± 0.081 0.852 ± 0.002	31.53 ± 0.078 0.813 ± 0.003
DnCNN (TARGET)	37.01 ± 0.078 0.924 ± 0.001	34.59 ± 0.079 0.884 ± 0.002	33.31 ± 0.083 0.858 ± 0.002	31.91 ± 0.083 0.824 ± 0.002
DnCNN (TAILORED)	36.78 ± 0.076 0.919 ± 0.001	34.65 ± 0.079 0.883 ± 0.001	33.27 ± 0.080 0.855 ± 0.002	31.81 ± 0.079 0.821 ± 0.002
DnCNN (TAILORED) w/ fine-tuning	37.01 ± 0.078 0.924 ± 0.001	34.85 ± 0.081 0.890 ± 0.001	33.47 ± 0.083 0.862 ± 0.002	32.02 ± 0.085 0.827 ± 0.002

The loss along the structure is far less visible in DnCNN (TAILORED) restoration than in the others, even though the PSNR / SSIM performance is fairly similar to that of produced by the TARGET dataset. Fine-tuning this model, on the other hand, manage to fix the oil-painting-like artifacts, while preserving the line connectivity, and yield the best PSNR / SSIM among all.

In Table 2, the restoration results on the external test images are separately reported, for the same noise levels present in Table 1. These images are visually less similar to the images in TARGET, when compared to FMDTest. In this case, one may expect that the performance of DnCNN (TARGET) would drop, perhaps to the levels of those given by the source datasets. This anticipation has partially been met in Table 2. For example, on HeLa and Fly Wings images, the DnCNN (TARGET) failed to be one of the three best performing models, except for the cases when $\sigma = 50$. Contrary to this, it produced the best PSNR and SSIM scores on Actin ($\sigma = 15$), Membrane ($\sigma = \{35, 50\}$), and Nucleus ($\sigma = 35$) images and experiments. For the rest of the experiments, predominant observation is that the DnCNN (TAILORED) and the DnCNN (TARGET) alternately shared the second place. When it comes to our DnCNN (TAILORED) with fine-tuning, it achieved the best restoration results with a few exceptions, even though its training required only 120 images of the target field. This achievement provides us insights on how the usage of the computationally chosen unrelated data can lower the need of the related data.

Paired sample t-tests are also carried out over 40 test images in FMDTest40 to determine if the PSNRs produced by the DnCNN models trained (w/ or w/o fine-tuning) on TAILORED dataset is significantly higher than the ones trained on the other datasets (including TARGET). The tests are performed for all noise levels ($\sigma = \{15, 25, 35, 50\}$) to assess the general significance of our approach. In each t-test, the null hypothesis (H0) is that the PSNR for both models is the same. The alternative hypothesis (H1) is that the PSNR produced by the winner model is larger than that produced by the competing model. In Table 3, we report the exact p values obtained for each pair of DnCNN models trained by using six different datasets (plus, one set of fine-tuning data). Each test has 159 degrees of freedom and the threshold is selected as $\alpha = 1e-07$. The p values larger than α are marked in blue, indicating that there is no significant difference ($p > 1e-07$) in PSNRs amongst the considered models. It can be seen that, when the considered models are the ones trained on two of the four source datasets, almost all tests fail to reject H0. On the other hand, the red color highlights the p values found by the t-tests run between our fine-tuned DnCNN (TAILORED), which is the best performing model, and each of the other models. We can deduce that the PSNR produced by our fine-tuned model is significantly larger ($p < 1e-07$) than that of all other models, including the TARGET. Moreover, the test carried out between DnCNN (TARGET) and DnCNN (TAILORED) shows that although the mean PSNR of the samples produced by using the TARGET data is higher, the difference is not so significant. This also supports the claim that the TAILORED dataset can be used as a substitute for target data, without even the need for fine-tuning, when the data are in short supply.

2) EXPERIMENT 2- MPG NOISE DENOISING

In the second group, we test our entire FM denoising framework. This time, we train DnCNN-B [20] (blind, trained for a range of noise levels, i.e., $\sigma \in [0, 50]$). The patch size is set to 50×50 for the data augmentation. The network depth is

TABLE 2. PSNR / SSIM comparisons on Gaussian noise denoising of external test set.

	Noise level (σ)	Observed Images	NLM [5]	BM3D [7]	K-SVD [4]	DnCNN (BSD500)	DnCNN (DTD)	DnCNN (PLACES)	DnCNN (UCMERCED)	DnCNN (TARGET)	DnCNN (TAILORED)	DnCNN (TAILORED) w/ fine-tuning
HeLa (CF) [25]	**15**	25.91 0.329	33.68 0.856	38.17 0.921	35.83 0.884	39.25 0.937	39.02 0.935	38.85 0.936	38.47 0.899	39.08 0.928	39.11 0.939	39.17 0.939
	25	21.64 0.157	33.19 0.798	36.44 0.902	33.53 0.836	36.56 0.905	36.71 0.908	36.44 0.905	36.53 0.906	36.52 0.899	36.82 0.909	36.88 0.914
	35	18.87 0.089	32.53 0.737	30.12 0.676	31.78 0.785	35.02 0.880	35.05 0.888	34.99 0.878	34.93 0.881	35.33 0.888	35.38 0.889	35.40 0.900
	50	16.00 0.046	24.53 0.532	25.15 0.584	28.89 0.681	32.88 0.784	33.54 0.859	33.37 0.847	32.99 0.813	33.73 0.867	33.73 0.870	33.78 0.872
Actin (WF) [40]	**15**	27.05 0.601	34.33 0.881	34.50 0.890	33.29 0.862	35.45 0.911	35.41 0.910	35.36 0.910	35.07 0.898	35.59 0.914	35.44 0.911	35.56 0.914
	25	22.86 0.392	32.12 0.831	33.43 0.875	31.10 0.811	33.65 0.883	33.61 0.881	33.46 0.877	33.57 0.882	33.65 0.883	33.66 0.883	33.77 0.885
	35	20.27 0.269	30.52 0.787	28.78 0.718	29.71 0.771	32.37 0.856	32.28 0.852	32.21 0.853	32.26 0.857	32.34 0.862	32.38 0.857	32.52 0.862
	50	17.87 0.174	28.74 0.732	24.43 0.624	27.94 0.713	30.86 0.825	30.89 0.822	30.75 0.818	30.78 0.823	30.89 0.823	30.90 0.821	31.04 0.828
Mitochondria (WF) [40]	**15**	28.67 0.604	36.04 0.919	36.16 0.932	35.12 0.898	37.43 0.939	37.47 0.945	37.33 0.933	37.03 0.921	37.60 0.950	37.53 0.946	37.66 0.951
	25	24.62 0.408	33.41 0.867	34.73 0.888	32.25 0.845	35.04 0.889	35.13 0.916	34.86 0.895	35.04 0.898	34.94 0.860	35.20 0.921	35.32 0.935
	35	22.14 0.293	29.78 0.548	29.89 0.683	30.69 0.803	33.36 0.844	33.41 0.871	33.26 0.880	33.37 0.864	33.51 0.898	33.56 0.898	33.72 0.918
	50	19.86 0.203	27.42 0.712	26.50 0.706	28.77 0.741	31.69 0.888	31.74 0.889	31.55 0.885	31.64 0.883	31.78 0.889	31.80 0.890	31.88 0.893
Membrane (WF) [40]	**15**	26.22 0.376	37.78 0.900	39.27 0.943	36.46 0.870	39.96 0.951	39.68 0.948	39.88 0.950	39.16 0.929	40.14 0.953	40.02 0.952	40.20 0.954
	25	22.29 0.214	34.74 0.827	37.15 0.922	33.85 0.800	37.70 0.931	37.62 0.931	37.58 0.929	37.38 0.928	37.74 0.929	37.83 0.934	38.05 0.939
	35	19.66 0.135	32.53 0.748	31.03 0.771	31.54 0.718	36.19 0.915	36.02 0.912	36.06 0.914	35.74 0.910	36.56 0.925	36.30 0.918	36.51 0.924
	50	16.79 0.072	25.53 0.629	22.47 0.442	28.73 0.602	34.12 0.866	34.46 0.894	34.34 0.890	34.01 0.875	34.89 0.905	34.58 0.895	34.85 0.904
Nucleus (WF) [40]	**15**	27.97 0.492	37.48 0.887	39.27 0.940	36.30 0.874	39.51 0.942	39.47 0.940	39.49 0.941	39.00 0.923	39.61 0.944	39.55 0.942	39.62 0.943
	25	24.43 0.367	34.94 0.830	37.77 0.924	34.03 0.818	38.08 0.930	38.06 0.929	37.94 0.926	37.96 0.929	38.09 0.930	38.10 0.930	38.23 0.932
	35	21.53 0.265	31.86 0.671	31.21 0.679	31.78 0.743	37.08 0.922	36.80 0.914	36.92 0.917	36.78 0.919	37.29 0.925	37.09 0.920	37.28 0.924
	50	18.36 0.158	29.98 0.674	22.37 0.614	29.51 0.667	35.48 0.897	35.69 0.905	35.49 0.900	35.39 0.900	36.06 0.910	35.74 0.904	36.06 0.912
Fly Wings (CF) [41]	**15**	27.41 0.710	31.04 0.869	30.76 0.880	30.56 0.850	32.96 0.918	32.60 0.909	32.88 0.916	32.94 0.917	32.86 0.910	32.95 0.916	33.15 0.918
	25	24.21 0.537	27.91 0.754	29.63 0.845	27.31 0.735	29.96 0.855	29.83 0.847	29.95 0.855	30.08 0.861	30.00 0.856	30.01 0.855	30.25 0.860
	35	22.58 0.427	26.25 0.660	25.27 0.599	25.80 0.657	27.86 0.778	27.84 0.772	27.92 0.784	28.00 0.785	27.92 0.777	28.00 0.786	28.24 0.794
	50	21.29 0.330	23.33 0.505	23.57 0.529	24.40 0.573	25.93 0.681	25.86 0.673	25.96 0.687	25.88 0.675	26.00 0.685	25.97 0.684	26.30 0.708
Planaria (CF) [41]	**15**	29.51 0.545	36.73 0.848	38.18 0.905	35.86 0.850	38.39 0.909	38.43 0.909	38.36 0.908	37.80 0.873	38.44 0.908	38.48 0.910	38.54 0.913
	25	27.46 0.459	34.64 0.811	36.73 0.889	34.09 0.807	37.17 0.894	37.23 0.893	37.00 0.889	37.05 0.894	37.10 0.891	37.25 0.894	37.30 0.899
	35	26.19 0.411	31.49 0.600	31.44 0.676	32.34 0.751	36.21 0.882	36.17 0.877	36.10 0.875	36.03 0.880	36.39 0.886	36.31 0.879	36.37 0.888
	50	25.16 0.386	31.28 0.686	32.19 0.791	30.26 0.674	34.60 0.842	34.96 0.875	34.83 0.854	34.51 0.849	35.19 0.866	35.07 0.857	35.13 0.872
U2OS cells (CF) [78]	**15**	27.40 0.622	36.65 0.932	37.94 0.966	34.25 0.914	38.61 0.969	38.55 0.967	38.43 0.967	38.21 0.965	39.00 0.969	38.66 0.969	39.00 0.971
	25	23.07 0.398	34.43 0.923	35.40 0.945	32.38 0.892	36.32 0.957	36.35 0.957	36.03 0.953	35.83 0.954	36.43 0.957	36.48 0.959	36.83 0.962
	35	20.40 0.263	31.07 0.784	31.22 0.892	31.12 0.866	34.70 0.948	34.63 0.944	34.45 0.945	34.13 0.941	35.35 0.955	34.87 0.950	35.43 0.956
	50	18.03 0.164	28.45 0.665	25.66 0.759	28.57 0.791	32.58 0.925	32.80 0.931	32.22 0.921	32.27 0.923	33.39 0.938	32.76 0.930	33.41 0.940

stepped up to 20, as suggested in [20]. Test set involves FM images with real MPG noise. For the set FMDTest40, different noise levels are obtained by averaging $S = \{1, 2, 4, 8, 16\}$ raw images of the same scene [35]. For the external test images, the Poisson noise dominated samples were imaged with short excitation times and/or reduced laser power (in confocal micrographs) [25], [40], [41], [78].

Before passing on to the comparative analyses, we explore the effectiveness of GAT usage prior to DnCNN denoising. We run fine-tuned DnCNN (TAILORED) model, with

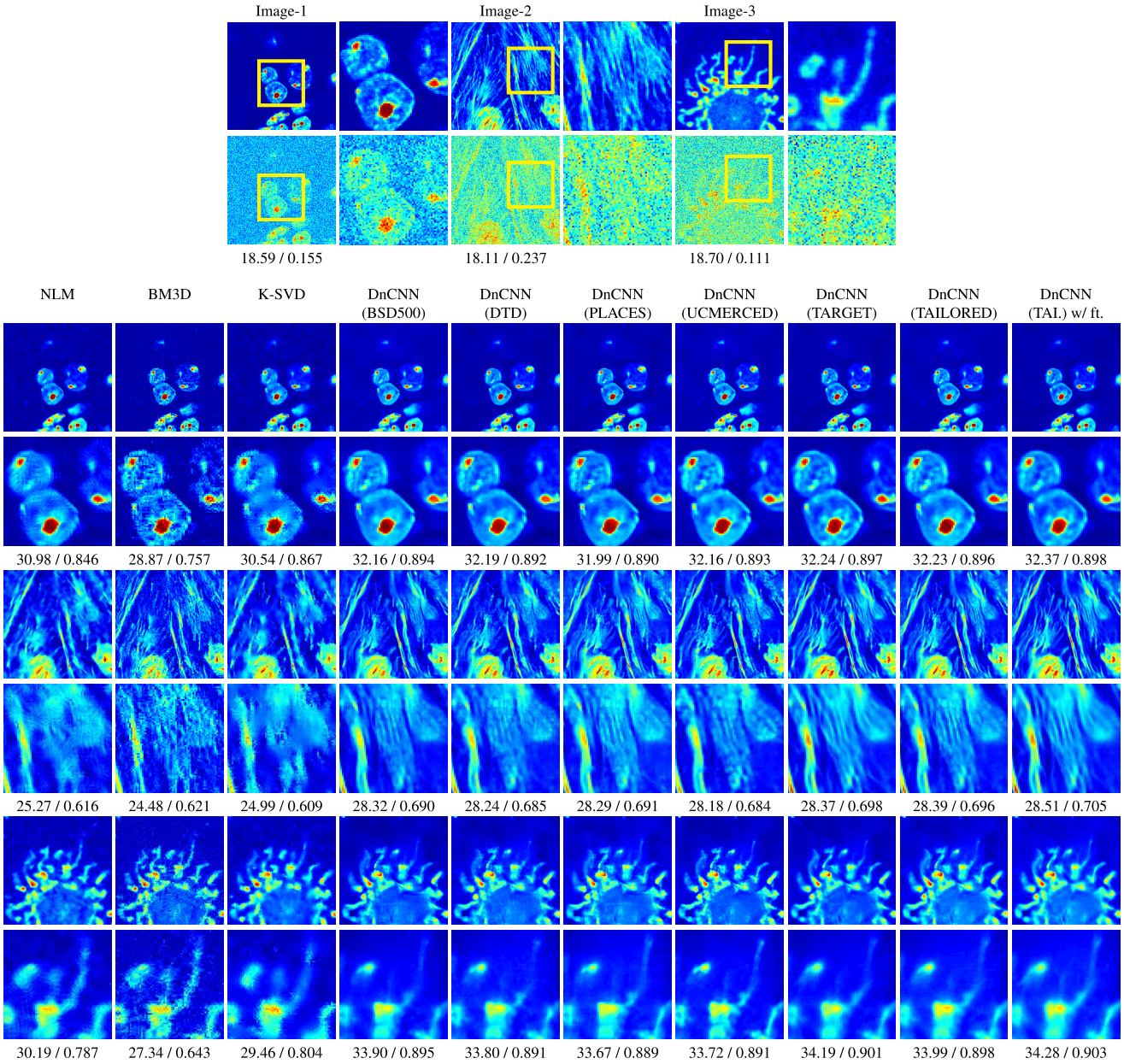


FIGURE 6. **Denoising of three different images from FMDTest40, degraded by Gaussian noise of $\sigma = 35$. The first and the second rows show clean and noisy images, respectively. The quantity pairs shown at the bottom of each image is correspoding to PSNR / SSIM values. The images are presented by using a different colormap to stress the details. The abbreviations TAI. and ft. stand for TAILORED and fine-tuning, respectively.**

and without GAT module, on FMDTest40 images with different MPG noise levels. Since there are five levels ($|S| = 5$), we have 200 different image instances in total. For each instance, Fig. 7 illustrates the difference between the PSNR scores produced by the fine-tuned DnCNN(TAILORED) model with and without GAT incorporation (i.e., gap = psnr(GAT+DnCNN(y),x) - psnr(DnCNN(y),x)). Fig. 7 (a) plots the gaps with respect to the estimated scaling coefficient of the detector, whereas Fig. 7 (b) plots them as a function of the variance estimate of the Gaussian noise. The same data points across two plots are marked in the same color. It is observed that the GAT usage shows quantitative improvement up to nearly 2.5 dB (0.47 dB on average) for 148 out of 200 instances (PSNR gap > 0). In 25 instances, the results are neither negatively nor positively affected (PSNR gap = 0). These are actually corresponding to the cases when the estimated value of the parameter a is negative. On the other hand, we observe 27 negatively valued PSNR gaps addressing to the situations where the GAT incorporation has failed (-0.29 dB on average). One may observe that, many data points with relatively large b values also have negative a values. In such cases,

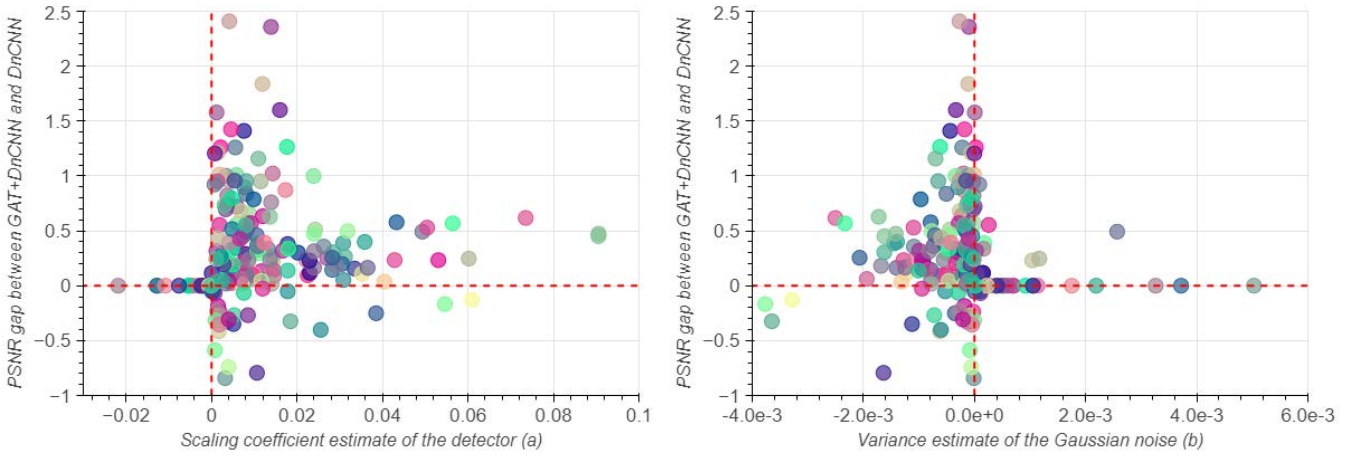


FIGURE 7. PSNR differences between GAT + DnCNN and DnCNN as a function of (a) scaling coefficient estimate of the detector and (b) variance estimate of the Gaussian noise, for each FMDTest40 image with MPG noise. The higher the gap value is, the more effective it is to incorporate variance stabilization. Negative gaps indicate the cases where the GAT incorporation has failed.

TABLE 3. Resulting p values of paired sample t-tests undertaken on the FMDTest40 images for four different Gaussian noise settings. Red highlights the significance of using our fine-tuned DnCNN (TAILORED) model. Blue marks results that show no statistical significance ($p > 1e-07$). Some dataset names are abbreviated using only the first three letters (e.g., TAI. stands for TAILORED).

	BSD.	DTD	PLA.	UCM.	TAR.	TAI.
TAI. (w/ ft.)	9.9e-29	1.4e-43	1.3e-44	8.6e-27	4.2e-22	4.8e-23
TAILORED	8.9e-16	1.8e-15	1.0e-36	1.2e-14	9.8e-06	
TARGET	2.9e-15	2.8e-22	5.1e-32	2.5e-16		
UCMERCED	2.0e-07	5.4e-07	4.9e-02			
PLACES	2.1e-07	3.5e-12				
DTD	9.3e-01					

our framework bypasses the GAT module, which results in no difference (PSNR gap = 0). Among the rest of the data points, the ones that satisfy the ''large a, small b'' condition mostly have satisfactory PSNR gap values, although some exceptions (PSNR gap < 0) exist.

As the baseline methods, we again test NLM, BM3D, and K-SVD in combination with GAT. We also involve PURE-LET[11] [30] as a representative of the state-of-the-art Poisson denoisers. The rest of the competing methods are all DnCNN-B models trained on different datasets, as in Experiment 1. The PSNR/SSIM scores on FMDTest40 are reported in Table 4. The columns are corresponding the number of raw images averaged. So the strength of the noise decreases as this number increases. Each row is corresponding to a different method or model. As in Table 1, each cell contains PSNR ± SE at the top and SSIM ± SE at the bottom. The text color red highlights the best, while the blue marks the second best. When the strength of the noise is high, the baseline methods are poorly functioning. The PSNR gap between the worst performing DnCNN model and the best performing baseline method PURE-LET is around 4.5 dB. As the noise settles down, this gap drops considerably. When the number of images averaged is 16, GAT + BM3D becomes the best performing baseline and produces even better PSNR and SSIM results than the DnCNN models trained on BSD500, DTD, and PLACES. Among the DnCNN models, the one trained on BSD500 produced the worst results for all noise levels. It seems the structure-rich content of the natural images is not a good fit for FMDTest40 data in this experimentation. Another structure-rich dataset PLACES seems more effective, but still be beaten by the texture-rich DTD. On the other hand, the reason why UCMERCED dataset worked better than the other source datasets might be the presence of both textured and homogeneous areas in its data. Although the results obtained by using our TAILORED dataset are consistently above the ones obtained by four source datasets, they are not as promising as those reported in Table 1. This is also the case for our fine tuned GAT + DnCNN (TAILORED) model, which falls behind GAT + DnCNN (TARGET) in this experimentation. It is observed that the performance gap between these two models increases at low noise levels, in parallel with the fact that the performance differences between the models become more apparent as the level of noise decreases. Our fine tuned model shows 0.9 dB improvement over the best-performing source model at the lowest noise level, whereas it falls 0.3 dB behind the TARGET model. This may seem in contrast to our hypothesis that the unrelated data with high resemblance to the low-level features of the target data may even do better. However, one should note that the set TARGET consists of exactly the same configurations as the FMDTest40. The results are actually remarkable, since the tailored dataset is successful enough to almost replace the TARGET dataset. This provides us a substantial facility that can be utilized when the data

[11] https://github.com/hijizhou/PureLetDeconv

TABLE 4. PSNR / SSIM comparisons on MPG noise denoising of the FMDTest40 [35].

# of images	1	2	4	8	16
Observed Images	23.55 ± 0.080 0.504 ± 0.004	26.25 ± 0.081 0.632 ± 0.004	28.97 ± 0.083 0.746 ± 0.003	31.65 ± 0.084 0.843 ± 0.002	34.86 ± 0.087 0.918 ± 0.001
GAT+NLM [5]	27.33 ± 0.121 0.764 ± 0.004	29.31 ± 0.107 0.827 ± 0.003	31.56 ± 0.103 0.872 ± 0.002	33.10 ± 0.095 0.898 ± 0.002	34.74 ± 0.089 0.921 ± 0.002
GAT + BM3D [7]	27.66 ± 0.126 0.777 ± 0.005	29.63 ± 0.110 0.834 ± 0.003	31.88 ± 0.104 0.877 ± 0.002	33.54 ± 0.097 0.903 ± 0.002	35.56 ± 0.095 0.928 ± 0.002
GAT + K-SVD [4]	26.55 ± 0.187 0.735 ± 0.006	28.87 ± 0.149 0.816 ± 0.004	31.19 ± 0.096 0.866 ± 0.002	33.08 ± 0.165 0.899 ± 0.004	35.44 ± 0.171 0.932 ± 0.002
PURE-LET [30]	28.23 ± 0.103 0.762 ± 0.004	29.72 ± 0.103 0.812 ± 0.003	31.13 ± 0.098 0.856 ± 0.002	32.62 ± 0.094 0.889 ± 0.002	34.44 ± 0.089 0.912 ± 0.002
GAT + DnCNN (BSD500)	32.69 ± 0.091 0.858 ± 0.002	33.45 ± 0.100 0.870 ± 0.002	33.86 ± 0.106 0.873 ± 0.002	34.20 ± 0.109 0.875 ± 0.002	34.69 ± 0.110 0.877 ± 0.002
GAT + DnCNN (DTD)	32.69 ± 0.096 0.865 ± 0.002	33.93 ± 0.093 0.889 ± 0.002	34.52 ± 0.103 0.894 ± 0.002	34.90 ± 0.105 0.895 ± 0.002	35.44 ± 0.106 0.896 ± 0.002
GAT + DnCNN (PLACES)	32.82 ± 0.093 0.867 ± 0.002	33.80 ± 0.099 0.882 ± 0.002	34.29 ± 0.106 0.886 ± 0.002	34.67 ± 0.108 0.887 ± 0.002	35.24 ± 0.108 0.890 ± 0.002
GAT + DnCNN (UCMERCED)	32.68 ± 0.096 0.864 ± 0.002	34.02 ± 0.098 0.887 ± 0.002	34.26 ± 0.105 0.893 ± 0.002	35.14 ± 0.108 0.896 ± 0.002	35.79 ± 0.109 0.898 ± 0.002
GAT + DnCNN (TARGET)	33.13 ± 0.094 0.871 ± 0.002	34.53 ± 0.094 0.905 ± 0.001	35.42 ± 0.099 0.917 ± 0.001	36.12 ± 0.103 0.923 ± 0.001	37.02 ± 0.104 0.929 ± 0.001
GAT + DnCNN (TAILORED)	32.94 ± 0.092 0.871 ± 0.002	34.16 ± 0.096 0.897 ± 0.002	34.84 ± 0.104 0.904 ± 0.002	35.36 ± 0.107 0.908 ± 0.002	36.05 ± 0.108 0.911 ± 0.002
GAT + DnCNN (TAILORED) w/ ft.	33.11 ± 0.097 0.877 ± 0.002	34.48 ± 0.094 0.904 ± 0.001	35.29 ± 0.101 0.914 ± 0.001	35.89 ± 0.104 0.917 ± 0.001	36.68 ± 0.104 0.920 ± 0.001

shortage is the case. Even though the fine tuned model takes the support of some target data, the number of fluorescence images is far less than the ones used by the target model.

As was done in the first set of experiments, we also tried MPG noise denoisers on the external test images. Table 5 reports PSNR and SSIM scores for each type of external test data produced by using competing methods and models. When the results are compared to those in Table 2, the benefit of using TAILORED dataset over TARGET dataset does not seem to be significant. Even so, our GAT + DnCNN (TAILORED) model almost always (with the exception of PSNR result obtained on Planaria) produced better results than the models trained on the source datasets. Moreover, in some cases (e.g., HeLa, Nucleus, Fly Wings), our model achieved better quantitative results (either PSNR or SSIM) than GAT + DnCNN (TARGET). These observations support that these unrelated images in TAILORED dataset can nearly substitute the target data. As was observed in the previous experiments, this gain even becomes higher when we fine-tune the model with a few fluorescence microscopy images. Except for the case of denoising raw Actin images, our training set tailoring approach (with/without fine-tuning) achieved competitive results against the approach of training on the fluorescence microscopy images only. Fig. 8 shows some detail results obtained by using baseline methods, whereas the detail results obtained by the DnCNN models are demonstrated in Fig. 9. The very first column of Fig. 8 shows the detail patches cropped from the input images suffering from MPG noise. The very last column of Fig. 9 shows the same detail images cropped from the ground-truth images. Note that, the PSNR / SSIM pairs given below are for the entire images rather than the cropped ones. This time, unlike the previous results shown in Fig. 6, the improvement brought by using TAILORED dataset with fine-tuning is visually less perceptible. By zooming in, one may realize that training on the TAILORED data tends to yield artificial structures, especially on the originally smooth regions, whereas training on the TARGET data tends to over-smooth some sharp contents. The fine-tuned GAT + DnCNN (TAILORED) model, on the other side, demonstrates slight visual improvements over using target data. However, it seems to be offering a good balance between the benefits of using TARGET and TAILORED datasets, aside from requiring a smaller amount of fluorescence data.

We run paired sample t-tests also for the MPG noise denoising experiments. The tests are conducted over 40 test images and for five different noise settings. The null and the alternative hypotheses are addressing to the same statements that are reported before. In Table 6, we report the p values obtained for each pair of DnCNN models trained by using

TABLE 5. PSNR / SSIM comparisons on MPG noise denoising of external test set.

	HeLa (CF)	Actin (WF)	Mito (WF)	Membrane (WF)	Nucleus (WF)	Fly Wings (CF)	Planaria (CF)	U2OS cells (CF)
Observed Images	20.23 0.239	22.54 0.416	25.08 0.522	30.14 0.689	26.66 0.509	23.23 0.478	26.99 0.571	18.40 0.267
GAT + NLM [5]	27.05 0.669	29.19 0.795	30.38 0.718	35.62 0.943	34.98 0.886	25.26 0.700	28.74 0.721	23.02 0.685
GAT + BM3D [7]	27.29 0.711	29.79 0.825	30.94 0.746	35.85 0.949	35.30 0.895	25.77 0.734	28.97 0.722	21.70 0.550
GAT + K-SVD [4]	27.36 0.695	28.05 0.742	30.09 0.830	34.18 0.890	36.53 0.910	26.92 0.780	31.43 0.729	20.29 0.404
PURE-LET [30]	25.53 0.644	28.33 0.793	30.20 0.752	33.17 0.941	35.28 0.897	25.51 0.712	30.90 0.857	25.33 0.808
GAT + DnCNN (BSD500)	29.05 0.840	30.76 0.833	31.97 0.892	34.71 0.887	36.82 0.903	27.19 0.780	32.21 0.826	26.04 0.782
GAT + DnCNN (DTD)	29.01 0.838	30.75 0.833	31.96 0.862	35.08 0.905	36.86 0.901	27.07 0.776	32.16 0.818	26.09 0.787
GAT + DnCNN (PLACES)	28.97 0.836	30.70 0.830	32.02 0.896	35.07 0.906	36.83 0.905	27.22 0.782	32.26 0.832	26.05 0.783
GAT + DnCNN (UCMERCED)	28.92 0.833	30.81 0.838	32.16 0.890	35.09 0.906	36.85 0.908	27.25 0.784	32.29 0.834	25.77 0.774
GAT + DnCNN (TARGET)	29.08 0.845	31.00 0.843	32.34 0.902	35.35 0.912	36.89 0.906	27.37 0.788	32.14 0.824	26.79 0.801
GAT + DnCNN (TAILORED)	29.13 0.842	30.92 0.838	32.14 0.895	35.25 0.912	37.07 0.913	27.33 0.789	32.21 0.843	26.24 0.784
GAT + DnCNN (TAILORED) w/ fine-tuning	29.18 0.847	30.98 0.840	32.39 0.901	35.37 0.914	36.93 0.910	27.41 0.788	32.86 0.838	26.93 0.808

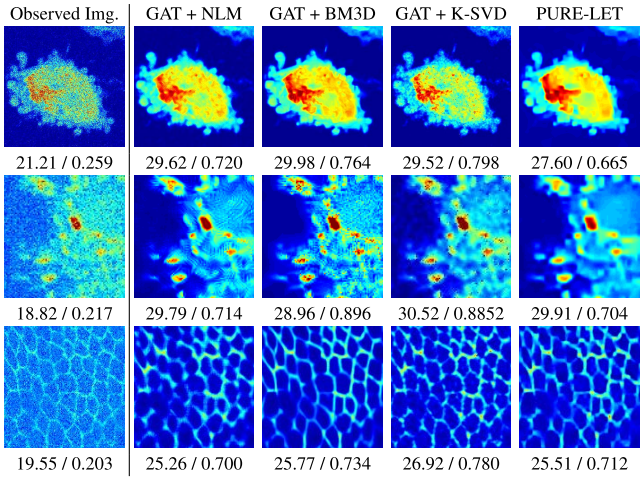


FIGURE 8. Denoising of short exposure external test images by using baseline methods. The images are the detail patches cropped from the outputs and presented by using a different colormap. First column shows the detail patches of the observed images with real MPG noise. The results with the DnCNN models are shown in Fig. 9.

six different datasets (plus, one set of fine-tuning data). This time, each test has 199 degrees of freedom. The threshold is selected as $\alpha = 1e-07$, as was done earlier. Additional to the blue and red markers, we also use underline to highlight that the TARGET's success compared to the TAILORED (with fine-tuning) seems statistically significant ($p > 1e-07$).

IV. DISCUSSION

Overall, we tackled denoising of fluorescence microscopy (FM) images. These images often suffer from the mixed Poisson-Gaussian (MPG) noise. That is one of the reasons why in the literature it is rare to come across deep denoisers used to restore fluorescence micrographs, since such networks mostly assume that the noise is Gaussian. Yet another and the major reason for the supervised deep denoisers is

TABLE 6. Resulting p values of paired sample t-tests undertaken on the FMDTest40 images for five different MPG noise settings. Red highlights the significance of using our fine-tuned GAT + DnCNN (TAILORED) model. Blue marks results that show no statistical significance ($p > 1e-07$). Underline shows the significance of using target data instead of our tailored dataset with fine-tuning. Some dataset names are abbreviated using only the first three letters (e.g., TAI. stands for TAILORED).

	BSD.	DTD	PLA.	UCM.	TAR.	TAI.
TAI. (w/ ft.)	9.5e-43	1.1e-54	2.4e-44	1.1e-25	4.6e-09	1.5e-35
TAILORED	1.6e-39	4.8e-34	4.1e-38	2.3e-02	2.1e-34	
TARGET	3.9e-44	4.3e-45	3.6e-43	4.5e-32		
UCMERCED	1.8e-24	7.7e-08	9.1e-14			
PLACES	5.9e-26	3.3e-04				
DTD	1.2e-15					

the lack of clean data. The images taken under high photon exposure, or the average images of the same scene may serve as ground-truths; however, the former may cause phototoxicity, while the later is pretty exhaustive. Plus, in both cases, the images acquired wouldn't be competent enough to represent the structural richness of the microscopic world. Here is where the transfer learning comes into play, which suggests to borrow some domain-invariant information from unrelated sources. This strategy is applied not only because the amount of data in the target domain is limited and calls for augmentation, but also to promote diversity for better generalization. In our work, we asked two questions: (1) could a dataset of unrelated images tailored by considering low-level similarities to the fluorescence data can substitude the real fluorescence images, and (2) could such a dataset even do better? We achieved tailoring a dataset (i.e., TAILORED) that partly confirms the former question. Our work showed that the target data (i.e., TARGET) and a tailored dataset of unrelated images with low-level similarities to the target data could produce comparable results with a few exceptions. Only when the noise was MPG and the test set was involving the same configurations with those in the target dataset (i.e., FMDTest40), the tailored dataset prominently fell behind the TARGET. Even so, its performance was still superior to the arbitrarily chosen source datasets. These visual and quantitative experiments show that our tailored dataset strikes a good balance between the denoising performance and the data availability, to which the success of a supervised deep denoiser largely depends on. Namely, our approach offers a computational way to choose training images from potentially large unrelated sources at the cost of slight decrease in denoising performance. This is a substantial facility since the amount of fluorescence data is often not sufficient. When it comes to our second question, we have shown that the model trained on our tailored dataset can produce better results with the aid of a few fluorescence images. Fine-tuning the DnCNN (TAILORED) model on a small dataset of target images (involving 120 fluorescence images in our case) came up with a significantly superior model against all of the other competing models in the majority of the cases. There exist exceptions (especially those reported in

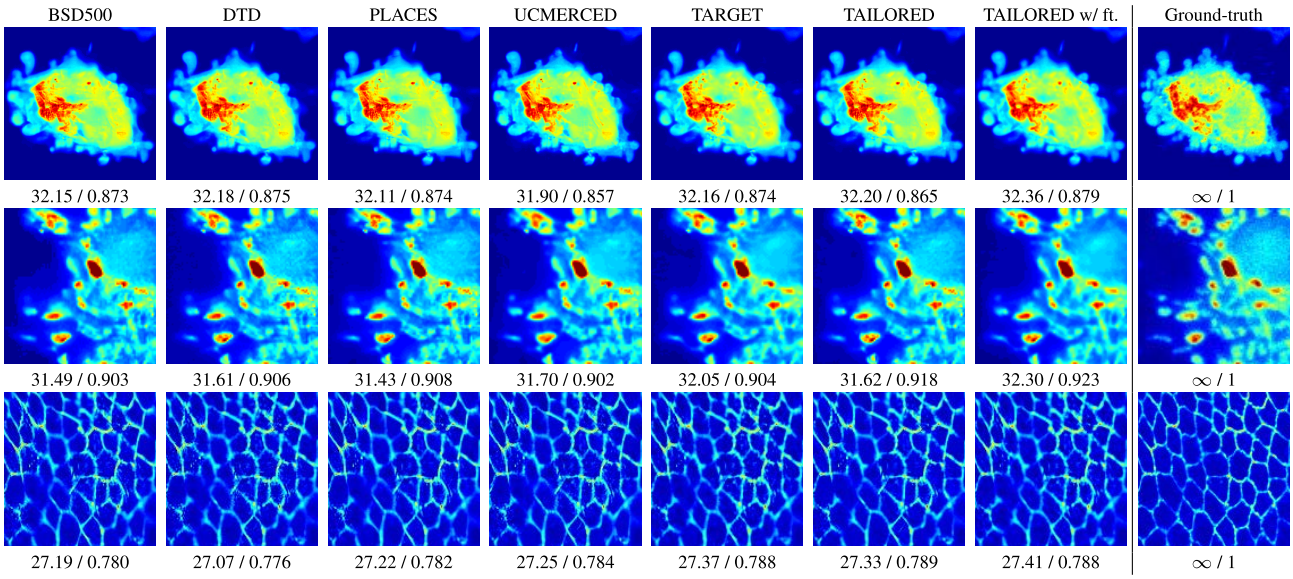


FIGURE 9. **(Continued from Fig. 8) Denoising of short exposure HeLa cell images by using GAT + DnCNN (·), where · stands for dataset name. The images are the detail patches cropped from the outputs and presented by using a different colormap. Last column shows the detail patches of the ground-truth images.**

Table 4), where our fine-tuned model performed comparable to DnCNN (TARGET). In this way, the need for fluorescence data has been considerably reduced, although not completely eliminated. In brief, our work gives insights on how the considered low-level features can be exploited to approximate inadequate target data, how beneficial they might be in balancing content-awareness and universality, and how discriminative they become in a restoration task. Recall that, while tailoring, we only took the low-level similarities into account by encoding descriptors for color, local symmetry, and directionality related features.

As we touched upon in Section I, various transfer learning approaches are employed in the field of microscopy imaging. However, the literature is dominated by detection/classification studies, rather than restoration. A Google Scholar search for *"restoration" or "denoising" and "microscopy" and "transfer learning"* returned 361 hits as of April 2022, but many of the returned papers were still on detection and recognition tasks (due to some coinciding keywords as in ''stacked denoising autoencoders''). We screened those papers by relevance and realized three very recent transfer learning models that we wouldn't finish without referring to. In [80], the authors propose a universal reconstruction scheme for structured illumination microscopy (SIM). Their model generalizes beyond the task by transferring knowledge from a large and diverse dataset of unrelated high-resolution images (i.e., DIV2K [81]). When compared to our strategy, one can see that they do not run a selective procedure. They generate the training data by transforming all the images in the source dataset, such that the imaging conditions of SIM are simulated. One interesting research direction here might involve the realization of this strategy in fluorescence microscopy domain as well, and combining it with our tailoring procedure. In [82], on the other side, the authors propose a scheme that transfers knowledge learned from synthetic noise data to the real noise domain. Another engaging study [83] uses transfer learning to combine supervised and self-supervised training, in order to eliminate the demand to the clean images. Our approach may further be reconsidered in combination with these studies as well.

In order not to finish discussion without referring to the state-of-the-art for fluorescence microscopy image restoration one more time, we want to draw the reader's attention to a recent preprint [45]. In this preprint, the authors discuss that the pixel-intensity distributions of the microscopy images differ from those of natural images and a successful denoiser should take this into account. They refer to the DnCNN as the state-of-the-art for photographic image noise reduction, which is not that good at denoising transmission electron microscope (TEM) images. They attribute its insufficiency to its very small field of view (41×41 pixels), and state that the electron micrographs call for larger field of view, due to the prominent global regularities that they exhibit. The U-net architecture [42] allows us to increase the field of view without dramatically increasing the number of parameters. Therefore, the success of CARE [41] and Noise2noise [24] in microscopy imaging might be due to their U-net backbone with a large field of view. Our training dataset tailoring approach may be considered as an attempt to promote the distinctive global regularities in fluorescence microscopy images. Another attempt may be made to further improve the performance by using a U-net architecture, instead of DnCNN. Our approach can be tried with CARE [41] at the cost of relaxing the strong prior on the visual characteristics of the task-specific data that the CARE networks aim to learn. It can even be exploited by Noise2noise [24] and its variants

to reduce the demand to the low-SNR fluorescence images as well.

V. CONCLUSION

Acquiring clean fluorescence microscopy (FM) images is an inconvenient process. This poses an obstacle for restoring noisy micrographs through the use of supervised deep denoisers. In this respect, this paper suggested to tailor a dataset of images coming from various unrelated repositories, with the hope of approximating the fluorescence data up to a point. While gathering images, we considered some low-level view-based features. Our tailoring process carefully chooses the images whose colors, basic image features, and the directional characteristics are similar to those of fluorescence images. On top of training a feed-forward denoising convolutional neural network (DnCNN) model with our tailored dataset, we also applied fine-tuning to encourage content-awareness. In order to combat MPG noise, we incorporated the generalized Anscombe transform (GAT), which is a variance stabilization technique. Finally, we assessed the performance of our approach on both Gaussian and MPG noise denoising problems. We performed tests on both unseen fluorescence images of the same configurations (sample + fluorescence microscopy type/settings + FoV) with those in the target dataset, and the images of entirely different configurations. Overall, the experiments prove that the proposed approach (with fine-tuning) produces either superior or comparable results against the models trained on the real fluorescence data. Furthermore, it outperforms all of the other models, trained on the arbitrary datasets. We can conclude that tailoring such a dataset strikes a good balance between the restoration quality and the data availability. Future work may involve many research directions. The tailored dataset may further be exposed to some transformations that simulate the imaging conditions of the fluorescence microscopy. The image descriptors used to define similarity might be enriched, maybe by inspecting the intermediate layers of a deep denoiser. Our approach might be tested on a different application field, where the data shortage is the case. Multiple datasets by considering different feature groups might be tailored and these datasets might be used to train multiple networks to be combined in an ensemble.

REFERENCES

[1] J. Pawley, *Handbook of Biological Confocal Microscopy*, vol. 236. New York, NY, USA: Springer, 2006.

[2] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, 1992.

[3] A. Chambolle, R. A. De Vore, N.-Y. Lee, and B. J. Lucier, "Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 319–335, Mar. 1998.

[4] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[5] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 60–65.

[6] G. Gilboa and S. Osher, "Nonlocal operators with applications to image processing," *Multiscale Model. Simul.*, vol. 7, no. 3, pp. 1005–1028, 2008.

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

[8] D. C. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Mitosis detection in breast cancer histology images with deep neural networks," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Springer, 2013, pp. 411–418.

[9] S. Chilamkurthy, R. Ghosh, S. Tanamala, and M. Biviji, "Deep learning algorithms for detection of critical findings in head CT scans: A retrospective study," *Lancet*, vol. 392, no. 10162, pp. 2388–2396, Dec. 2018.

[10] C. Angermueller, H. J. Lee, W. Reik, and O. Stegle, "DeepCpG: Accurate prediction of single-cell DNA methylation states using deep learning," *Genome Biol.*, vol. 18, no. 1, pp. 1–13, Dec. 2017.

[11] R. Singh, J. Lanchantin, G. Robins, and Y. Qi, "DeepChrome: Deep-learning for predicting gene expression from histone modifications," *Bioinformatics*, vol. 32, no. 17, pp. i639–i648, Sep. 2016.

[12] B. P. Nguyen, "Prediction of FMN binding sites in electron transport chains based on 2-D CNN and PSSM profiles," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 18, no. 6, pp. 2189–2197, Nov./Dec. 2021.

[13] N. Q. K. Le and Q.-T. Ho, "Deep transformers and convolutional neural network in identifying DNA N6-methyladenine sites in cross-species genomes," *Methods*, vol. 204, pp. 199–206, Aug. 2022.

[14] J. Jin, J. Q. Lu, Y. Wen, P. Tian, and X. Hu, "Deep learning of diffraction image patterns for accurate classification of five cell types," *J. Biophoton.*, vol. 13, no. 3, Mar. 2020, Art. no. e201900242.

[15] C. L. Chen, A. Mahjoubfar, L.-C. Tai, I. K. Blaby, A. Huang, K. R. Niazi, and B. Jalali, "Deep learning in label-free cell classification," *Sci. Rep.*, vol. 6, no. 1, pp. 1–16, Mar. 2016.

[16] F. Qin, N. Gao, Y. Peng, Z. Wu, S. Shen, and A. Grudtsin, "Fine-grained leukocyte classification with deep residual learning for microscopic images," *Comput. Methods Programs Biomed.*, vol. 162, pp. 243–252, Aug. 2018.

[17] V. Jain and S. Seung, "Natural image denoising with convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 769–776.

[18] J. Xie, L. Xu, and E. Chen, "Image denoising and inpainting with deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 341–349.

[19] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1256–1272, Jun. 2017.

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

[21] Y. Tai, J. Yang, X. Liu, and C. Xu, "MemNet: A persistent memory network for image restoration," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4539–4547.

[22] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for CNN-based image denoising," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4608–4622, Sep. 2018.

[23] S. Lefkimmiatis, "Universal denoising networks: A novel CNN architecture for image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3204–3213.

[24] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila, "Noise2Noise: Learning image restoration without clean data," 2018, *arXiv:1803.04189*.

[25] W. Meiniel, J.-C. Olivo-Marin, and E. D. Angelini, "Denoising of microscopy images: A review of the state-of-the-art, and a new sparsity-based method," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3842–3856, Aug. 2018.

[26] A. Jezierska, C. Chaux, J.-C. Pesquet, and H. Talbot, "An EM approach for Poisson-Gaussian noise modeling," in *Proc. 19th Eur. Signal Process. Conf.*, Aug./Sep. 2011, pp. 2244–2248.

[27] F. J. Anscombe, "The transformation of Poisson, binomial and negative-binomial data," *Biometrika*, vol. 35, nos. 3–4, pp. 246–254, 1948.

[28] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data," *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1737–1754, Oct. 2008.

[29] M. Makitalo and A. Foi, "Optimal inversion of the generalized Anscombe transformation for Poisson-Gaussian noise," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 91–103, Jan. 2013.

[30] F. Luisier, T. Blu, and M. Unser, "Image denoising in mixed Poisson–Gaussian noise," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 696–708, Mar. 2011.
[31] F. Luisier, C. Vonesch, T. Blu, and M. Unser, "Fast interscale wavelet denoising of Poisson-corrupted images," *Signal Process.*, vol. 90, no. 2, pp. 415–427, Feb. 2010.
[32] J. Li, F. Luisier, and T. Blu, "PURE-LET image deconvolution," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 92–105, Jan. 2018.
[33] V. Pronina, F. Kokkinos, D. V. Dylov, and S. Lefkimmiatis, "Microscopy image restoration with deep Wiener-Kolmogorov filters," 2019, *arXiv:1911.10989*.
[34] J. Roels, J. Aelterman, H. Luong, S. Lippens, A. Pižurica, Y. Saeys, and W. Philips, "An overview of state-of-the-art image restoration in electron microscopy," *J. Microsc.*, vol. 271, no. 3, pp. 239–254, 2018.
[35] Y. Zhang, Y. Zhu, E. Nichols, Q. Wang, S. Zhang, C. Smith, and S. Howard, "A Poisson-Gaussian denoising dataset with real fluorescence microscopy images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11710–11718.
[36] A. Krull, T.-O. Buchholz, and F. Jug, "Noise2Void-learning denoising from single noisy images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 2129–2137.
[37] A. Krull, T. Vičar, M. Prakash, M. Lalit, and F. Jug, "Probabilistic Noise2 Void: Unsupervised content-aware denoising," *Frontiers Comput. Sci.*, vol. 2, p. 5, Feb. 2020.
[38] V. Mannam, Y. Zhang, Y. Zhu, E. Nichols, Q. Wang, V. Sundaresan, S. Zhang, C. Smith, P. W. Bohn, and S. S. Howard, "Real-time image denoising of mixed Poisson–Gaussian noise in fluorescence microscopy images using ImageJ," *Optica*, vol. 9, no. 4, pp. 335–345, 2022.
[39] J. Batson and L. Royer, "Noise2Self: Blind denoising by self-supervision," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 524–533.
[40] G. M. Hagen, J. Bendesky, R. Machado, T.-A. Nguyen, T. Kumar, and J. Ventura, "Fluorescence microscopy datasets for training deep neural networks," *GigaScience*, vol. 10, no. 5, May 2021, Art. no. giab032.
[41] M. Weigert, U. Schmidt, T. Boothe, A. Müller, A. Dibrov, A. Jain, B. Wilhelm, D. Schmidt, C. Broaddus, and S. Culley, "Content-aware image restoration: Pushing the limits of fluorescence microscopy," *Nature Methods*, vol. 15, no. 12, p. 1090, 2018.
[42] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Berlin, Germany: Springer, 2015, pp. 234–241.
[43] E. Moen, D. Bannon, T. Kudo, W. Graf, and M. Covert, "Deep learning for cellular image analysis," *Nature*, vol. 16, no. 12, pp. 1233–1246, May 2019.
[44] T.-O. Buchholz, M. Jordan, G. Pigino, and F. Jug, "Cryo-CARE: Content-aware image restoration for cryo-transmission electron microscopy data," in *Proc. IEEE 16th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2019, pp. 502–506.
[45] S. Mohan, R. Manzorro, J. L. Vincent, B. Tang, D. Yashpal Sheth, E. P. Simoncelli, D. S. Matteson, P. A. Crozier, and C. Fernandez-Granda, "Deep denoising for scientific discovery: A case study in electron microscopy," 2020, *arXiv:2010.12970*.
[46] V. Lempitsky, A. Vedaldi, and D. Ulyanov, "Deep image prior," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9446–9454.
[47] T. Vu, A. DiSpirito, D. Li, Z. Wang, X. Zhu, M. Chen, L. Jiang, D. Zhang, J. Luo, Y. S. Zhang, Q. Zhou, R. Horstmeyer, and J. Yao, "Deep image prior for undersampling high-speed photoacoustic microscopy," *Photoacoustics*, vol. 22, Jun. 2021, Art. no. 100266.
[48] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2223–2232.
[49] S. Lim, H. Park, S.-E. Lee, S. Chang, B. Sim, and J. C. Ye, "CycleGAN with a blur kernel for deconvolution microscopy: Optimal transport geometry," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 1127–1138, 2020.
[50] T. Schlegl, J. Ofner, and G. Langs, "Unsupervised pre-training across image domains improves lung tissue classification," in *Proc. Int. MICCAI Workshop Med. Comput. Vis.* Berlin, Germany: Springer, 2014, pp. 82–93.
[51] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 215–223.
[52] A. Menegola, M. Fornaciali, R. Pires, F. V. Bittencourt, S. Avila, and E. Valle, "Knowledge transfer for melanoma screening with deep learning," in *Proc. IEEE 14th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2017, pp. 297–300.
[53] B. Graham, "Kaggle diabetic retinopathy detection competition report," Univ. Warwick, Coventry, U.K., Tech. Rep., 2015.
[54] E. Ribeiro, M. Hafner, G. Wimmer, T. Tamaki, J. J. W. Tischendorf, S. Yoshida, S. Tanaka, and A. Uhl, "Exploring texture transfer learning for colonic polyp classification via convolutional neural networks," in *Proc. IEEE 14th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2017, pp. 1044–1048.
[55] J.-L. Starck, F. D. Murtagh, and A. Bijaoui, *Image Processing and Data Analysis: The Multiscale Approach*. Cambridge, U.K.: Cambridge Univ. Press, 1998.
[56] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
[57] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2014, pp. 818–833.
[58] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, Vancouver, BC, Canada, Jul. 2001, pp. 416–423.
[59] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3606–3613.
[60] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1452–1464, Jun. 2018.
[61] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2010, pp. 270–279.
[62] S. Arslan, T. Ersahin, R. Cetin-Atalay, and C. Gunduz-Demir, "Attributed relational graphs for cell nucleus segmentation in fluorescence microscopy images," *IEEE Trans. Med. Imag.*, vol. 32, no. 6, pp. 1121–1131, Jun. 2013.
[63] C. F. Koyuncu, R. Cetin-Atalay, and C. Gunduz-Demir, "Object-oriented segmentation of cell nuclei in fluorescence microscopy images," *Cytometry A*, vol. 93, no. 10, pp. 1019–1028, Oct. 2018.
[64] Y. Al-Kofahi, A. Zaltsman, R. Graves, W. Marshall, and M. Rusu, "A deep learning-based algorithm for 2-D cell segmentation in microscopy images," *BMC Bioinf.*, vol. 19, no. 1, pp. 1–11, Dec. 2018.
[65] L. D. Griffin, M. Lillholm, M. Crosier, and J. Van Sande, "Basic image features (BIFs) arising from approximate symmetry type," in *Proc. Int. Conf. Scale Space Variational Methods Comput. Vis.* Berlin, Germany: Springer, 2009, pp. 343–355.
[66] J.-L. Starck, E. J. Candès, and D. L. Donoho, "The curvelet transform for image denoising," *IEEE Trans. Image Process.*, vol. 11, no. 6, pp. 670–684, Dec. 2002.
[67] J. Frikel, "Sparse regularization in limited angle tomography," *Appl. Comput. Harmonic Anal.*, vol. 34, no. 1, pp. 117–141, 2013.
[68] I. Bayram and M. E. Kamasak, "Directional total variation," *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 781–784, Dec. 2012.
[69] E. Demircan-Tureyen and M. E. Kamasak, "Adaptive direction-guided structure tensor total variation," *Signal Process., Image Commun.*, vol. 99, Nov. 2021, Art. no. 116497.
[70] E. Demircan-Tureyen and M. E. Kamasak, "Nonlocal adaptive direction-guided structure tensor total variation for image recovery," *Signal, Image Video Process.*, vol. 15, pp. 1517–1525, Apr. 2021.
[71] A. A. Bharath and J. Ng, "A steerable complex wavelet construction and its application to image denoising," *IEEE Trans. Image Process.*, vol. 14, no. 7, pp. 948–959, Jul. 2005.
[72] S. Lefkimmiatis, A. Roussos, P. Maragos, and M. Unser, "Structure tensor total variation," *SIAM J. Imag. Sci.*, vol. 8, no. 2, pp. 1090–1122, 2015.
[73] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 886–893.
[74] J. Bigun and S. M. Karlsson, "Histogram of directions by the structure tensor," in *Proc. 4th Int. Symp. Appl. Sci. Biomed. Commun. Technol. (ISABEL)*, 2011, pp. 1–5.
[75] L. D. Griffin, M. H. Wahab, and A. J. Newell, "Distributional learning of appearance," *PLoS ONE*, vol. 8, no. 2, Feb. 2013, Art. no. e58074.
[76] L. D. Griffin. (2015). *Basic Image Features (BIFs) Implementation*. [Online]. Available: https://github.com/GriffinLab/BIFs
[77] A. Vedaldi and K. Lenc, "MatConvNet: Convolutional neural networks for MATLAB," in *Proc. 23rd ACM Int. Conf. Multimedia*, 2015, pp. 689–692.
[78] V. Ljosa, K. L. Sokolnicki, and A. E. Carpenter, "Annotated high-throughput microscopy image sets for validation," *Nat. Methods*, vol. 9, no. 7, p. 637, 2012.

[79] 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.
[80] C. N. Christensen, E. N. Ward, M. Lu, P. Lio, and C. F. Kaminski, "ML-SIM: Universal reconstruction of structured illumination microscopy images using transfer learning," *Biomed. Opt. Exp.*, vol. 12, no. 5, pp. 2720–2733, 2021.
[81] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 126–135.
[82] Y. Kim, J. W. Soh, G. Y. Park, and N. I. Cho, "Transfer learning from synthetic to real-noise denoising with adaptive instance normalization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3482–3492.
[83] Y. Wang, H. Pinkard, E. Khwaja, S. Zhou, L. Waller, and B. Huang, "Image denoising for fluorescence microscopy by supervised to self-supervised transfer learning," *Opt. Exp.*, vol. 29, no. 25, pp. 41303–41312, 2021.

EZGI DEMIRCAN-TUREYEN (Member, IEEE) received the B.Sc. degree in computer engineering from Istanbul Kültür University, Istanbul, Turkey, in 2010, and the M.Sc. degree in computer engineering from Istanbul Technical University, Istanbul, in 2015, where she is currently pursuing the Ph.D. degree with the Department of Computer Engineering. She does research on directional regularization-based variational models for image restoration tasks at the Department of Computer Engineering, Istanbul Technical University. She also works with the Department of Computer Engineering, Istanbul Kültür University, as a Lecturer. Her research interests include inverse imaging problems, convex optimization, medical imaging, computer vision, and machine learning.

FATMA PATLAR AKBULUT (Member, IEEE) received the B.S. and M.S. degrees in computer engineering from Istanbul Kültür University, Istanbul, Turkey, and the Ph.D. degree in biomedical engineering from Istanbul University, Istanbul, in 2017. Her doctoral work focused on developing machine learning enabled wearable systems for long-term cardiovascular disease monitoring. She is currently an Assistant Professor with the Computer Engineering Department, Istanbul Kültür University (IKU). Prior to IKU, she has worked as a Postdoctoral Researcher with the Computer Science Department and the Advanced Self-Powered Systems of Integrated Sensors and Technologies (ASSIST) Center, North Carolina State University (NCSU), from 2017 to 2019. Her research interests include biomedical signal processing, affective computing, data analytics, and wearable systems for healthcare.

MUSTAFA E. KAMASAK (Member, IEEE) received the B.S.E.E. and M.S.E.E. degrees from Boğazi çi University, Istanbul, Turkey, in 1997 and 1999, respectively, and the Ph.D. degree from Purdue University, West Lafayette, IN, USA, in 2005. He is currently a Professor with the Department of Computer Engineering, Istanbul Technical University, Istanbul. His research interests include medical informatics, medical imaging, and image processing.