Deep Learning for Denoising of Fluorescence Microscopy Images

Tram-Anh Nguyen
George Mason University
Fairfax, VA
tnguy150@gmu.edu

Guy Hagen and **Jonathan Ventura**
University of Colorado Colorado Springs
Colorado Springs, CO
ghagen@uccs.edu, jventura@uccs.edu

Abstract

Fluorescence microscopy images are often taken at low light and short exposure times to preserve the integrity of cell samples. However, imaging under these conditions leads to severely degraded images with low signal to noise ratios. To computationally restore these images, we introduce novel loss functions to denoise microscopy images. These loss functions will be folded into the CARE algorithm. The results produced by this modification will also be compared against its CARE predecessor using standard image quality metrics.

Introduction

Fluorescence microscopy is vital for understanding processes and structures at the cellular level. Because imaging at the cellular level under strong lighting conditions or long exposure times may damage the cell sample through phototoxicity, fluorescence microscopy images need to be restored. A safe way to image a cell is to use low light conditions and/or low exposure times, which unfortunately lowers the signal to noise ratio (Xing et al. 2017).

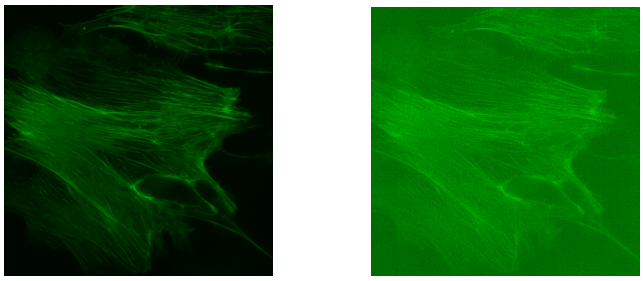

Figure 1: Fluorescence microscopy images taken of actin under high light (left) and low light conditions (right)

Noise in an image depends on a combination of factors, including exposure time and physical experimental conditions. In fluorescence microscopy, noise is typically described by a Poison-Gaussian model. There has been extensive work done in image restoration through filtering noise from microscopy images. These filtering techniques have limitations, which call for a more generalized solution.

Deep learning methods have been successful in restoring corrupted images, as well as in other image processing tasks such as classification, segmentation, and object detection. A widely used deep learning architecture in image processing is a convolutional neural network (CNN). A CNN consists of an input layer, hidden layers, and an output layer. Another popular network in image processing is an autoencoder. Typically, autoencoders are used for denoising and reducing the number of dimensions of input data (Xing et al. 2017).

Related Work

Deep learning approaches to image deblurring may involve blind and non-blind image deconvolution. There are a wealth of studies devoted to the non-blind image deconvolution approach, but these networks are limited, as they rely on information about the non-blurry image beforehand. By contrast, blind deblurring models are more flexible since information about the non-blurry image is not required for the network to deblur an input image.

In 2014, Xu et al. introduced a natural image deconvolution that is data-driven and does not rely on traditional assumptions. For example, generative models tend to assume that noise in an image is identically and independently distributed, even if this assumption is not necessarily true. Instead, this CNN was trained on images that were not deblurred ahead of time and the network learned the deconvolution operation without requiring information about the original image. The main contribution of Xu et al. was developing this deep convolutional neural network (DCNN) that consisted of two sub models—one for deconvolution and the other for denoising. The models perform inverse filtering using large 1D kernels and the former sub model is pre-trained to mimic Weiner deconvolution (Xu et al. 2014).

A major drawback of the previously discussed blind DCNN was that it failed if the original image was not blurry. To address this shortcoming, Conti et.al introduced a convolutional neural network that consisted of a regularization term in the cost function. This improved model was able to denoise a blurry image and maintain the quality of images that are not noisy. The major modifications made by Conti et al. were that they used single 2D convolutional layers rather than 1D kernels for deblurring. The regularized cost function was built using the results of a classification network trained to distinguish blurry and non-blurry images that had roughly 80 percent accuracy when evaluated(Conti,

Minucci, and Derakhshan 2017).

More recently, Weigert et al. published a series of image restoration methods that succeeded in restoring seven images of various organisms (i.e. planaria flatworm, fruit fly wings). The major contributions of this work include: generating training data without requiring manual labeling, replicating live imaging for organisms in which live imaging had once been near-impossible, and restoring microscopy images even when lighting conditions are reduced by 60-fold. Weigert et al. demonstrated high quality results with restorations of images containing *Tribolium castaneum* (red flour beetle) and *Schmidtea mediterranea* (a flatworm commonly known as planaria). This Content-Aware Image Restoration (CARE) network is based on the U-Net network, which consists of an encoder-decoder architecture. The only difference between the CARE network and the U-net algorithm is that the former outputs a per-pixel Laplace distribution whereas the latter outputs one value per pixel. Although these restoration methods are promising, each pair of image content and corruption requires a unique data set. Each model must be retrained for an image of that particular content and corruption to be successfully restored (Weigert et al. 2017).

Modifying network architecture is a popular strategy to achieve better performance. Another widely used approach to this goal is data augmentation. Data augmentation is widely used for image classification(Paulin et al. 2014), as applying random transformations to training data effectively provides more data for a network to learn from. Not surprisingly, training with large data sets produces high-quality image restorations (Burger, Schuler, and Harmeling 2012). Typical transformations for data augmentation include translation, rotation, and scaling. Augmenting data is a manual process in which these image transformations are specified by humans. To automate this process, Jain et al. developed an unsupervised learning procedure that generated training samples using different noise models (Jain and Seung 2009). Likewise, Hauberg et al. developed a learned augmentation scheme that outperforms manual augmentation of MNIST data when used as training for a multilayer perceptron and a CNN (Hauberg et al. 2016).

Lastly, altering the loss function is a viable strategy, though this approach tends to be overlooked (Zhao et al. 2017). Typically, the mean absolute error and mean squared error loss functions are employed in image processing networks (Zhao et al. 2017; Burger, Schuler, and Harmeling 2012; Agostinelli, Anderson, and Lee 2013; Chen et al. 2018). In 2017, Zhao et al. introduced a new loss function for image restoration that combined the multi-scale SSIM (MS-SSIM) metric with L_1 loss. Without changing network architecture, Zhao et al. demonstrated that by using this mixed loss function, their fully convolutional neural network outperformed state-of-the-art networks on tasks such as joint denoising and demosaicking (Zhao et al. 2017). Drawing inspiration from this approach, we will replace the existing Laplace loss function of the CARE network with a novel loss function that enhances edge restoration in fluorescence microscopy images.

Research Questions and Hypothesis

The questions we will address in this study are:

1. How well does the CARE network perform on our microscopy data set?
2. How can we produce image restorations that are sharper than those produced by CARE?
3. How does the network perform when it is trained on one kind of sample and tested on a different kind of sample?
4. Can we reliably restore live cell images?

We hypothesize that our improvement on the method of Weigert et al. will restore microscopy images that are less blurry and more detailed than the restorations of the original CARE model. Thus, our model will be a more faithful solution compared to the denoising approach of Weigert et al.

Proposed Implementation

We modified the loss function of the CARE network in hopes of producing restorations that are more faithful to ground truth images. Ultimately, the objective of this study is to restore microscopy images free of artifacts and without loss of fine details.

Experimental Setup

The data set we used to train the standard CARE model is High Low, which consists of over 400 fluorescent images of actin and mitochondria, in addition to 170 images of dendra. All images in the High Low data set were taken using an Olympus IX83 with 60X/1.3NA objective lens. The Andor Zyla CMOS camera was used to image cell organelles.

For all of our experiments, we first observed the behavior of the network and assessed the quality of the network without loss function modifications. These results were then compared to results produced by training the CARE network using our FFT and bandpass cost functions. These experiments include restorations of actin imaged at 1 millisecond and 10 milliseconds, restorations of mitochondria imaged at 1 millisecond, model mismatch experiments, and restorations of dendra imaged at 10 milliseconds. We use the term model mismatch to indicate experiments in which images of one type of cell content are used for training while images of another type of cell content are used for testing. For example, we used images of mitochondria as training data for the CARE algorithm, and subsequently tested the model using noisy images of actin. Our most recent experiments involved restorations of dendra imaged at 10 milliseconds using dendra imaged at 10 and 400 milliseconds as the training set. The results of these experiments were evaluated using peak signal to noise (PSNR) and structural similarity (SSIM) image quality metrics.

To conduct our experiments, we used the default configurations of the standard CARE model. The training batch size was 16 images, the number of training epochs was 100, the initial learning rate was 0.0004, and the iterations per epoch (training steps) was 400. The training images were 2048 pixels wide, and 2048 pixels high, with 1 grayscale channel. In sampling the training images, 800 patches per image of size 64 pixels by 64 pixels were used to train the

CARE model. In all experiments, images were split according to the ratio 4:1 for training and validation respectively. Nine or ten images were used for testing in all experiments. Table 1 provides an overview of the experiments we performed along with their abbreviations.

table

Method

The denoising method of Weigert et al. is successful when restoring images with up to 60-fold reduction in light exposure. Beyond that range, however, we found the CARE algorithm is not able to restore images with fine details. In

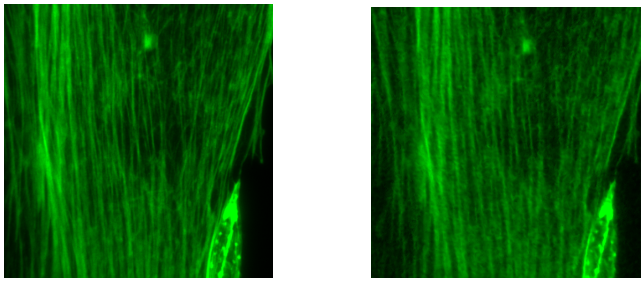

Figure 2: Ground truth image of actin (left) and CARE restoration of actin using Laplace loss (right)

the CARE network, the popular stochastic gradient descent Adam optimizer is used to minimize a Laplace loss function.

$$L_{laplace}(\theta) = \frac{1}{T}\frac{1}{N}\sum_{t=1}^{T}\sum_{i=1}^{N}\frac{|\ y_i^t - \mu_\theta(x^t)_i|}{\sigma_\theta(x^t)_i} + \log\sigma_\theta(x^t)_i$$

where T indicates the number of training images, N indicates the number of pixels per image, y^t corresponds to the ground truth pixel value, and x^t corresponds to the input pixel. μ and σ correspond to the mean and variance of the predicted pixel distribution.

Loss function modification

The restored images produced by this algorithm suffered from blurriness and lack of fine details. In response to this issue, we tested various loss functions tailored to preserve edges. Despite the strong performance of the combined MS-SSIM and L_1 loss function introduced by Zhao et al. (2017), we found a similar combined MS-SSIM and L_1 loss function yielded poor results when applied to the CARE network. Incorporating the SSIM metric into the loss function resulted in slightly better performance according to the SSIM metric, which was expected. In addition to SSIM-based loss functions, edge detection techniques were applied to the loss function. We obtained poor results by using the Sobel operator in the loss function. The loss functions that we introduce in this study are the FFT and bandpass loss functions. So far, these two loss functions have produced the most faithful results for denoising images in our High Low data set compared to other loss functions we designed.

The FFT loss function is similar to mean absolute error with one key difference. Instead of taking differences between corresponding pixel values, the FFT loss function considers differences between per-pixel frequencies represented in the 2D Fourier transforms of the restored image and its corresponding ground-truth image.

$$L_{FFT} = \frac{1}{N \times M}\sum_{j=1}^{M}\sum_{i=1}^{N} |\ f_{i,j}^r - fi, j^t|$$

where f represents the frequency at the (i,j) pixel in the Fourier transforms of the restored and ground truth images (represented by r and t, respectively). M and N represent the M x N pixels in an image. To preserve edges, the bandpass loss function was designed to emphasize high frequencies. This loss function (L_b) consists of taking a difference of Gaussians, with $\sigma_1 = 0.5$ and $\sigma_2 = 5$. These sigma values were chosen arbitrarily, and may be adjusted through trial and error. In the equation below, G^t_{i,j,σ_1} denotes Gaussian blur applied to the ground truth image with standard deviation σ_1 and G^t_{i,j,σ_2} denotes Gaussian blur applied to the ground truth image with standard deviation σ_2. Likewise, G^r_{i,j,σ_1} denotes Gaussian blur applied to the restored image with standard deviation σ_1 and G^r_{i,j,σ_2} denotes Gaussian blur applied to the restored image with standard deviation σ_2

$$L_b = \frac{1}{N \times M}\sum_{j=1}^{M}\sum_{i=1}^{N} |\ (G^t_{i,j,\sigma_1} - G^t_{i,j,\sigma_2}) - (G^r_{i,j,\sigma_1} - G^r_{i,j,\sigma_2})|$$

Results

Restoration of actin

To restore actin imaged using 1 millisecond of exposure time, the CARE algorithm was trained using sixty pairs of actin images taken using 1 millisecond and 100 milliseconds of exposure time. After the model was trained, nine testing images of actin taken at 1 millisecond of exposure time were restored using the CARE prediction function. The following table displays peak signal to noise ratios and structural similarity measurements of the initial input images and the corresponding restored images. In the following tables, the PSNR and SSIM values of the input image are displayed in the Input column , with the rest of the column headings indicating the loss function used (Laplace, FFT, and bandpass).

Actin	Input	Laplace	FFT	BP
0	26.690	**35.869**	35.676	35.304
1	28.322	35.256	33.500	**35.559**
2	29.074	**39.915**	39.747	39.833
3	33.395	42.284	42.140	**43.218**
4	31.766	39.827	**40.741**	40.657
5	32.937	39.892	**40.534**	40.406
6	31.452	38.235	36.589	**38.297**
7	27.677	**36.458**	33.306	35.642
8	32.679	**41.278**	39.607	40.980
Mean	30.444	38.779	37.982	**38.877**

Table 2: PSNR values produced by loss functions (AA)

Encoding	Training Set (Exposure Time)		Input	Ground Truth (Exposure Time)
	Low SNR	High SNR		
MM	Mitochondria (1 ms)	Mitochondria (100 ms)	Mitochondria (1 ms)	Mitochondria (100 ms)
AA	Actin (1 ms)	Actin (100 ms)	Actin (1 ms)	Actin (100 ms)
MA	Mitochondria (1 ms)	Mitochondria (100 ms)	Actin (1 ms)	Actin (100 ms)
AM	Actin (1 ms)	Actin (100 ms)	Mitochondria (1 ms)	Mitochondria (100 ms)
DDS	Dendra (10 ms)	Dendra (400 ms)	Dendra (10 ms)	Dendra (400 ms)
AAS	Actin (10 ms)	Actin (400 ms)	Actin (10 ms)	Actin (400 ms)
AAE	Actin (1 ms)	Actin (100 ms)	Actin (1 ms)	Actin (100 ms)

Table 1: Summary of experiments

Actin	Input	Laplace	FFT	BP
0	0.447	0.895	0.907	**0.914**
1	0.838	0.951	0.947	**0.954**
2	0.707	0.967	**0.970**	0.966
3	0.829	0.976	**0.981**	0.979
4	0.831	0.976	**0.978**	0.977
5	0.870	0.977	**0.980**	0.979
6	0.895	0.972	0.968	**0.977**
7	0.846	**0.956**	0.930	0.953
8	0.889	**0.980**	0.979	0.977
Mean	0.795	0.961	0.960	**0.964**

Table 3: SSIM values produced by loss functions (AA)

Mitochondria	Input	Laplace	FFT	BP
0	0.805	0.983	0.985	**0.985**
1	0.907	0.981	**0.985**	0.977
2	0.797	0.981	0.983	**0.984**
3	0.605	0.957	**0.965**	0.961
4	0.886	0.988	**0.989**	0.988
5	0.844	0.977	0.977	**0.978**
6	0.830	**0.985**	0.983	0.985
7	0.895	0.984	**0.986**	0.983
8	0.844	**0.987**	0.984	0.986
Mean	0.824	0.980	**0.982**	0.981

Table 5: SSIM values produced by loss functions (MM)

Restoration of mitochondria

An identical experiment was performed to restore mitochondria images taken with 1 millisecond of exposure time. Sixty pairs of mitochondria were used for training and validation while ten pairs of mitochondria images were used for testing. The following tables display results of this experiment.

Mitochondria	Input	Laplace	FFT	BP
0	32.073	**41.059**	40.644	40.213
1	31.624	34.233	34.474	**36.011**
2	32.167	39.965	**40.171**	39.937
3	27.757	35.261	**36.709**	35.068
4	34.548	41.057	**41.583**	40.292
5	31.943	35.940	**36.585**	35.206
6	32.995	40.704	**41.148**	39.873
7	34.169	39.387	**40.401**	39.163
8	33.385	40.702	**40.773**	39.253
Mean	32.296	38.701	39.165	38.335

Table 4: PSNR values produced by loss functions (MM)

Model mismatch to restore actin

Actin imaged at 1 millisecond was restored using the standard CARE model trained with mitochondria images. As CARE is a content-aware network, the PSNR and SSIM values produced by the algorithm were less faithful to ground truth images compared to previous experiments (MM and AA). The following tables display results of this model mismatch experiment.

Actin	Input	Laplace	FFT	BP
0	26.690	**36.666**	36.060	35.363
1	28.322	30.095	30.557	**30.661**
2	29.074	37.591	**38.431**	36.413
3	33.395	38.911	**40.440**	37.258
4	31.766	36.436	**38.631**	36.627
5	32.937	36.125	**39.027**	35.660
6	31.452	33.647	**34.482**	33.029
7	27.677	28.937	**29.406**	27.911
8	32.679	35.851	**36.631**	35.440
Mean	30.444	34.918	**35.963**	34.262

Table 6: PSNR values produced by loss functions (MA)

Actin	Input	Laplace	FFT	BP
0	0.448	0.936	0.930	**0.937**
1	0.838	0.937	**0.954**	0.941
2	0.707	0.957	**0.963**	0.958
3	0.829	0.967	**0.974**	0.962
4	0.831	0.958	**0.972**	0.962
5	0.870	0.959	**0.975**	0.956
6	0.895	0.964	**0.974**	0.962
7	0.846	0.919	**0.939**	0.895
8	0.889	0.971	**0.976**	0.967
Mean	0.795	0.952	**0.962**	0.949

Table 7: SSIM values produced by loss functions (MA)

Model mismatch to restore mitochondria

Mitochondria imaged at 1 millisecond were restored using the standard CARE model trained with actin images. The following tables display results of the second model mismatch experiment.

Mitochondria	Input	Laplace	FFT	BP
0	32.073	39.803	**40.825**	40.779
1	31.624	**35.938**	34.370	35.862
2	32.167	39.416	**39.808**	39.666
3	27.757	34.052	35.616	**35.641**
4	34.548	40.982	40.920	**41.938**
5	31.943	36.055	36.485	**36.923**
6	32.995	38.463	**40.084**	39.956
7	34.169	40.389	40.016	**41.037**
8	33.385	39.696	40.299	**40.401**
Mean	32.396	38.311	38.714	**39.134**

Table 8: PSNR values produced by loss functions (AM)

Mitochondria	Input	Laplace	FFT	BP
0	0.805	0.977	0.982	**0.983**
1	0.907	0.978	**0.984**	0.977
2	0.797	0.977	**0.981**	0.980
3	0.605	0.942	0.953	**0.955**
4	0.886	0.985	**0.987**	0.987
5	0.844	0.974	0.977	**0.978**
6	0.830	0.971	**0.982**	0.980
7	0.895	0.983	**0.985**	0.984
8	0.844	0.974	**0.983**	0.982
Mean	0.824	0.973	**0.979**	0.979

Table 9: SSIM values produced by loss functions (AM)

Restoration of dendra

Dendra imaged at 10 milliseconds were restored using the standard CARE model trained with dendra images. The dendra samples were imaged at 200 timesteps, with each timestep lasting 400 milliseconds. The following table displays average PSNR and SSIM results of denoising using the Laplace, FFT, and bandpass loss functions.

	Input	Laplace	FFT	BP
PSNR	26.639	32.691	28.534	32.758
SSIM	0.570	0.897	0.846	0.909

Restoration of actin sequence

Actin imaged at 10 milliseconds were restored using the standard CARE model trained with actin images. The actin samples were imaged at 200 timesteps, with each timestep lasting 400 milliseconds. Due to the large number of samples restored (200 images), the following table displays average PSNR and SSIM results of denoising using the Laplace, FFT, and bandpass loss functions.

	Input	Laplace	FFT	BP
PSNR	25.672	25.917	26.023	26.002
SSIM	0.837	0.872	0.874	0.874

Table 10: Average PSNR and SSIM values produced by loss functions (AAS)

Restoration of extremely noisy actin

In our AAE experiment, actin imaged at 1 millisecond were restored using the standard CARE model trained with actin images. Microscope settings were altered prior to imaging these actin samples to induce significant noise in these images (taken with 1 millisecond of exposure). The accompanying table that displays these results will be included in a future revision.

Discussion

The results of our study were analyzed using paired sample t-tests with an alpha significance value of 0.05. We demonstrate statistically significant results with respect to SSIM measurements, particularly in model mismatch experiments. The sequence of actin restorations (AAS) also demonstrated statistically significant results. The table below summarizes t-test results (p-values) for each experiment. The PSNR and SSIM values obtained using the FFT and bandpass loss functions were each evaluated against PSNR and SSIM values obtained by the original Laplace loss function.

Experiment	FFT/Laplace		BP/Laplace	
	PSNR	SSIM	PSNR	SSIM
MM	0.03	0.150	0.264	0.426
AA	0.110	0.201	0.640	0.710
MA	0.016	0.007	0.311	0.311
AM	0.260	0.0008	0.002	0.008
DDS	—	—	—	—
AAS	1.509e-232	5.811e-205	4.055e-232	5.458e-199
AAE	—	—	—	—

The following table summarizes the average change in PSNR and SSIM values organized by experiment description and loss function.

Metric	Loss	MM	AA	MA	AM	DD
$\Delta PSNR$	Laplace	6.405	8.336	4.474	6.015	DD
	FFT	6.870	7.539	5.519	6.418	–
	BP	6.039	8.434	3.819	6.838	–
$\Delta SSIM$	Laplace	0.157	0.166	0.157	0.150	DD
	FFT	0.158	0.165	0.167	0.156	–
	BP	0.157	0.169	0.154	0.155	–

Table 11: Average Increase in PSNR and SSIM

Further Research

To further evaluate our deep learning approach, we will employ traditional filtering techniques (total variation and non-local means) and compare our results with these conventional practices using standard metrics (PSNR and SSIM). The total variation minimization (TV) technique restores images by minimizing a cost function. The TV method smooths excess details while maintaining sharp edges. The non-local means (NL means) filtering technique finds the average of all pixels in an image, and makes each pixel a linear combination of patches. Similar patches are weighted more heavily than dissimilar patches (Buades, Coll, and Morel 2005).

In the near future, we will train the CARE network while varying the patch size of the training images to 32 x 32 and 128 x 128. By introducing different patch sizes for training, the CARE network performance may improve. We would also like to implement combined loss functions (FFT loss combined with bandpass loss). Perhaps an adaptive, GAN-based loss function may better outperform the current state-of-the-art. To test the limits of our computational restoration method, we will conduct future studies to determine what is the lowest amount of light that can be used when imaging a sample such that the image can successfully be restored by our implementation?

Conclusion

The purpose and major contribution of this research is to modify and improve existing restoration methods for fluorescence microscopy imaging. Compared to the Laplace loss function, the results of this study indicate that there were statistically significant improvements in image denoising using FFT loss and bandpass loss to train the CARE network. Our model mismatch and actin sequence restoration experiments yielded the most prominent statistically significant results, which confirms that the CARE model generalizes poorly when it is content-unaware. By developing ways to denoise fluorescence microscopy images faithfully, significantly less time and resources will be required to image 2D structures.

Acknowledgements

I would like to thank my advisers Dr. Ventura and Dr. Hagen for their generosity and kind support throughout the REU program. I would also like to thank members of my REU cohort: Lee Sharma, Chance Hamilton, and Kayleigh Migdol for all the laughs, lovely conversations, and late-night hangouts this summer.

This material is based upon work supported by the National Science Foundation under Grant No. 1659788. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Agostinelli, F.; Anderson, M. R.; and Lee, H. 2013. Adaptive multi-column deep neural networks with application to robust image denoising. In *Advances in Neural Information Processing Systems*, 1493–1501.

Buades, A.; Coll, B.; and Morel, J.-M. 2005. A non-local algorithm for image denoising. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005.*, volume 2, 60–65. IEEE.

Burger, H. C.; Schuler, C. J.; and Harmeling, S. 2012. Image denoising: Can plain neural networks compete with bm3d? In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2392–2399. IEEE.

Chen, C.; Chen, Q.; Xu, J.; and Koltun, V. 2018. Learning to see in the dark. *arXiv preprint arXiv:1805.01934*.

Conti, F. L.; Minucci, G.; and Derakhshan, N. 2017. A regularized deep learning approach for image de-blurring.

Hauberg, S.; Freifeld, O.; Larsen, A. B. L.; Fisher, J.; and Hansen, L. 2016. Dreaming more data: Class-dependent distributions over diffeomorphisms for learned data augmentation. In *Artificial Intelligence and Statistics*, 342–350.

Jain, V., and Seung, S. 2009. Natural image denoising with convolutional networks. In *Advances in Neural Information Processing Systems*, 769–776.

Paulin, M.; Revaud, J.; Harchaoui, Z.; Perronnin, F.; and Schmid, C. 2014. Transformation pursuit for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, 3646–3653. IEEE.

Weigert, M.; Schmidt, U.; Boothe, T.; Andreas, M.; Dibrov, A.; Jain, A.; Wilhelm, B.; Schmidt, D.; Broaddus, C.; Culley, S.; et al. 2017. Content-aware image restoration: Pushing the limits of fluorescence microscopy. *bioRxiv* 236463.

Xing, F.; Xie, Y.; Su, H.; Liu, F.; and Yang, L. 2017. Deep learning in microscopy image analysis: A survey. *IEEE Transactions on Neural Networks and Learning Systems* 119.

Xu, L.; Ren, J. S.; Liu, C.; and Jia, J. 2014. Deep convolutional neural network for image deconvolution. In *Advances in Neural Information Processing Systems*, 1790–1798.

Zhao, H.; Gallo, O.; Frosio, I.; and Kautz, J. 2017. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging* 3(1):47–57.