

SEMI-BLIND SPATIALLY-VARIANT DECONVOLUTION IN OPTICAL MICROSCOPY WITH LOCAL POINT SPREAD FUNCTION ESTIMATION BY USE OF CONVOLUTIONAL NEURAL NETWORKS

Adrian Shajkofci^{1,2}, Michael Liebling^{1,3}

¹Idiap Research Institute, CH-1920 Martigny, Switzerland

²École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland

³Electrical & Computer Engineering, University of California, Santa Barbara, CA 93106, USA

ABSTRACT

We present a semi-blind, spatially-variant deconvolution technique aimed at optical microscopy that combines a local estimation step of the point spread function (PSF) and deconvolution using a spatially variant, regularized Richardson-Lucy algorithm [1]. To find the local PSF map in a computationally tractable way, we train a convolutional neural network to perform regression of an optical parametric model on synthetically blurred image patches. We deconvolved both synthetic and experimentally-acquired data, and achieved an improvement of image SNR of 1.00 dB on average, compared to other deconvolution algorithms.

Index Terms— Microscopy, blind deconvolution, point spread function, convolutional neural networks

1. INTRODUCTION

Optical microscopy is a powerful tool to comprehend biological systems, enabling researchers and physicians to acquire qualitative and quantitative data about cellular function, organ development, or diseases. However, light traveling through any imaging system undergoes diffraction, which leads to image blur [2]. This represents an intrinsic limit and the determining factor for the resolution of an optical instrument, and thus limits visual access to details. Indeed, the optical system only collects a fraction of the light emitted by any one point on the object, and cannot focus the light into a perfect image. Instead, the light spreads into a three-dimensional diffraction pattern. Image formation can be modeled as the convolution of the original object with a PSF, which sums up the optical aberrations [3]. For thin, yet not flat, samples, the PSF remains shift-invariant within small areas of the 2D image, but the three-dimensional depth of the imaged object produces a local blur. Using a PSF corresponding to the blur in a deconvolution algorithm can be used to restore details in the image [4]. Deconvolution techniques can be categorized into three classes: (1) Non-blind methods, (2) entirely blind methods, and (3) parametric semi-blind algorithms. Non-blind methods require knowledge of the PSF [5]. One of the main difficulties in

practice is to determine the original PSF that characterizes the actual optical system without discovering it empirically by acquiring a 3D image of a fluorescent bead, which is a tedious and time-consuming calibration step. The two latter classes fall into blind deconvolution (BD) techniques, which improve the image without prior knowledge of the PSF, the object or other optical parameters. Entirely blind algorithms, such as [6] are based the optimization of a penalty function or a maximum a posteriori (MAP) estimation of the latent image or kernel [7]. However, these methods typically use strong constraints such as sharpness along the object edges and do not always generalize to unexpected or noisy types of data [8], which are common in microscopy images. Also, many BD techniques are computationally expensive, especially for larger convolution kernels, and assume spatially invariant PSFs. Finally, parametric or semi-blind algorithms are blind methods that are constrained by knowledge about the transfer function distribution, such as a diffraction model or a prior on the shape of the PSF ([9], [10]). Parametric models allow reducing the complexity of the optimization problem, increasing the overall robustness, and avoiding issues such as over-fitting. However, it remains hard to estimate the parameters from experimental data. We will focus on this third class of deconvolution methods, by addressing the central problem of how to best infer the parameters without measuring any of them experimentally. Machine learning recently improved the ability to classify images [11], detect objects, or describe content [12]. Convolutional Neural Networks (CNNs) [13], in particular, are built for learning new optimal representations of image data and perform self-regulating feature extraction [14]. Because of their ability to learn correlations between high- and low-resolution training samples, CNNs appear well adapted to our problem of determining the blur kernel. A similar reasoning led to recent results in [15] and [16], where the direction and amplitude of motion blur was determined by a CNN classifier from images blurred with a Gaussian kernel.

Here we present a spatially-variant BD technique aimed at microscopy of thin, yet non-flat objects. Our method combines local determination of the PSF and spatially-variant deconvolution using a regularized Richardson-Lucy (RL) algorithm [1]. To find the PSF in a computationally tractable way, we train a CNN to perform regression of model parameters on

synthetically blurred image patches. The novel aspects of our approach are: (1) Our method does not require the experimental measurement of a PSF, only synthetic training data is necessary. (2) Compared to non-parametric BD, the problem complexity remains low and therefore is more easily amenable to optimization. (3) Parameters with a physical meaning are inferred from the image itself. (4) The algorithm is computationally efficient, resulting in a near real-time kernel regression and mapping at the expense of a much longer, yet straightforward, training phase.

In Section 2, we describe each step of the method. In Section 3, we present experiments with synthetic and acquired data. In Section 4, we conclude.

2. METHOD

We describe a deconvolution method for a 2D image in 3D space, such as a thin sample suspended in a gel. Blur in the acquired 2D image y can be determined by an impulse response h . It is assumed that h varies along the two lateral coordinates \mathbf{s} of the image. Thus, the optical system can be represented by the following image formation equation:

$$y(\mathbf{s}) = \iint x(\boldsymbol{\xi})h(\mathbf{s}, \boldsymbol{\xi})d\boldsymbol{\xi}. \quad (1)$$

In a nutshell, we propose the following steps to infer the parameters of the spatially-dependent function $h(\mathbf{s}, \boldsymbol{\xi}) = h_{\mathbf{s}}(\boldsymbol{\xi})$:

1. We develop a parametric model for $h_{\mathbf{s}}(\boldsymbol{\xi})$ allowing the generation of PSF/parameters pairs. We gather a training library of microscopy images, degraded by the space-invariant convolution of synthetic PSFs.
2. We train a CNN on the above dataset to recognize the PSF model parameters given a unique blurred patch (Figure 1).
3. Using a sliding window, we extract small patches from a large input image. For each patch, we regress the PSF parameters using the CNN. We combine the results and generate a map of PSFs over the input image.
4. We recover a de-blurred image using a Total Variation (TV) regularized space-variant RL algorithm from the estimated map of PSFs.

In the following subsections, we provide details on each of these four steps.

2.1. Training library generation

Optical abnormalities, such as defocus, astigmatism, or spherical aberrations, can be modeled with superpositions of Zernike polynomials in the expansion of the pupil function W [17]:

$$W_{\mathbf{a}}(\mathbf{s}) = \sum_{n=1}^N a_n \cdot Z_n(\mathbf{s}), \quad (2)$$

where \mathbf{s} denotes the two-vector of spatial coordinates, N the maximal order of considered aberrations, and $\mathbf{a} = (a_0 \ a_1 \ \dots \ a_N)$ the parameters corresponding to each

Zernike polynomial Z_n . The pupil function can ultimately be converted, using a Fourier transform, into a PSF $h_{\mathbf{a}}(\mathbf{s}) = |\mathcal{F}(W_{\mathbf{a}}(\mathbf{s}))|^2$, with \mathcal{F} the Fourier transform. A training dataset is created by convolving K images x_k , $k = 1, \dots, K$, with K generated PSFs $h_{\mathbf{a}_k}$ of parameters \mathbf{a}_k drawn from an uniform distribution. The degraded training images are obtained as follows:

$$\Psi_k(\mathbf{s}) = (h_{\mathbf{a}_k} * x_k)(\mathbf{s}) = \mathcal{F}^{-1}[\mathcal{F}(h_{\mathbf{a}_k})\mathcal{F}(x_k)](\mathbf{s}). \quad (3)$$

We crop the images Ψ_k to size $K_{\psi} \times L_{\psi}$ and pair them with their respective \mathbf{a}_k in order to form the training set $T = \{(\psi_k, \mathbf{a}_k)\}_{k=1}^K$ (see Figure 1, part 2.1).

Implementation: We gathered a library of microscopy images from publicly available sources, augmented with synthetic cell images. We blurred the images by 127×127 PSFs generated with parameters a_k drawn from an uniform distribution. We generated PSFs with $N = 1$ and $N = 2$. When $N = 1$, the covered aberration was the focus, when $N = 2$, the aberrations were the former plus the astigmatism. The images were cropped to 128×128 . We added Poisson noise and subtracted the pixel-wise mean from every patch. Images that did not comply with a minimal variance and white pixel ratio were rejected from the library. We applied image rotation to further augment K to 2 million pairs.

2.2. CNN training for PSF parameter regression

Given an image patch as input, we wish to estimate the parameters of the PSF that was used to degrade the patch. For that purpose, we train a CNN for regression of these parameters from the established training set T (Figure 1). We aim at training the neural network to learn the Zernike parameters that specify the the PSF by minimizing the Euclidean loss function:

$$E_k = \frac{1}{2N} \sum_{n=1}^N |a_{k,n} - \tilde{a}_{k,n}|^2, \quad (4)$$

where $\tilde{a}_{k,n}$ is the current estimate of the n -th Zernike parameter on the training image ψ_k and $a_{k,n}$ its ground-truth as stored in T .

Implementation: We trained two CNN models that we selected based on their good performance in the ImageNet competition: AlexNet [11], adapted to gray-scale 128×128 input images, and a 34-layer ResNet [18]. We modified the last fully-connected layer to have N outputs. We trained both networks with the Adam optimizer in Caffe [19] for 10 epochs.

2.3. Blind spatially variant kernel estimation of local PSF parameter map

Given the trained model able to recover the degradation parameters from a single image patch, we now turn to the problem of locally estimating the parameter of the PSF that degraded a larger input image. To achieve this, we use an overlapping sliding window over the input image with stride t that is fed into the locally invariant regression CNN trained in 2.2 (see

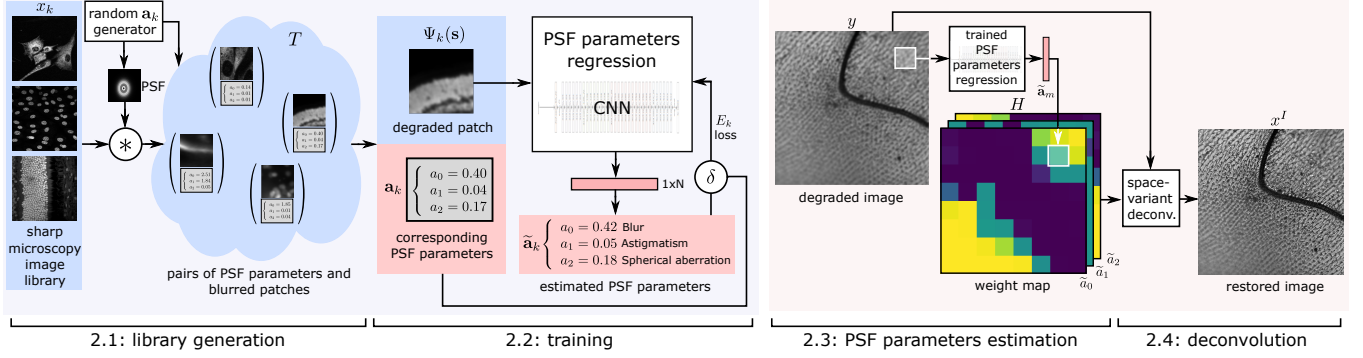


Fig. 1: Left: method to estimate the parameters of the PSF that degraded an image patch. Right: method to estimate a map of the local PSF that degraded an image that can be fed into a spatially variant deconvolution algorithm.

Figure 1, part 2.3). We store the resulting parameters in a map $H = \{h_m\}_{m=1}^M$, where h_m is the PSF corresponding to the patch m with parameters \mathbf{a}_m and M the total number of patches and thus local PSF kernels.

Implementation: The window size was the same as the training patches ψ_k . For example, a 1024×1024 input image using 128×128 patches and $t = 64$ yields $M = 169$ spatially-dependent PSF kernels. In order to avoid disparities and artifacts, we smoothed the parameter map with median filtering.

2.4. Spatially variant deconvolution

Given the degraded image and a local map of PSF parameters, we can now restore the input using TV-RL deconvolution. RL is an iterative maximum-likelihood approach derived from Bayes’s theorem and assumes that the noise follows a Poisson distribution [20], which is well adapted for microscopy. The method is subject to noise amplification, which can, however, be counterbalanced by a regularization term that penalizes the l_1 norm of the gradient of the signal [1]. Here, we assume that the PSF is spatially invariant in small parts of the image. We detail these steps in the sub-sections below.

2.4.1. Spatially variant filtering

Spatially variant convolution techniques have been extensively reviewed by Denis et al. (2015) [21]. Hirsch et al. have shown that our local invariance assumption can be improved by convolving the input with every local PSF and then reconstructing the image using interpolation [6]. We extend this method by using it as the basic convolution block in the TV-RL algorithm. Rather than interpolating deconvolved images, the method interpolates the PSF for each point in the image space [22]. The idea for this pseudo space-variant filtering is: (i) to cover the image with overlapping patches using smooth interpolation, (ii) to apply to each patch a different PSF, (iii) to add the patches to obtain a single large image. The equivalent for convolution can be written as:

$$x(\mathbf{s}) = \mathcal{F}^{-1} \left[\sum_m^M \mathcal{F}(h_m(\mathbf{s})) \cdot \mathcal{F}(\varphi_m(\mathbf{s}) \cdot y(\mathbf{s})) \right], \quad (5)$$

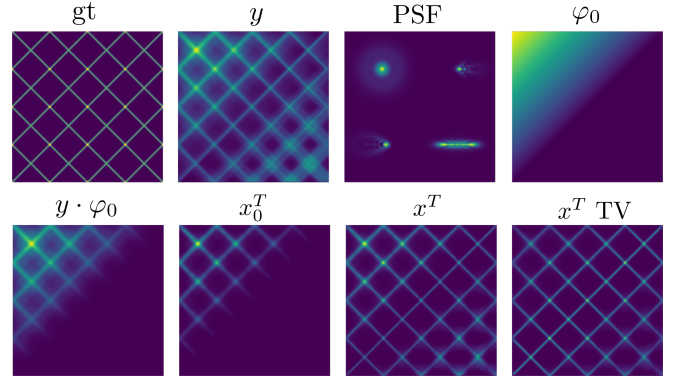


Fig. 2: Synthetic experiment (3.1). Starting from a ground truth image (gt), a local map of PSF (smooth interpolation between the 4 shown PSFs) and local weight combination φ , we generated a blurred image y . The deconvolution method (similar to [6]) starts from a map of locally estimated PSFs to deconvolve and recombine into a single image x^T and x^T TV.

where $\varphi_m(\mathbf{s})$ is the masking operator of the m th patch. We illustrated the masking and deconvolution steps in Figure 2.

2.4.2. Regularization of the RL algorithm

In noisy images, the RL algorithm tends to exacerbate small variations. TV regularization is used to obtain a smooth solution while preserving the borders [1]. The resulting image at each RL iteration becomes:

$$x^{i+1}(\mathbf{s}) = \sum_m^M \left[\frac{(h_m * (\varphi_m \cdot y))(\cdot)}{(h_m * x_m^i)(\cdot)} * h_m(-\cdot) \right] (\mathbf{s}) \cdot \frac{x_m^i(\mathbf{s})}{1 - \lambda_{TV} \operatorname{div} \left(\frac{\nabla x_m^i(\mathbf{s})}{|\nabla x_m^i(\mathbf{s})|} \right)}, \quad (6)$$

with y the blurry image, x^i the deconvolved image at iteration $i = 1, \dots, I$, x_m^i the $K_y \times L_y$ deconvolved patch at iteration i , M the number of patches (and different PSFs) in one image x , h_m the $K_h \times L_h$ PSF for patch m and λ_{TV} the regularization factor. $\nabla x_m^i(\mathbf{s})$ is the finite difference operator, which approximates the spatial gradient.

Implementation: Here, φ_m was a bilinear interpolating function. If λ_{TV} is small, RL is dominated by the data, and if $\lambda_{TV} \sim 1$, RL is dominated by the regularization [1]. Therefore, we used a balanced $\lambda_{TV} = 0.001$. Since using FFT-based calculations implies that h is circulant, we had to take into account the field extension to prevent spatial aliasing. The number of RL iterations, I , was fixed to 20.

3. EXPERIMENTAL RESULTS

We validated the method using synthetic and live microscopy experiments. The results were compared using the SNR and SSIM [26] with three BD methods : 1. MATLAB’s deconvblind [23]; 2. the non-CNN deconvolution method [24], and 3. the space-variant BD described in [25]. Due to gain difference between methods and boundary effects, we computed the maximum SNR obtainable following linear scaling and offset of the gray values.

3.1. Synthetic data

We generated a 252×252 gray-scale grid pattern, as shown in Figure 2. Using the technique in equation (5), we degraded each quadrant of the input image with a specific, randomly-generated 127×127 PSFs using parameters a drawn from a uniform random distribution. We subsequently inferred a PSF map via the CNN from the blurry image as described in 2.3. Finally, we deconvolved the image and determined the match R^2 between the Zernike parameters used for PSF generation and the recovered parameters, as well as the reconstruction quality. Results in Table 1 indicate an improvement of the spatially variant BD in comparison to invariant BD.

3.2. Acquired data

We used a wide-field microscope to acquire the image of a housefly wing. The sample was tilted 20 degrees so that only the central part of the image was in focus. The goal was to recover the in- and out-of-focus PSFs in order to restore the whole image. Our method gives visual results that are similar to [24] (Figure 3). With $M = 64$ regions, we generated the map H in 18 milliseconds with a consumer-grade GPU. The additional time necessary for determining the PSF maps is negligible compared to the time necessary for deconvolution.

4. DISCUSSION

Space-varying BD is a more complex process than space-invariant deconvolution. However, our experiments showed that it is possible to add parametric constraints to the problem

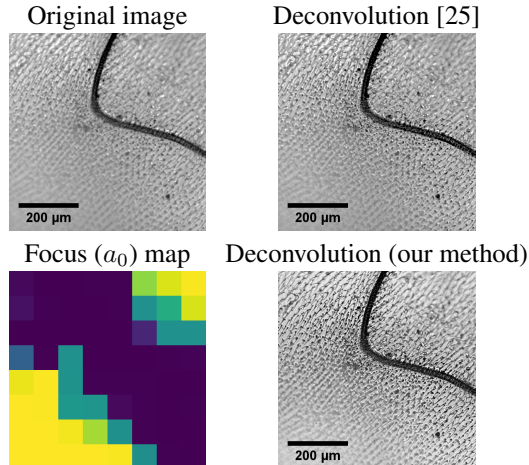


Fig. 3: Deconvolution of a tilted housefly wing using our algorithm and the method from [25]. Our pipeline inferred a 8×8 map of a coefficients. Here, a yellow color represents $a_0 = 2$, while a dark blue color represents $a_0 = 0$.

and thereby allowing to map patches of spatially-invariant PSFs onto the input image. With our CNN-based method, we were able to detect their original blur kernel with an average accuracy of 91.1% (Table 1), given only synthetic images as the training input (no experimental measurement of the PSF was necessary). From these results, we have been able to deconvolve these image with an SNR on average 1.00 dB higher than other blind deconvolution techniques. We used the method on acquired experimental data and observed a visual improvement similar or, in some regions, better than when we used other methods. The main limitation of our approach is that the number of regressed Zernike polynomials has an important influence on the performance. For example, the parameter “focus” is well recovered, but other parameters, such as coma, have a very low regression accuracy. We will focus future efforts into researching a CNN architectures that are more specific to the problem. In contrast to end-to-end deblurring methods such as [27], users may be able to easily customize our algorithm with other parametric models, CNN networks or deconvolution techniques for specific applications. Our method is therefore suitable to other applications than image enhancement, such as auto-focus or depth determination from focus [28]. But mostly, our method could be used to simplify procedures that required physical measurements of PSFs such as [29], where deconvolution improved PALM/STORM microscopy using PSFs from a set of calibration images.

Code and documentation are available at this address: <https://github.com/idiap/semibindpsfdeconv>.

R^2_{focus}	$R^2_{\text{focus, ast}}$	SNR _{degraded}	SNR _{ours}	SNR _[23]	SNR _[24]	SNR _[25]	SSIM _{degraded}	SSIM _{ours}	SSIM _[23]	SSIM _[24]	SSIM _[25]
0.91	0.58	1.90 dB	4.48 dB	3.48 dB	1.4 dB	1.54 dB	0.10	0.51	0.28	0.29	0.52

Table 1: Image recovery benchmark with 100 synthetic input images. We computed R^2 , between the generated PSF parameters and the recovered PSF parameters, as well as SNR and SSIM.

5. REFERENCES

- [1] N. Dey, L. Blanc-Féraud, C. Zimmer, P. Roux, Z. Kam, J.-C. Olivo-Marin, and J. Zerubia, “Richardson–Lucy algorithm with total variation regularization for 3D confocal microscope deconvolution,” *Microsc. Res. Tech.*, vol. 69, no. 4, pp. 260–266, Apr. 2006.
- [2] S. F. Gibson and F. Lanni, “Diffraction by a circular aperture as a model for three-dimensional optical microscopy,” *J. Opt. Soc. Am. A*, vol. 6, no. 9, pp. 1357–1367, Sept. 1989.
- [3] D. Sage, L. Donati, F. Soulez, D. Fortun, G. Schmit, A. Seitz, R. Guiet, C. Vonesch, and M. Unser, “DeconvolutionLab2: An open-source software for deconvolution microscopy,” vol. 115, pp. 28–41, Feb. 2017.
- [4] A. Griffla, N. Garin, and D. Sage, “Comparison of deconvolution software in 3D microscopy: A user point of view—part 1,” *GIT Imaging Microsc.*, vol. 12, no. 1, pp. 43–45, 2010.
- [5] F. Soulez, “A “Learn 2D, Apply 3D” method for 3D deconvolution microscopy,” in *IEEE ISBI 2014*, Apr. 2014, pp. 1075–1078.
- [6] M. Hirsch, S. Sra, B. Scholkopf, and S. Harmeling, “Efficient filter flow for space-variant multiframe blind deconvolution,” in *IEEE CVPR 2010*, San Francisco, CA, June 2010, pp. 607–614.
- [7] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, “Understanding blind deconvolution algorithms,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2354–2367, Dec. 2011.
- [8] D. A. Fish, A. M. Brinicombe, E. R. Pike, and J. G. Walker, “Blind deconvolution by means of the Richardson–Lucy algorithm,” *J. Opt. Soc. Am. A*, vol. 12, no. 1, pp. 58–65, Jan. 1995.
- [9] F. Aguet, D. Van De Ville, and M. Unser, “Model-based 2.5-D deconvolution for extended depth of field in brightfield microscopy,” *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1144–1153, July 2008.
- [10] R. Morin, S. Bidon, A. Basarab, and D. Kouamé, “Semi-blind deconvolution for resolution enhancement in ultrasound imaging,” in *IEEE ICIP 2013*, Sept. 2013, pp. 1413–1417.
- [11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep Convolutional Neural Networks,” in *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc., 2012.
- [12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *IEEE CVPR 2014*, 2014.
- [13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [14] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” vol. 521, no. 7553, pp. 436–444, May 2015.
- [15] J. Sun, W. Cao, Z. Xu, and J. Ponce, “Learning a Convolutional Neural Network for non-uniform motion blur removal,” in *IEEE CVPR 2015*, 2015.
- [16] S. Nah, T. H. Kim, and K. M. Lee, “Deep multi-scale convolutional neural network for dynamic scene deblurring,” in *IEEE CVPR 2017*, 2017.
- [17] F. von Zernike, “Beugungstheorie des schneidenverfahrens und seiner verbesserten form, der phasenkontrastmethode,” vol. 1, no. 7, pp. 689–704, May 1934.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *IEEE CVPR 2016*, 2016, pp. 770–778.
- [19] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” *ArXiv14085093 Cs*, June 2014.
- [20] W. H. Richardson, “Bayesian-based iterative method of image restoration,” vol. 62, no. 1, pp. 55–59, Jan. 1972.
- [21] L. Denis, E. Thiébaud, F. Soulez, J.-M. Becker, and R. Mourya, “Fast approximations of shift-variant blur,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 253–278, Dec. 2015.
- [22] M. Temerinac Ott, O. Ronneberger, R. Nitschke, W. Driever, and H. Burkhardt, “Spatially variant Lucy-Richardson deconvolution for multiview fusion of microscopical 3D images,” in *IEEE ISBI 2011*, Mar. 2011, pp. 899–904.
- [23] T. J. Holmes, S. Bhattacharyya, J. A. Cooper, D. Hanzel, V. Krishnamurthi, W.-c. Lin, B. Roysam, D. H. Szarowski, and J. N. Turner, “Light microscopic images reconstructed by maximum likelihood deconvolution,” in *Handbook of Biological Confocal Microscopy*, pp. 389–402. Springer, Boston, MA, 1995.
- [24] J. Kotera, F. Šroubek, and P. Milanfar, “Blind deconvolution using alternating maximum a posteriori estimation with heavy-tailed priors,” in *International Conference on Computer Analysis of Images and Patterns*. 2013, pp. 59–66, Springer.
- [25] O. Whyte, J. Sivic, and A. Zisserman, “Deblurring shaken and partially saturated images,” *Int. J. Comput. Vis.*, vol. 110, no. 2, pp. 185–201, Nov. 2014.
- [26] 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.
- [27] P. Wieschollek, M. Hirsch, B. Schölkopf, and H. P. Lensch, “Learning blind motion deblurring,” in *IEEE ICCV 2017*, 2017, pp. 231–240.
- [28] V. M. Bove, “Entropy-based depth from focus,” *J. Optical Soc. Amer. A*, vol. 10, no. 4, pp. 561–566, 1993.
- [29] S. Quirin, S. R. P. Pavani, and R. Piestun, “Optimal 3D single-molecule localization for superresolution microscopy with aberrations and engineered point spread functions,” *Proc. Natl. Acad. Sci.*, vol. 109, no. 3, pp. 675–679, 2012.