

Spatially-Variant CNN-based Point Spread Function Estimation for Blind Deconvolution and Depth Estimation in Optical Microscopy

Adrian Shajkofci, *Graduate Student Member, IEEE*, Michael Liebling, *Member, IEEE*

Abstract—Optical microscopy is an essential tool in biology and medicine. Imaging thin, yet non-flat objects in a single shot (without relying on more sophisticated sectioning setups) remains challenging as the shallow depth of field that comes with high-resolution microscopes leads to unsharp image regions and makes depth localization and quantitative image interpretation difficult. Here, we present a method that improves the resolution of light microscopy images of such objects by locally estimating image distortion while jointly estimating object distance to the focal plane. Specifically, we estimate the parameters of a spatially-variant Point Spread Function (PSF) model using a Convolutional Neural Network (CNN), which does not require instrument- or object-specific calibration. Our method recovers PSF parameters from the image itself with up to a squared Pearson correlation coefficient of 0.99 in ideal conditions, while remaining robust to object rotation, illumination variations, or photon noise. When the recovered PSFs are used with a spatially-variant and regularized Richardson-Lucy (RL) deconvolution algorithm, we observed up to 2.1 dB better Signal-to-Noise Ratio (SNR) compared to other Blind Deconvolution (BD) techniques. Following microscope-specific calibration, we further demonstrate that the recovered PSF model parameters permit estimating surface depth with a precision of 2 micrometers and over an extended range when using engineered PSFs. Our method opens up multiple possibilities for enhancing images of non-flat objects with minimal need for a priori knowledge about the optical setup.

Index Terms—Microscopy, point spread function estimation, convolutional neural networks, blind deconvolution, depth from focus

I. INTRODUCTION

RESEARCHERS and physicians intensively use optical microscopes to observe and quantify cellular function, organ development, or disease mechanisms. Despite the availability of many volumetric imaging methods (in particular, optical sectioning methods), single-shot wide-field microscopy remains an important tool to image small and relatively shallow objects. However, non-flat areas, which are out of focus, lead to unsharp regions in the image, making localization and visual interpretation difficult. Image formation in a microscope can be modeled by light diffraction, which causes sharp point-like objects to appear blurry [1]. Because the optical system only

collects a fraction of the light emanating from a point on the object, it cannot focus the light into a perfect point and, instead, spreads the light into a three-dimensional diffraction pattern described by the Point Spread Function (PSF). As the image is formed by superposing the contribution of all points in the object, knowledge of the local diffraction pattern, which sums up the optical system and its aberrations, can be used to estimate a sharper image [2]. For thin, yet not flat samples, image formation can be modeled as a superposition of 2D PSFs. These are shaped both by the optical system and the three-dimensional depth of the object. Knowledge of the local PSF could therefore both be used to recover the image and estimate its depth, which usually requires careful camera calibration and ad-hoc focus estimation [3], acquisition of focal depth stacks ([4], [5]), or coherent imaging, such as digital holographic microscopy [6], to numerically refocus the image. Using an adequate PSF, i.e. one that corresponds to the blur, in a deconvolution algorithm can restore details in the image [7]. PSF estimation can be achieved by many techniques [8], but most of them are either dependent on a tedious calibration step, such as the experimental measurement of the PSF, or are sensitive to noise or image variability. Blind Deconvolution (BD) techniques are methods able to recover features in the image without prior knowledge of the PSF.

Here, we aim at estimating the local PSF only from the acquired image and use it to reverse the local degradation due to the optical system. Furthermore, we aim at estimating the depth of any location on the surface of a thin object with respect to the focal plane. We rely on a model-based approach that retrieves the PSF given a degraded image patch via a machine learning approach.

Machine learning technologies have improved our ability to classify images [9], detect objects [10], describe content [11], and estimate image quality [12]. Convolutional Neural Networks (CNNs), in particular, have the ability to learn correlations between an image input and a defined outcome and appear well adapted to determining the degradation kernel directly from the image texture. A similar reasoning led to recent results by Zhu et al. [13], Sun et al. [14], Gong et al. [15] and Nah et al. [16], where the direction and amplitude of motion blur was determined by a CNN classifier from images blurred with a Gaussian kernel. Our approach is similar to that of [14] but with PSF models that are tailored to the specificities of microscopy, a concept that we initially introduced in [17] and that has since been used by other groups such as [18]. In particular, we considered a more generic physical model that

Adrian Shajkofci is with the Idiap Research Institute, CH-1920 Martigny, Switzerland and École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland.

Michael Liebling is with the Idiap Research Institute and the Electrical & Computer Engineering Department, University of California, Santa Barbara, CA 93106, USA.

This work was supported by the Swiss National Science Foundation grant 206021_164022.

Manuscript received XXXXX XX, XXXX; revised XXXXXXXX XX, XXXX.

can accommodate large-support PSFs. CNNs were also used in a end-to-end manner to enhance details in biological images by performing supervised interpolation [19]–[21] or to emulate confocal stacks of sparse 3D structures from widefield images [22].

In this paper, we propose a method to:

- 1) Find the spatially-variant PSF of the degraded image of a thin, non-flat object directly from the image texture without any instrument-specific calibration step. The PSF determination technique is derived from the one we proposed in [17], which recovers local Zernike moments of the PSF. We focus here on improving the degradation model and quantitatively assess the robustness of the method.
- 2) Deconvolve the image in a blind and spatially-variant manner, using a regularized Richardson-Lucy algorithm with an overlap-add approach.
- 3) Extract the depth of a three-dimensional surface from a single two-dimensional image using combinations of Zernike moments.

This technique allows us to enhance the acquired image and recover the three-dimensional structure of a two-dimensional manifold in a 3D space using a single 2D image as an input. This paper is organized as follows. In Section II, we present the method, comprising the image formation model, the degradation model, the data set generation process, the different neural networks to be trained, the PSF mapping, the deconvolution algorithm, and the depth from focus algorithm. Then, in Section III, we characterize the regression performance of the CNN for different modalities, as well as the gain in resolution from the deconvolution, and the precision of depth detection. We then discuss our findings in Section IV and conclude in Section V.

II. METHODS

A. Object and image formation model

We consider a two-dimensional manifold in 3D-space with local intensity $x(\mathbf{r})$, e.g. an infinitely thin sample suspended in a gel, which can be parameterized by the lateral coordinates $\mathbf{r} = (r_1, r_2)$ and axial coordinate $z(\mathbf{r})$. We express the resulting three-dimensional object as:

$$x_{3D}(\mathbf{r}, r_3) = x(\mathbf{r})\delta(r_3 - z(\mathbf{r})), \quad (1)$$

where $(\mathbf{r}, r_3) = (r_1, r_2, r_3)$ are coordinates in 3D object space. We further consider an optical imaging system with camera coordinates $\mathbf{s} = (s_1, s_2)$ and axial position s_3 , characterized by a spatially-varying point spread function $h_{3D}(\mathbf{s}, s_3, \mathbf{r}, r_3)$ (see Fig. 1). For a fixed axial camera position s_3 , the measured intensity by a pixel at position \mathbf{s} is given by the convolution (Fig. 1 (a)) [23]:

$$y_{3D}(\mathbf{s}, s_3) = \iiint x_{3D}(\mathbf{r}, r_3)h_{3D}(\mathbf{s}, s_3, \mathbf{r}, r_3)d\mathbf{r}dr_3, \quad (2)$$

where we assumed, to simplify the notation, that the magnification is 1. When the microscope is focused at the origin ($s_3 = 0$)

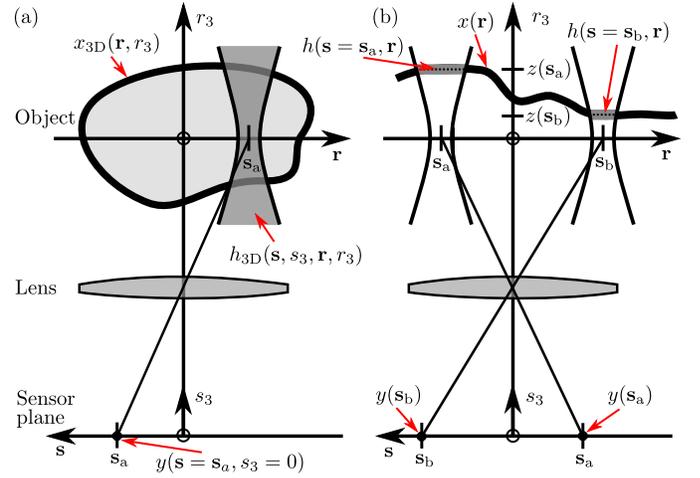


Fig. 1. Object and image formation model. (a) General thick 3D object case, (b) Approximate manifold case.

we define the 2D image $y(\mathbf{s}) = y_{3D}(\mathbf{s}, s_3)|_{s_3=0}$, which can be obtained via the expression:

$$y(\mathbf{s}) = \iiint x(\mathbf{r})\delta(r_3 - z(\mathbf{r}))h_{3D}(\mathbf{s}, s_3 = 0, \mathbf{r}, r_3)d\mathbf{r}dr_3 \quad (3)$$

$$= \iint x(\mathbf{r})h_{3D}(\mathbf{s}, s_3 = 0, \mathbf{r}, r_3 = z(\mathbf{r}))d\mathbf{r} \quad (4)$$

$$= \iint x(\mathbf{r})h(\mathbf{s}, \mathbf{r})d\mathbf{r}, \quad (5)$$

where $h(\mathbf{s}, \mathbf{r}) = h_{3D}(\mathbf{s}, s_3 = 0, \mathbf{r}, r_3 = z(\mathbf{r}))$ is a 2D point spread function that incorporates both the local (3D) variations of the optical system and the variable depth of the thin sample. We further assume that $h(\mathbf{s}, \mathbf{r})$ can be approximated by a parametric function $\tilde{h}_{\mathbf{a}(\mathbf{s})}(\mathbf{r})$, where the N parameters $\mathbf{a}(\mathbf{s}) = (a_1(\mathbf{s}) \ a_2(\mathbf{s}) \ \dots \ a_N(\mathbf{s}))$ can vary for every 2D location \mathbf{s} of the image (Fig. 1 (b)).

B. Parametric degradation models

There are many methods for estimating the PSF of a degraded image. Such methods can be categorized into two classes: direct PSF estimation or parametric modeling. In works such as those by Grossmann et al. [3], the PSFs are estimated directly from the image (e.g. using edge detection and a regression model [24], a Maximum a Posteriori (MAP) prediction [25]), or via a camera calibration using images of a defined and known pattern [26] [27]. Levin et al. [28] showed that a MAP approach to recover the blur kernel is well constrained, but that the MAP global optimum for the recovered image is a blurred image because the strong constraints do not always generalize to unexpected or noisy types of data [29], which are common in microscopy images. Full pixel-wise PSFs can also be estimated using dictionary learning [30], or CNNs [31]. However, this latter kind of estimation is not well constrained and can generate over-fitting artifacts.

In contrast, parametric modeling of the PSF allows to reduce the dimensionality of the optimization problem and to attach a physical meaning to the parameters, such as the relative distance

from the focal point, or optical aberrations such as astigmatism. There are many mathematical models to represent the PSF of a microscope. They can take into account both physical characteristics of the objectives (for example numerical aperture, correction types, etc.) and of the experimental conditions (focal distance and immersion medium) [32]. In many cases, the model parameters correspond to physical design conditions, such as optical distances, aperture diameters, or foci. A simple PSF model can be obtained from the Fraunhofer diffraction theory to calculate the diffraction of a circular aperture [33]. The *Gibson & Lanni* model accounts for the immersion medium, the cover-slip, the sample layers, different medium numerical apertures, and the properties of the objective [1]. Despite their theoretical relevance, in practice, numerical values for these parameters may not be available, as detailed information about all experimental and design conditions may be lacking. Even if the parameters are accessible, they may be missing if they were not recorded along with the image.

Since we aim to recover the PSF from the image itself, with minimal knowledge of the imaging conditions, we focused on models specified by only a small number of parameters or whose complexity can be adjusted progressively by considering approximations with a subset of the complete set of parameters. Specifically, we considered PSF models based on Zernike polynomials, which are used to describe the wavefront function of lenses such as the eye [34], as well as anisotropic Gaussian models.

1) *Zernike polynomial decomposition of the pupil function*: Optical abnormalities, such as de-focus, astigmatism, or spherical aberrations, can be modeled with a superposition of Zernike polynomials $Z_n(\xi)$ in the expansion of the microscope objective's pupil function $W_{\mathbf{a}(s)}$ [35]:

$$W_{\mathbf{a}(s)}(\xi) = \sum_{n=1}^N Z_n(\xi, a_n(s)), \quad (6)$$

where ξ denotes the two-vector of spatial coordinates in the pupil plane perpendicular to the optical axis, N the maximal order of considered aberrations, and $a_n(s)$ the parameter corresponding to each Zernike term Z_n . In our experiments, Z_1 describes the de-focus term, Z_2 describes the power of the astigmatism (cylinder), and Z_3 describes the astigmatism angle (axis). The pupil function can ultimately be converted into a PSF [36]:

$$\tilde{h}_{\mathbf{a}(s)}(\mathbf{r}) \propto \left| \mathcal{F} \left\{ W_{\mathbf{a}(s)} \left(\frac{\mathbf{r}}{\lambda} \right) \right\} \right|^2, \quad (7)$$

with \mathcal{F} the Fourier transform and λ the wavelength of the light.

2) *Anisotropic Gaussian model*: For many applications, Gaussian distributions are sufficiently accurate approximations of the diffraction-limited PSF of wide field microscopes [37]. We extend the model by allowing for anisotropy, which we require to describe astigmatic aberrations, or if the spatial resolution in one lateral direction is different from that in the other. The detection method is in that case similar to the one described in [14]. The PSF is then defined by the anisotropic

zero-centered normal probability density function:

$$\tilde{h}_{\mathbf{a}(s)}(\mathbf{r}) = \frac{1}{\sqrt{2\pi a_1(s)}} \exp\left(-\frac{1}{2} \frac{r_1^2}{a_1(s)}\right) \cdot \frac{1}{\sqrt{2\pi a_2(s)}} \exp\left(-\frac{1}{2} \frac{r_2^2}{a_2(s)}\right), \quad (8)$$

where a_1 and a_2 are the variances of the Gaussian in the x and y axes, respectively.

C. Problem statement

We aim to solve several problems. First, given only an observed degraded image $y(s)$, we want to estimate the PSF model $\tilde{h}_{\mathbf{a}(s)}(\mathbf{r})$ closest to the effective PSF of the imaging system $h(s, \mathbf{r})$ for any point s *without* requiring additional information on the microscope or any further calibration images acquired with that microscope. Specifically, we want to infer the model parameters $\mathbf{a}(s)$, first locally, then globally. Next, given the local PSF parameters $\mathbf{a}(s)$ and the blurred image $y(s)$, we want to recover an estimate of the non-degraded image $x(\mathbf{r})$. Finally, we want to infer the local depth $z(\mathbf{r})$ along the axis of the object at any position \mathbf{r} in the plane perpendicular to the optical axis thereby allowing us to build $x_{3D}(\mathbf{r}, z(\mathbf{r}))$.

D. Method overview

For each of the problems, we summarize the following main steps:

- 1) Shift-invariant PSF parameter estimation given an image patch (see Section II-E2)
 - a) Select a parametric degradation model for $h_{\mathbf{a}}(\mathbf{r})$ allowing the generation of PSF/parameters pairs.
 - b) Gather a training library of microscopy images, degrade each image via a *spatially-invariant* convolution with its corresponding PSF, corrupt it with synthetic noise.
 - c) Train a CNN that takes a degraded image patch as input and returns the corresponding degradation model parameters, via regression.
- 2) Local PSF estimation given a full degraded microscopy image (see Section II-F)
 - a) Given a full microscopy image as input, locally extract a patch, then regress the PSF parameters using the steps above.
 - b) Repeat in all regions of the image.
 - c) Combine the estimated PSF parameters to generate the map $\mathbf{a}(s)$ of the local PSF model parameters.
- 3) Application 1: spatially-variant blind deconvolution (see Section II-G)
 - a) Given an input image and the map of estimated local PSF parameters, generate local PSFs $\tilde{h}_{\mathbf{a}(s)}(\mathbf{r})$.
 - b) Use the generated PSFs in a Total Variation regularized Richardson-Lucy (TV-RL) deconvolution algorithm to recover an estimate for $x(\mathbf{r})$.
- 4) Application 2: estimate depth from focus using PSF engineering (see Section II-H)
 - a) Given an image acquired with a PSF-engineered optical system $y(s)$ [38]), deduce the depth map $z(\mathbf{r})$ from the local parameters $\mathbf{a}(s)$.

b) Generate $x_{3D}(\mathbf{r}, z(\mathbf{r}))$.

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

E. PSF parameter estimation in image patches (shift-invariant image formation model)

1) *Data set generation for CNN training:* Given an image patch as input, we wish to estimate the degradation model parameters corresponding to the spatially-invariant PSF that degraded the patch. Since neural networks are trained by adjusting their internal weights using backpropagation of the derivative of a loss function between the ground truth of the training data and the output of the network [39], we establish a training set T (Fig. 2). For that purpose, we gather input images $x^{(k)}$. Since reliable public image data sets have limited size, we augment the size of the training sets by rotating each image by an angle of $\pm 90^\circ$, so we have K images, $x^{(k)}$, $k = 0, \dots, K-1$. We induce a synthetic degradation by convolving the images with K generated PSFs $h_{\mathbf{a}^{(k)}}$ of model parameters $\mathbf{a}^{(k)}$ drawn from a normalized and scaled uniform distribution:

$$\psi^{(k)}(\mathbf{s}) = (h_{\mathbf{a}^{(k)}} * x^{(k)})(\mathbf{s}). \quad (9)$$

We also consider the two predominant sources of noise in digital image acquisition: the stochastic nature of the quantum effects of the photoconversion process and the intrinsic thermal and electronic fluctuations in the CCD camera [40]. The first source of noise comes from physical constraints such as a low-power light source or short exposure time, while the second is signal-independent. This motivates the noise model as a mixed Gaussian-Poisson noise process. Therefore, we define noise with the two following components:

- A random variable $n_p(\mathbf{s}) \sim \mathcal{P}(\lambda = \psi^{(k)}(\mathbf{s}))$ following a Poisson distribution of probability $P\{n_p(\mathbf{s}) = i\} = e^{-\lambda} \lambda^i / i!$.
- A random variable $b(\mathbf{s}) \sim \mathcal{N}(0, \sigma^2)$ following a Gaussian distribution with zero-mean and variance σ^2 .

The image noise model for data set generation is then:

$$\psi_{\text{noisy}}^{(k)}(\mathbf{s}) = \beta n_p(\mathbf{s}) + b(\mathbf{s}), \quad (10)$$

with β a number between 0 and 1 reflecting the quantum efficiency of the CCD [41]. Images that did not comply with a minimal variance and white pixel ratio were tagged as invalid, i.e. $a_0^{(k)} = 1$ (see Section II-E2).

We cropped the images $\psi_{\text{noisy}}^{(k)}(\mathbf{s})$ to a size $K_\psi \times L_\psi$ by randomly selecting the position of a region of interest of that size. We then paired these image patches with their respective PSF parameter vector $\mathbf{a}^{(k)}$ in order to form the training set $T = \{(\psi_{\text{noisy}}^{(k)}(\mathbf{s}), \mathbf{a}^{(k)})\}_{k=0}^{K-1}$.

2) *CNN training modalities:* We considered several neural networks (whose architectures we further describe below) and trained them to learn the PSF model parameters described in Section II-B. The task of the network is to estimate, only from the k^{th} input image patch $\Psi^{(k)}(\mathbf{s})$, the parameters $\tilde{\mathbf{a}}^{(k)}(\mathbf{s})$ that have been used by the PSF model to degrade that input image. Since there are cases where the PSF estimation is not possible, e.g. where the sample lacks texture, such as in uniformly black

or grey areas, we added a boolean parameter $a_0^{(k)}$ (whose values can be either 0 or 1), which indicates the legitimacy of the sample. The total number of estimated parameters is then $N + 1$. We aim at minimizing the distance between the output of the network $\tilde{\mathbf{a}}^{(k)}$ and the ground-truth PSF parameters $\mathbf{a}^{(k)}$. Therefore, in the training phase, we updated the weights of the CNN using the modified Euclidean loss function:

$$E^{(k)} = \gamma \left(a_0^{(k)} - \tilde{a}_0^{(k)} \right)^2 + \frac{1 - a_0^{(k)}}{2N} \sum_{n=1}^N \left(a_n^{(k)} - \tilde{a}_n^{(k)} \right)^2, \quad (11)$$

with γ a hyperparameter regulating the importance of the validity parameter, that we set to 1 in our further experiments. We choose to rely on networks that showed good performance in the ImageNet competition, which is a benchmark in object classification on hundreds of categories [42], [43]. Donahue et al. [44] showed that deep convolutional representations can be applied to a variety of tasks and detection of visual features, which drove our selection for estimating optical aberrations. Hendrycks et al. [45] extensively discussed whether the networks were robust to changes in input illumination, noise, and blur. While residual networks that use skip-connections such as ResNet [46] appear to be more robust to input noise than primitive feedforward networks such as AlexNet [9], their performance appears to be surpassed by newer multibranch models such as ResNeXt [47] or Densenet [48]. In the context of our specific task, we compared the performance (see Section III-B) and the robustness to degradation (see Section III-C) of several of the above architectures.

After training, the networks can regress the spatially-invariant PSF parameters $\tilde{\mathbf{a}}^{(k)}(\mathbf{s})$ from a single input image patch $\Psi^{(k)}(\mathbf{s})$.

F. Spatially-variant PSF parameter mapping

Given a trained CNN that is able to recover the degradation parameters from a single image patch, we now turn to the problem of locally estimating the parameters of the different PSFs that degraded a larger input image. To achieve this, we use an overlapping sliding window over the input image $x(\mathbf{r})$ with stride t that is fed into the locally invariant regression CNN trained in Section II-E2 (see Fig. 3). We store the resulting parameters in the map $\tilde{\mathbf{A}} = (\tilde{\mathbf{a}}^{(0)} \quad \tilde{\mathbf{a}}^{(1)} \quad \dots \quad \tilde{\mathbf{a}}^{(M)})$, where $\tilde{\mathbf{a}}^{(m)}$ is the output of the neural network for patch m and M is the total number of patches. Using the PSF model, we generate from $\tilde{\mathbf{A}}$ a spatially-variant map of local PSF kernels defined as $\tilde{\mathbf{H}} = (\tilde{h}^{(0)}(\mathbf{r}) \quad \tilde{h}^{(1)}(\mathbf{r}) \quad \dots \quad \tilde{h}^{(M)}(\mathbf{r}))$.

The overlapping window over the input image yields a map of $(\lfloor (K_x - K_\psi)/t \rfloor + 1) \times (\lfloor (L_x - L_\psi)/t \rfloor + 1)$ kernels, with K_x , L_x , and K_ψ , L_ψ being the width and height of the input image and the window step size, respectively. For example, a 1024×1024 pixel input image using 128×128 pixel patches and $t = 64$ yields $M = 13 \times 13$ spatially-dependent PSF kernels. We fill every patch with a validity parameter $a_0 = 1$ (i.e. invalid) with the content of the inverse Euclidean distance-weighted average of the four-connected nearest neighbors using K-Nearest Neighbor regression [49] in order to avoid boxing artifacts during the later deconvolution process.

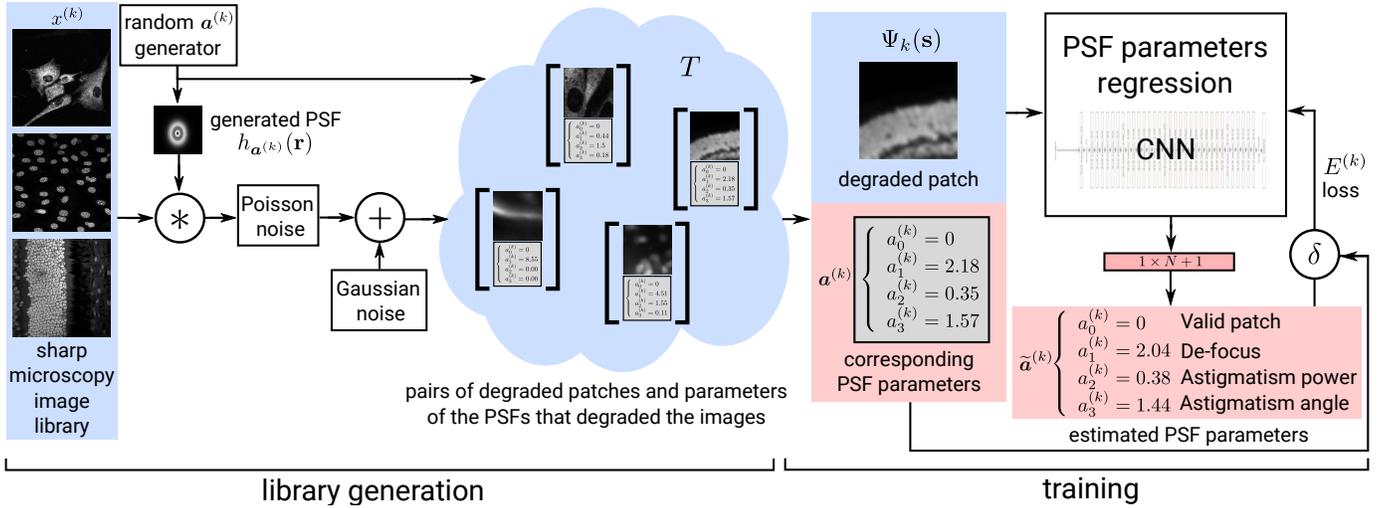


Fig. 2. Data set gathering (left) and CNN training (right) pipelines. From a large library of sharp microscopy images, small patches are created, blurred with a PSF generated from random parameters, and degraded with a Poisson-Gaussian noise mixture (see Section II-E1). The resulting patches and the parameters are stored in the training set T , that is used for training the CNN. Using backpropagation of the loss function, the CNN output is trained towards the prediction of the PSF model parameters (Section II-E2).

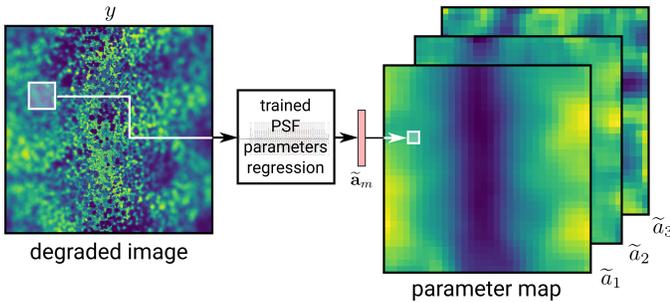


Fig. 3. Spatially-variant PSF parameter mapping using a sliding window over the acquired image, as input of a convolutional neural network.

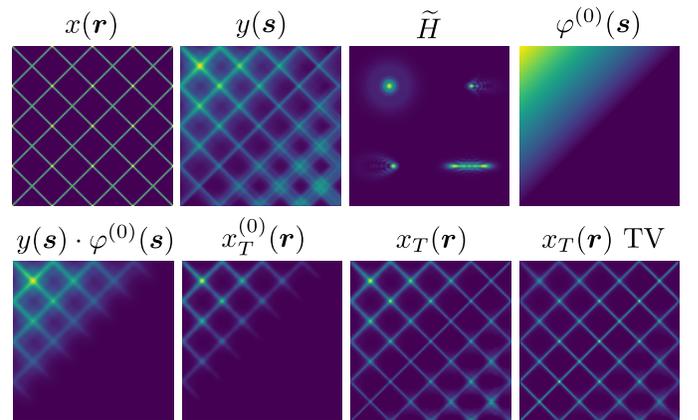


Fig. 4. Synthetic experiment involving variables and results of Eq. (13). Starting from a ground truth image ($x(\mathbf{r})$), a local map of PSF (smooth interpolation between the 4 shown PSFs) and local weight combination $\varphi(\mathbf{s})$, we generated a blurred image y . The deconvolution method (similar to [50] and [51]) starts from a map of locally estimated PSFs to deconvolve and recombine into a single image $x_T(\mathbf{r})$ and $x_T(\mathbf{r})$ TV.

G. Application 1: spatially-variant blind deconvolution

We now turn to the problem of deconvolving the image. Existing 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 full knowledge of the PSF ([30], [52]), while the latter two classes aim at improving the image without prior knowledge of the PSF, the object, or other optical parameters. Entirely blind algorithms, such as [50] are based on optimization and estimation of the latent image or kernel [28]. Many blind deconvolution techniques are computationally expensive, especially for larger convolution kernels, and assume spatially invariant PSFs.

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 ([4], [53]). Parametric models allow reducing the complexity of the optimization problem, increasing the overall robustness, and avoiding issues such as over-fitting. However, it remains difficult to estimate the parameters from experimental data. We will focus on this third class of deconvolution methods,

since our local PSF estimation method allows inferring their parameters without measuring any of them experimentally. Since we set a diffraction model defining the shape of the local PSF in a large image, we describe our deconvolution method as semi-blind.

Given the degraded image and a local map of PSF parameters, we restore the input using TV-RL deconvolution. Richardson-Lucy (RL) is an iterative maximum-likelihood approach and assumes that the noise follows a Poisson distribution [54], 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 ([55], [51]). Here, we assume that the PSF is spatially invariant in small parts of the image. Spatially-variant convolution techniques have been extensively reviewed by Denis et al. [56]. Hirsch et al. [50] have shown that the local

invariance assumption can be improved by filtering the input with every local PSF and then reconstructing the image using interpolation. We extend this method by its inclusion in the TV-RL algorithm. Rather than interpolating deconvolved images, the overlap-add filtering method, as described in [50], [57], interpolates the PSF for each point in the image space. The idea for such method is: (i) to cover the image with overlapping patches using smooth interpolation, (ii) to deconvolve each patch with a different PSF, (iii) to add the patches to obtain a single large image. The equivalent for convolution can be written as:

$$x(\mathbf{r}) = \sum_{m=0}^M (\tilde{h}^{(m)} * (\varphi^{(m)} \odot y))(\mathbf{r}), \quad (12)$$

where $\varphi^{(m)}(\mathbf{s})$ is the masking operator of the m th patch. We illustrated the masking and deconvolution steps in Fig. 4. Since the RL algorithm tends to exacerbate edges and small variations such as noise, we use Total Variation (TV) regularization to obtain a smooth solution while preserving the borders [51]. The image at each RL iteration becomes:

$$x_{i+1}(\mathbf{r}) = \sum_{m=0}^M \left[\frac{(\tilde{h}^{(m)} * (\varphi^{(m)} \cdot y))(\mathbf{r})}{(\tilde{h}^{(m)} * x_i^{(m)}) (\mathbf{r})} * \tilde{h}^{(m)}(-\cdot) \right] \cdot \frac{x_i^{(m)}(\mathbf{r})}{1 - \lambda_{TV} \operatorname{div} \left(\frac{\nabla x_i^{(m)}(\mathbf{r})}{|\nabla x_i^{(m)}(\mathbf{r})|} \right)}, \quad (13)$$

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

H. Application 2: depth from focus using astigmatism

The spatially-variant PSF parameter mappings obtained in Section II-F yield local parameters, such as the blur, that are a function of the distance of the object to the focal plane. However, due to the symmetry of the PSF in depth, this function is ambiguous about the sign of the distance map $z(\mathbf{s})$ (above or below the focal plane). That is why we now aim at estimating the depth map $z(\mathbf{s})$ for every lateral pixel \mathbf{s} of the 2D manifold in 3D space using our trained neural network and one single image as input. To achieve this, we use a local combination of Zernike polynomial coefficients $\mathbf{a}(\mathbf{s})$. The de-focus coefficient $a_1(\mathbf{s})$ is linked to the distance of the object to the focal plane, but there is no information about whether the object is in the front or behind the focal point. To address this limitation, we took inspiration from several methods to retrieve the relative position of a particle by encoding it in the shape of its PSF (either via use of astigmatic lenses ([38], [58]) or by use of a deformable mirror to generate more precise and complex PSF shapes [59]).

We used two cylindrical lenses of focal length $-400 \mu\text{m}$ and $400 \mu\text{m}$, separated by 3.4 cm thereby giving a combined focal length of $f = 6000 \text{ mm}$ and placed them in the infinity space of a microscope to generate an imaging system with an astigmatic

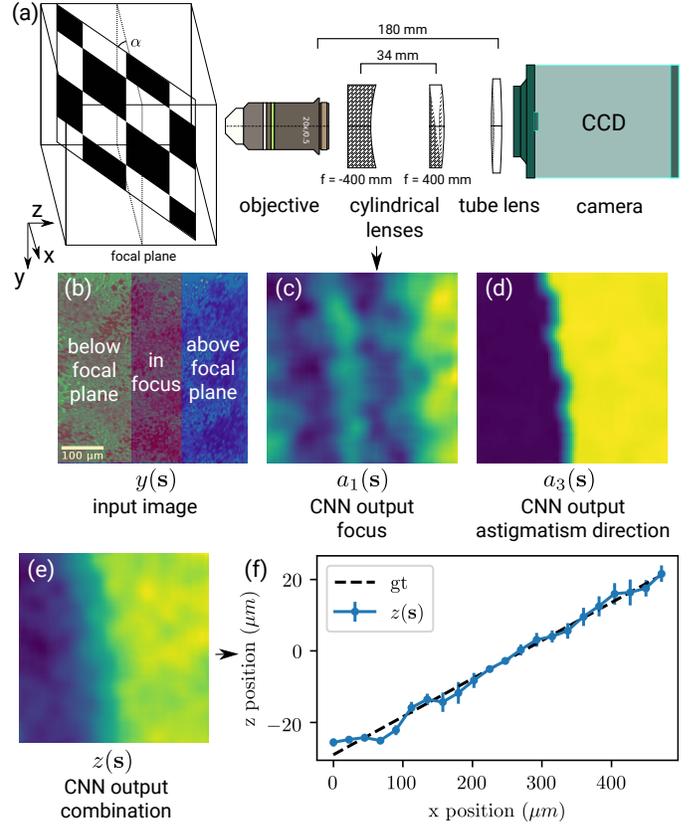


Fig. 5. Depth estimation of a plane using controlled astigmatic aberrations by use of cylindrical lenses in the infinite tube space to induce astigmatism. (a) Optical system with using cylindrical lenses in the infinite tube space to induce astigmatism. (b) Image of a grid taken from the camera, with highlighted parts of the surface above and below the focal plane. (c),(d) Output of the CNN using (b) as input. The resolution of the map is $M = 31 \times 31$ different PSFs. (e) Weighted combination of (c) and (d) to form the depth map. (f) Projection of the depth of the surface over the y axis and comparison to the ground truth data.

PSF (Fig. 5). We used the networks trained in Section II-E1 using 2D-Zernike models to infer the depth map $z(\mathbf{s})$ from the 2D image $y(\mathbf{s})$ of the tridimensional surface $x_{3D}(\mathbf{r}, z(\mathbf{r}))$. We defined a distance metric by multiplying the output focus parameter and the normalized and zero-centered astigmatism direction:

$$z(\mathbf{s}) = a_1(\mathbf{s}) \left(\frac{2a_3(\mathbf{s})}{\pi} - 1 \right), \quad (14)$$

with $a_1(\mathbf{s})$ the spatially local de-focus Zernike coefficient, and $a_3(\mathbf{s}) \in (0, \pi)$ the spatially local Zernike coefficient encoding the direction of astigmatism.

III. EXPERIMENTS

In this section we report our efforts to characterize the performance of our method as well as its dependency to several hyper-parameters, such as the choice of the PSF model, the neural network architecture, the training set size, or its content. We furthermore tested the regression performance of the PSF parameter regression and its robustness to Signal-to-Noise Ratio (SNR) degradation and the absence of texturing. Finally, we also assessed the quality of deconvolved images and the accuracy of the estimated depths.

TABLE I
NAME AND SIZE OF THE DIFFERENT TRAINING, VALIDATION AND TEST
DATA SETS AS INPUT OF THE CNN.

Data set	K_{train}	K_{valid}	K_{test}
syn	440,000	10,000	10,000
poi	330,000	10,000	10,000
micr	2,700,000	10,000	10,000
micrsm	270,000	10,000	10,000
nat	2,700,000	10,000	10,000
micr-syn-poi	3,470,000	10,000	10,000

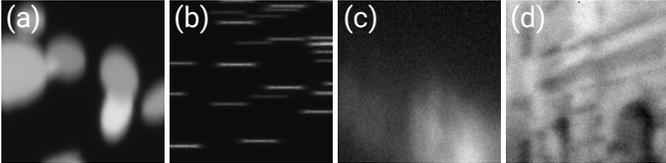


Fig. 6. Examples of degraded input patches from different data sources: (a) synthetic cells [syn], (b) synthetic points [poi], (c) microscopy images [micr], (d) natural images [nat]. The images have been degraded by the Zernike-3 PSF model (see Table II) and noise as described in Section II-E1

A. Infrastructure

Our PSF parameter estimation method depends on three main variables: the content and size of training data sets, the PSF parametric model and the neural network architecture. We briefly describe the different options below.

1) *Training, validation and test data sets for CNN regression performance:* We gathered images from four different data sources:

- 1) [**micr**] microscopy images collected from [60], [61] and [62],
- 2) [**nat**] common images from the MIT Places365 data set [63] that gathers natural and man-made photographs,
- 3) [**poi**] synthetic images of points on a black background (Fig. 6),
- 4) [**syn**] synthetic images of cells (Fig. 6).

The rationale for using natural and synthetic images is that these data sources are much more abundant than microscopy images, often sharper and royalty free, making it possible to quickly assemble a large dataset. We combined these data to generate six different data sets (Table I) and prepared the library as described in Section II-E1. We randomly selected two times 10,000 images to form a validation set and an test dataset that the networks never use during the training process. The validation dataset is used for selecting the best learning rate and early stopping epoch for every training, while the testing set is used for performance assessment. We added synthetic black images to every data set to avoid misdetection of non-textured parts of the image and explicitly set $a_0^{(k)} = 1$ for these samples.

2) *PSF models and parameters:* We considered two different PSF model types: Zernike polynomials (Section II-B1) with $N = 1, 2$, or 3 parameters, and Gaussian (Section II-B2), with either $N = 1$ or 2 parameters, as described in Table II.

3) *CNN architectures and training modalities:* We compared two residual neural networks architectures trained from scratch: 34-layer ResNet [46] and 50-layer ResNeXt [47]. They were

TABLE II
PSF MODELS SELECTED FOR DATA SET GENERATION (SECTION II-E1),
WITH THE NUMBER AND NAME OF FREE PARAMETERS.

PSF model	N	Parameters
Zernike-1 (Z-1)	1	focus
Zernike-2 (Z-2)	2	cylinder, axis
Zernike-3 (Z-3)	3	focus, cylinder, axis
Gaussian-1 (G-1)	1	width
Gaussian-2 (G-2)	2	width x axis, width y axis

adapted for accepting normalized gray-scale input image patches of size $W_\psi \times H_\psi = 128 \times 128$ pixels. Additionally, we fine-tuned, using our training dataset, the same network already trained on the ImageNet data set (available on the PyTorch website). For the latter model, we re-scaled the input images to the network input size using bilinear interpolation. We trained the models for 20 epochs with PyTorch 1.0 using the Adam optimizer [64] and a learning rate between 0.001 and 0.01 defined by the validation set performance.

B. Characterization of the CNN regression performance

We analyzed the performance of our system for regressing the PSF parameters. The metrics we used to assess the performance of the network is the goodness-of-fit of the parameter estimation compared to the ground truth. We quantified it in terms of the squared Pearson correlation coefficient R^2 averaged over all PSF parameters:

$$R^2 = \frac{1}{N} \sum_{n=0}^N \left(1 - \frac{\sum_{k=0}^{K_{test}-1} (a_n^{(k)} - \tilde{a}_n^{(k)})^2}{\sum_{k=0}^{K_{test}-1} (a_n^{(k)} - \bar{a}_n)^2} \right), \quad (15)$$

with K_{test} the number of samples in the test set. We calculated the correlation coefficient only for samples that contained texture in the ground-truth (i.e. when $a_0^{(k)} = 0$) and discarded the others.

1) *Characterization of CNN regression performance when training and test data set types are the same:* We started by assessing the performance of the CNNs when the test set is made of the same image type as the training set. Table III summarizes the performance of the regression of test data for every combination of training data sets (Table I), PSF models (Table II) and CNN architectures.

Variables: data set type, PSF model type, network architectures.

Fixed: the data set type is the same for training and testing.

Evaluation criterion: R^2 between the degradation parameters used to generate the test image and the parameters recovered by the CNN.

In most cases, the correlation coefficient is superior to 80%, which indicates a very good degree of overall correlation. The worst cases are with models trained for Zernike-3, that yield $0.61 < R^2 < 0.96$. We notice a few differences in the regression performances between Gaussian and Zernike models. Indeed, images blurred with a Gaussian model tend to

TABLE III

RESULTS OF REGRESSION ANALYSES (IN TERMS OF R^2) FOR $N = 10,000$ TEST IMAGES IN DATA SETS SHOWN IN TABLE I USING THE SAME DATA TYPE FOR TRAINING, VALIDATION (MODEL SELECTION) AND TEST.

		ResNet-34					
		[syn]	[poi]	[micr]	[micrsm]	[nat]	[micr-syn-poi]
PSF models	Z-1	0.99	0.99	0.98	0.73	0.98	0.68
	Z-2	0.67	0.99	0.81	0.80	0.80	0.79
	Z-3	0.95	0.98	0.78	0.58	0.84	0.88
	G-1	0.99	0.99	0.98	0.92	0.92	0.99
	G-2	0.99	0.99	0.99	0.97	0.91	0.99
		ResNet-34-pretrained					
		[syn]	[poi]	[micr]	[micrsm]	[nat]	[micr-syn-poi]
PSF models	Z-1	0.99	0.99	0.99	0.89	0.99	0.81
	Z-2	0.93	0.97	0.92	0.81	0.94	0.90
	Z-3	0.97	0.89	0.95	0.77	0.80	0.89
	G-1	0.94	0.99	0.98	0.99	0.94	0.99
	G-2	0.99	0.99	0.99	0.98	0.95	0.99
		ResNext-50					
		[syn]	[poi]	[micr]	[micrsm]	[nat]	[micr-syn-poi]
PSF models	Z-1	0.98	0.99	0.97	0.69	0.97	0.65
	Z-2	0.69	0.98	0.85	0.74	0.90	0.85
	Z-3	0.64	0.94	0.90	0.72	0.61	0.85
	G-1	0.99	0.99	0.97	0.80	0.91	0.98
	G-2	0.99	0.99	0.99	0.92	0.97	0.98

be better recognized by the neural network, with $R^2 > 0.90$, than images blurred with a Zernike model that fluctuates around $0.60 < R^2 < 1.00$. When looking at the performance of a smaller [micr] training data set compared to the full [micr] data set, we notice that the performance of the smaller data set is always worse or equal, no matter which CNN model or PSF model used. Finally, we observe that the overall performance of ResNext-50 is lower than the performance of both ResNets.

2) *Characterization of CNN regression performance when the type of training and test data set differ:* We assessed the robustness of our regression method when the system is tested on image types other than those it has been trained for.

Variables: data set types for both training and test sets.
Fixed: the network architecture (ResNet-34), the PSF model (Gaussian-2). The model is already trained and selected using an independent validation dataset.
Evaluation criterion: R^2 between the degradation parameters used to generate the test input and the parameters recovered by the CNN.

Table IV gathers the regression performance obtained using a Gaussian-2 PSF model and the ResNet-34 network, with training and testing data sets of different types. The regression is robust to different train and test data set types ($R^2 < 0.90$) except when the CNN is trained with [poi] and, to a lesser extent, with [syn]. Surprisingly, networks trained on natural ([nat]) images perform as well as networks trained on microscopy images.

TABLE IV

EVALUATION (IN TERMS OF R^2) FOR $N = 10,000$ IMAGES INPUT INTO A RESNET-34-PRETRAINED NETWORK TRAINED FOR REGRESSION OF THE GAUSSIAN-2 PSF MODEL PARAMETERS WITH DIFFERENT TRAINING SET / TEST SETS PAIRS. THE LEARNING RATE AND EPOCH WERE SELECTED USING RESULTS FROM A TEST SET OF $N = 10,000$ SEPARATE IMAGES.

		Test data sets					
		[syn]	[poi]	[micr]	[micrsm]	[nat]	[micr-syn-poi]
Training data sets	[syn]	1	0.93	0.76	0.83	0.81	0.85
	[poi]	0	1	0	0	0	0
	[micr]	0.97	0.97	0.99	0.99	0.99	0.99
	[micrsm]	0.97	0.95	0.95	0.98	0.89	0.97
	[nat]	0.99	0.80	0.96	0.93	0.96	0.96
	[micr-syn-poi]	1	0.99	0.99	0.99	0.94	0.99

C. Robustness of PSF regression against input degradation

Degradations on the input images are unavoidable in biological environments. Indeed, microscopes are often used for a variety of sample types and preparations and are calibrated by different people. Settings such as illumination brightness, exposure time, and contrast frequently change or are operator-dependent. Furthermore, as described in Section II-E1, low light and electronics induce noise in the acquired image. Since we aim at training a regression network that is not specific to any defined acquisition condition, we characterized the robustness of the neural network to extrinsic modifications of the image quality. The list of handled degradations is the following:

- global illumination level,
- non-uniform illumination (e.g. caused by poorly adjusted Köhler illumination),
- zero-mean Gaussian noise,
- signal-dependent Poisson noise,
- mixed Gaussian-Poisson noise.

Variables: degradation strength, degradation type, three different CNNs, two different PSF models (G-1, Z-1).
Fixed: the CNNs are already trained with [micr-syn-poi] and selected using an independent test dataset. The test data set is common to all modalities.
Evaluation criterion: R^2 between the degradation parameters that were used to generate the test image and the parameters recovered by the CNN.

Fig. 7 summarizes the performance of the CNN networks as a function of degradation strength. All networks are robust to partial or full brightness changes in the input image. The addition of Gaussian noise to the input results in a slow and linear decay in performance, whereas the application of Poisson noise to the data decrease the performance much faster as the noise strength increases. Using CNNs trained for regression of Zernike polynomial PSF model parameters, the regression performance is decaying linearly as a function of the amount of noise we apply in the input picture. For Gaussian models, the parameter estimation usually breaks with less noise than with the Zernike polynomial model. Without any degradation, networks regressing Zernike polynomials are less accurate than networks for Gaussian PSF models, but they appear to be more robust when the input is noisy. Indeed, with a very

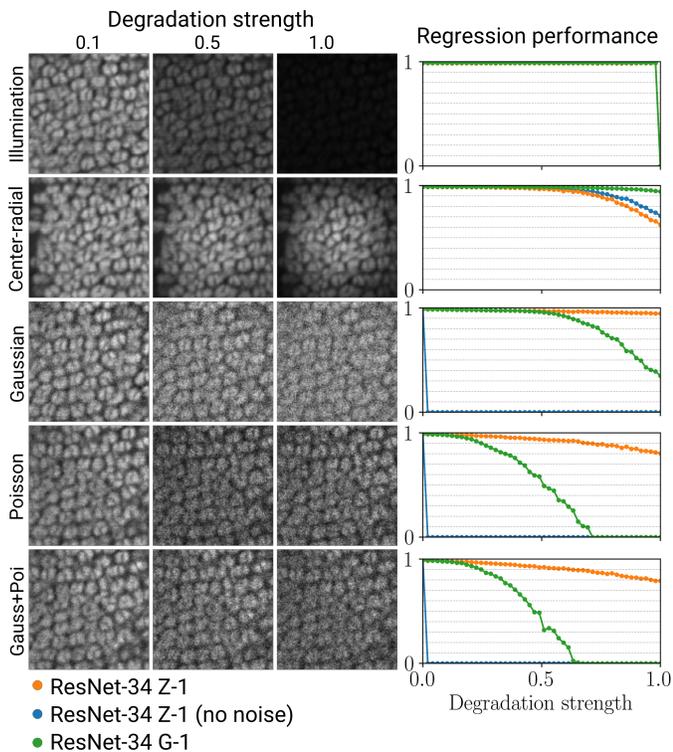


Fig. 7. Regression performance of CNNs architectures trained on [micr] and evaluated on the [micr] test set with various types of degradation and variable strength. The regression performance is shown in terms of R^2 using $N = 10,000$ images. We trained as well a CNN without adding noise in the training set (blue). All three networks exhibits the same performance for the illumination degradation (top right).

strong (strength of 0.5 in Fig. 7) Gaussian and Poisson noise, Zernike-1 CNNs dropped from $R^2 = 0.95$ to $R^2 = 0.85$, as opposed to the Gaussian-1 CNNs, which dropped from $R^2 = 0.99$ to less than $R^2 = 0.60$. Surprisingly, contrary to the findings in a recent benchmark [45], we found that CNNs based on ResNeXt performed worse than their ResNet counterparts. Finally, we trained new models without adding synthetic noise to the training dataset (see Eq. (10)). Performance of these networks was the same as their counterparts for the illumination degradations, but dropped to $R^2 = 0$ when the test images contained even only moderate Poisson and Gaussian noise.

D. Application 1: spatially-variant blind deconvolution

We next wanted to verify that the parameters recovered by the CNN were producing PSFs that are sufficiently accurate to be usable to enhance the details in the image, despite not being specifically measured. To this end, we devised a deconvolution experiment to compare images deconvolved by our method with those obtained by other blind deconvolution techniques. As test input, we used 256×256 pixels image patches from the [micr] data set (see Section II-E1). Using Eq. (10), we degraded each quadrant of the input image with a specific, randomly-generated 127×127 pixels PSFs using parameters $\alpha(s)$ drawn from a uniform random distribution allowing us to systematically explore the parameter space. We subsequently inferred a PSFs map via the CNN from the blurry image as

described in Section II-F. Finally, we deconvolved the image by applying the method described in Section II-G. In the experiment, we set $\varphi^{(m)}$ as a bilinear interpolating function and, similarly to [51], $\lambda_{TV} = 0.1$. Since using FFT-based calculations implies that the PSF is circulant, we took into account field extension to prevent spatial aliasing. We fixed the number I of RL iterations to 20.

We assessed the reconstruction quality by computing the SNR and Structural Similarity (SSIM) [65]. We compared the deconvolution results to spatially-invariant blind deconvolution techniques [66], [67] and [68], and the spatially-variant method from [69]. In the latter cases, we used the estimated PSF in the TV-RL algorithm with the same number of iterations and λ_{TV} . Since the estimation of a full PSF by these methods would take more than 20 minutes per sample, we constrained the support of the PSF to 31×31 pixels. We computed the scores by taking the difference between the “ground truth” SNR and SSIM of images deconvolved using the PSFs actually used to degrade the images, and the deconvolution results using PSFs regressed with the CNN or other BD techniques. These values are therefore reported as ΔSNR and ΔSSIM .

Finally, in order to recover details lost due to the aberrations of actual microscope objectives (such as out-of-focus blur and astigmatism), we acquired different fixed samples (HeLa cells actin (Alexa Fluor 635) and HeLa cells anti- α -catenin (Alexa Fluor 488) with a $10\times/0.3$ air objective, *Convallaria majalis* bulb autofluorescence with a $20\times/0.7$ air objective) both in focus and slightly out of focus. Then, starting from a 256×256 patch of the out-of-focus picture only, we sought to retrieve a sharper picture containing the details of the in-focus picture. We compared qualitatively the in-focus image, the out-of-focus image, our method with four PSFs detected with a 128×128 stride, [66], [69], and the imaged obtained via a “sharpen” high-pass filter.

Variables: network architecture, two different PSF models (Z-1 and G-1) for the degradation and detection parts.

Fixed: CNNs are already trained, the test data set [micr] is fixed.

Evaluation criterion: difference of SNR and SSIM between the ground truth image and the deconvolved image.

Results in Table V indicate an average improvement of both SNR (1.88 dB) and SSIM (0.09) of our spatially variant BD. In comparison to spatially-invariant BD and other spatially-variant BD techniques improves the image by 1.55 dB SNR and 0.08 SSIM. Deconvolution results are equivalent when the degradation and detection models are mismatched. The qualitative results shown in Fig. 8 highlight the stability of our method, which improves the degraded image to a detail level similar or better than the one of the in-focus image. Using the algorithm of Kotera et al. [66], the blurry features are well recovered, but the images have less detail. Furthermore, this algorithm converges to an aberrated image (Fig. 8 (b)) when the image contains long filaments. The method from Whyte et al. [69] enhances the contrast of the blurry image, however,

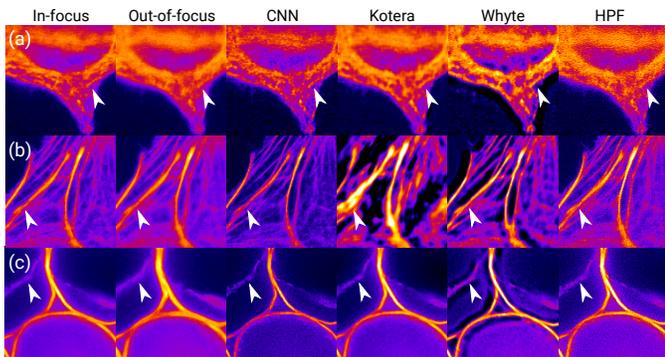


Fig. 8. We acquired different fixed samples ((a) HeLa cells actin (Alexa Fluor 635) and (b) HeLa cells anti- α -catenin (Alexa Fluor 488) with a $10\times/0.3$ air objective, (c) *Convallaria majalis* bulb autofluorescence with a $20\times/0.7$ air objective) both in-focus and slightly out-of-focus. Then, starting from a out-of-focus 256×256 patch only, we seek to retrieve a sharper picture containing the details of the in-focus picture. We compared qualitative results of (from left to right), the in-focus image, the out-of-focus image, the proposed method (CNN) with four PSFs detected with a 128×128 stride, [66], [69], and a "sharpen" high-pass filter. Arrow edges indicate regions with features of particular interest.

it creates hallucinations near edges, which were not part of the original image. Finally, the high-pass filter, as expected, enhances both high-frequency features and noise.

E. Application 2: depth from focus using astigmatism

The PSF parameter describing the local blur that we obtained from the image in Section II-F lacks information about the relative direction of the object from the focal plane. To infer the local axial distance map $z(s)$ in the sample, we applied the method described in Section II-H using the astigmatism created by a cylindrical lens. As an imaging sample, we used a grid of $200\mu\text{m} \times 200\mu\text{m}$ squares which we laser-printed on a transparent plastic foil. We placed the grid towards the focal plane and tilted it by 3° , 6° , or 10° , so that the in-focus position was in the middle of the field of view (Fig. 5 (a)). With a field of view of $655\mu\text{m} \times 655\mu\text{m}$, such a rotation yielded depth ranges of $48.2\mu\text{m}$, $94.7\mu\text{m}$, or $159.9\mu\text{m}$, respectively. We were able to retrieve the local Zernike coefficient parameters of focus (a_1), cylinder (a_2), and axis (a_3). We inferred the depth map $z(s)$ using Eq. (14) for $N = 30$ acquired images in total (see Fig. 5 (e)).

Variables: input images of surfaces with a varying tilt angle from the focal plane.

Fixed: the network architecture (ResNet-34-pretrained for regression of Zernike parameters with astigmatism). CNNs are already trained, the test data set is fixed.

Evaluation criterion: ℓ_1 -error between the actual axial position of the surface and the position extracted from the image.

Using a CNN ResNet-34 trained with [micr], we obtained a correlation coefficient (R^2) between the average slope and a line fit of more than 0.96. From this line fit we calibrated the

system to spatial units so that we could build a depth map. The method was accurate with an absolute average ℓ_1 -error of $1.81 \pm 1.39\mu\text{m}$ in depth (corresponding to a $1.61 \pm 1.23\%$ of the relative depth boundaries), obtained by comparing the error between the known position of the object in depth and the calibrated distance. Results in Table VI reveal that the relative error increases as the maximum depth of the object increases. Depth estimation is thus more precise around the focal plane.

IV. DISCUSSION

Hereafter, we discuss the results of the experiments described in Section III.

A. Characterization of the CNN regression performance

Many regression accuracies are above $R^2 = 0.90$ in Table III, which shows that our neural networks can accurately regress PSF model parameters (in particular, when images are textured). The recovered parameters can be used to generate synthetic PSFs that are similar to the ones that degraded the image.

Our network is most accurate when applied to images of the same type as the ones used for training and the performance scales with the size of the training set. Therefore, the more data we gather, the more precise and robust the predictions are. However, adding synthetic training data to augment a natural images data set does not increase the efficacy of PSF estimations. The network fails to predict the PSF parameters if the network is trained with a very narrow type of data and more variety increases generalization potential. Nevertheless, when learning from a data set of images containing texture, even if different from the test set type, the model remains as accurate as when training the data set using microscopy images. This suggests that one could avoid the need to gather costly microscopy ground-truth data, or that it might be possible to learn from images of other microscope types (e.g. confocal microscopes) and use the trained models with wide field microscopy images. Furthermore, the high correlation score ($R^2 > 0.8$) obtained for [micr] test images using networks trained with [nat] suggests that the networks did not undergo overfitting and were able to generalize on other data types.

In comparison to ResNet-34, ResNeXt-50 requires a larger number of images to be accurate since the regression accuracy drops drastically (from $R^2 = 0.97$ to $R^2 = 0.79$) using a smaller data set. It is consistent with the general idea that the amount of training data must scale with the depth of the network to be able to generalize well.

Networks trained for Gaussian PSFs estimate parameters with a better accuracy than networks trained to find Zernike polynomial parameters. This could be explained by the fact that although the Zernike polynomial parameters are independent when describing the pupil function, they can compensate each other when forming the PSF (which is obtained by a non-linear operation on the pupil function, see Eq. (7)).

Using an NVIDIA GeForce GTX 1080 GPU, the estimation of the PSF parameters of a 1024×1024 px image with 64 PSFs takes around 5.9 ± 0.1 ms, which is in the same scale as the usual camera exposure time. This suggests that real-time

TABLE V

IMAGE DECONVOLUTION BENCHMARK WITH 350 INPUT IMAGES (256×256) FROM THE TEST SET [MICR] FOR EACH EXPERIMENT. WE COMPUTED THE DIFFERENCE IN SNR AND SSIM BETWEEN THE GROUND TRUTH IMAGE AND THE DECONVOLVED IMAGE. WE COMPARED THE FOLLOWING METHODS: TV-RL WITH FOUR KNOWN PSFs (GT), TV-RL WITH FOUR PSFs ESTIMATED BY THE PROPOSED CNN ($\text{CNN}_{\text{BLIND}}$), KOTERA ET AL., 2013 [66], WHYTE ET AL., 2014 [69], DONG ET AL. [67] AND JIN ET AL. [68]. THE CNN IS A RESNET-34 TRAINED ON DATA SET [MICR].

		blurry	gt	$\text{CNN}_{\text{blind}}$	Kotera [66]	Whyte [69]	Dong [67]	Jin [68]	
Degradation	Detection			$\Delta\text{SNR}_{\text{gt}}$	$\Delta\text{SNR}_{\text{CNN-blind}}$	$\Delta\text{SNR}_{[66]}$	$\Delta\text{SNR}_{[69]}$	$\Delta\text{SNR}_{[67]}$	$\Delta\text{SNR}_{[68]}$
Z-1	Z-1			2.14 ± 0.71	1.65 ± 0.62	-0.81 ± 0.71	0.12 ± 0.31	0.61 ± 0.58	1.34 ± 0.62
Z-1	G-1			2.14 ± 0.71	1.72 ± 0.71	-0.61 ± 0.67	0.02 ± 0.31	0.83 ± 0.59	1.48 ± 0.67
G-1	Z-1			2.24 ± 0.73	2.02 ± 1.04	-1.20 ± 0.97	-0.01 ± 0.55	0.89 ± 0.74	1.78 ± 0.76
G-1	G-1			2.24 ± 0.73	2.11 ± 0.72	-1.25 ± 1.03	-0.31 ± 0.87	0.68 ± 0.9	1.58 ± 0.70
				$\Delta\text{SSIM}_{\text{gt}}$	$\Delta\text{SSIM}_{\text{CNN-blind}}$	$\Delta\text{SSIM}_{[66]}$	$\Delta\text{SSIM}_{[69]}$	$\Delta\text{SSIM}_{[67]}$	$\Delta\text{SSIM}_{[68]}$
Z-1	Z-1			0.10 ± 0.03	0.10 ± 0.04	0.05 ± 0.04	0.02 ± 0.02	0.09 ± 0.04	0.09 ± 0.04
Z-1	G-1			0.10 ± 0.03	0.09 ± 0.03	0.03 ± 0.05	0.02 ± 0.01	0.08 ± 0.04	0.08 ± 0.03
G-1	Z-1			0.08 ± 0.03	0.08 ± 0.06	0.01 ± 0.05	0.01 ± 0.03	0.07 ± 0.04	0.08 ± 0.03
G-1	G-1			0.08 ± 0.03	0.08 ± 0.04	0.01 ± 0.06	0.03 ± 0.03	0.07 ± 0.04	0.08 ± 0.04

TABLE VI

ANALYSIS OF THE PRECISION OF THE DEPTH RECOVERY OF A PLANE USING CONTROLLED ASTIGMATIC ABERRATIONS BY USE OF CYLINDRICAL LENSES. THE TEST IMAGES ARE ACQUISITIONS OF $N = 30$ PRINTED GRIDS TILTED 3° , 6° AND 10° . THE NETWORK IS A RESNET-34 TRAINED ON DATA SET [MICR].

Tilt angle	R^2	Absolute error	Relative error
10°	0.967	$3.50 \pm 2.62 \mu\text{m}$	$2.18 \pm 1.64\%$
6°	0.988	$1.31 \pm 1.15 \mu\text{m}$	$1.38 \pm 1.21\%$
3°	0.989	$0.61 \pm 0.41 \mu\text{m}$	$1.26 \pm 0.85\%$

applications in a microscope could be feasible.

While transfer learning (i.e. networks trained with ImageNet for image detection prior to training) does not help to improve the final accuracy of the regression task because the network was already able to learn from the original data set in a reasonable amount of time, training networks by starting from pre-trained models tends to speed up convergence during learning.

B. Robustness analysis against input degradation

We noticed that all our models are invariant to changes in illumination, certainly due to inherent normalization steps in the CNN architecture, and are overall robust to small to medium amounts of noise. However, when the signal-to-noise ratio strongly decreases, the correlation coefficient tends to decrease as well.

Additionally, we observed that, in comparison to ResNet, CNNs based on ResNeXt perform generally worse when noise is applied. This result contradicts observations reported in a recent benchmark [45], where ResNext is more robust than ResNet to Gaussian noise. However, this publication scores the network's accuracy for a classification task into image type categories, which is a different application than our regression of aberrations and might explain the discrepancy.

We found the synthetic degradations we added in the training set (Eq. (10)) to be a necessary step to achieve robustness to noise. Indeed, the performance dropped when this step was omitted.

C. Application 1: spatially-variant blind deconvolution

Using the spatially-variant PSF map inferred from the PSF output, we have been able to reconstruct details in a degraded image without any prior information on the image content or the optical system. Given that our method does not require adjusting parameters or experimentally measuring a PSF (which is labor intensive), it leads to results faster than non-blind deconvolution methods.

We noticed that, because we use a constrained PSF model, our deconvolution method does not suffer from drawbacks sometimes associated to other deconvolution and super-resolution methods. For example, techniques based on MAP optimization sometimes converge to exotic forms of PSFs that are not consistent with the physics of optics, causing image deformation or loss of features [70]. We could illustrate this by the example in Fig.8 (b) for the method from Kotera et al. [66], which diverges when directed filaments are shown to the algorithm and creates artifacts.

Similarly, the use of denoising CNNs for image enhancement can lead to phantom details that could falsify underlying biological features, or discard high-frequency features that are mistaken for noise [71]. When using constrained PSF models such as the ones we use, deconvolution algorithms such as RL will still produce a reasonable image even if the predicted PSF is not exactly matching the PSF corresponding to the blur. This is likely due to the inherent constraint of a model with a small number of parameters that enforces the shape of the PSF. The outcome is a higher average SNR and SSIM of the reconstruction using our method compared to other

BD algorithms. Another advantage of using constrained PSF models is that they can model PSFs with very large supports (sizes). Classical BD only allows for a smaller support, as using more pixels creates higher complexity. Nevertheless, our models are currently unlikely suitable for some types of degradation, where BD methods were successfully applied, such as for compensating for motion blur with rotation [72]. Deconvolution results are equivalently efficient both in terms of SNR and SSIM when the degradation and detection models are not similar (e.g. degradation using a Gaussian PSF and estimation of the PSF using a Zernike model). This particular point is relevant since it confirms the robustness of the image enhancement process when there is a mismatch between the degradation PSF that we want to model (i.e. the optical system PSF) and the model itself.

Finally, we observed that methods [69], [67] and [68], due to their multiscale optimization approach, were considerably slower than the one we propose, taking up to 4 minutes to deblur a 256×256 image, whereas our method takes less than 3 seconds using the same machine to both estimate the PSF and perform TV-RL. The difference in run times can be explained by our GPU implementation, but as well because of the inherent nature of traditional optimization algorithms that alternate kernel and image estimation, which limits the parallelizability of the calculations.

D. Application 2: depth from focus using astigmatism

We showed that the focus parameter of the PSF models is a function of the distance between the sample and the focal plane, and that the sign of the axial distance could be recovered from higher Zernike coefficients when using engineered PSFs. Furthermore, we found that the depth function at any point in the image could be obtained from an affine function that combines two Zernike coefficients.

We recovered the relative depth of both sides of the printed grid in real microscopy acquisitions, which extends the idea of PSF engineering introduced in [58] for point-like structures, to work for fully textured images. Using a textured plane image, we have been able to recover the depth to up to $160 \mu\text{m}$. However, this accuracy decreases when the imaged objects lack texture. Using other shapes of engineered PSFs (e.g. quadripoles [59]) could potentially lead to improved depth accuracy.

Since use of an engineered PSF degrades the image, simultaneous depth retrieval and high-resolution might best be carried out by splitting the acquisition line (e.g. with a beam splitter) to record the image on one side and a PSF engineered image on the other.

Depth retrieval is also possible using the parameters of a Gaussian-2 PSF model, but the precision is improved using the PSF model based on Zernike polynomials.

V. CONCLUSION

In this work, we have shown that CNNs, in particular residual networks, can be used to extract local blur characteristics from microscopy images in the form of parameters of a PSF model with only minimal knowledge about the optical setup. Our system is robust to signal perturbations and does not need to

be trained specifically on images of the target imaging system. This flexibility allows the user to perform, without taking measurements beyond the images of interest, a wide range of tasks in microscopy image processing, including deblurring or obtaining a depth map.

Our method opens up the possibility to improve the resolution of non-flat objects with minimal *a priori* knowledge of the optical setup and obtain its tridimensional shape in a single shot.

APPENDIX

Software sources, documentation, trained models, and the acquired test dataset (Fig. 8) will be made available upon publication. We thank Arne Seitz and José Artacho from EPFL BIOP for providing the fixed sample slides, reused from experiments that were approved by the EPFL Ethics Committee.

REFERENCES

- [1] S. F. Gibson and F. Lanni, "Diffraction by a circular aperture as a model for three-dimensional optical microscopy," vol. 6, no. 9, pp. 1357–1367, 1989.
- [2] D. Sage *et al.*, "DeconvolutionLab2: An open-source software for deconvolution microscopy," *Methods*, vol. 115, pp. 28–41, 2017.
- [3] P. Grossmann, "Depth from focus," *Pattern Recognit. Lett.*, vol. 5, no. 1, pp. 63–69, 1987.
- [4] 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, 2008.
- [5] A. Shihavuddin *et al.*, "Smooth 2D manifold extraction from 3D image stack," *Nat Commun*, vol. 8, p. 15554, 2017.
- [6] E. Cuhe, P. Marquet, and C. Depeursinge, "Simultaneous amplitude-contrast and quantitative phase-contrast microscopy by numerical reconstruction of Fresnel off-axis holograms," *Appl. Opt.*, vol. 38, no. 34, p. 6994, 1999.
- [7] A. Griffa, N. Garin, and D. Sage, "Comparison of deconvolution software in 3D microscopy: A user point of view—part 1," *G.I.T. Imaging & Microscopy*, vol. 12, no. 1, pp. 43–45, 2010.
- [8] J.-A. Conchello and J. W. Lichtman, "Optical sectioning microscopy," *Nat Methods*, vol. 2, no. 12, pp. 920–931, 2005.
- [9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep Convolutional Neural Networks," in *NIPS*, Curran Associates, Inc., 2012, pp. 1097–1105.
- [10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017.
- [11] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE CVPR*, 2014, pp. 580–587.
- [12] S. J. Yang *et al.*, "Assessing microscope image focus quality with deep learning," *BMC Bioinformatics*, vol. 19, no. 1, 2018.

- [13] X. Zhu, S. Cohen, S. Schiller, and P. Milanfar, "Estimating Spatially Varying Defocus Blur From A Single Image," *IEEE Trans. on Image Process.*, vol. 22, no. 12, pp. 4879–4891, 2013.
- [14] 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.
- [15] D. Gong *et al.*, "From Motion Blur to Motion Flow: A Deep Learning Solution for Removing Heterogeneous Motion Blur," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI: IEEE, 2017, pp. 3806–3815.
- [16] S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *IEEE CVPR*, 2017.
- [17] A. Shajkofci and M. Liebling, "Semi-blind spatially-variant deconvolution in optical microscopy with local point spread function estimation by use of convolutional neural networks," in *IEEE ICIP*, 2018, pp. 3818–3822.
- [18] D. He, D. Cai, J. Zhou, J. Luo, and S.-L. Chen, "Restoration of Out-of-Focus Fluorescence Microscopy Images Using Learning-Based Depth-Variant Deconvolution," *IEEE Photonics Journal*, pp. 1–1, 2020, Conference Name: IEEE Photonics Journal.
- [19] M. Weigert *et al.*, "Content-aware image restoration: Pushing the limits of fluorescence microscopy," *Nat Methods*, vol. 15, no. 12, pp. 1090–1097, 2018.
- [20] Y. Rivenson, Y. Zhang, H. Günaydin, D. Teng, and A. Ozcan, "Phase recovery and holographic image reconstruction using deep learning in neural networks," *Light Sci Appl*, vol. 7, no. 2, p. 17 141, 2018.
- [21] Y. Wu *et al.*, "Three-dimensional virtual refocusing of fluorescence microscopy images using deep learning," *Nat Methods*, 2019.
- [22] H. Wang *et al.*, "Deep learning enables cross-modality super-resolution in fluorescence microscopy," *Nat Methods*, vol. 16, no. 1, pp. 103–110, 2019.
- [23] J. W. Goodman, "Chapter 6-1," in *Introduction To Fourier Optics*, 2nd Revised edition, Englewood, Colo: W.H.Freeman & Co Ltd, 2005, p. 129.
- [24] B. Chalmond, "PSF estimation for image deblurring," *CVGIP*, vol. 53, no. 4, pp. 364–372, 1991.
- [25] N. Joshi, R. Szeliski, and D. J. Kriegman, "PSF estimation using sharp edge prediction," in *IEEE CVPR*, 2008, pp. 1–8.
- [26] A. Reuter, H.-P. Seidel, and I. Ihrke, "BlurTags: Spatially varying PSF estimation with out-of-focus patterns," in *WSCG*, 2012, p. 9.
- [27] J. Brauers, C. Seiler, and T. Aach, "Direct PSF estimation using a random noise target," in *SPIE Electronic Imaging*, 2010, 75370B.
- [28] 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, 2011.
- [29] 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, 1995.
- [30] F. Soulez, "A "Learn 2D, Apply 3D" method for 3D deconvolution microscopy," in *IEEE ISBI*, 2014, pp. 1075–1078.
- [31] J. Herbel, T. Kacprzak, A. Amara, A. Refregier, and A. Lucchi, "Fast point spread function modeling with deep learning," *J. Cosmol. Astropart. Phys.*, vol. 2018, no. 07, pp. 054–054, 2018.
- [32] F. Aguet, D. Van De Ville, and M. Unser, "An accurate PSF model with few parameters for axially shift-variant deconvolution," in *IEEE ISBI*, 2008, pp. 157–160.
- [33] G. B. Airy, "On the diffraction of an object-glass with circular aperture," *Trans. Cambridge Philos. Soc.*, vol. 5, p. 283, 1835.
- [34] D. R. Iskander, M. J. Collins, and B. Davis, "Optimal modeling of corneal surfaces with Zernike polynomials," *IEEE Trans Biomed Eng*, vol. 48, no. 1, pp. 87–95, 2001.
- [35] F. von Zernike, "Beugungstheorie des schneidenverfahrens und seiner verbesserten form, der phasenkontrastmethode," *Physica*, vol. 1, no. 7, pp. 689–704, 1934.
- [36] J. W. Goodman, *Introduction to Fourier Optics*. Roberts and Company Publishers, 2005, 520 pp.
- [37] B. Zhang, J. Zerubia, and J.-C. Olivo-Marin, "Gaussian approximations of fluorescence microscope point-spread function models," *Appl. Opt.*, vol. 46, no. 10, pp. 1819–1829, 2007.
- [38] H. P. Kao and A. S. Verkman, "Tracking of single fluorescent particles in three dimensions: Use of cylindrical optics to encode particle position," *Biophys. J.*, vol. 67, no. 3, pp. 1291–1300, 1994.
- [39] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient backprop," in *Neural Networks: Tricks of the Trade*, Springer, 2012, pp. 9–48.
- [40] 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, 2011.
- [41] E. N. Gilbert and H. O. Pollak, "Amplitude distribution of shot noise," *Bell System Technical Journal*, vol. 39, no. 2, pp. 333–350, 1960.
- [42] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *IEEE CVPR*, 2009, p. 8.
- [43] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int J Comput Vis*, vol. 115, no. 3, pp. 211–252, 2015.
- [44] J. Donahue *et al.*, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition," in *ICML*, 2014, pp. 647–655.
- [45] D. Hendrycks and T. G. Dietterich, "Benchmarking neural network robustness to common corruptions and surface variations," in *ICLR*, 2019.
- [46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE CVPR*, 2016, pp. 770–778.
- [47] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated Residual Transformations for Deep Neural Networks," in *IEEE CVPR*, 2017, pp. 5987–5995.
- [48] G. Huang, Z. Liu, L. v d Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *IEEE CVPR*, 2017, pp. 2261–2269.
- [49] K. Hechenbichler and K. Schliep, "Weighted k-nearest-neighbor techniques and ordinal classifi-

- cation,” Ludwig-Maximilians-Universität München, Working Paper 399, 2004.
- [50] M. Hirsch, S. Sra, B. Scholkopf, and S. Harmeling, “Efficient filter flow for space-variant multiframe blind deconvolution,” in *IEEE CVPR*, 2010, pp. 607–614.
- [51] N. Dey *et al.*, “Richardson–Lucy algorithm with total variation regularization for 3D confocal microscope deconvolution,” *Microsc. Res. Tech.*, vol. 69, no. 4, pp. 260–266, 2006.
- [52] M. Weigert, L. Royer, F. Jug, and G. Myers, “Isotropic Reconstruction of 3D Fluorescence Microscopy Images Using Convolutional Neural Networks,” in *MICCAI*, vol. 10434, 2017, pp. 126–134.
- [53] R. Morin, S. Bidon, A. Basarab, and D. Kouamé, “Semi-blind deconvolution for resolution enhancement in ultrasound imaging,” in *IEEE ICIP*, 2013, pp. 1413–1417.
- [54] W. H. Richardson, “Bayesian-based iterative method of image restoration,” *J. Opt. Soc. Am.*, vol. 62, no. 1, pp. 55–59, 1972.
- [55] T. Chan and Chiu-Kwong Wong, “Total variation blind deconvolution,” *IEEE Trans. on Image Process.*, vol. 7, no. 3, pp. 370–375, 1998.
- [56] 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, 2015.
- [57] 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, pp. 899–904.
- [58] B. Huang, W. Wang, M. Bates, and X. Zhuang, “Three-dimensional super-resolution imaging by stochastic optical reconstruction microscopy,” *Science*, vol. 319, no. 5864, pp. 810–813, 2008.
- [59] A. Aristov, B. Lelandais, E. Rensen, and C. Zimmer, “ZOLA-3D allows flexible 3D localization microscopy over an adjustable axial range,” *Nat Commun*, vol. 9, no. 1, p. 2409, 2018.
- [60] M.-A. Bray, A. N. Fraser, T. P. Hasaka, and A. E. Carpenter, “Workflow and metrics for image quality control in large-scale high-content screens,” *J Biomol Screen*, vol. 17, no. 2, pp. 266–274, 2012.
- [61] V. Ljosa, K. L. Sokolnicki, and A. E. Carpenter, “Annotated high-throughput microscopy image sets for validation,” *Nature Methods*, vol. 9, no. 7, p. 637, 2012.
- [62] E. Williams *et al.*, “The image data resource: A bioimage data integration and publication platform,” *Nat Methods*, vol. 14, no. 8, pp. 775–781, 2017.
- [63] 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, 2018.
- [64] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” in *ICLR*, 2015.
- [65] Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multi-scale structural similarity for image quality assessment,” in *IEEE ACSSC*, vol. 2, 2004, pp. 1398–1402.
- [66] J. Kotera, F. Šroubek, and P. Milanfar, “Blind deconvolution using alternating maximum a posteriori estimation with heavy-tailed priors,” in *CAIP*, Springer, 2013, pp. 59–66.
- [67] J. Dong, J. Pan, Z. Su, and M.-H. Yang, “Blind Image Deblurring with Outlier Handling,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2497–2505.
- [68] M. Jin, S. Roth, and P. Favaro, “Normalized Blind Deconvolution,” presented at the ECCV 2018, vol. 11211, Cham: Springer International Publishing, 2018, pp. 694–711.
- [69] O. Whyte, J. Sivic, and A. Zisserman, “Deblurring shaken and partially saturated images,” *Int. J. Comput. Vis*, vol. 110, no. 2, pp. 185–201, 2014.
- [70] D. Perrone and P. Favaro, “Total Variation Blind Deconvolution: The Devil Is in the Details,” in *IEEE CVPR*, 2014, pp. 2909–2916.
- [71] D. Liu, B. Wen, X. Liu, Z. Wang, and T. Huang, “When Image Denoising Meets High-Level Vision Tasks: A Deep Learning Approach,” in *IJCAI*, 2018, pp. 842–848.
- [72] Q. Shan, J. Jia, and A. Agarwala, “High-quality motion deblurring from a single image,” *ACM Trans. Graph.*, vol. 27, no. 3, p. 1, 2008.