Biological Imaging (2024), 4: e13
doi:10.1017/S2633903X24000096

RESEARCH ARTICLE

Deep-blur: Blind identification and deblurring with convolutional neural networks

Valentin Debarnot[1] and Pierre Weiss[2,3]

[1]Departement Mathematics and computer science, Basel University, Basel, Switzerland
[2]Institut de Recherche en Informatique de Toulouse (IRIT), CNRS & Université de Toulouse, Toulouse, France
[3]Centre de Biologie Intégrative (CBI), Laboratoire de biologie Moléculaire, Cellulaire et du Développement (MCD), CNRS & Université de Toulouse, Toulouse, France
Corresponding author: Pierre Weiss; Email: pierre.weiss@cnrs.fr

Received: 12 September 2023; **Revised:** 21 May 2024; **Accepted:** 15 June 2024

Keywords: blind deblurring; deep learning; identification network; spatially variant blur; unrolled network

Abstract

We propose a neural network architecture and a training procedure to estimate blurring operators and deblur images from a single degraded image. Our key assumption is that the forward operators can be parameterized by a low-dimensional vector. The models we consider include a description of the point spread function with Zernike polynomials in the pupil plane or product-convolution expansions, which incorporate space-varying operators. Numerical experiments show that the proposed method can accurately and robustly recover the blur parameters even for large noise levels. For a convolution model, the average signal-to-noise ratio of the recovered point spread function ranges from 13 dB in the noiseless regime to 8 dB in the high-noise regime. In comparison, the tested alternatives yield negative values. This operator estimate can then be used as an input for an unrolled neural network to deblur the image. Quantitative experiments on synthetic data demonstrate that this method outperforms other commonly used methods both perceptually and in terms of SSIM. The algorithm can process a 512 × 512 image under a second on a consumer graphics card and does not require any human interaction once the operator parameterization has been set up.[1]

Impact Statement

The prospect of restoring blurred images with a wave of the digital wand is undeniably seductive in microscopy. However, the reality currently appears less satisfying, as handcrafted algorithms often offer only minimal gains at the price of long parameter tuning.

In this article, we combine physical models of the blur and artificial intelligence to design an interpretable blind deblurring method. A first neural network is trained to estimate the point spread function of the optical system, while a second network leverages this estimate to improve image quality. This approach provides a fully automated tool, capable of improving the image quality in seconds. The proposed methodology yields point spread function estimates with a quality that is superior by 10 dB to other popular methods, which also leads to better and more reliable deblurring results.

1. Introduction

Image deblurring and superresolution consist of recovering a sharp image $\bar{\mathbf{x}}$ from its blurred and subsampled version $\mathbf{y} = \mathcal{P}(\bar{\mathbf{A}}\bar{\mathbf{x}})$, where $\bar{\mathbf{A}} \in \mathbb{R}^{M \times N}$ is a discretized linear integral operator describing

[1] A preliminary version of this work was published in IEEE ISBI 2021.[30]

the acquisition process, and $\mathcal{P}:\mathbb{R}^M \to \mathbb{R}^M$ is some perturbation modeling noise, quantization, and saturation. It plays an important role in biomedical and astronomical imaging, where physical phenomena such as diffraction and turbulence strongly reduce the achievable resolution. It also received a constant attention in the field of computer vision, where moving or out-of-focus objects create artifacts. When the operator $\bar{\mathbf{A}}$ describing the optical system is available, this problem can be solved with mature variational inverse problem solvers[16] or data-driven approaches.[8]

However, deriving a precise forward model requires specific calibration procedures, well-controlled imaging environments. and/or highly qualified staff. In addition, model mismatches result in distorted reconstructions. This can lead to dramatic performance loss, especially for superresolution applications.[41,84]

An alternative to a careful calibration step consists of solving the problem blindly: the forward model $\bar{\mathbf{A}}$ is estimated together with the sharp image $\bar{\mathbf{x}}$. Unfortunately, this blind inverse problem is highly degenerate. There is no hope to recover the sharp image without prior assumptions on $\bar{\mathbf{x}}$ and $\bar{\mathbf{A}}$. For instance, assume that $\bar{\mathbf{A}}$ is a discrete convolution operator with some kernel $\bar{\mathbf{h}}$, that is, $\mathbf{y} = \bar{\mathbf{h}} \star \bar{\mathbf{x}}$. Then, the couple $(\bar{\mathbf{h}}, \bar{\mathbf{x}})$ can be recovered only up to a large group of transformations.[79] For instance, the identity and blurred image are a trivial solution, and the image and kernels can be shifted in opposite directions or scaled with inverse factors. Therefore, it is critical to introduce regularization terms both for the operator $\bar{\mathbf{A}}$ and the signal $\bar{\mathbf{x}}$.

The main objective of this work is to design a blind inverse problem solver under the two assumptions below:

- The operator $\bar{\mathbf{A}}$ can be parameterized by a low-dimensional vector. In what follows, we let $\mathbf{A}: \mathbb{R}^K \to \mathbb{R}^{M\times N}$ denote the operator mapping and we assume that $\bar{\mathbf{A}} = \mathbf{A}(\bar{\boldsymbol{\gamma}})$ for some $\bar{\boldsymbol{\gamma}} \in \mathbb{R}^K$.
- The signal $\bar{\mathbf{x}}$ lives in a family $\mathcal{X} \subseteq \mathbb{R}^N$ with some known distribution $\mathcal{L}_{\mathcal{X}}$.

We propose a specific convolutional neural architecture and a training procedure to recover the couple $(\bar{\boldsymbol{\gamma}}, \bar{\mathbf{x}})$ from the degraded data $\mathbf{y}$ and the mapping $\mathbf{A}(\cdot)$. A first network identifies the parameterization $\bar{\boldsymbol{\gamma}}$, while the second uses this parameterization to estimate the image $\bar{\mathbf{y}}$. This results in an efficient algorithm to sequentially estimate the blur operator and the sharp image $\bar{\mathbf{x}}$. The network architecture is shown on Figure 1. At a formal level, the work can be adapted to arbitrary inverse problems beyond image deblurring. We, however, showcase its efficiency only for challenging deblurring tasks involving convolutions but also more advanced space-varying operators.

1.1. Related works

Solving blind deblurring problems is a challenging task that started being studied in the 1970s.[80] Fifty years later, it seems impossible to perform an exhaustive review of existing methods and the following description will be lacunary. We refer the interested reader to[18] for a general overview of this field and to[74] for a survey more focused on microscopy. The prevailing approach is to estimate the original signal and the blur operator by solving variational problems of the form:

$$\inf_{\mathbf{A}\in\mathbb{R}^{M\times N}, \mathbf{x}\in\mathbb{R}^N} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + R_A(\mathbf{A}) + R_x(\mathbf{x}), \tag{1}$$

where $R_A: \mathbb{R}^{M\times N} \to \mathbb{R}\cup\{+\infty\}$ and $R_x: \mathbb{R}^N \to \mathbb{R}\cup\{+\infty\}$ are regularization terms for the operator and the signal respectively. This problem arises when considering maximum a posteriori (MAP) estimators.[50] It can be attacked with various types of alternating minimization procedures.[11] Before the advent of data-driven approaches, the regularizers were carefully designed to target specific features. The point spread functions can be considered as sparse and compactly supported for motion deblurring.[15,25,27,34,49,69,70,82] They are smooth for diffraction-limited systems[17] and can also be parameterized with Zernike polynomials in the pupil plane.[7,40,47,74,78,79] The images can sometimes

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

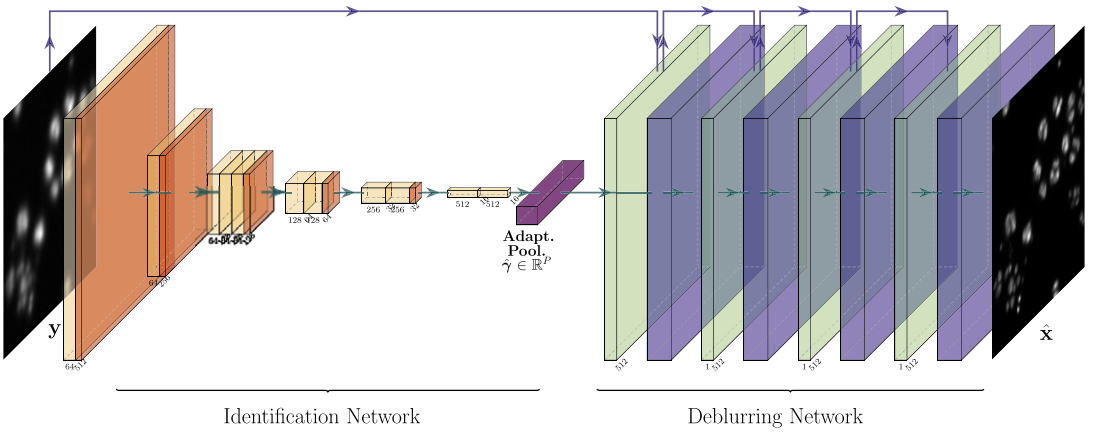


Figure 1. *The deep-blur architecture. The first part of the network identifies the parameter* $\hat{\boldsymbol{\gamma}}$*. In this article, we use a ResNet architecture. The estimated parameter* $\hat{\boldsymbol{\gamma}}$ *is given as an input of a second deblurring network. This one is an unrolled Douglas–Rachford algorithm. The yellow blocks are convolution layers with ReLU and batch normalization. The red ones are average pooling layers. The green ones are regularized inverse layers of the form* $\mathbf{x}_{t+1} = (\mathbf{A}^*(\hat{\boldsymbol{\gamma}})\mathbf{A}(\hat{\boldsymbol{\gamma}}) + \lambda\mathbf{I})^{-1}\mathbf{A}(\hat{\boldsymbol{\gamma}})\mathbf{y}$*. The violet blocks are U-Net-like neural networks with weights learned to provide a sharp image* $\hat{\mathbf{x}}$*.*

be considered as sparse in microscopy and astronomical imaging[31,60] or piecewise constant for natural images. The typical regularizer R_x is then the total variation, or more advanced priors on the image gradient.[9,15,17,68–70] Some authors also advocate for the use of priors on the image spectrum,[38,86] which transform the blind deconvolution problem into a phase retrieval problem under ideal conditions.

The most recent variants of these approaches can provide excellent results (see e.g.,[66,88]). However, they strongly rely on the detection of specific features (points, edges, textures) which may be absent or inaccurate models of the typical image features. In addition, problem (1) or its derivatives is usually highly nonconvex, and the initialization must be chosen carefully to ensure local convergence to the right minimizer. As a result, these methods require a substantial know-how to be successfully applied to a specific field.

In the most recent years, machine learning approaches have emerged and now seem to outperform carefully handcrafted ones, at least under well-controlled conditions. These approaches can be divided into two categories. The first category concerns methods that directly estimate the reconstructed image from the observation.[3,20,55–57,61,64,75] The second category contains approaches that produce an estimation of the blur operator. This estimate can then be used to deblur the original image. These approaches are specifically tuned for applications in computer vision[14,39,51,75,81] (motion and out-of-focus blurs) or diffraction-limited systems.[26,58,73,76,77,85] Our work rather falls in the second category.

In this list of references, a few authors propose ideas closely related to the ones developed hereafter. In particular,[26,73,85] propose to estimate the pupil function of a microscope from images of point sources using neural networks. This idea is similar to the identification network in Figure 1. The two underlying assumptions are a space invariant system and the observation of a single point source. The idea closest to ours is from Shajkofci and Liebling.[76,77] Therein, the authors estimate a decomposition of the point spread function from a single image using a low-dimensional parameterization such as a decomposition over Zernike polynomials. The spatial variations are then estimated by splitting the observation domain in patches where the blur is assumed locally invariant. The image can then be deblurred using a Richardson–Lucy algorithm based on the estimated operator.

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

1.2. Contributions

In this work, we propose to use a pair of convolutional neural networks to first estimate the operator parameterization $\bar{\boldsymbol{\gamma}} \in \mathbb{R}^K$ and then use this parameterization to estimate the sharp image $\bar{\mathbf{x}} \in \mathbb{R}^N$ with a second convolutional neural network. The first network is the popular ResNet[45] as in [77]. The second network has the structure of an unrolled algorithm, which offers the advantage of adapting to the forward operator.[1,2,59] We call the resulting algorithm deep-blur, see Figure 1. This work contains various original features:

- It includes space-varying blur operators that are accurately and efficiently encoded using product-convolution expansions as illustrated in [32,33]. In particular, we show that this approach is compatible with the characterization of an optical system as a low-dimensional subspace of operators proposed in [28,29]. Most approaches in the literature decompose the observation space into patches and treat each patch independently. In this work, we consider operators with an impulse response that varies continuously in the field of view.
- The resulting deblurring network is able to adapt to different forward models and to handle model mismatches naturally. This issue is an important concern for the use of model-based inverse problem solvers.[6,36,65] As will be discussed later, our approach can be seen as an intermediate step between the plug-and-play algorithms[83,87] and the unrolled algorithms.[1]
- We evaluate the efficiency, robustness, and stability of the proposed approach on various challenging problems, showing that the method is reliable and accurate.

The *PyTorch* implementation of our method is available on demand. We are currently integrating it into the DeepInv package.

2. Methods

In this article, we assume that the degraded signal $\mathbf{y} \in \mathbb{R}^M$ is generated according to the following equation:

$$\mathbf{y} = \mathcal{P}(\mathbf{A}(\bar{\gamma})\bar{\mathbf{x}}), \tag{2}$$

where $\mathbf{A}(\boldsymbol{\gamma}) : \mathbb{R}^N \to \mathbb{R}^M$ is a linear operator describing the optical system. It depends on an unknown parameter $\bar{\boldsymbol{\gamma}} \in \mathbb{R}^P$. The mapping $\mathcal{P} : \mathbb{R}^N \to \mathbb{R}^N$ can model various deterministic or stochastic perturbations occurring in real systems such as additive white Gaussian noise, Poisson noise, and quantization. In this article, we will use a Poisson-Gaussian noise approximation detailed in [35]. It is known to accurately model microscopes, except in the very low photon count regime. Other more complex models could be easily incorporated into the proposed framework at the learning stage. A critical aspect of this article is the parameterization of the forward operator $\mathbf{A}$. We discuss this aspect below.

2.1. Modeling the blur operators

We consider both space-invariant and space-varying blur operators and linear or nonlinear parameterization.

2.1.1. Linear parameterization

We may assume that $\mathbf{A}$ belongs to a subspace of operators.

2.1.2. Convolution models and eigen-PSF bases

By far, the most widespread blurring model in imaging is based on convolution operators: the point spread function is identical whatever the position in space. This model is accurate for small fields of view, which are widespread in applications. Assuming that there is no subsampling in the model, we can set $M = N$ and $\mathbf{Ax} = \mathbf{h} \star \mathbf{x}$ for some unknown convolution kernel $\mathbf{h}$.

The convolution model strongly simplifies the blur identification problem since we are now looking for a vector of size N instead of a huge $N \times N$ matrix. Yet, the blind deconvolution problem is known to

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

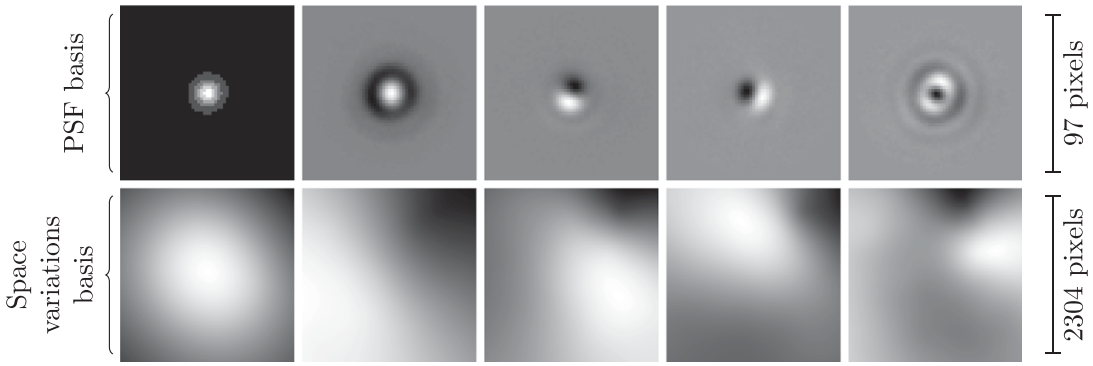


Figure 2. *Examples of eigen-PSF and eigen-space variation bases for a wide-field microscope.*[28]

suffer from many degeneracies and possesses a huge number of possible solutions, see for example [79]. To further restrict the space of admissible operators and therefore improve the identifiability, we can expand the kernel **h** in an *eigen-PSF basis.* This leads to the following low-dimensional model.

Model 2.1 (Convolution and eigen-PSFs). *We assume that*

$$\mathbf{A}(\boldsymbol{\gamma})\mathbf{x} = \sum_{k=1}^{K} \boldsymbol{\gamma}[k]\mathbf{e}_k \star \mathbf{x}, \forall \mathbf{x} \in \mathbb{R}^N,$$

where $(\mathbf{e}_k)$ is an orthogonal family of convolution kernels called eigen-PSF basis.

Defining an eigen-PSF basis can be achieved by computing a principal component analysis of a family of observed or theoretical point spread functions.[37] An example of an experimental eigen-PSF basis obtained in [28] is shown on Figure 2, top.

2.1.3. Space-variant models and product-convolution expansions

The convolution model 2.1 can only capture *space-invariant* impulse responses. When dealing with large field of views, this model becomes inaccurate. One way to overcome this limitation is to use product-convolution expansions,[28,32,33] which efficiently encode space-varying systems.

Model 2.2 (Product-convolution expansions). *Let* $(\mathbf{e}_i)_{1 \le i \le I}$ and $(\mathbf{f}_j)_{1 \le j \le J}$ *define two orthogonal families of* $\mathbb{R}^N$. The action of a product-convolution **A** *operator reads:*

$$\mathbf{A}\mathbf{x} = \sum_{i,j} \mathbf{e}_i * (\mathbf{f}_j \odot \mathbf{x}), \forall \mathbf{x} \in \mathbb{R}^N, \tag{3}$$

where $\odot$ indicates the coordinate-wise (Hadamard) product.

In the above model, the basis $(\mathbf{e}_i)$ can still be interpreted as an eigen-PSF basis. Indeed, we …have for all locations $z \in \{1, \ldots, N\}$:

$$\mathbf{A}(\boldsymbol{\gamma})\boldsymbol{\delta}_z = \sum_{i=1}^{I} \left(\sum_{j=1}^{J} \boldsymbol{\gamma}[i,j]\mathbf{f}_j[z] \right) \mathbf{e}_i[\cdot - z].$$

Hence, we see that each impulse response is expressed in the basis $(\mathbf{e}_i)$. The basis $(\mathbf{f}_j)$, on its side, can be interpreted as an eigen-space variation basis: it describes how the point spread functions can vary in space. It can be estimated by interpolation of the coefficient of a few scattered PSF in the eigen-PSF basis $(\mathbf{e}_i)$. In optical devices such as microscopes, the estimation of the families $(\mathbf{e}_i)$ and $(\mathbf{f}_j)$ can be accomplished by observing several images of microbeads.[10,28] An example of the experimental product-convolution family is shown in Figure 2 for a wide-field microscope. In that case, the dimension K of the

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

subspace is $K = I \cdot J = 16$. Airy pattern oscillations are found in the first eigen-PSFs and intensity variations, such as nonhomogeneous illuminations/vignetting, in the spatial variation maps.

2.1.4. Nonlinear parameterization and Zernike polynomials

An alternative to the linear models is given by the theory of diffraction. A popular and effective model in microscopy and astronomy consists of using the Fresnel/Fraunhofer theory. We can approximate the pupil function with a finite number of Zernike polynomials.[40,44] This model leads to some of the state-of-the-art algorithms for blind deconvolution and superresolution in microscopy and astronomy.[7,62,72,78]

Model 2.3 (Fresnel approximation and a Zernike basis). *We assume that the forward model is a convolution with a slice of a continuous 3D kernel $h(x,y,z)$. The 3D kernel can be expressed through the 2D pupil function ϕ as*

$$h(x,y,z) = \left| \int_{B(0,f_c)} \phi(w_1,w_2) \exp(2i\pi z d(w_1,w_2)) \exp(2i\pi(w_1 x + w_2 y)) dw_1 dw_2 \right|^2,$$

where $f_c = n/\lambda$ is the cutoff frequency, n is the refractive index of the immersion medium, and λ is the wavelength of the observation light and

$$d(w_1,w_2) = \sqrt{f_c^2 - (w_1+w_2)^2}.$$

The complex pupil function ϕ can be expanded with Zernike polynomials Z_k:

$$\phi = \exp\left(2i \sum_{k=4}^{K+4} \gamma[k] Z_k \right),$$

where the coefficients $\gamma[k] \in \mathbb{R}$ are real number.[2]

A few examples of slices of point spread functions generated with Model 2.3 are displayed in Figure 3. Notice that we do not use the first three Zernike polynomials (piston, tip, and tilt) as they do not influence the shape of the PSF. In our experiments, we used $K = 7$ Zernike polynomials. In the Noll nomenclature, they are referred to as Z_4: defocus, Z_5-Z_6: primary astigmatism, Z_7-Z_8: primary coma, and Z_9-Z_{10}: trefoil. We set the coefficients γ as uniform random variables with an amplitude smaller than 0.15. As can be seen, a rich variety of impulse responses can be generated with this low-dimensional model.

2.2. The deep-blur architecture

We propose to train two different neural networks IN and DN sequentially:

- IN is an *identification network*. It depends on weights $\boldsymbol{\theta}$. The mapping $\mathrm{IN}(\boldsymbol{\theta}) : \mathbb{R}^M \to \mathbb{R}^K$ takes as an input a degraded image $\mathbf{y} \in \mathbb{R}^M$ and provides an estimate $\hat{\boldsymbol{\gamma}}$ of $\bar{\boldsymbol{\gamma}}$ in $\mathbb{R}^K$.
- DN is a *deblurring network*. It depends on weights $\boldsymbol{\xi}$. The mapping $\mathrm{DN}(\boldsymbol{\xi}) : \left(\mathbb{R}^M, \mathbb{R}^K\right) \to \mathbb{R}^N$ takes as input parameters the blurry image $\mathbf{y}$ and the operator coefficient $\boldsymbol{\gamma}$. It outputs an estimate $\hat{\mathbf{x}}$ of the sharp image $\bar{\mathbf{x}}$.

2.2.1. The identification network

Traditional estimation of a blur kernel relies on the detection of cues in the image such as points (direct observation[10,26,31,73]), edges in different orientations (Radon transform of the kernel[49]), or textures (power spectrum[38]) followed by adapted inversion procedures. This whole process can be modeled by a set of linear operations (filtering) and nonlinear operations (e.g., thresholding). A convolutional neural network, composed of similar operations, should therefore be expressive enough to estimate the blur

[2] In this model, only the phase of the pupil function varies. In all generality, the amplitude could vary as well at a slower rate. Most models in the literature assume a constant amplitude.

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

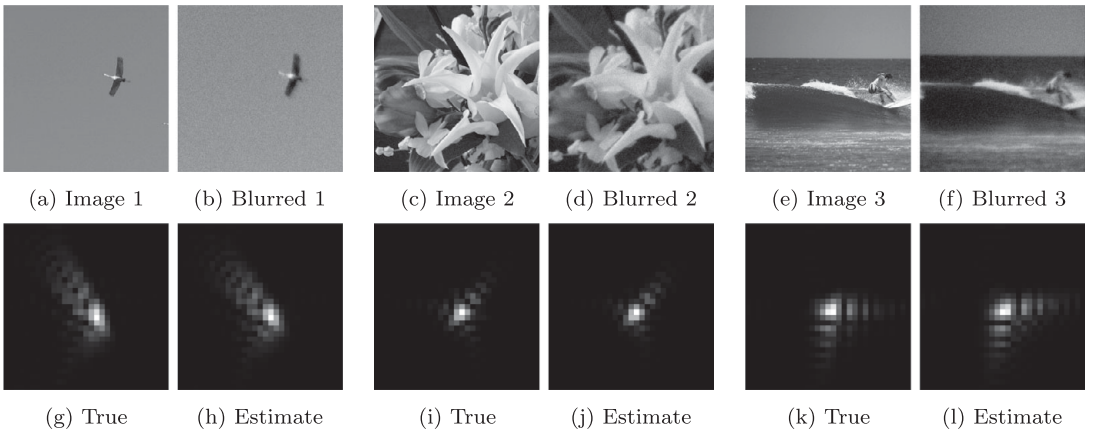


Figure 3. *Examples of results for the identification network with convolution kernels defined through Fresnel approximation. Top: the original and blurred and noisy* 400×400 *images. Bottom: the true* 31×31 *kernel used to generate the blurry image and the corresponding estimation by the neural network. Notice that there is a large amount of white Gaussian noise added to the blurred image. The image boundaries have been discarded from the estimation process to prevent the neural network from using information that would not be present in real images.*

parameters. This is the case for the deep-blur identification architecture, a ResNet encoder,[45] as shown in Figure 1, left. It consists of a succession of convolutions, ReLU activation, batch normalization, and average pooling layers, which sequentially reduce the image dimensions. The last layer is an adaptive average pooling layer, mapping the output of the penultimate layer to a vector of constant size K. In our experiments, the total number of trainable parameters, which includes the weights of the ResNet, that is, the convolution kernels in the convolution layers, the biases in the convolution layers and the weights of the adaptive pooling layer, is $|\boldsymbol{\theta}| = 11{,}178{,}448$. The encoder structure has been proven to be particularly effective for a large panel of signal processing tasks.[89]

2.2.2. The deblurring network

The proposed deblurring network mimics a Douglas–Rachford algorithm.[21] It is sometimes called an unrolled or unfolded network. This type of network currently achieves near state-of-the-art performance for a wide range of inverse problems (see e.g. [59]). It has the advantages of having a natural interpretation as an approximate solution of a variational problem and naturally adapts to changes of the observation operators.

Deep unrolling. For $\lambda > 0$, let $\mathbf{R}_{\gamma,\lambda}$ denote the following regularized inverse:

$$\mathbf{R}_{\boldsymbol{\gamma},\lambda} = \left(\mathbf{A}(\boldsymbol{\gamma})^T \mathbf{A}(\boldsymbol{\gamma}) + \lambda \mathbf{I}\right)^{-1} \mathbf{A}(\boldsymbol{\gamma})^T.$$

For a parameter $\boldsymbol{\gamma}$ describing the forward operator and an input image $\mathbf{y}$, the Douglas–Rachford algorithm can be described by the following sequence of operations, ran from $t = 0$ to $t = T - 1$ with $T \in \mathbb{N}$.

Algorithm 1 The Douglas–Rachford deblurring network DN.

Require: iteration number $T \in \mathbb{N}$, operator $\boldsymbol{\gamma}$, scale $\lambda \in \mathbb{R}_+$
  $\mathbf{z}_0 = \mathbf{R}_{\gamma,\lambda}(\mathbf{y})$
  for all $t = 0 \to T - 1$ **do**
    $\mathbf{x}_{t+1} = \mathrm{PN}_t(\mathbf{z}_t)$
    $\mathbf{z}_{t+1} = \mathbf{z}_t + \mathbf{R}_{\gamma,\lambda}(2\mathbf{x}_t - \mathbf{z}_t) - \mathbf{x}_t$
  end for

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

The initial guess $\mathbf{z}_0$ corresponds to the solution of

$$\mathbf{z}_0 = \underset{\mathbf{z}\in\mathbb{R}^N}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{A}(\boldsymbol{\gamma})\mathbf{z}-\mathbf{y}\|_2^2 + \frac{\lambda}{2}\|\mathbf{z}\|_2^2.$$

It can be evaluated approximately with a conjugate gradient algorithm run for a few iterations (20 in our implementation).

The mapping $\mathrm{PN}_t(\boldsymbol{\xi}_t):\mathbb{R}^N\to\mathbb{R}^N$ can be interpreted as a "proximal neural network." Proximal operators[21] have been used massively in the last 20 years to regularize inverse problems. A popular example is the soft-thresholding operator, which is known to promote sparse solutions. Here, we propose to learn the regularizer as a neural network denoted PN_t, which may change from one iteration to the next. It corresponds to the green layers in Figure 1.

The parameters $\boldsymbol{\xi}$ that are learned are the weights $\boldsymbol{\xi}_t$ defining the tth proximal neural network PN_t. In our experiments, the networks PN_t have the same architecture for all $1\le t\le T$. We used the current state-of-the-art network used in plug-and-play algorithms called DRUNet.[46,87] We set $T=4$ iterations. Each of the 4 proximal networks contain 8,159,808 parameters, resulting in a total of $|\boldsymbol{\xi}| = 32{,}639{,}232$ parameters to be trained.

2.3. Training

We propose to first train the identification network $\mathrm{IN}(\boldsymbol{\theta})$ alone and then train the deblurring network $\mathrm{DN}(\boldsymbol{\xi})$ with the output of the identification network as an input parameter. This sequential approach presents two advantages:

- The memory consumption is lower. The automatic differentiation only needs to store the parameters of the individual networks, instead of both. This reduces the memory footprint.
- The identification network can be used independently of the other, and it is therefore tempting to train it separately. In metrology applications, for instance, where the aim is to follow the state of an optical system through time, the identification network IN is the most relevant brick. In some applications, such as superresolution from single molecules, the deblurring network could be replaced by a more standard total variation-based solver,[13] once the operator is estimated.

In what follows, we let $\mathcal{X}\subset\mathbb{R}^N$ denote a dataset of admissible images/signals and $\mathcal{L}_{\mathcal{X}}$ denote a sampling distribution over $\mathcal{X}$. We let $\mathcal{L}_\Gamma$ denote a sampling distribution on the set $\mathbb{R}^K$ of blur parameters. In our experiments, the perturbation $\mathcal{P}$ in Equation (2) is assumed to be an approximation of the Poisson-Gaussian noise.[35] We assume that $\mathbf{y}=\mathbf{A}(\bar{\boldsymbol{\gamma}})\bar{\mathbf{x}}+\mathbf{b}$, where $\mathbf{b}[z]\sim\boldsymbol{\sigma}[z]\boldsymbol{\eta}[z]$, $\eta\sim\mathcal{N}(0,\mathbf{I}_M)$, and $\boldsymbol{\sigma}[z]=\sqrt{\alpha(\mathbf{A}(\bar{\boldsymbol{\gamma}})\bar{\mathbf{x}})[z]+\beta}$. The parameters α and β are set uniformly at random in the ranges $\alpha\in[0,0.05]$ and $\beta\in[0,0.15]$. In what follows, we let $\mathcal{L}_b$ denote the noise distribution that we just described.

We propose to train both the identification and the deblurring networks using the empirical risk minimization. First, the identification network is trained by solving:

$$\inf_{\theta\in\mathbb{R}^{|\theta|}} \mathop{\mathbb{E}}_{\substack{\mathbf{x}\sim\mathcal{L}_{\mathcal{X}}\\ \boldsymbol{\gamma}\sim\mathcal{L}_\Gamma\\ \mathbf{b}\sim\mathcal{L}_b}} \left[\frac{1}{2}\|\mathrm{IN}(\boldsymbol{\theta})(\mathbf{A}(\boldsymbol{\gamma})\mathbf{x}+\mathbf{b})-\boldsymbol{\gamma}\|_2^2\right]. \tag{4}$$

Once the identification network IN is trained, we turn to the deblurring network by solving the following optimization problem:

$$\inf_{\xi\in\mathbb{R}^{|\xi|}} \mathop{\mathbb{E}}_{\substack{\mathbf{x}\sim\mathcal{L}_{\mathcal{X}}\\ \boldsymbol{\gamma}\sim\mathcal{L}_\Gamma\\ \mathbf{b}\sim\mathcal{L}_b}} \left[\frac{1}{2}\|\mathrm{DN}(\boldsymbol{\xi})(\mathbf{y},\hat{\boldsymbol{\gamma}})-\mathbf{x}\|_2^2\right], \tag{5}$$

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

where $\mathbf{y} = \mathbf{A}(\boldsymbol{\gamma})\mathbf{x} + \mathbf{b}$ is the degraded image and $\hat{\boldsymbol{\gamma}} = \mathrm{IN}(\boldsymbol{\theta})(\mathbf{y})$ is the estimated parameter. Of importance, notice that we do not plug the true parameter $\boldsymbol{\gamma}$ in 5, but rather the estimated one $\hat{\boldsymbol{\gamma}}$. This way, the deblurring network DN can learn to correct model mismatches that may occur at the estimation step.

The two problems above consist in constructing minimum mean square estimators (MMSE). At the end of the training procedure – under technical assumptions[41] – we can consider that the networks approximate a conditional expectation:

$$\mathrm{IN}(\boldsymbol{\theta})(\mathbf{y}) \approx \mathbb{E}[\boldsymbol{\gamma}|\mathbf{y}]$$
$$\mathrm{DN}(\boldsymbol{\xi})(\mathbf{y},\boldsymbol{\gamma}) \approx \mathbb{E}[\mathbf{x}|\mathbf{y},\boldsymbol{\gamma}].$$

This is – by construction – the best estimators that can be generated on average. This approach is therefore really different from most alternatives in the literature, which consist of constructing MAP estimators. MMSE estimators can be expressed as integrals, which depend heavily on the operator distributions $\mathcal{L}_\Gamma$ and on the image distribution $\mathcal{L}_\mathcal{X}$. They should therefore be constructed carefully depending on the physical knowledge of the observation system (resp. observed sample). By using the general computer vision database COCO, we hope to cover a wide range of image contents, leading to a wide-purpose method for identification. The performance could likely be improved using more specific databases. For instance, we could simulate the images according to realistic processes for specific applications such as single-molecule localization. This is out of the scope of this article. For $\mathcal{L}_\Gamma$, we sample a large set of realistic parameters uniformly at random in our experiments.

The above optimization problems are solved approximately using stochastic gradient descent-type algorithms. In our experiments, we used the Adam optimizer[48] with the default parameters: the learning rate is set to 0.001, betas are (0.9, 0.999), epsilon is 1e-8, weight decay is 0, and amsgrad is false.

3. Results

Let us illustrate the different ideas proposed in this article. In all our experiments, we trained the neural networks using the MS COCO dataset.[53] It contains 118,287 images in the training set and 40,670 images in the test set. It is composed of images of everyday scenes, capturing objects in various indoor and outdoor environments. It presents substantial differences with typical microscopy images, but the high diversity and quality of the images makes it possible to construct efficient generic image priors. This was already observed in [77].

3.1. Convolution operators

We evaluate the accuracy of the identification and deblurring networks for convolution (i.e., space invariant) operators. We assess them for images generated with point spread functions expanded in Zernike polynomial.

3.1.1. Identifying convolution operators

We assess the ability of a residual network to identify the point spread function generated by the Fresnel diffraction Model 2.3. A similar study was carried out in [76] with $K = 3$ coefficients. Here, we extend the study to $K = 7$ coefficients allowing us to represent the following aberrations in the Noll nomenclature[63]: defocus, primary astigmatism, primary coma, trefoil, and primary spherical.

We generate random PSFs by drawing the coefficients $\boldsymbol{\gamma}[k]$ (see Model 2.3) uniformly in the range $[-\eta, \eta]$. The higher η, the more spread and oscillating the PSFs. Hence, η can be interpreted as a measure of PSF complexity. The model was trained for a value of $\eta = 0.15$.

In the first experiment, we simply used additive white Gaussian noise (i.e. $\beta = 0$) of standard deviation α. Figure 3 shows the identification results for 3 images taken at random from the test set and 3 operators taken at random in the operator set. On these examples, the network provides faithful estimates despite a substantial noise level and images with little contents. To further characterize the network efficiency, we

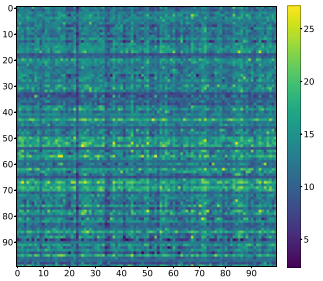
https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

measure the distribution of signal-to-noise-ratio (SNR) in the noiseless regime. For a kernel $\mathbf{h}$, the error of the estimated kernel $\hat{\mathbf{h}}$ is defined by

$$\mathrm{SNR}\left(\mathbf{h},\hat{\mathbf{h}}\right) = -10\log_{10}\left(\frac{\|\hat{\mathbf{h}}-\mathbf{h}\|_2^2}{\|\mathbf{h}\|_2^2}\right). \tag{6}$$

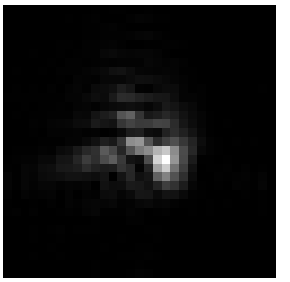
Figure 4 summarizes the conclusions. In average, the identification network outputs estimates with a relative error below 5%.

Finally, we study the stability to the noise level α in Figure 5a – and to the PSF complexity η in Figure 5b. As can be seen, the identification outputs predictions with less than 10% error with a probability larger than 0.5 up to a large noise level of $\alpha = 0.1$ for images in the range $[0,1]$. The dependency on the kernel's complexity, measured through the Zernike polynomials amplitude η is very clear with typical errors below 2% for $\eta < 0.1$ and then a relatively fast increase. It is nonetheless remarkable that the identification returns estimates with less than 15% error for $\eta = 0.2$, which produces more complex PSFs than those observed during the training phase, showing some ability of the network to generalize.



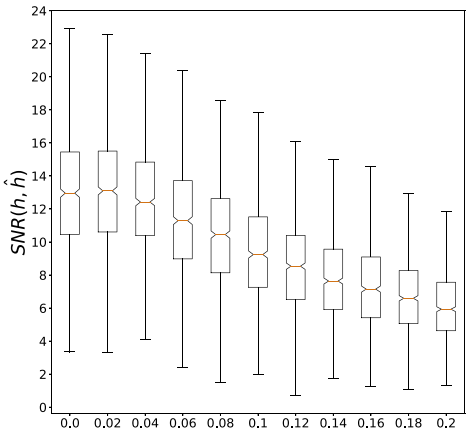
(a) $\mathrm{SNR}(\mathbf{h},\hat{\mathbf{h}})$

(b) A sneaky image

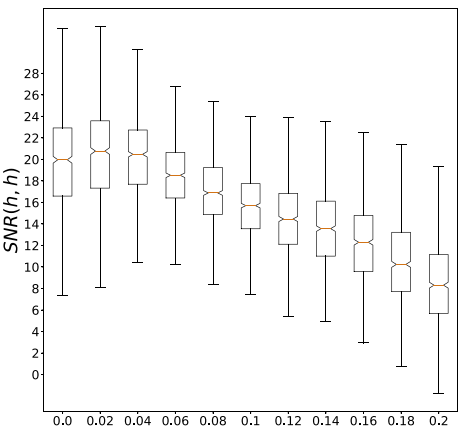


(c) A hard kernel

Figure 4. *On the left: a* 100×100 *table representing the SNR of the PSF. In this table, we evaluated the identification network for 100 images (left to right) and 100 kernels (top to bottom) with no noise. As can be seen, there are horizontal and vertical stripes. This means that some images and some kernels make the identification problem easier or harder. In the middle: an image making the identification problem hard (column 23). On the right: a kernel making the identification harder (row 65).*



(a) Noise level α



(b) Zernike amplitude

Figure 5. *Stability of the kernel estimation with respect to noise level (left) and amplitude of the Zernike coefficients in the noiseless regime (right).*

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

3.1.2. Evaluating the deblurring network

We evaluate the performance of the proposed deblurring network for convolution operators defined using the Fresnel approximation. Figures 6–8 display some deconvolution results for different methods. The corresponding image quality measures are displayed in Table 1.

Notice that this problem is particularly involved: there is a complete loss of information in the high frequencies since the convolution kernels are bandlimited and we treat different noise levels up to rather

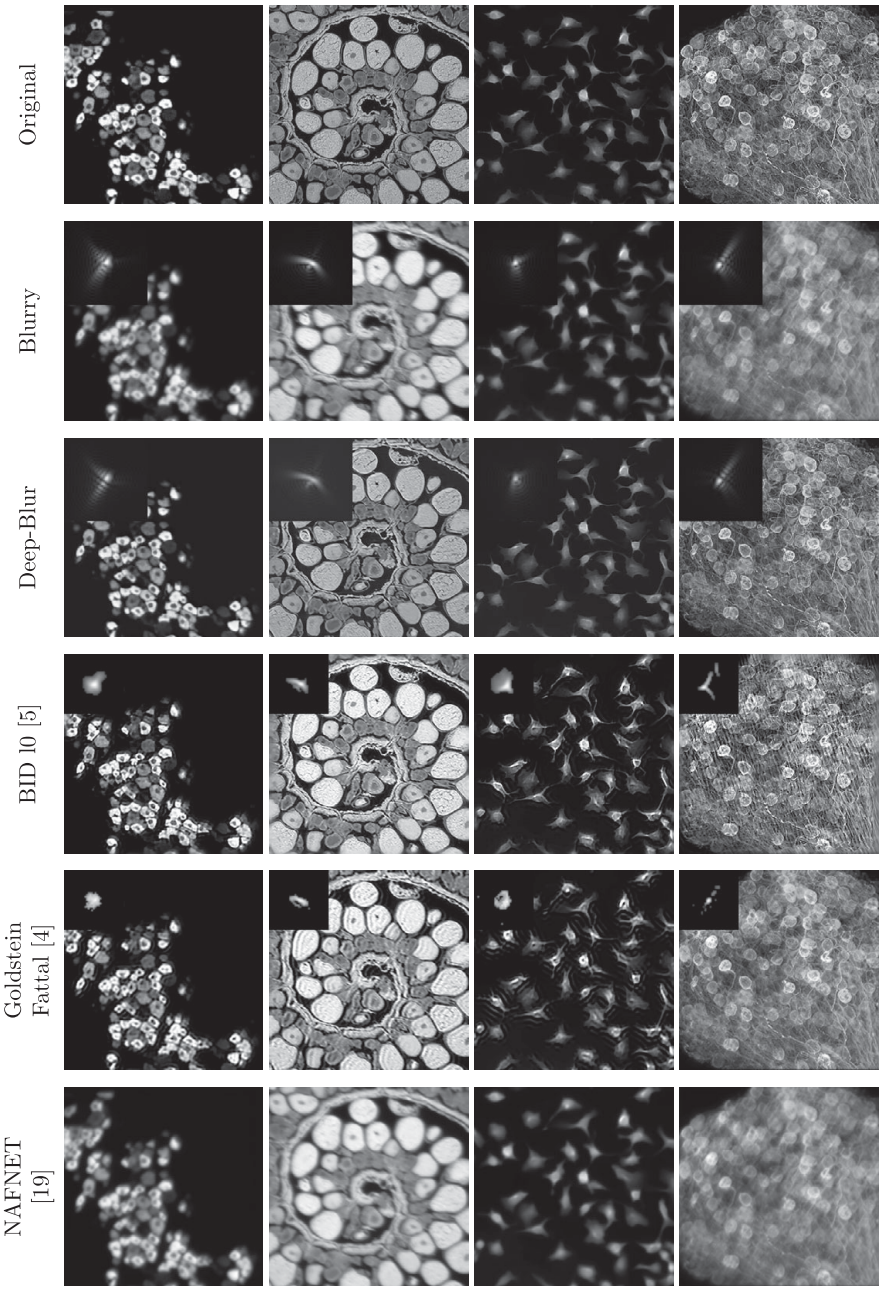


Figure 6. Deep-blur in action in the noiseless setting. Quantitative evaluations are reported in Table 1. When available, the estimated blur kernel is displayed at the bottom right. First row: original images. Second row: blurry-noisy images. Third row: deep-blur. Fourth row: [5] Fifth row: [4] Sixth row: [19].

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

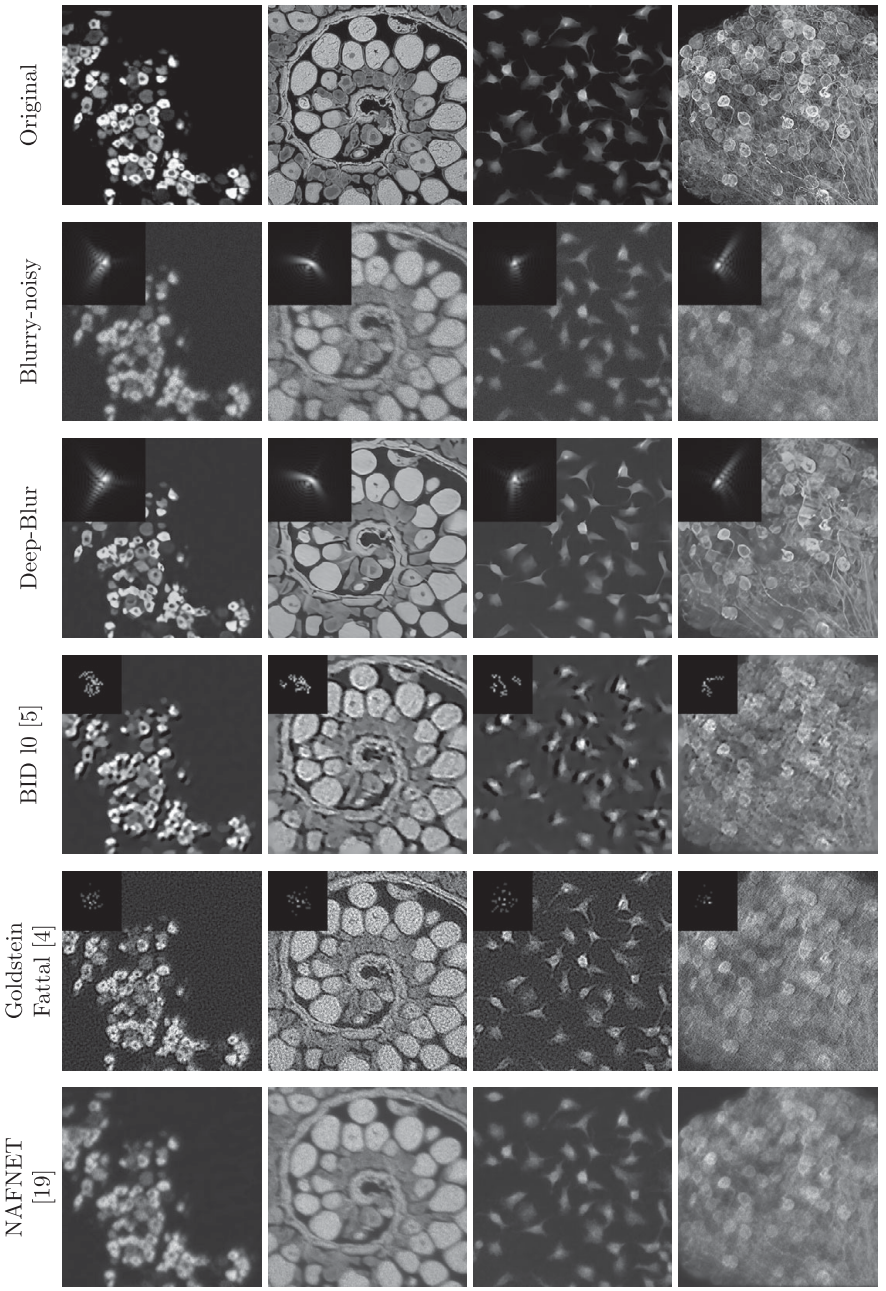


Figure 7. *Deep-blur in action with a medium noise level ($\alpha = 0.025$, $\beta = 0.05$). Quantitative evaluations are reported in Table 1. When available, the estimated blur kernel is displayed at the bottom right. First row: original images. Second row: blurry-noisy images. Third row: deep-blur. Fourth row: [5] Fifth row: [4] Sixth row: [19].*

high values (here $\alpha = 0.12$, $\beta = 0.24$ for images in the range $[0, 1]$). Despite this challenging setting, it can be seen both perceptually and from the SSIM (structural similarity index measure) that the image quality is improved whatever the noise level. It is also remarkable to observe that the proposed network architecture allows us to treat images with different noise levels. This is an important feature of the DRUNet used as a proximal network.[87]

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

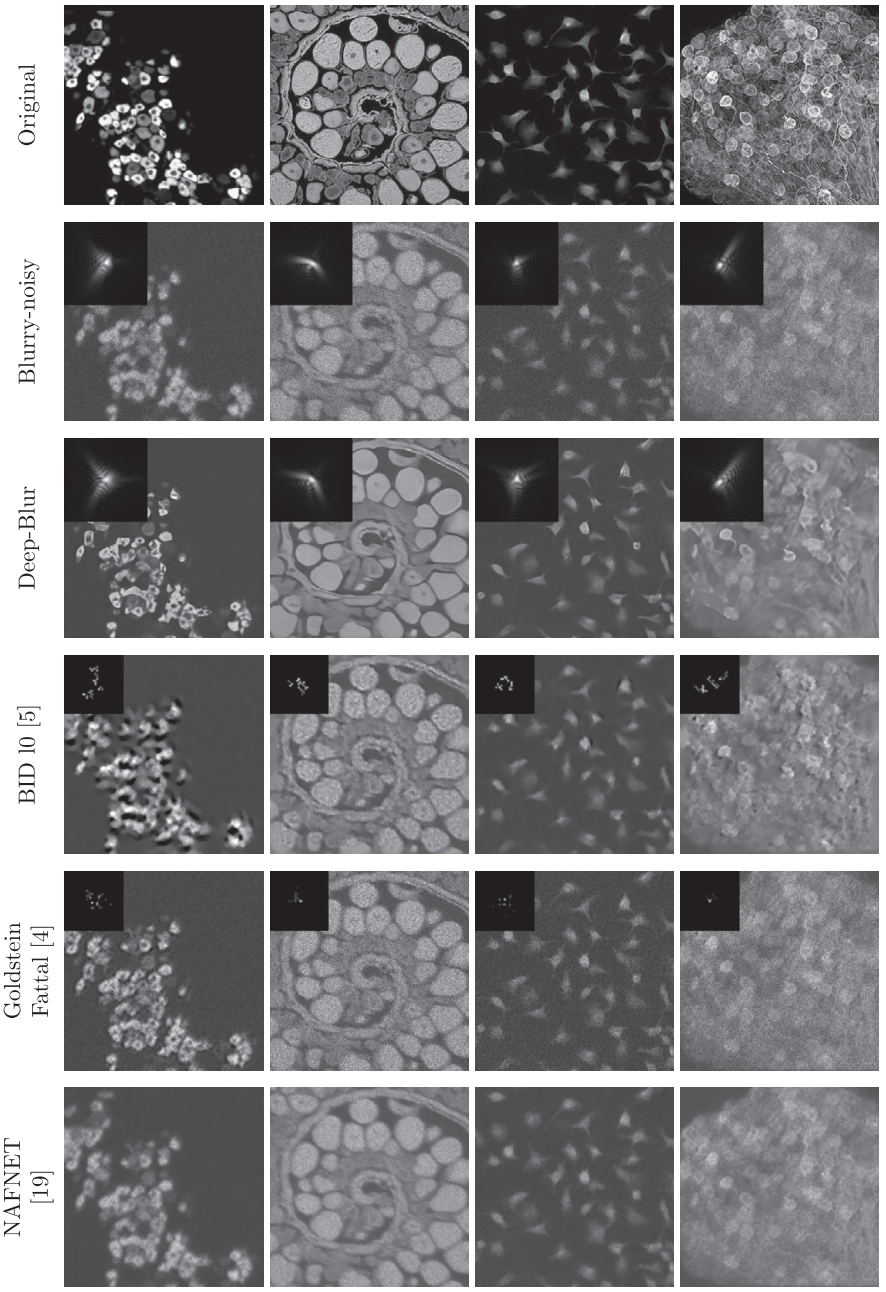


Figure 8. *Deep-blur in action in a high-noise regime ($\alpha = 0.12$, $\beta = 0.24$). Quantitative evaluations are reported in Table 1. When available, the estimated blur kernel is displayed at the bottom right. First row: original images. Second row: blurry-noisy images. Third row: deep-blur. Fourth row: [5] Fifth row: [4] Sixth row: [19].*

We also propose some comparisons with other methods from the literature. Whenever possible, we optimized the hyperparameters by hand for each noise level to produce the best possible output. We chose the following methods:

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

Table 1. *Reconstruction results for different noise levels and different methods*

		$\alpha=0,\beta=0$	$\alpha=0.025,\beta=0.12$	$\alpha=0.05,\beta=0.24$
SSIM$\hat{\mathbf{x}}$	$\mathbf{y}$	0.76 ± 0.17	0.31 ± 0.10	0.19 ± 0.07
	Deep-blur	$\mathbf{0.85\pm0.16}$	$\mathbf{0.53\pm0.16}$	$\mathbf{0.40\pm0.16}$
	(5)	0.48 ± 0.27	0.33 ± 0.13	0.27 ± 0.11
	(4)	0.64 ± 0.2	0.21 ± 0.08	0.21 ± 0.08
	(19)	0.66 ± 0.2	0.43 ± 0.16	0.33 ± 0.13
SNR$\hat{\mathbf{h}}$	Deep-blur	$\mathbf{12.9\pm4.4}$	$\mathbf{10.8\pm4.3}$	$\mathbf{7.9\pm3.7}$
	(5)	-6.6 ± 2.4	-3.6 ± 1.6	-3.1 ± 1.3
	(4)	-5.7 ± 1.9	-1.6 ± 0.6	-1.26 ± 0.45

Note: The standard deviation is given after the symbol ±.
Bold numbers indicate the best performing method.

- The ℓ^0-gradient prior.[66,67] This method is one of the state-of-the-art handcrafted blind deblurring methods. An efficient implementation was recently proposed in [5].
- In [38], the authors proposed a kernel estimation method based on the assumption that the image spectrum amplitude has a specific decaying distribution in the Fourier plane. The kernel estimation then boils down to a phase retrieval problem. An efficient implementation was recently proposed in [4].
- We also tested two state-of-the-art neural network approaches. The first one was a past leader of the Go-Pro deblurring challenge called NAFNET.[19]

The deep learning method is retrained on the same dataset as our method. As can be seen from Table 1 and the perceptual results in Figures 6–8, deep-blur outperforms the other three methods that we considered by a large margin. The PSF is recovered with an average accuracy varying between 12.9 dB in the noiseless regime to 7.9 dB in the high-noise regime using deep-blur. The image quality is improved in terms of SSIM by 0.1 in the noiseless regime to 0.2 in the high-noise regime.

All the other methods yield negative SNR for the PSF. At a perceptual level, handcrafted methods (Goldstein-Fattal and ℓ^0 gradient prior) still recover the PSF shape approximately. The recovered image is also sharpened, but its SSIM quality is actually lowered by more than 0.1 in the noiseless regime, and improved by 0.1 in the high-noise regime. The SSIM is always lower than the one of deep-blur.

Experiments on real images. In Figure 9, we provide a few deep-blur results on real microscopy images from the dataset.[43] We used the sample images available on the following link. As can be seen, the reconstructed images are denoised and have a better visual contrast In this experiment, we do not have a ground-truth deblurred result and the quality can only be assessed by visual inspection. Validating the estimation requires careful optics experiments, which we leave as an open topic for now.

Training on the true or estimated operators. At training time, we can feed the unrolled deblurring network with the operator that was used to synthesize the blurry image, or the one estimated using the identification network. The potential advantage of the second option is to train the proximal networks to correct model mismatches. We tested both solutions on two different operator families. It turns out that they led to nearly indistinguishable results overall in average. The most likely explanation for this phenomenon is that the model mismatches produced by the identification network cannot be corrected with the proximal networks.

Memory and computing times. The model contains about $11\cdot10^6$ parameters for the identification part and $32\cdot10^6$ parameters for the deblurring part. This is a total of $43\cdot10^6$ trainable parameters. This size is comparable to the usual computer vision models available in TorchVision. For example, it is slightly

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

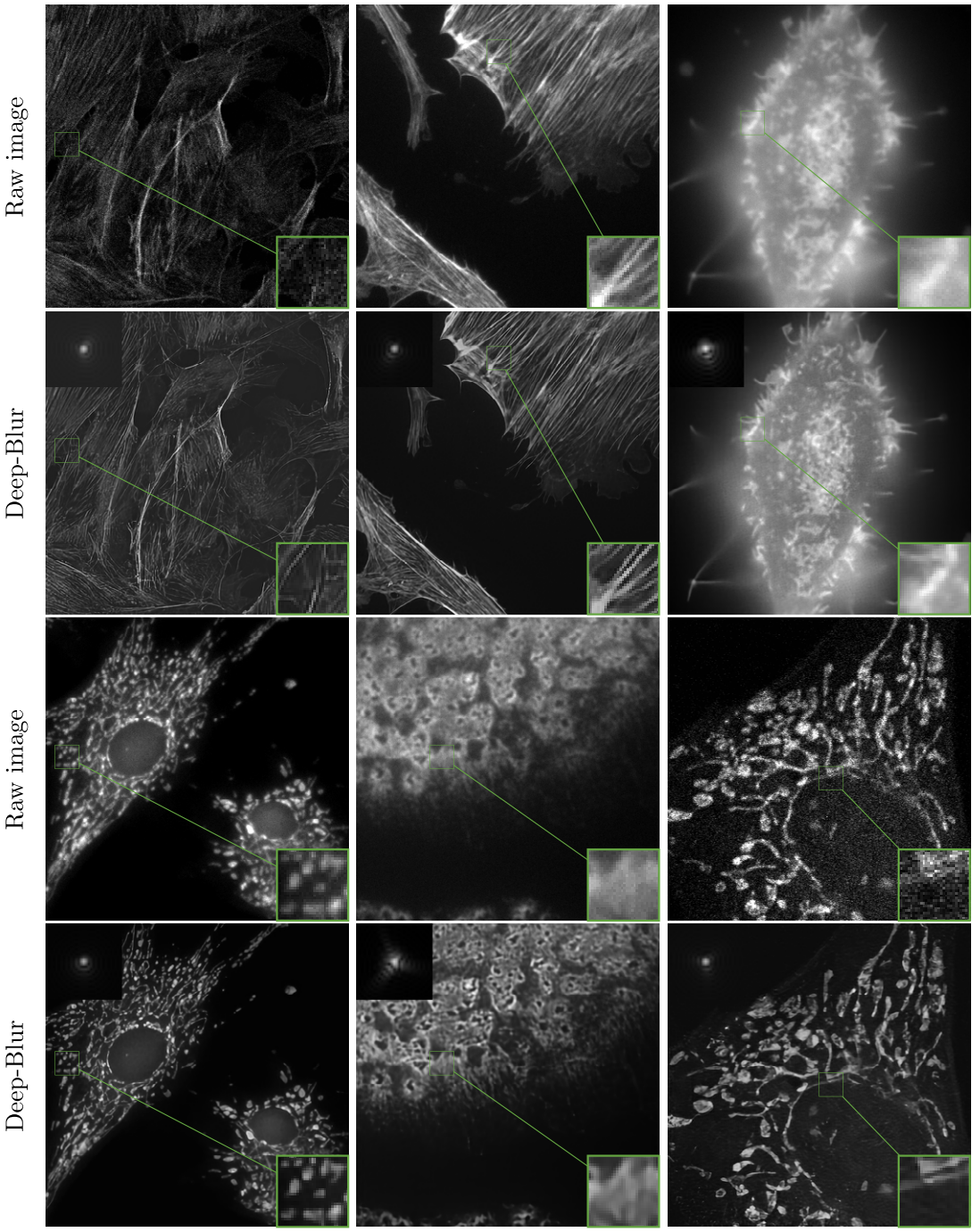


Figure 9. Blind deblurring examples on real images taken from [43], see the samples for more details. In this experiment, only the noise level was set manually, the rest of the process is fully automatized. For this experiment, no ground truth is available and the results have to be assessed by visual inspection.

smaller than a ResNet101 classifier. The deep-blur model uses about 1 GB of GPU memory at test time, which can be considered lightweight, since it fits on most consumer graphics cards.

After training, it takes 0.3 seconds to identify a kernel and deblur an image of size 400×400 on an Nvidia RTX 8000 with 16 TFlops. For comparison, the handcrafted models used in our numerical

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

comparisons take between 5 seconds and a few minutes to perform the same task on the CPU. No GPU implementation is provided.

3.2. Product-convolution operators

To finish the numerical experiments, we illustrate how the proposed ideas perform on product-convolution operators.

We first illustrate the performance of the identification network. We trained the identification network on natural images from the MS COCO dataset, but evaluate it on biological images from microscopes. We selected 6 images: *ImBio 1* is an histopathology of angiolipoma,[22] *ImBio 2* is an histopathology of reactive gastropathy,[23], *ImBio 3 and 4* are actin filaments within a cell,[24] *ImBio 5* is an slice of a spheroid from [54], and *ImBio 6* is a crop of a podosome obtain on a wide-field microscope.[12]

The blur operators are generated by Model 2.2 using $K = 16$ parameters. The blur model is obtained following the procedure described in [28]. To compute the product-convolution decomposition described in Model 2.2, we collected 18 stacks of 21 images of microbeads spaced by 200 nm on a wide-field microscope with a ×100 objective lens (CFI SR APO 100XH NA 1,49 DT 0,12 Nikon) mounted on a Nikon Eclipse Ti-E and a Hamamatsu sCMOS camera (ORCA FLASH4.0 LT). Figure 10 shows the identification results. The blur coefficients predicted by the deep-blur identification are accurate estimates in all cases. On average, the SNR is much higher than in the previous experiment, which can likely be explained by a smaller dimensionality of the operators' family. In all cases, the image quality is improved despite an additive white Gaussian noise with $\alpha = 1 \cdot 10^{-2}$ and $\beta = 0$. This is remarkable since this type of image is different from the typical computer vision images found in the MS COCO dataset.

4. Discussion

We proposed an efficient and lightweight network architecture for solving challenging blind deblurring problems in optics. An encoder first identifies a low-dimensional parameterization of the optical system from the blurry image. A second network with an unrolled architecture exploits this information to efficiently deblur the image. The performance of the overall architecture compares favorably with alternative approaches designed in the field of computer vision. The principal reason is that our network is trained using fine physical models obtained using Fresnel diffraction theory or experimental data providing accurate space-varying models. A second reason is that the unrolled architecture proposed herein emerges as a state-of-the-art competitor for a wide range of inverse problems. Overall, we believe that the proposed network, trained carefully on a large collection of blurs and images could provide a universal tool to deblur microscope images. In the future, we would like to carry out specific optical experiments to ensure that the results obtained with synthetic data are reproducible and trustworthy with real images. The initial results, obtained without reference images for comparisons are however really encouraging.

Differences with plug and play and deep unrolling. The proposed unrolled architecture follows closely the usual unrolled algorithms.[1,2,51,52,59] There is, however, a major difference: traditionally, these unrolled architectures are trained to invert a *single operator*. In this article, we train the network with a *family* of operators. The results we obtained confirm some results obtained in [41]: this approach only results in marginal performance loss for a given operator if the family is sufficiently small while providing a massive improvement in adaptivity to all the operators. In a sense, the proposed approach can be seen as an intermediate step between the plug-and-play priors (related to diffusion models[42]), which are designed to adapt to all possible operators, and the traditional unrolled algorithm is adapted to a single one.

The limits of a low-dimensional parameterization. The models considered are sufficiently rich to describe most optical devices accurately. In microscopy, they can capture defocus, refractive index

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

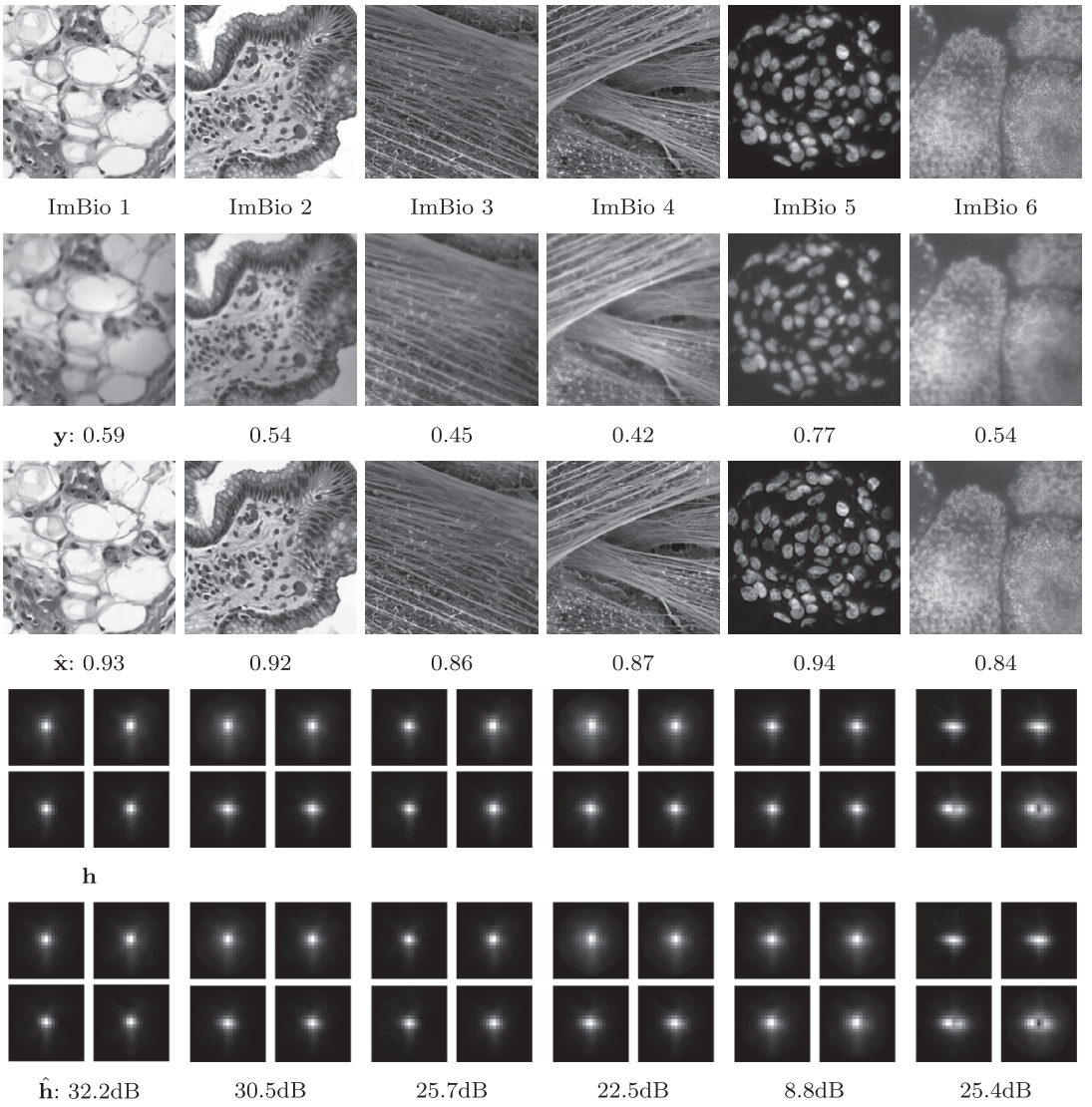


Figure 10. *Deep-blur applied to spatially varying blur operators on microscopy images (not seen during training). The blur operators are sampled from a family estimated using a real wide-field microscope. First row: the original images. Second row: blurry-noisy images. Third row: the blind deblurring result with deep-blur. The SSIM of the resulting deblurred image is displayed below. Fourth row: The true blur operator. We display 4 evenly spaced impulse responses in the field of view. Fifth row: The estimated blur operator. The SNR of the estimated kernel is displayed in the caption in dB.*

mismatches, changes of temperature, tilts of optical components, usual optical aberrations with a parameter dimension K smaller than 20.

Notice however that some phenomena can hardly be modeled by low-dimensional parameterization. In microscopy, for instance, diffraction by the sample itself can lead to extremely complicated and diverse forward models better described by nonlinear equations (see e.g., [71]). Similarly, in computer vision,

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

motion and defocus blurs can vary abruptly with the movement and depth of the objects. The resulting operators would likely require a large number of parameters, which is out of the scope of this article.

Overall, we see that the proposed contribution is well adapted to the correction of systems with slowly varying point spread functions but probably does not extend easily to fast variations that can be induced by some complex biological samples.

5. Conclusion

We proposed a specific neural network architecture to solve blind deblurring problems where distortions come from the optical elements. We evaluated its performance carefully on blind deblurring problems with space invariant and space-varying operators. A key assumption is to have access to a forward model that depends on a set of parameters. The network first estimates the unknown parameters describing the forward model from the measurements with a ResNet architecture. In a second step, an unrolled algorithm solves the inverse problem with a forward model that was estimated at the previous step. After designing a careful training procedure, we showed an advantage of the proposed approach in terms of robustness to noise levels and adaptivity to a vast family of operators and conditions not seen during the training phase.

Acknowledgments. The authors wish to thank Emmanuel Soubies and Thomas Mangeat for fruitful discussions. They thank the associate editor and the anonymous reviewers for their comments and advices.

Data availability statement. The deep-blur architecture and the trained weights can be provided on explicit demand by e-mail at pierre.weiss@cnrs.fr.

Author contribution. Both authors contributed equally to all parts (conception, writing, programming, training, and experiments) of this article. P.W. was responsible for finding the funding.

Funding statement. This work was supported by the ANR Micro-Blind ANR-21-CE48–0008 and by the ANR LabEx CIMI (grant ANR-11-LABX-0040) within the French State Program "Investissements d'Avenir". The authors acknowledge the support of AI Interdisciplinary Institute ANITI funding, through the French "Investing for the Future—PIA3" program under the Grant Agreement ANR-19-PI3A-0004. This work was performed using HPC resources from GENCI-IDRIS (Grant 2021-AD011012210R1).

Competing interest. The authors declare none.

References

1. Adler J and Öktem O (2017) Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems* **33** (12), 124007.
2. Adler J and Oktem O (June 2018) Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging* **37**(6), 1322–1332.
3. Aljadaany R, Pal DK and Savvides M (2019) Douglas-Rachford networks: learning both the image prior and data fidelity terms for blind image deconvolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10235–10244.
4. Anger J, Facciolo G and Delbracio M (2018) Estimating an image's blur kernel using natural image statistics, and deblurring it: An analysis of the Goldstein-Fattal method. *Image Processing On Line* **8**, 282–304. https://doi.org/10.5201/ipol.2018.211.
5. Anger J, Facciolo G and Delbracio M (2019) Blind image deblurring using the l0 gradient prior. *Image Processing On Line* **9**, 124–142. https://doi.org/10.5201/ipol.2019.243.
6. Antun V, Renna F, Poon C, Adcock B and Hansen AC (2020) On instabilities of deep learning in image reconstruction and the potential costs of AI. *Proceedings of the National Academy of Sciences* **117**(48), 30088–30095.
7. Aristov A, Lelandais B, Rensen E and Zimmer C (2018) ZOLA-3D allows flexible 3D localization microscopy over an adjustable axial range. *Nature Communications* **9**(1), 1–8.
8. Arridge S, Maass P, Öktem O and Schönlieb C-B (2019) Solving inverse problems using data-driven models. *Acta Numerica* **28**, 1–174.
9. Bar L, Sochen N and Kiryati N (2006) Semi-blind image restoration via Mumford-Shah regularization. *IEEE Transactions on Image Processing* **15**(2), 483–493.
10. Bigot J, Escande P and Weiss P (2019) Estimation of linear operators from scattered impulse responses. *Applied and Computational Harmonic Analysis* **47**(3), 730–758.

11. Bolte J, Sabach S and Teboulle M (2014) Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming* **146**(1), 459–494.
12. Bouissou A, Proag A, Bourg N, Pingris K, Cabriel C, Balor S, Mangeat T, Thibault C, Vieu C, Dupuis G, Fort E, Lévêque-Fort S, Maridonneau-Parini I and Poincloux R (2017) Podosome force generation machinery: a local balance between protrusion at the core and traction at the ring. *ACS Nano* **11**(4), 4028–4040.
13. Bredies K and Pikkarainen HK (2013) Inverse problems in spaces of measures. *ESAIM: Control, Optimisation and Calculus of Variations* **19**(1), 190–218.
14. Chakrabarti A (2016). A neural approach to blind motion deblurring. In *European Conference on Computer Vision*. Springer, pp. 221–235.
15. Chakrabarti A, Zickler T and Freeman WT (2010) Analyzing spatially-varying blur. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 2512–2519.
16. Chambolle A and Pock T (2016) An introduction to continuous optimization for imaging. *Acta Numerica* **25**, 161–319.
17. Chan TF and Wong C-K (1998) Total variation blind deconvolution. *IEEE Transactions on Image Processing* **7**(3), 370–375.
18. Chaudhuri S, Velmurugan R and Rameshan R (2016) *Blind Image Deconvolution*. Springer.
19. Chen L, Chu X, Zhang X and Sun J (2022) Simple baselines for image restoration. In *European Conference on Computer Vision*. Springer, pp. 17–33.
20. Cho S-J, Ji S-W, Hong J-P, Jung S-W and Ko S-J (2021) Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*,pp. 4641–4650.
21. Combettes PL and Pesquet J-C (2011) Proximal splitting methods in signal processing. In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. New York: Springer, pp. 185–212.
22. Wikimedia Commons (2020) File:Histopathology of angiolipoma.jpg — Wikimedia Commons, the free media repository (accessed 9 May 2022). https://commons.wikimedia.org/wiki/File:Histopathology_of_angiolipoma.jpg
23. Wikimedia Commons (2020) File:Histopathology of reactive gastropathy.jpg — Wikimedia Commons, the free media repository (accessed 9 May 2022). https://commons.wikimedia.org/wiki/File:Histopathology_of_reactive_gastropathy.jpg.
24. Wikimedia Commons (2020) File:STD Depth Coded Stack Phallodin Stained Actin Filaments.png — Wikimedia Commons, the free media repository (accessed 9 May 2022). https://commons.wikimedia.org/wiki/File:STD_Depth_Coded_Stack_Phallodin_Stained_Actin_Filaments.png.
25. Couzinie-Devy F, Sun J, Alahari K and Ponce J (2013) Learning to estimate and remove non-uniform image blur. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1075–1082.
26. Cumming BP and Gu M (2020) Direct determination of aberration functions in microscopy by an artificial neural network. *Optics Express* **28**(10), 14511–14521.
27. Dai S and Wu Y (2008) Motion from blur. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1–8.
28. Debarnot V, Escande P, Mangeat T and Weiss P (2021) Learning low-dimensional models of microscopes. *IEEE Transactions on Computational Imaging* **7**, 178–190.
29. Debarnot V, Escande P and Weiss P (2019) A scalable estimator of sets of integral operators. *Inverse Problems* **35**, 105011.
30. Debarnot V and Weiss P (2021) Deepblur: Blind identification of space variant PSF. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*. IEEE, pp. 1544–1547.
31. Debarnot V and Weiss P (2022) Blind inverse problems with isolated spikes. *Information and Inference: A Journal of the IMA* **12**(1), 26–71.
32. Denis L, Thiébaut E, Soulez F, Becker J-M and Mourya R (2015) Fast approximations of shift-variant blur. *International Journal of Computer Vision* **115**(3), 253–278.
33. Escande P and Weiss P (2017) Approximation of integral operators using product-convolution expansions. *Journal of Mathematical Imaging and Vision* **58**(3), 333–348.
34. Fergus R, Singh B, Hertzmann A, Roweis ST and Freeman WT (2006) Removing camera shake from a single photograph. In *ACM SIGGRAPH 2006 Papers on - SIGGRAPH '06*. ACM Press, 787–794.
35. Foi A, Trimeche M, Katkovnik V and Egiazarian K (2008) Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing* **17**(10), 1737–1754.
36. Genzel M, Macdonald J and Marz M (2022). Solving inverse problems with deep neural networks-robustness included. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(1), 1119–1134.
37. Gibson SF and Lanni F (1989) Diffraction by a circular aperture as a model for three-dimensional optical microscopy. *Journal of the Optical Society of America A* **6**(9), 1357–1367.
38. Goldstein A and Fattal R (2012) Blur-kernel estimation from spectral irregularities. In *European Conference on Computer Vision*. Springer, pp. 622–635.
39. Gong D, Yang J, Liu L, Zhang Y, Reid I, Shen C, Hengel AVD and Shi Q (2017) From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2319–2328.
40. Goodman JW (2005) *Introduction to Fourier Optics*. Roberts and Company Publishers.
41. Gossard A and Weiss P (2024) Training adaptive reconstruction networks for blind inverse problems. *SIAM Imaging Science* **17**, 1314–1346.

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

42. Graikos A, Malkin N, Jojic N and Samaras D (2022) Diffusion models as plug-and-play priors. *Advances in Neural Information Processing Systems* **35**, 14715–14728.
43. Hagen GM, Bendesky J, Machado R, Nguyen T-A, Kumar T and Ventura J (2021) Fluorescence microscopy datasets for training deep neural networks. *GigaScience* **10**(5), giab032.
44. Hanser BM, Gustafsson MGL, Agard DA and Sedat JW (2004) Phase-retrieved pupil functions in wide-field fluorescence microscopy. *Journal of Microscopy* **216**(1), 32–48.
45. He K, Zhang X, Ren S and Sun J (2016) Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 770–778.
46. Hurault S, Leclaire A and Papadakis N (2022) Gradient step denoiser for convergent plug-and-play. In *International Conference on Learning Representations (ICLR'22)*, International Conference on Learning Representations, Online, United States.
47. Keuper M, Schmidt T, Temerinac-Ott M, Padeken J, Heun P, Ronneberger O and Brox T (2013) Blind deconvolution of widefield fluorescence microscopic data by regularization of the optical transfer function (OTF). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2179–2186.
48. Kingma D and Ba J (2015) Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diega, CA.
49. Krahmer F, Lin Y, McAdoo B, Ott K, Wang J, Widemann D and Wohlberg B (2006) Blind image deconvolution: Motion blur estimation. Master research report.
50. Levin A, Weiss Y, Durand F and Freeman WT (2009) Understanding and evaluating blind deconvolution algorithms. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1964–1971.
51. Li Y, Tofighi M, Geng J, Monga V and Eldar YC (2020) Efficient and interpretable deep blind image deblurring via algorithm unrolling. *IEEE Transactions on Computational Imaging* **6**, 666–681.
52. Li Y, Tofighi M, Monga V and Eldar YC (2019) An algorithm unrolling approach to deep image deblurring. In *ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 7675–7679.
53. Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P and Zitnick CL (2014) Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*. Springer, pp. 740–755.
54. Lorenzo C, Frongia C, Jorand R, Fehrenbach J, Weiss P, Maandhui A, Gay G, Ducommun B and Lobjois V (2011) Live cell division dynamics monitoring in 3D large spheroid tumor models using light sheet microscopy. *Cell Division* **6**(1), 1–8.
55. Lucas A, Iliadis M, Molina R and Katsaggelos AK (2018) Using deep neural networks for inverse problems in imaging: beyond analytical methods. *IEEE Signal Processing Magazine* **35**(1), 20–36.
56. McCann MT, Jin KH and Unser M (2017) Convolutional neural networks for inverse problems in imaging: A review. *IEEE Signal Processing Magazine* **34**(6), 85–95.
57. Mehri A, Ardakani PB and Sappa AD (2021) MPRNet: Multi-path residual network for lightweight image super resolution. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2704–2713.
58. Möckl L, Petrov PN and Moerner WE (2019) Accurate phase retrieval of complex 3D point spread functions with deep residual neural networks. *Applied Physics Letters* **115**(25), 251106.
59. Monga V, Li Y and Eldar YC (2021) Algorithm unrolling: Interpretable efficient deep, learning for signal and image processing. *IEEE Signal Processing Magazine* **38**(2), 18–44.
60. Mourya R, Denis L, Becker J-M and Thiébaut E (2015) A blind deblurring and image decomposition approach for astronomical image restoration. In *2015 23rd European Signal Processing Conference (EUSIPCO)*. IEEE, pp. 1636–1640.
61. Nah S, Kim TH and Lee KM (2017) Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3883–3891.
62. Nehme E, Freedman D, Gordon R, Ferdman B, Weiss LE, Alalouf O, Naor T, Orange R, Michaeli T and Shechtman Y (2020) DeepSTORM3D: Dense 3D localization microscopy and PSF design by deep learning. *Nature Methods* **17**(7):734–740.
63. Noll RJ (1976) Zernike polynomials and atmospheric turbulence. *Journal of the Optical Society of America* **66**(3), 207–211.
64. Noroozi M, Chandramouli P and Favaro P (2017) Motion deblurring in the wild. In *German Conference on Pattern Recognition*. Springer, pp. 65–77.
65. Ongie G, Jalal A, Metzler CA, Baraniuk RG, Dimakis AG and Willett R (2020) Deep learning techniques for inverse problems in imaging. *IEEE Journal on Selected Areas in Information Theory* **1**(1), 39–56.
66. Pan J, Hu Z, Su Z and Yang M-H (2014) Deblurring text images via l0-regularized intensity and gradient prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2901–2908.
67. Pan J, Sun D, Pfister H and Yang M-H (2016) Blind image deblurring using dark channel prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1628–1636.
68. Pankajakshan P, Zhang B, Blanc-Féraud L, Kam Z, Olivo-Marin J-C and Zerubia J (2009) Blind deconvolution for thin-layered confocal imaging. *Applied Optics* **48**(22), 4437–4448.
69. Perrone D and Favaro P (2014) Total variation blind deconvolution: the devil is in the details. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2909–2916.
70. Peyrin F, Toma A, Sixou B, Denis L, Burghardt A and Pialat J-B (2015) Semi-blind joint super-resolution/segmentation of 3D trabecular bone images by a tv box approach. In *2015 23rd European Signal Processing Conference (EUSIPCO)*. IEEE, pp. 2811–2815.

https://doi.org/10.1017/S2633903X24000096 Published online by Cambridge University Press

71. Pham T, Soubies E, Ayoub A, Lim J, Psaltis D and Unser M (2020) Three-dimensional optical diffraction tomography with Lippmann-Schwinger model. *IEEE Transactions on Computational Imaging* **6**, 727–738.
72. Sage D, Donati L, Soulez F, Fortun D, Schmit G, Seitz A, Guiet R, Vonesch C and Unser M (2017) DeconvolutionLab2: An open-source software for deconvolution microscopy. *Methods* **115**, 28–41.
73. Saha D, Schmidt U, Zhang Q, Barbotin A, Hu Q, Ji N, Booth MJ, Weigert M and Myers EW (2020) Practical sensorless aberration estimation for 3D microscopy with deep learning. *Optics Express* **28**(20), 29044–29053.
74. Sarder P and Nehorai A (2006) Deconvolution methods for 3D fluorescence microscopy images. *IEEE signal processing magazine* **23**(3), 32–45.
75. Schuler CJ, Hirsch M, Harmeling S and Schölkopf B (2015) Learning to deblur. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **38**(7), 1439–1451.
76. Shajkofci A and Liebling M (2020) DeepFocus: a few-shot microscope slide auto-focus using a sample invariant CNN-based sharpness function. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*. IEEE, pp. 164–168.
77. Shajkofci A and Liebling M (2020) Spatially-variant CNN-based point spread function estimation for blind deconvolution and Depth estimation in optical microscopy. *IEEE Transactions on Image Processing* **29**, 5848–5861.
78. Soulez F, Denis L, Tourneur Y and Thiébaut E (2012) Blind deconvolution of 3D data in wide field fluorescence microscopy. In *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*. IEEE, pp. 1735–1738.
79. Soulez F and Unser M (2016) Superresolution with optically-motivated blind deconvolution. In *Laser Applications to Chemical Security and, Environmental Analysis*. Optical Society of America, **180**, 2520.
80. Stockham TG, Cannon TM and Ingebretsen RB (1975) Blind deconvolution through digital signal processing. *Proceedings of the IEEE* **63**(4), 678–692.
81. Sun J, Cao W, Xu Z and Ponce J (2015) Learning a convolutional neural network for non-uniform motion blur removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 769–777.
82. Sun L, Cho S, Wang J and Hays J (2013) Edge-based blur kernel estimation using patch priors. In *IEEE International Conference on Computational Photography (ICCP)*. IEEE, pp. 1–8.
83. Venkatakrishnan SV, Bouman CA and Wohlberg B (2013) Plug-and-play priors for model based reconstruction. In *2013 IEEE Global Conference on Signal and Information Processing*. IEEE, pp. 945–948
84. von Diezmann A, Lee MY, Lew MD and Moerner WE (2015) Correcting field-dependent aberrations with nanoscale accuracy in three-dimensional single-molecule localization microscopy. *Optica* **2**(11), 985–993.
85. Wang Y, Wang H, Li Y, Hu C, Yang H and Gu M (2022) High-accuracy, direct aberration determination using self-attention-armed deep convolutional neural networks. *Journal of Microscopy* **286**(1), 13–21.
86. Zachevsky I and Zeevi YY (2018) Blind deblurring of natural stochastic textures using an anisotropic fractal model and phase retrieval algorithm. *IEEE Transactions on Image Processing* **28**(2), 937–951.
87. Zhang KZ, Li Y, Zuo W, Zhang L, Van Gool L and Timofte R (2022) Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **44**(10), 6360–6376.
88. Zhang M, Fang Y, Ni G and Zeng T (2022) Pixel screening based intermediate correction for blind deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5892–5900.
89. Zhang R, Isola P, Efros AA, Shechtman E and Wang O (2018) The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595.

Cite this article: Debarnot V & Weiss P (2024). Deep-blur: Blind identification and deblurring with convolutional neural networks. *Biological Imaging*, 4: e13. doi:https://doi.org/10.1017/S2633903X24000096