

1. Feb. 2024

Preprint-Series: Department of Mathematics - Applied Mathematics

ECALL: Expectation-calibrated learning for
unsupervised blind deconvolution

M. Haltmeier, G. Hwang

AppliedMathematics

Technikerstraße 13 - 6020 Innsbruck - Austria
Tel.: +43 512 507 53803 Fax: +43 512 507 53898
<https://applied-math.uibk.ac.at>

ECALL: Expectation-calibrated learning for unsupervised blind deconvolution

Markus Haltmeier

Department of Mathematics, University of Innsbruck
Technikerstrasse 13, 6020 Innsbruck, Austria
E-mail: markus.haltmeier@uibk.ac.at

Gyeongha Hwang

Department of Mathematics, Yeungnam University
280 Daehak-Ro, Gyeongsan, Gyeongbuk 38541, South Korea
E-mail: ghhwang@yu.ac.kr

January 29, 2024

Abstract

Blind deconvolution aims to recover an original image from a blurred version in the case where the blurring kernel is unknown. It has wide applications in diverse fields such as astronomy, microscopy, and medical imaging. Blind deconvolution is a challenging ill-posed problem that suffers from significant non-uniqueness. Solution methods therefore require the integration of appropriate prior information. Early approaches rely on hand-crafted priors for the original image and the kernel. Recently, deep learning methods have shown excellent performance in addressing this challenge. However, most existing learning methods for blind deconvolution require a paired dataset of original and blurred images, which is often difficult to obtain. In this paper, we present a novel unsupervised learning approach named ECALL (Expectation-CALibrated Learning) that uses separate unpaired collections of original and blurred images. Key features of the proposed loss function are cycle consistency involving the kernel and associated reconstruction operator, and terms that use expectation values of data distributions to obtain information about the kernel. Numerical results are used to support ECALL.

Key words: Inverse problems, unsupervised learning, unknown forward operator, blind deconvolution.

MSC codes: 65F22; 68T07

1 Introduction

Blind deconvolution addresses the problem of recovering the original unknown potentially vector valued image $\mathbf{x}: \mathbb{R}^d \rightarrow \mathbb{R}^c$ from its blurry and noisy observation

$$\mathbf{y} = \mathbf{k}^* * \mathbf{x} + \boldsymbol{\delta}, \quad (1.1)$$

where $\mathbf{k}^*: \mathbb{R}^d \rightarrow \mathbb{R}$ is the unknown convolution kernel and $\boldsymbol{\delta}$ is the unknown noise. Since the kernel needs to be at least partially estimated along with the recovery of the original image, the problem (1.1) is severely underdetermined and ill-posed. Thus, strong additional assumptions on the original image and the kernel are required for a proper solution. While classical variational methods have been developed for this purpose [3, 22, 14, 23], in this work we introduce a novel unsupervised learning approach to tackle this challenging problem.

Among others, blind deconvolution is relevant in medical imaging, astronomy, microscopy, signal processing, radar imaging, remote sensing, and computer vision (see [14] and references therein). In astronomy, for example, blind deconvolution is used to improve the resolution of images obtained by telescopes. In medical imaging, it is used to remove blur caused by motion or the imaging process. In microscopy, the image of a specimen is often blurred due to imperfections in the optics, aberrations, and scattering of light. Non-blur deconvolution is simpler and better understood, but requires precise knowledge of the blurring kernel. In the above applications, however, the kernel is often at least partially unknown.

1.1 Prior work

Blind deconvolution has been studied extensively for several decades. Due to its highly ill-posed nature, blind deconvolution is usually approached with strong additional assumptions on \mathbf{x} and \mathbf{k}^* (see [4, 7, 12, 13, 16, 17, 32, 26, 37, 38, 39, 22]). Typically, \mathbf{x} is assumed to have a sparse representation in some basis or dictionary, and \mathbf{k}^* is assumed to be a smooth function or to have a compact support. In general, variational regularization [30] provides a general framework for integrating prior information in blind deconvolution. However, they typically use hand-crafted priors that are not well represented in natural images. They also require time-consuming iterative minimization and may suffer from local minima.

Deep learning based approaches have recently shown excellent results for blind deconvolution tasks [2, 8, 15, 24, 25, 27, 33, 31, 34, 40]. Most of these approaches are supervised and require a paired dataset of original and blurred images. Existing methods fall into two broad categories: Two-step approaches with kernel estimation in a first step, and end-to-end approaches for estimating only the \mathbf{x} . See [41] for a detailed review of such supervised

learning approaches in blind deconvolution. Paired data however is in many cases difficult to obtain.

In contrast, unsupervised methods overcome the challenging problem of collecting corresponding pairs of original and blurred images. For example, in [28] a method inspired by the principles of Deep Image Prior (DIP) and Double-DIP [6, 1, 35] is presented. In [18], a method is proposed to approximate the maximum a posteriori estimate of the blurring kernel using Monte Carlo sampling. Another approach [21] is based on Generative Adversarial Networks (GAN) [5]. While GAN-based methods suffer from the instability of GAN training, the DIP-based method suffers from long inference time and sensitivity to hyperparameters.

1.2 Main contributions

In this paper, we present a novel simple unsupervised learning approach to blind deconvolution that overcomes the above problems. Specifically, we consider blind deconvolution with an unknown kernel \mathbf{k}^* when separate unpaired collections of original and blurred images are given. It is proposed to estimate an unknown kernel \mathbf{k}^* and the reconstruction operator $\mathbf{R}^*: \mathbf{y} \rightarrow \mathbf{x}$ via unsupervised learning by comparing expectations of the Fourier transform of given separate unpaired collections of original and blurred images along with a cycle consistency term. Our method is particularly useful in various applications where the acquisition of a paired dataset is challenging. As our main contribution, we motivate and introduce a novel loss function that estimates the kernel and reconstruction operator. The loss function includes a cycle constancy term, which requires \mathbf{R}^* to be a near-inverse of the convolution with \mathbf{k}^* , and an expectation calibration term to constrain \mathbf{k}^* . We refer to the proposed method as ECALL (Expectation-CALibrated Learning). We present numerical results that demonstrate the success of ECALL.

1.3 Outline

The rest of the paper is organized as follows: In Section 2 we present the problem formulation and our proposed unsupervised learning method ECALL for blind deconvolution. In particular, the design of the proposed loss function is presented in detail. Numerical aspects and numerical results are presented in Section 3. The paper ends with conclusions and an outlook on future work presented in Section 4.

2 Proposed ECALL method

2.1 Problem formulation

We consider the blind deconvolution problem (1.1) in the case when \mathbf{x} and $\boldsymbol{\delta}$ are independent random variables in $L^2(\mathbb{R}^d)$ modelling original images and noise, respectively, $\mathbf{k}^* \in L^1(\mathbb{R}^d)$ is a deterministic kernel and \mathbf{y} are the given data representing noisy blurred images. We assume the noise to have zero mean. We write \mathcal{F} and \mathcal{F}^{-1} for the Fourier transform on \mathbb{R}^d and its inverse, and write $\mathbb{E}[\cdot]$ for the expectation. At some places we will use the abbreviation $\widehat{\mathbf{x}} := \mathcal{F}(\mathbf{x})$. We assume that the convolution is well defined by $\mathcal{F}(\mathbf{k}^* * \mathbf{x}) = \mathcal{F}(\mathbf{k}^*) \cdot \mathcal{F}(\mathbf{x})$ and given by point-wise multiplication in Fourier space.

Let $p_{\mathbf{x}}$, $p_{\mathbf{y}}$ and $p_{\boldsymbol{\delta}}$ be the distributions of \mathbf{x} , \mathbf{y} and $\boldsymbol{\delta}$, respectively. Our goal is to find the kernel \mathbf{k}^* together with a reconstruction operator $\mathbf{R}^*: \mathbf{y} \rightarrow \mathbf{x}$ mapping noisy blurred images to original ones based on (1.1). More precisely, we consider the following problem.

Problem 2.1 (Unsupervised blind deconvolution). *For given distributions $p_{\mathbf{x}}$, $p_{\mathbf{y}}$ and $p_{\boldsymbol{\delta}}$ subject to model (1.1) determine the kernel \mathbf{k}^* and the reconstruction operator*

$$\mathbf{R}^* := \arg \min_{\mathbf{R}} \mathbb{E}[\|\mathbf{R}(\mathbf{k}^* * \mathbf{x} + \boldsymbol{\delta}) - \mathbf{x}\|^2]. \quad (2.1)$$

Since, in practice, only a finite data set can be collected, we will actually address the following empirical version.

Problem 2.2 (Empirical unsupervised blind deconvolution). *For given unpaired collections of original images $(\mathbf{x}_i)_{i=1,\dots,N} \sim p_{\mathbf{x}}$, data $(\mathbf{y}_i)_{i=1,\dots,N} \sim p_{\mathbf{y}}$ and noises $(\boldsymbol{\delta}_i)_{i=1,\dots,N} \sim p_{\boldsymbol{\delta}}$, estimate the kernel \mathbf{k}^* and the reconstruction operator \mathbf{R}^* defined by (2.1).*

Note that in Problem 2.2 we assume no dependence among $(\mathbf{x}_i)_{i=1}^N$, $(\mathbf{y}_i)_{i=1}^N$ and $(\boldsymbol{\delta}_i)_{i=1}^N$.

Remark 2.3. *In the supervised learning setting [36] the ideal reconstruction operator \mathbf{R}^* is given as the minimizer of the expected loss $\mathbb{E}[\|\mathbf{R}(\mathbf{y}) - \mathbf{x}\|^2]$. Its implementation, however, requires paired data, which we don't have. If \mathbf{k}^* would be known, we could create paired data using samples of \mathbf{x} and the model $\mathbf{k}^* * \mathbf{x} + \boldsymbol{\delta}$. Our strategy will therefore include kernel estimation, which will then allow the creation of virtual supervised training data based on the kernel estimate.*

To solve Problem 2.1, we set up an expected loss functional that determines \mathbf{k}^* based on $p_{\mathbf{x}}$ and $p_{\mathbf{y}}$ (expectation calibration) and \mathbf{R}^* based on virtual supervised data using the estimated kernel. To stabilize the whole process, we integrate several additional terms. In particular, the joint estimation of \mathbf{k}^* and \mathbf{R}^* improves the estimation of each. Problem

2.2 then uses the empirical version of the expected loss together with specific network architectures for approximating (k^*, \mathbf{R}^*) .

2.2 Theory

Let us start with the theoretical Problem 2.1. Our ECALL method is motivated by the following simple result, which constitutes a constructive uniqueness theorem for the problem of determining (k^*, \mathbf{R}^*) .

Theorem 2.4 (Uniqueness results). *Suppose $\mathbb{E}[\widehat{\mathbf{x}}](\xi) \neq 0$ for almost every $\xi \in \text{supp}(\widehat{\mathbf{k}}^*)$. Then $(\mathbf{k}^*, \mathbf{R}^*)$ is the unique solution of*

$$\mathbf{k}^* = \arg \min_{\mathbf{k}} \|\mathbb{E}[\widehat{\mathbf{y}}] - \mathbb{E}[\mathcal{F}(\mathbf{k} * \mathbf{x})]\|_1 \quad (2.2)$$

$$\mathbf{R}^* = \arg \min_{\mathbf{R}} \mathbb{E} \|\mathbf{x} - \mathbf{R}(\mathbf{k}^* * \mathbf{x} + \delta)\|_2^2. \quad (2.3)$$

Proof. By the convolution theorem and equation (1.1) we have $\widehat{\mathbf{y}} = \widehat{\mathbf{k}}^* \cdot \widehat{\mathbf{x}} + \widehat{\delta}$. By applying expectation values, using that \mathbf{k}^* is a deterministic quantity, and that the noise has zero mean and is independent of the clean image, we obtain (2.2). Having determined the kernel \mathbf{k}^* , identity (2.3) is then simply the definition of \mathbf{R}^* . \square

Based on Theorem 2.4, we set up a loss function that includes terms for estimating the kernel \mathbf{k}^* and terms for estimating the reconstruction operator \mathbf{R}^* . To increase accuracy and stability, we introduce regularization. More specifically, we consider

$$\mathcal{L}(\mathbf{k}, \mathbf{R}) = \mathcal{L}_A(\mathbf{k}) + \mathcal{L}_B(\mathbf{k}, \mathbf{R}) + \mathcal{L}_C(\mathbf{k}, \mathbf{R}) \quad (2.4)$$

where the three terms \mathcal{L}_A , \mathcal{L}_B , \mathcal{L}_C are described next.

- (A) EXPECTATION CALIBRATION: Based on (2.2) we introduce a term for determining the kernel \mathbf{k}^* . Under the assumption that $\mathbb{E}[\widehat{\mathbf{x}}](\xi) \neq 0$, minimizing the term $\|\mathbb{E}[\widehat{\mathbf{y}}] - \mathbb{E}[\mathcal{F}(\mathbf{k} * \mathbf{x} + \delta)]\|_1$ is theoretically sufficient for that purpose. However, when $\mathbb{E}[\widehat{\mathbf{x}}]$ is close to zero, then the estimation becomes unstable. We thus add a second term involving expected values of Fourier magnitudes. Thus we take

$$\mathcal{L}_A(\mathbf{k}) = \lambda_{A,1} \|\mathbb{E}[\widehat{\mathbf{y}}] - \mathbb{E}[\mathcal{F}(\mathbf{k} * \mathbf{x} + \delta)]\|_1 + \lambda_{A,2} \|\mathbb{E}[|\widehat{\mathbf{y}}|] - \mathbb{E}[|\mathcal{F}(\mathbf{k} * \mathbf{x} + \delta)|]\|_1. \quad (2.5)$$

Loss (2.5) is the main term allowing to estimate the unknown convolution kernel. While theoretically the first term would be sufficient the second term turns out to significantly increase stability.

(B) **CYCLE CONSISTENCY:** Based on (2.3) we construct a term to obtain the reconstruction operator \mathbf{R}^* . If the exact kernel $\mathbf{k} = \mathbf{k}^*$ would be known, the term $\mathbb{E}[\|\mathbf{x} - \mathbf{R}(\mathbf{k} * \mathbf{x} + \boldsymbol{\delta})\|^2]$ would be sufficient for that purpose. However, as the kernel is estimated simultaneously with the reconstruction operator we found that adding $\mathbb{E}[\|\mathbf{y} - \mathbf{k} * (\mathbf{R}\mathbf{y})\|^2]$ significantly improves results and, in particular, also stabilizes the kernel estimation. Thus we consider the term

$$\mathcal{L}_B(\mathbf{k}, \mathbf{R}) = \lambda_{B,1} \mathbb{E}[\|\mathbf{y} - \mathbf{k} * (\mathbf{R}\mathbf{y})\|^2] + \lambda_{B,2} \mathbb{E}[\|\mathbf{x} - \mathbf{R}(\mathbf{k} * \mathbf{x} + \boldsymbol{\delta})\|^2]. \quad (2.6)$$

It implements the reconstruction property (2.3) together with data consistency. Making \mathcal{L}_B small requires $\mathbf{k} * (\mathbf{R}\mathbf{y}) \simeq \mathbf{y}$ and $\mathbf{R}(\mathbf{k} * \mathbf{x}) \simeq \mathbf{x}$ and thus resembles the cycle loss commonly considered for unpaired image-to-image translation [42].

(C) **REGULARIZATION:** To avoid overfitting, we consider simple L^2 -regularization $\|\mathbf{k}\|_2^2$. To stabilize estimating the reconstruction operator, we add the difference between the expectations of \mathbf{x} and $\mathbf{R}\mathbf{y}$ resulting in

$$\mathcal{L}_C(\mathbf{k}, \mathbf{R}) = \lambda_{C,1} \|\mathbf{k}\|_2^2 + \lambda_{C,2} \|\mathbb{E}[\mathbf{R}\mathbf{y}] - \mathbb{E}[\mathbf{x}]\|_2^2. \quad (2.7)$$

Together with the expectation calibration (2.5) and cycle consistency (2.6) the regularization term (2.7) forms the loss function (2.4).

2.3 Empirical estimation

Next we turn over to the practically more important Problem 2.2 of blind deconvolution using empirical data (\mathbf{x}_i) , (\mathbf{y}_i^δ) and $(\boldsymbol{\delta}_i)$. For that purpose we replace all expectation values in (2.4) by empirical counterparts. The resulting functional is minimized over a parameterized class of convolution kernels $(\mathbf{k}_w)_{w \in W}$ for the kernel and a convolutional neural network $(\mathbf{R}_\theta)_{\theta \in \Theta}$ for the reconstruction operator. Thus we consider

$$\mathcal{L}^N(w, \theta) = \mathcal{L}_A^N(w) + \mathcal{L}_B^N(w, \theta) + \mathcal{L}_C^N(w, \theta) \quad (2.8)$$

where

$$\begin{aligned} \mathcal{L}_A^N(w) &= \frac{\lambda_{A,1}}{N} \left\| \sum_{i=1}^N \widehat{\mathbf{y}}_i^\delta - \sum_{i=1}^N \mathcal{F}(\mathbf{k}_w * \mathbf{x}_i + \delta_{\sigma(i)}) \right\|_1 \\ &\quad + \frac{\lambda_{A,2}}{N} \left\| \sum_{i=1}^N |\widehat{\mathbf{y}}_i^\delta| - \sum_{i=1}^N |\mathcal{F}(\mathbf{k}_w * \mathbf{x}_i + \delta_{\sigma(i)})| \right\|_1 \\ \mathcal{L}_B^N(w, \theta) &= \frac{\lambda_{B,1}}{N} \sum_{i=1}^N \left\| \mathbf{y}_i^\delta - \mathbf{k}_w * (\mathbf{R}_\theta \mathbf{y}_i^\delta) \right\|_2^2 \end{aligned} \quad (2.9)$$

$$+ \frac{\lambda_{B,2}}{N} \sum_{i=1}^N \left\| \mathbf{x}_i - \mathbf{R}_\theta(\mathbf{k}_w * \mathbf{x}_i + \delta_{\sigma(i)}) \right\|_2^2 \quad (2.10)$$

$$\mathcal{L}_C^N(w, \theta) = \lambda_{C,1} \|w\|_2^2 + \frac{\lambda_{C,2}}{N} \left\| \sum_{i=1}^N \mathbf{x}_i - \sum_{i=1}^N \mathbf{R}_\theta(\mathbf{y}_i^\delta) \right\|_2^2. \quad (2.11)$$

Here $(\delta_{\sigma(i)})$ denotes random sampling from the given collection of noises.

Details on the implementation and the network architecture are given below.

3 Numerical simulations

In this section, we present details of implementation and experimental results. We will work with color images with size of 256×256 having three color channels. The convolution kernel \mathbf{k}^* is approximated by a parameterised family of convolutions $(\mathbf{k}_w)_{w \in W}$ realised as CNN with one convolution layer with kernel size 31×31 . The entries of the kernel are taken as parameters of the network. The reconstruction operator \mathbf{R}^* is approximated by the U-Net architecture $(\mathbf{R}_\theta)_{\theta \in \Theta}$ which has proven to be highly effective for a variety of image processing tasks [29]. The U-Net structure used in our study is shown in Figure 3.1.

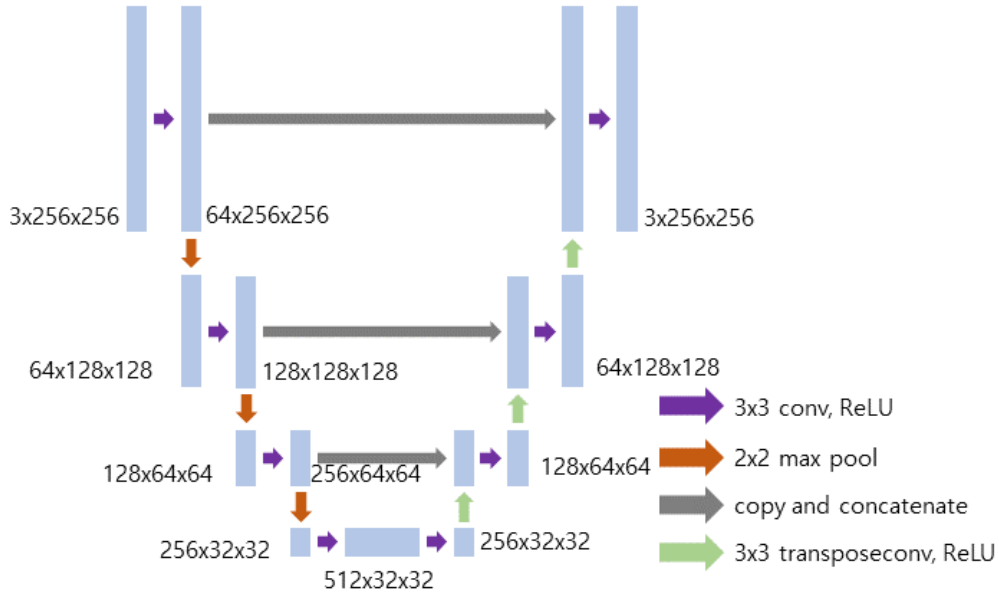


Figure 3.1: U-net architecture $(\mathbf{R}_\theta)_{\theta \in \Theta}$ for representing the reconstruction operator \mathbf{R}^* used in all presented numerical results.

3.1 Datasets and data generation

We perform experiments using $3 * 10^3$ images of the FFHQ dataset [10] where 10^3 samples are taken as original images, other 10^3 images to generate data samples and another 10^3 images as taken as test set.

Blurry images are created by applying three Gaussian filters, referred to as Broad, Medium, and Narrow. The window size of each Gaussian filter is truncated to have size 31. The Gaussian kernels are centered at zero and have different standard deviation, which determines the degree of blurring applied to the image. The Broad filter has the smallest standard deviation of 0.5, which will result in a mild blurring effect. The Medium filter has a standard deviation of 1, which will result in a moderate blurring effect. Finally, the Narrow filter has the largest standard deviation of 2, which will result in a strong blurring effect. For the noisy data, we add additive Gaussian white noise with standard deviation equal to 1% of the maximal value of the original data. The Fourier transform in the loss function is realized by the FFT applied with periodic padding [19].

3.2 Training

For minimizing (2.8) we use the AdamW optimizer [20] which is a stochastic optimization method that modifies the conventional weight decay implementation in the Adam [11] by decoupling weight decay from the gradient update process. The batch size for the neural network \mathbf{k}_w is taken as 10^3 . The large batch helps to evaluate and update the expectation effectively. We train the neural network \mathbf{R}_θ using a batch size of 2, which helps to reduce the gradient variance and improve model generalization.

The specific training strategy based on (2.8) consists of three phases:

[Phase 1] We first train the kernel \mathbf{k}_w with the objective function (2.8) using coefficients $\lambda_{A,1} = 1$, $\lambda_{C,1} = 5$ and $\lambda_{A,2} = \lambda_{B,1} = \lambda_{B,2} = \lambda_{C,2} = 0$ with the learning rate of 10^{-3} for \mathbf{k}_w . The iteration number is 10^3 .

[Phase 2] Train the neural networks \mathbf{k}_w and \mathbf{R}_θ with the objective function (2.8) which the coefficients $\lambda_{A,1} = \lambda_{A,2} = 10$, $\lambda_{B,1} = \lambda_{B,2} = 1$, $\lambda_{C,1} = 5$ and $\lambda_{C,2} = 10$ with the learning rate of 10^{-4} for \mathbf{k}_w and 10^{-3} for \mathbf{R}_θ . The iteration number is 10^4 .

[Phase 3] In the final fine-tuning phase we train the neural network \mathbf{R}_θ with the objective function (2.8) with $\lambda_{A,1} = \lambda_{A,2} = \lambda_{B,1} = \lambda_{C,1} = 0$, $\lambda_{B,2} = 1$ and $\lambda_{C,2} = 10$ with the learning rate of 10^{-3} for \mathbf{R}_θ . The iteration number is 10^4 .

To prevent getting stuck in a local minimum, we modify \mathcal{L}_A^N as

$$\begin{aligned} \mathcal{L}_A^N(w) = & \frac{\lambda_{A,1}}{N} \left\| \sum_{i=1}^N \chi \odot \widehat{\mathbf{y}}_i^\delta - \sum_{i=1}^N \chi \odot \mathcal{F}(\mathbf{k}_w * \mathbf{x}_i + \delta_{\sigma(i)}) \right\|_1 \\ & + \frac{\lambda_{A,2}}{N} \left\| \sum_{i=1}^N |\chi \odot \widehat{\mathbf{y}}_i^\delta| - \sum_{i=1}^N |\chi \odot \mathcal{F}(\mathbf{k}_w * \mathbf{x}_i + \delta_{\sigma(i)})| \right\|_1 \end{aligned} \quad (3.1)$$

where χ is a random mask that sets 20% of the pixels to zero and keeps the other pixels unchanged, and \odot denotes the Hadamard product.

3.3 Results

The experiments for proposed unsupervised blind deconvolution are made with noiseless data as well as noisy data. Kernel estimation is the most challenging part. Due to the empirical data this task in noiseless and noisy case suffers from inexact data; see (2.9). For comparison purpose we also present results with supervised learning where we minimise the supervised loss

$$\begin{aligned} \mathcal{L}_{N,\text{super}}(w, \theta) := & \frac{1}{N} \sum_{i=1}^N \|\mathbf{k}_w(\mathbf{x}_i) - (\mathbf{k}^* * \mathbf{x}_i + \delta_i)\|_2^2 \\ & + \frac{1}{N} \sum_{i=1}^N \|\mathbf{x}_i - \mathbf{R}_\theta(\mathbf{k}^* * \mathbf{x}_i + \delta_i)\|_2^2 + 5\|\mathbf{k}_w\|_2^2, \end{aligned}$$

using the same architectures as in the unsupervised case, learning rates of 10^{-4} and 10^{-3} for \mathbf{k}_w and \mathbf{R}_θ respectively, and iteration number $2 * 10^4$.

The estimated kernels in the supervised and unsupervised (ECALL) case are shown in Table 1. Both methods are able to well recover the kernel especially the broad and medium kernel. Table 2 shows typical example of corresponding blind deconvolution of the images.



















	Noiseless			Noisy		
	Broad	Medium	Narrow	Broad	Medium	Narrow
Ground Truth						
Supervised						
ECALL						

Table 1: Ground truth kernels (top) and reconstructed kernels using the supervised (middle) and proposed unsupervised (bottom) learning.

















































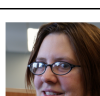





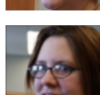





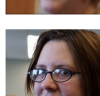





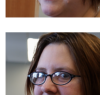





	Noiseless			Noisy		
	Broad	Medium	Narrow	Broad	Medium	Narrow
Original						
Observed						
Supervised						
ECALL						
Original						
Observed						
Supervised						
ECALL						
Original						
Observed						
Supervised						
ECALL						

Table 2: Blind deconvolution results using the supervised and proposed unsupervised learning.

For quantitative evaluation of the kernel estimation, we use the relative L^2 -norm error $L2err := \|\mathbf{k}_w - \mathbf{k}^*\|_2 / \|\mathbf{k}^*\|_2$ and the maximum of normalized convolution [9] $MNC := \max[\mathbf{k}_w * \mathbf{k}^* / (\|\mathbf{k}_w\|_2 \|\mathbf{k}^*\|_2)]$. The quantitative analysis of the kernel estimation is shown in Table 3. To evaluate the quality of the deconvolved images we use structural similarity index measure (SSIM) and peak signal-to-noise ratio (PSNR). Results are shown in Tables 3 and 4. Overall, we can see that ECALL performs comparably to supervised learning.

	Noiseless			Noisy		
	Broad	Medium	Narrow	Broad	Medium	Narrow
Measurements	L2err ↓ MNC↑	L2err ↓ MNC↑	L2err ↓ MNC↑	L2err ↓ MNC↑	L2err ↓ MNC↑	L2err ↓ MNC↑
Supervised	0.0765 0.9584	0.0129 0.9950	0.0029 0.9937	0.0765 0.9589	0.0124 0.9926	0.0050 0.9892
ECALL	0.0434 0.9991	0.0224 0.9997	0.0119 0.9999	0.0571 0.9984	0.0598 0.9982	0.0879 0.9962

Table 3: Quantitative evaluation of kernel estimation using L2err (relative L^2 -error) and MNC (maximum of normalized convolution). The arrows \uparrow and \downarrow indicate whether higher or lower values indicate better performance.

	Noiseless			Noisy		
	Broad	Medium	Narrow	Broad	Medium	Narrow
Measurements	SSIM↑ PSNR↑	SSIM↑ PSNR↑	SSIM↑ PSNR↑	SSIM↑ PSNR↑	SSIM↑ PSNR↑	SSIM↑ PSNR↑
Supervised	0.9437 33.43dB	0.8636 29.42dB	0.7080 25.11dB	0.9065 31.70dB	0.8129 27.98dB	0.6684 24.31dB
ECALL	0.9435 33.49dB	0.8549 29.17dB	0.6956 24.88dB	0.9061 31.67dB	0.8133 28.03dB	0.6720 24.31dB

Table 4: Quantitative evaluation of reconstruction results using SSIM and PSNR. The arrows \uparrow and \downarrow indicate whether higher or lower values indicate better performance.

4 Conclusion

Blind deconvolution consists in recovering an original image \mathbf{x} from its noisy blurred observation $\mathbf{y} = \mathbf{k}^* * \mathbf{x} + \boldsymbol{\delta}$. Blind deconvolution is a highly ill-posed problem, since both the original image \mathbf{x} and the kernel \mathbf{k}^* need to be estimated. To address the ill-posed nature of the problem, blind deconvolution is typically approached with additional prior information on \mathbf{x} and \mathbf{k}^* . Recently, deep learning-based approaches have shown great promise in blind

deconvolution, but most of these approaches require paired datasets of original and blurred images, which are often difficult to obtain. To address this, in this work we propose an unsupervised learning approach (ECALL) to blind deconvolution that uses separate unpaired collections of original and blurred images. Experimental results demonstrate the feasibility and performance of the proposed algorithm.

In short, the main novelty of ECALL is to construct special statistics using the distributions of \mathbf{x} and \mathbf{y} , respectively, which allows us to determine the kernel from which we can then create virtual supervised data pairs. In particular, we use the expectation of the Fourier transform and its absolute value for kernel estimation. There are a number of interesting follow-ups to our work, including generalisation to other inverse problems with unknown forward operator, development of better statistics, analysis of the influence of empirical data and architecture on the estimates, and comparison with GAN-based methods for blind image deconvolution.

5 Acknowledgments

G. Hwang was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (NRF-2021R1F1A1048120).

References

- [1] Daniel Otero Baguer, Johannes Leuschner, and Maximilian Schmidt. Computed tomography reconstruction using deep image prior and learned reconstruction methods. *Inverse Problems*, 36(9):094004, 2020.
- [2] Ayan Chakrabarti. A neural approach to blind motion deblurring. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part III 14*, pages 221–235. Springer, 2016.
- [3] Tony F Chan and Chiu-Kwong Wong. Total variation blind deconvolution. *IEEE transactions on Image Processing*, 7(3):370–375, 1998.
- [4] Sunghyun Cho and Seungyong Lee. Fast motion deblurring. In *ACM SIGGRAPH Asia 2009 papers*, pages 1–8. 2009.
- [5] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018.

- [6] Yosef Gandelsman, Assaf Shocher, and Michal Irani. “Double-DIP”: Unsupervised image decomposition via coupled deep-image-priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11026–11035, 2019.
- [7] Dong Gong, Mingkui Tan, Yanning Zhang, Anton Van den Hengel, and Qinfeng Shi. Blind image deconvolution by automatic gradient activation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1827–1836, 2016.
- [8] Dong Gong, Jie Yang, Lingqiao Liu, Yanning Zhang, Ian Reid, Chunhua Shen, Anton Van Den Hengel, and Qinfeng Shi. 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*, pages 2319–2328, 2017.
- [9] Zhe Hu and Ming-Hsuan Yang. Good regions to deblur. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7–13, 2012, Proceedings, Part V 12*, pages 59–72. Springer, 2012.
- [10] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.
- [11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [12] Dilip Krishnan and Rob Fergus. Fast image deconvolution using hyper-laplacian priors. *Advances in neural information processing systems*, 22, 2009.
- [13] Dilip Krishnan, Terence Tay, and Rob Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR 2011*, pages 233–240. IEEE, 2011.
- [14] Deepa Kundur and Dimitrios Hatzinakos. Blind image deconvolution. *IEEE signal processing magazine*, 13(3):43–64, 1996.
- [15] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8183–8192, 2018.
- [16] Anat Levin, Yair Weiss, Fredo Durand, and William T Freeman. Understanding and evaluating blind deconvolution algorithms. In *2009 IEEE conference on computer vision and pattern recognition*, pages 1964–1971. IEEE, 2009.

- [17] Anat Levin, Yair Weiss, Fredo Durand, and William T Freeman. Efficient marginal likelihood optimization in blind deconvolution. In *CVPR 2011*, pages 2657–2664. IEEE, 2011.
- [18] Ji Li, Yuesong Nan, and Hui Ji. Un-supervised learning for blind image deconvolution via monte-carlo sampling. *Inverse Problems*, 38(3):035012, 2022.
- [19] Renting Liu and Jiaya Jia. Reducing boundary artifacts in image deconvolution. In *2008 15th IEEE International Conference on Image Processing*, pages 505–508. IEEE, 2008.
- [20] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- [21] Boyu Lu, Jun-Cheng Chen, and Rama Chellappa. Unsupervised domain-specific deblurring via disentangled representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10225–10234, 2019.
- [22] Xiao-Guang Lv, Fang Li, and Tiejong Zeng. Convex blind image deconvolution with inverse filtering. *Inverse Problems*, 34(3):035003, 2018.
- [23] Antonio Marquina. Nonlinear inverse scale space methods for total variation blind deconvolution. *SIAM Journal on Imaging Sciences*, 2(1):64–83, 2009.
- [24] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3883–3891, 2017.
- [25] Mehdi Noroozi, Paramanand Chandramouli, and Paolo Favaro. Motion deblurring in the wild. In *Pattern Recognition: 39th German Conference, GCPR 2017, Basel, Switzerland, September 12–15, 2017, Proceedings 39*, pages 65–77. Springer, 2017.
- [26] Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. Blind image deblurring using dark channel prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1628–1636, 2016.
- [27] Sainandan Ramakrishnan, Shubham Pachori, Aalok Gangopadhyay, and Shanmuganathan Raman. Deep generative filter for motion deblurring. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 2993–3000, 2017.
- [28] Dongwei Ren, Kai Zhang, Qilong Wang, Qinghua Hu, and Wangmeng Zuo. Neural blind deconvolution using deep priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3341–3350, 2020.

- [29] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015.
- [30] Otmar Scherzer, Markus Grasmair, Harald Grossauer, Markus Haltmeier, and Frank Lenzen. *Variational methods in imaging*, volume 167. Springer, 2009.
- [31] Christian J Schuler, Michael Hirsch, Stefan Harmeling, and Bernhard Schölkopf. Learning to deblur. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1439–1451, 2015.
- [32] Qi Shan, Jiaya Jia, and Aseem Agarwala. High-quality motion deblurring from a single image. *Acm transactions on graphics (tog)*, 27(3):1–10, 2008.
- [33] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 769–777, 2015.
- [34] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8174–8182, 2018.
- [35] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9446–9454, 2018.
- [36] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1999.
- [37] David Wipf and Haichao Zhang. Revisiting bayesian blind deconvolution. *Journal of Machine Learning Research (JMLR)*, 2014.
- [38] Li Xu and Jiaya Jia. Two-phase kernel estimation for robust motion deblurring. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part I 11*, pages 157–170. Springer, 2010.
- [39] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1107–1114, 2013.
- [40] Jiawei Zhang, Jinshan Pan, Jimmy Ren, Yibing Song, Linchao Bao, Rynson WH Lau, and Ming-Hsuan Yang. Dynamic scene deblurring using spatially variant recurrent

neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2521–2529, 2018.

- [41] Kaihao Zhang, Wenqi Ren, Wenhan Luo, Wei-Sheng Lai, Björn Stenger, Ming-Hsuan Yang, and Hongdong Li. Deep image deblurring: A survey. *International Journal of Computer Vision*, 130(9):2103–2130, 2022.
- [42] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.