Plug-and-play approach to class-adapted blind image deblurring

Marina Ljubenović¹ · Mário A. T. Figueiredo¹

Received: 16 January 2018 / Revised: 31 January 2019 / Accepted: 20 February 2019 / Published online: 2 March 2019
© The Author(s) 2019

Abstract

Most of the existing single-image blind deblurring methods are tailored for natural images. However, in many important applications (e.g., document analysis, forensics), the image being recovered belongs to a specific class (e.g., text, faces, fingerprints) or contains two or more classes. To deal with these images, we propose a class-adapted blind deblurring framework, based on the plug-and-play scheme, which allows forward models of imaging systems to be combined with state-of-the-art denoisers. We consider three patch-based denoisers, two suitable for images that belong to a specific class and a general purpose one. Additionally, for images with two or more classes, we propose two approaches: a direct one, and one that uses a patch classification step before denoising. The proposed deblurring framework includes two priors on the blurring filter: a sparsity-inducing prior, suitable for motion blur and a weak prior, for a variety of filters. The results show the state-of-the-art performance of the proposed framework when applied to images that belong to a specific class (text, face, fingerprints), or contain two classes (text and face). For images with two classes, we show that the deblurring performance is improved by using the classification step. For these images, we choose to test one instance of the proposed framework suitable for text and faces, which is a natural test ground for the proposed framework. With the proper (dictionary and/or classifier) learning procedure, the framework can be adapted to other problems. For text images, we show that, in most cases, the proposed deblurring framework improves OCR accuracy.

Keywords Blind deblurring · Class-specific image priors · Plug-and-play · ADMM · Patch-based image processing

1 Introduction

Blind image deblurring (BID) is an inverse problem where an observed image is usually modeled as resulting from the convolution with a blurring filter, often followed by additive noise, and the goal is to estimate both the blurring filter and the underlying sharp image. As there is an infinite number of solutions, the problem is severely ill-posed. Furthermore, since the convolution operator is itself typi-

cally ill-conditioned, the problem is sensitive to inaccurate blurring filter estimates and the presence of noise.

The single-image blind deblurring problem has been widely investigated in recent years, mostly considering generic natural images [13,18,23–25,37,41]. To deal with the ill-posed nature of BID, most methods use priors on both the blurring filter and the sharp image. Much of the success of the state-of-the-art algorithms can be ascribed to the use of image priors that exploit the statistics of natural images [13,18,23,25,31,41] and/or are based on restoration of salient edges [5,42,48,49]. Previous research in natural image statistics has shown that images of real-world scenes obey heavy-tailed distributions in their gradients [18,19]. However, in many applications, the image being recovered belongs to some specific class (e.g., text, faces, fingerprints, medical structures) or contains two or more classes (e.g., document images that contain text and faces). For these types of images, which in some cases do not follow the heavy-tailed gradient statistics of natural images (e.g., text [12]), methods that use priors capturing the properties of a specific class of images are more likely to provide better results (e.g., text

---

The research leading to these results has received funding from the European Union's H2020 Framework Programme (H2020-MSCA-ITN-2014) under Grant Agreement No. 642685 MacSeNet and was partially supported by the Fundação para a Ciência e Tecnologia, Grant UID/EEA/5008/2013.

✉ Marina Ljubenović
marina.ljubenovic@lx.it.pt

Mário A. T. Figueiredo
mario.figueiredo@lx.it.pt

¹ Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal

[12,26,36], or face images [32,35]). Furthermore, images that belong to different specific classes may have different characteristics that are challenging to capture with a unique prior. For instance, text and face images have completely different structures, namely sharp transitions in text versus smooth and textured areas in faces. For images that contain two or more classes, we should consider a prior which is appropriate for all existing classes present in those images.

We propose a method that uses patch-based image priors learned off-line from sets of clean images of specific classes of interest. The proposed priors can be learned from a set that contains just one class (e.g., text) or from two or more sets that contain different classes (e.g., text and faces). The method is based on the so-called *plug-and-play* approach, recently introduced in [46], which allows forward models of imaging systems to be combined with state-of-the-art denoising methods. In particular, we use denoisers based on Gaussian mixture models (GMM) or dictionaries, learned from patches of clean images from specific classes. These types of denoisers can be used for images that contain only one class, as presented in [27], or two or more classes [28].

In addition to a class-specific image prior, we consider different priors on the blurring filter. Earlier methods typically impose constraint for the, arguably, most relevant type of generic motion blurs using sparsity priors [10,13,18,20,41,47]. To impose sparsity, in this work, we use the ℓ_1 -norm, to tackle motion blur. Furthermore, previous works have shown that good results can be obtained by using weak priors (limited support) on the blurring filter [5,27]. The same prior is used here, next to the ℓ_1 -norm, for a variety of blurring filters.

Finally, we should clarify the motivation behind considering images that contain two classes (e.g., identification cards and magazines). Concerning how realistic motion (or out-of-focus) blur scenarios are, when acquiring images of identification cards (or other documents), it suffices to think of the very common current practice of photographing documents with a handheld mobile phone camera. Unlike a desktop scanner or fixed-camera-based document acquisition system, a handheld mobile phone camera, nowadays so commonly used to photograph all sorts of documents for personal use, may yield poorly focused images, especially in low-light situations. Consequently, this type of scenario is realistic and worthy of consideration. Nevertheless, the main goal of our work is to introduce the modular framework itself that provides different possibilities according to the problem at hand.

1.1 Related work and contributions

Although most state-of-the-art BID methods are tailored to natural images, recent years have witnessed significant advances in class-specific single-image deblurring, mostly considering just one class of interest. Text image deblur-

ring has attracted considerable attention, mostly due to its wide range of applications [12,29,36]. For example, for document images, Chen et al. [11] propose a new prior based on the image intensity instead of the widely used heavy-tailed gradient prior of natural scenes; nevertheless, the proposed black-and-white document image deblurring method is limited in that it cannot handle images with complex backgrounds. Cho et al. develop a method to incorporate the specific properties of text images (i.e., a sharp contrast between text and background, uniform gradient within text, and background gradient following natural image statistics) for BID [12]. The performance of the method strongly depends on the accuracy of the *stroke width transform* (SWT), which classifies the image into text and non-text regions. Pan et al. propose a ℓ_0 regularizer based on intensity and gradient, for text image deblurring [36]; the proposed method can also be effectively applied to non-document text images and low-illumination scenes with saturated regions. The same authors go beyond text images by proposing a method suitable for both text and natural images [38]. Both methods [36] and [38] show limited performance when dealing with text images corrupted with strong and/or unknown noise. Recently, with the development of deep neural networks, new methods arise for solving BID of text images [22].

In numerous applications, images contain human faces. Although, faces belong to the class of natural images, they can also be considered as a part of a specific class due to their specific structure. In this case, there is not much texture in the blurred image, and thus, methods that rely on an implicit or explicit restoration of salient edges for kernel estimation are less effective as only a few edges can be used. For face image deblurring, a few methods have been proposed with the goal of increasing the performance of face recognition algorithms [32,51]. Zhang et al. [51] propose a joint image restoration and recognition method, based on a sparse representation prior suitable only for simple motion blurs. Recently, HaCohen et al. [21] propose a deblurring method with exemplars, reference images with the same content as the input images. That approach, although giving state-of-the-art results, can be used in a limited number of applications, depending on an available image dataset. Pan et al. [35] use a similar approach, but do not require the exemplars to be similar to the input. The blurred face image can be of a different person and have a different background compared to any exemplar images.

Instead of focusing only on one class of images, a few recent methods exploit class-specific image priors suitable for different image classes for various restoration tasks. Niknejad et al. introduce a new image denoising method, tailored to specific classes of images, for Gaussian and Poisson noise, based on an importance sampling approach [33,34]. Similarly, in [8], Anwar et al. use an external database of

images that belong to the specific class to extract the set of “support patches” used in the restoration process. Remez et al. approach class-aware image denoising by using deep neural networks [39]; they first perform image classification to classify images into classes, followed by denoising. Teodoro et al. propose using a Gaussian mixture model as a class-specific patch-based prior, for solving two image inverse problems, namely non-blind deblurring and compressive imaging [45]. In prior work, we have shown that class-specific GMM-based image prior can be used for BID [27]. That approach allows handling situations where the image being processed contains regions of different classes, as done by Teodoro et al. for denoising and non-blind deblurring [44]. Additionally, we show that a similar framework with a dictionary-based prior can be used for BID when multiple classes are present in the same image [28]. Anwar et al. exploit the potential of class-specific image priors for recovering spatial frequencies attenuated by the blurring process, by capturing the Fourier magnitude spectrum of an image class across all frequency bands [7] and that method achieves state-of-the-art results for images that belong to some specific classes (e.g., faces, animals, common objects), but there is no straightforward extension of the method for images that contain two or more classes.

As mentioned above, in [27], we introduced the class-adapted blind deblurring method based on the *plug-and-play* approach and GMM-based denoiser. That method is suitable for images that belong to a specific class (text, faces, or fingerprints) and a variety of blurring filters. Furthermore, in [28], we show how the previous method can be extended to tackle images that contain two classes and introduce different class-specific denoisers based on dictionaries trained from a clean image set. This paper combines and significantly extends the previous work reported in [27] and [28]. Namely, we introduce the patch classification step, yielding a simultaneous segmentation/deblurring method, which exploits the synergy between these two tasks: by identifying the most likely type of contents of each patch, the most adequate denoiser is used at that location, while a better deblurred image facilitates segmentation. Additionally, the experimental results reported in this paper considerably exceed those reported in [27] and [28].

In this work, we present a method based on the *plug-and-play* framework [46], which has two central features: (i) simplified and modular algorithmic integration and (ii) possibility of using state-of-the-art denoising methods that are not explicitly formulated as optimization problems. Furthermore, the proposed method handles images that belong to a specific class (e.g., text, faces, fingerprints), but also images that contain two or more image classes with significantly different structures (e.g., text and faces). We also show that by using a classification step, the results can be slightly improved. The approach may be seen as simultane-

ously performing segmentation and restoration, representing a step forward toward overcoming the gap between image restoration and analysis. As one of the steps of the proposed algorithm is a strong denoiser, the proposed method can handle images of a certain class (e.g., text) corrupted with high noise levels. In addition, with a weak prior on the blurring filter, the proposed method can be used for a variety of blurring filters.

The main contributions of the proposed work can be summarized as follows:

- (1) The modular BID framework itself. One of the reasons behind the modularity of the proposed framework is the use of the *alternating direction method of multipliers* (ADMM), an optimization algorithm that will be explained in more detail below.
- (2) The proposed framework can be used for different image classes having different properties. It is able to handle challenging text images, with a narrow space between letters or corrupted with large blurs. We show how class-aware denoisers yield better results when compared with generic ones.
- (3) It can be easily extended for images that contain two (or more) classes that have a different structure. We show how the classification step can be inserted into the framework, thus improving the results and representing a way of combining image restoration and analysis. Additionally, to the best of our knowledge, this is the first paper to deal with blind deblurring of images that contain two classes.
- (4) The method is able to tackle text images corrupted with strong Gaussian noise or with unknown noise (non-Gaussian) present in some real blurred images.

1.2 Outline

In Sect. 2, we briefly introduce the observation model. Sections 2.1 and 2.2 review the *alternating direction method of multipliers* (ADMM) and the *plug-and-play* framework. After presenting the necessary building blocks, Sect. 3 introduces our approach to class-adapted BID for images with one or more specific classes. The experimental evaluation of our method is reported in Sect. 4, and Sect. 5 concludes the manuscript.

2 Observation model

We assume the standard linear observation model, formulated as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \quad (1)$$

where $\mathbf{y} \in \mathbb{R}^n$ is a vector containing all the pixels (lexicographically ordered) of the observed image, $\mathbf{H} \in \mathbb{R}^{n \times n}$ is a matrix representing the convolution of a vectorized unknown original image $\mathbf{x} \in \mathbb{R}^n$ with an unknown blur operator $\mathbf{h}$, followed by additive noise $\mathbf{n}$, assumed to be Gaussian, with zero mean and known variance σ^2 .

The image $\mathbf{x}$ and the blur operator $\mathbf{H}$ (equivalently, the blurring filter $\mathbf{h}$) are estimated by solving the optimization problem with cost function

$$O_{\lambda, \gamma}(\mathbf{x}, \mathbf{h}) = \frac{1}{2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \lambda \Phi(\mathbf{x}) + \gamma \Psi(\mathbf{h}), \quad (2)$$

where the function Φ embodies the prior used to promote characteristics that the original sharp image is assumed to have and the function Ψ represents the prior on the blurring filter. Regularization parameters, $\lambda \geq 0$ and $\gamma \geq 0$, control the trade-off between the data-fidelity term and the regularizers.

As (2) is a non-convex objective function, one way to tackle it is by alternating estimation of the underlying image and the blurring filter [4,5], as shown in Algorithm 1. Both steps in Algorithm 1 (lines 3 and 4) are performed by using the *alternating direction method of multipliers* (ADMM) [5].

Algorithm 1 Generic BID Algorithm

```

1: Initialization: Set  $\hat{\mathbf{x}} = \mathbf{y}$ ,  $\hat{\mathbf{h}}$  as the identity filter,  $\lambda \geq 0$ , and  $\gamma \geq 0$ 
2: repeat
3:    $\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} O_{\lambda, \gamma}(\mathbf{x}, \hat{\mathbf{h}})$ 
4:    $\hat{\mathbf{h}} = \underset{\mathbf{h}}{\operatorname{argmin}} O_{\lambda, \gamma}(\hat{\mathbf{x}}, \mathbf{h})$ 
5: until stopping criterion is satisfied

```

2.1 ADMM

As discussed in the previous section, to estimate both the image and the blurring filter, we use the ADMM algorithm, which we now briefly review. Consider the following unconstrained optimization problem,

$$\min_{\mathbf{z}} f_1(\mathbf{z}) + f_2(\mathbf{z}). \quad (3)$$

Variable splitting is a simple procedure where a new variable $\mathbf{v}$ is introduced as the argument of f_2 , under the constraint $\mathbf{z} = \mathbf{v}$, i.e., the problem is written as

$$\min_{\mathbf{z}, \mathbf{v}} f_1(\mathbf{z}) + f_2(\mathbf{v}) \quad \text{subject to} \quad \mathbf{z} = \mathbf{v}. \quad (4)$$

The rationale is that it may be easier to solve the constrained problem (4) than its equivalent unconstrained counterpart

(3). One of the ways to tackle (4) is by forming the so-called augmented Lagrangian

$$L_{\rho}(\mathbf{z}, \mathbf{v}, \mathbf{d}) = f_1(\mathbf{z}) + f_2(\mathbf{v}) + \mathbf{d}^T(\mathbf{z} - \mathbf{v}) + \frac{\rho}{2} \|\mathbf{z} - \mathbf{v}\|_2^2, \quad (5)$$

and applying ADMM [9], which consists of alternating minimization of (5) with respect to $\mathbf{z}$ and $\mathbf{v}$ and updating the vector of Lagrange multipliers $\mathbf{d}$; in (5), $\rho \geq 0$ represents the penalty parameter. In summary, the steps of ADMM are as shown in Algorithm 2.

Algorithm 2 ADMM

```

1: Initialization: Set  $k = 0$ ,  $\rho > 0$ , initialize  $\mathbf{v}_0$  and  $\mathbf{d}_0$ 
2: repeat
3:    $\mathbf{z}_{k+1} \leftarrow \underset{\mathbf{z}}{\operatorname{argmin}} f_1(\mathbf{z}) + \frac{\rho}{2} \|\mathbf{z} - (\mathbf{v}_k + \mathbf{d}_k)\|_2^2$ 
4:    $\mathbf{v}_{k+1} \leftarrow \underset{\mathbf{v}}{\operatorname{argmin}} \frac{\rho}{2} \|(\mathbf{z}_{k+1} - \mathbf{d}_k) - \mathbf{v}\|_2^2 + f_2(\mathbf{v})$ 
5:    $\mathbf{d}_{k+1} \leftarrow \mathbf{d}_k - (\mathbf{z}_{k+1} - \mathbf{v}_{k+1})$ 
6:    $k \leftarrow k + 1$ 
7: until stopping criterion is satisfied

```

Recalling that the *proximity operator* (PO) of a convex function g , computed at $\mathbf{p}$, is defined as [14]

$$\operatorname{prox}_g(\mathbf{p}) = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{x} - \mathbf{p}\|_2^2 + g(\mathbf{x}), \quad (6)$$

it is clear that in Algorithm 2, lines 3 and 4 are the PO of functions f_1 and f_2 , computed at $\mathbf{v}_k + \mathbf{d}_k$ and $\mathbf{z}_{k+1} - \mathbf{d}_k$, respectively. Furthermore, a PO can be interpreted as a denoising operation, with its argument as the noisy observation and g as the regularizer.

2.2 Plug-and-play

As shown above, line 4 of Algorithm 2 can be seen as the solution to a denoising problem, suggesting that instead of using the PO of a convex regularizer, we can use a state-of-the-art denoiser. This approach, known as *plug-and-play* (PnP), was recently introduced [46]. In this work, instead of using standard denoisers, such as BM3D [16] or K-SVD [17], we consider patch-based class-adapted denoisers tailored to images that contain one or more specific classes (e.g., text, faces, fingerprints).

2.3 GMM-based denoiser

In recent years, it has been shown that state-of-the-art denoising results can be achieved by methods that use probabilistic patch models based on Gaussian mixtures [50,52]. Zoran et al. show that clean image patches are well modeled by a Gaussian mixture model (GMM) estimated from a collection of clean images using the expectation-maximization

(EM) algorithm [52]. Furthermore, for a GMM-based prior, the corresponding *minimum mean squared error* (MMSE) estimate can be obtained in closed form [43].

If we consider that $\mathbf{x}$ in (1) denotes one of the image patches and $\mathbf{H} = \mathbf{I}$ (where $\mathbf{I}$ is the identity matrix), the MMSE estimate of $\mathbf{x}$ is well known to be the posterior expectation,

$$\hat{\mathbf{x}} = \mathbb{E}(\mathbf{x}|\mathbf{y}) = \int \mathbf{x} \frac{p_{Y|X}(\mathbf{y}|\mathbf{x})p_X(\mathbf{x})}{p_Y(\mathbf{y})} d\mathbf{x}. \quad (7)$$

In (7), $p_{Y|X}(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}; \mathbf{x}, \sigma^2 \mathbf{I})$ represents a Gaussian density, with mean $\mathbf{x}$ and covariance matrix $\sigma^2 \mathbf{I}$, and $p_X(\mathbf{x})$ is a prior density on the clean patch.

A GMM has the form

$$p_X(\mathbf{x}|\theta) = \sum_{i=1}^K \omega_i \mathcal{N}(\mathbf{x}; \mu_i, \mathbf{C}_i), \quad (8)$$

where μ_i and $\mathbf{C}_i$ are the mean and covariance of the i -th component, respectively, ω_i is its weight ($\omega_i \geq 0$ and $\sum_{i=1}^K \omega_i = 1$), and $\theta = \{\mu_i, \mathbf{C}_i, \omega_i, i = 1, \dots, K\}$. GMM priors have the important feature that the MMSE estimate under Gaussian noise has a simple closed form expression

$$\hat{\mathbf{x}} = \sum_{i=1}^n \xi_i(\mathbf{y}) v_i(\mathbf{y}), \quad (9)$$

where

$$\xi_i(\mathbf{y}) = \frac{\omega_i \mathcal{N}(\mathbf{y}; \mu_i, \mathbf{C}_i + \sigma^2 \mathbf{I})}{\sum_{j=1}^K \omega_j \mathcal{N}(\mathbf{y}; \mu_j, \mathbf{C}_j + \sigma^2 \mathbf{I})}, \quad (10)$$

and

$$v_i(\mathbf{y}) = \left(\mathbf{C}_i^{-1} + \frac{1}{\sigma^2} \mathbf{I} \right)^{-1} \left(\mathbf{C}_i^{-1} \mu_i + \frac{1}{\sigma^2} \mathbf{y} \right). \quad (11)$$

We use the above-mentioned facts to obtain a GMM-based prior learned from a set of clean images that belong to a specific class. The rationale behind this approach is that the class-adapted image prior may achieve better performance than a fixed, generic denoiser, when processing images that do belong to that specific class (e.g., text, faces, fingerprints).

2.4 Dictionary-based denoiser

Several patch-based image denoising methods work by finding a sparse representation of the image patches in a learned dictionary [3,17,30,40].

As in the previous section, if we consider the linear model in (1) with $\mathbf{H} = \mathbf{I}$, we have a denoising problem. Furthermore, $\mathbf{x} \in \mathbb{R}^n$ is constructed from image patches of size

$\sqrt{n} \times \sqrt{n}$ pixels, ordered as column vectors and a dictionary $\mathbf{D} \in \mathbb{R}^{n \times k}$ (usually with $k \geq n$), assumed to be known and fixed. A dictionary-based denoising model suggests that every patch, $\mathbf{x}$, can be described by a sparse linear combination of dictionary atoms (columns of $\mathbf{D}$), i.e., as the solution of

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\boldsymbol{\alpha}\|_0 \text{ subject to } \mathbf{D}\boldsymbol{\alpha} \approx \mathbf{x}. \quad (12)$$

The so-called l_0 -norm stands for the count of nonzero elements in $\boldsymbol{\alpha}$.

The dictionary $\mathbf{D}$ can be learned (and/or updated) from an observed image itself, but it can also be learned from a set of clean images [3]. Similarly, like for Gaussian mixtures, instead of learning a dictionary from a set of clean generic images, we learn it from a set that contains clean images of a specific class. Furthermore, we can learn different dictionaries from sets that contain two or more image classes. After learning the dictionaries, we can combine them to tackle images that contain two or more image classes.

2.5 Standard denoisers

In order to avoid learning, one may choose to use a standard general purpose denoiser instead of class-specific denoisers. BM3D is a state-of-the-art denoising method based on collaborative filtering in 3D transform domain, combining sliding-window transform processing with block-matching [16]. The method is based on non-local similarity of image patches and thus can be used for any image that contains this structure.

3 Proposed method

In the proposed method, the image estimation problem in line 3 of Algorithm 1 is solved by using the PnP-ADMM algorithm described in the previous sections.

3.1 Image estimation

The image estimation problem (line 3 of Algorithm 1) can be formulated as

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \lambda \Phi(\mathbf{x}). \quad (13)$$

Furthermore, this problem can be written in the form (3) by setting $f_1(\mathbf{x}) = \frac{1}{2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2$ and $f_2(\mathbf{x}) = \lambda \Phi(\mathbf{x})$. The steps of the ADMM algorithm applied to problem (13) are then

$$\begin{aligned}
\mathbf{x}_{k+1} &\leftarrow \operatorname{argmin}_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{H} \mathbf{x}\|_2^2 + \frac{\rho}{2} \|\mathbf{x} - (\mathbf{v}_k - \mathbf{d}_k)\|_2^2 \\
\mathbf{v}_{k+1} &\leftarrow \operatorname{argmin}_{\mathbf{v}} \frac{\rho}{2} \|(\mathbf{x}_{k+1} - \mathbf{d}_k) - \mathbf{v}\|_2^2 + \lambda \Phi(\mathbf{v}) \\
\mathbf{d} &\leftarrow \mathbf{d}_{k+1} - (\mathbf{x}_{k+1} - \mathbf{v}_{k+1}).
\end{aligned} \quad (14)$$

The first step of ADMM in (14) is a quadratic optimization problem, which has a linear solution:

$$\mathbf{x}_{k+1} \leftarrow (\mathbf{H}^T \mathbf{H} + \rho \mathbf{I})^{-1} (\mathbf{H}^T \mathbf{y} + \rho(\mathbf{v}_k + \mathbf{d}_k)). \quad (15)$$

The matrix inversion in (15) can be efficiently computed in the discrete Fourier transform (DFT) domain in the case of cyclic deblurring, which we consider in this paper (for more information, see [1,2,5] and references therein). The second step is, as explained above, the PO of function Φ computed at $(\mathbf{x}_{k+1} - \mathbf{d}_k)$ and is replaced with one of the denoisers described above, following the PnP framework.

3.2 Blur estimation

The blur estimation problem (line 4 of Algorithm 1) can be formulated as

$$\hat{\mathbf{h}} = \operatorname{argmin}_{\mathbf{h}} \frac{1}{2} \|\mathbf{y} - \mathbf{X} \mathbf{h}\|_2^2 + \gamma \Psi(\mathbf{h}), \quad (16)$$

where $\mathbf{h} \in \mathbb{R}^n$ is the vector containing the blurring filter elements (lexicographically ordered) and $\mathbf{X} \in \mathbb{R}^{n \times n}$ is the matrix representing the convolution of the image $\mathbf{x}$ and the filter $\mathbf{h}$. As explained for image estimation, the problem can be written in the form (3) by setting $f_1(\mathbf{h}) = \frac{1}{2} \|\mathbf{y} - \mathbf{X} \mathbf{h}\|_2^2$ and $f_2(\mathbf{h}) = \gamma \Psi(\mathbf{h})$, leading to the following steps in the ADMM algorithm:

$$\begin{aligned}
\mathbf{h}_{k+1} &\leftarrow \operatorname{argmin}_{\mathbf{h}} \frac{1}{2} \|\mathbf{y} - \mathbf{X} \mathbf{h}\|_2^2 + \frac{\rho}{2} \|\mathbf{h} - (\mathbf{v}_k - \mathbf{d}_k)\|_2^2 \\
\mathbf{v}_{k+1} &\leftarrow \operatorname{argmin}_{\mathbf{v}} \frac{\rho}{2} \|(\mathbf{h}_{k+1} - \mathbf{d}_k) - \mathbf{v}\|_2^2 + \gamma \Psi(\mathbf{v}) \\
\mathbf{d} &\leftarrow \mathbf{d}_{k+1} - (\mathbf{h}_{k+1} - \mathbf{v}_{k+1}).
\end{aligned} \quad (17)$$

The first step in (17) has the same form as (15),

$$\mathbf{h}_{k+1} \leftarrow (\mathbf{X}^T \mathbf{X} + \rho \mathbf{I})^{-1} (\mathbf{X}^T \mathbf{y} + \rho(\mathbf{v}_k - \mathbf{d}_k)), \quad (18)$$

and, as previously explained, the matrix inversion can be efficiently computed in the DFT domain, using the fast Fourier transform (FFT). The second step in (17) depends on the choice of the regularizer. In this work, we use two types of regularizer on the blurring filter, which will be briefly explained next.

Fig. 1 Example of various blurring filters representing Gaussian, linear motion, out-of-focus, uniform, and nonlinear motion blur, respectively

3.2.1 Positivity and support

To cover a wide variety of blurring filters that have different characteristics (examples shown in Fig. 1), we simply assume the prior on the blur to be the indicator function $\Psi(\mathbf{h}) = \mathbb{1}_{S^+}(\mathbf{h})$, where S^+ is the set of filters with positive entries on a given support, that is,

$$\mathbb{1}_{S^+}(\mathbf{u}) = \begin{cases} 0 & \text{if } \mathbf{u} \in S^+ \\ \infty & \text{if } \mathbf{u} \notin S^+. \end{cases} \quad (19)$$

The rationale behind using a weak prior on the blurring filter is to cover a variety of filters with different characteristics. For example, a motion blur is sparse and as such can be described by a sparsity-inducing prior, but the same does not hold for out-of-focus or uniform blurs. To handle these variability, in this work we use a weak prior that covers common characteristics (positivity and limited support) of different types of blurring filters.

By introducing a weak constraint, which is an indicator function, the value of the regularization parameter γ becomes irrelevant, and the second step of ADMM (17) becomes

$$\mathbf{v}_{k+1} = \operatorname{argmin}_{\mathbf{v}} \frac{\rho}{2} \|(\mathbf{h}_{k+1} - \mathbf{d}_k) - \mathbf{v}\|_2^2 + \mathbb{1}_{S^+}(\mathbf{v}). \quad (20)$$

Problem (20) corresponds to the orthogonal projection on S^+ ,

$$\mathbf{v}_{k+1} = \operatorname{prox}_{\mathbb{1}_{S^+}}(\mathbf{u}) = P_{S^+}(\mathbf{u}), \text{ with } \mathbf{u} = (\mathbf{h}_{k+1} - \mathbf{d}_k), \quad (21)$$

which simply consists of setting zero to any negative entries and those outside the given support.

3.2.2 Sparsity-inducing prior

Generic motion blurs, which have a sparse support, can be tackled by encouraging sparsity using $\Psi(\mathbf{h}) = \gamma \|\mathbf{h}\|_1$, where $\|\cdot\|_1$ denotes the ℓ_1 -norm. Examples of motion blur filters, from a benchmark dataset [25], are shown in Fig. 2.

The second step of the ADMM algorithm in (17), in this case, becomes

$$\mathbf{v}_{k+1} = \operatorname{argmin}_{\mathbf{v}} \frac{\rho}{2} \|(\mathbf{h}_{k+1} - \mathbf{d}_k) - \mathbf{v}\|_2^2 + \gamma \|\mathbf{h}\|_1, \quad (22)$$

Fig. 2 Motion blur filters from the benchmark image dataset proposed by Levin et al. [25]

and corresponds to a *soft-thresholding* [1], e.g.,

$$\mathbf{v}_{k+1} = \text{soft}_{\rho}(\mathbf{h}_{k+1} - \mathbf{d}_k). \quad (23)$$

3.3 Two or more image classes

Additionally, in this work, we also consider images that contain two or more classes (e.g., document images that contain text and face), which can have completely different characteristics. For example, face images do not contain many strong edges and text images have specific structure due to the contents of interest being mainly in two tones. To tackle this problem, in the proposed BID framework, we use two approaches to perform dictionary-based denoising, which will be explained next.

3.3.1 Direct approach

To tackle a problem where we have two (or more) classes in an image, we can learn dictionaries from sets that contain images of these two (or more) different classes and then concatenate them. If we have two dictionaries learned from two classes (e.g., text and face), $\mathbf{D}_1 \in \mathbb{R}^{m \times k_1}$ and $\mathbf{D}_2 \in \mathbb{R}^{m \times k_2}$, where m represents size of vectorized image patch, and k_1 and k_2 , number of dictionary atoms (columns), a resulting dictionary, $\mathbf{D} \in \mathbb{R}^{m \times (k_1 + k_2)}$, will contain atoms that correspond to image patches from these two classes (e.g., a resulting dictionary will contain both text and face patches). To perform dictionary-based denoising, instead of using a dictionary trained from images of a single class, we use the resulting dictionary which combines images from two (or more) classes. Figure 3 shows parts of text and face dictionaries with noticeable differences between them.

To make it clearer, Fig. 4 illustrate the so-called direct approach to a dictionary-based denoising.

3.3.2 Patch classification

Instead of using the direct approach explained above to deal with images that contain regions from different classes,

FACE DICTIONARY

TEXT DICTIONARY

Fig. 3 Parts of learned dictionaries (100 atoms) from sets containing clean face and text images

we introduced a classification step before the patch-based denoising step. In the proposed procedure, each patch is first classified into one of the classes and then denoised with the corresponding dictionary.

Classification of image patches can be seen as performing a segmentation task, where we classify regions in the image that contain different classes (Fig. 5). Notice that image segmentation is not the main goal of the proposed method, and instead, it is a step that can boost the performance of BID.

To classify the patches, we consider two popular classification approaches:

- Each patch is classified using the *k-nearest neighbors* (kNN) classifier [6], which is a simple classical procedure where every patch is classified by a majority vote of its neighbors, with the patch being assigned to the class most common among its k -nearest neighbors.
- The second classifier tested in this work is a *support vector machine* (SVM) [15].

Both classification approaches mentioned above require a learning phase, which we perform using clean image patches from known classes.¹ Figure 6 illustrates the denoising part of the proposed framework with the classification step included. Note that instead of using dictionary-based denoising, we may choose GMM-based denoising. Again, for the direct approach, a new mixture can be formed from two (or more) class-adapted GMMs. In this work, we choose dictionary-based denoising simply because it is easier to implement.

4 Experiments and results

In all the experiments, we used the following settings for the two ADMM algorithms to perform image and blur estimation: (i) The image estimate is computed with 20 iterations,

¹ The classifier learning process is performed using the Matlab Toolbox: Classification Learner.

Fig. 4 Direct approach to dictionary-based denoising

Fig. 5 BID with a classification step performed on the image that contains text (black regions in the segmented image) and a face (white regions in the segmented image)

initialized with the estimate from the previous iteration, $\mathbf{d}_0 = 0$, and λ hand-tuned for the best visual results or best ISNR (improvement in SNR [5]: $\text{ISNR} = 10\log_{10}(\|\mathbf{x} - \mathbf{y}\|^2 / \|\mathbf{x} - \hat{\mathbf{x}}\|^2)$) on synthetic data; (ii) the blur estimate is obtained with 4 iterations, when using the weak prior (explained in Sect. 3.2.1), and 10 iterations, when using the sparsity prior (explained in Sect. 3.2.2), initialized with the blur estimate from the previous iteration, $\mathbf{d}_0 = 0$, and γ (with the ℓ_1 regularizer) hand-tuned for the best results. Default values of regularization parameters are set to $\lambda = 0.08$ (with $\rho = \lambda$ for the image estimate ADMM) and $\gamma = 0.05$ (with $\rho = 0.01$ for the blur estimate ADMM).²

Furthermore, we use three different datasets to perform the experiments and to train the GMMs and/or dictionaries:

- a dataset with 10 *text* images, available from Luo et al. [29] (one for testing and nine for training),
- a dataset with 100 *face* images from the same source as the text dataset (10 for testing and 90 for training),
- a dataset with 128 *fingerprints* from the publicly available UPEK database.

The GMM-based prior is obtained by using patches of size 6×6 pixels and a 20-component mixture. The dictionary-based prior is obtained by using the same size patches and the number of dictionary atoms is set to 1000, with 15 iterations

of the K-SVD algorithm. In all the experiments, the number of outer iterations is set to 100.

We compare our results with several state-of-the-art methods for natural images: Almeida et al. [5]; Krishnan et al. [23]; Xu et al. [48]; Xu et al. [49]; Pan et al. [37]. Almeida et al. tackle the realistic case of blind deblurring with unknown boundary conditions by using edge detectors to preserve important edges in the image. Krishnan et al. [23] use image regularization (ratio of the l_1 -norm to the l_2 -norm on the high frequencies of an image) that favors sharp images over blurry ones. Xu et al. [48,49] propose l_0 -based sparse representation of an intermediate image used to estimate the blurring kernel.³ Pan et al. [37] use the fact that a dark channel (smallest values in a local neighborhood) of blurred images is less sparse. Additionally, we compare our results with the following methods tailored for text images: Cho et al. [12] and Pan et al. [36]. Cho et al. [12] rely on specific properties of text images. Pan et al. [36] use an l_0 -based prior on the intensity and gradient for text image deblurring. Note that these text deblurring methods are not designed for images corrupted with strong or unknown noise.

We use several instances of the proposed framework: PlugBM3D refers to the proposed algorithm with the generic denoiser explained in Sect. 2.5; PlugGMM uses the class-specific GMM-based denoiser (Sect. 2.3); PlugDictionary uses a class-specific dictionary-based denoiser (Sect. 2.4) suitable for images that contain one or two classes. We will mention when PlugDictionary is used with the classification step.

4.1 Results: one class

For images that contain one class (e.g., text, faces, fingerprints), we performed several experiments with different types of blurs and different noise levels. To show that the proposed method can be used for various types of blurring filters, we created 10 test images containing text and faces

² Matlab demo code: <https://github.com/mljubenovic/Class-adapted-BID>.

³ Results of Xu et al. [48,49] obtained by using Robust Deblurring software available from the authors' webpage. Note that software related with the work proposed in [49] does not give the estimated kernel as output.

Fig. 6 Dictionary-based denoising with a classification step

Table 1 ISNR values (in dB) obtained by four algorithms: Almeida et al. [5], Krishnan et al. [23], PlugBM3D, and PlugGMM

Blur type	BSNR (dB)	TEXT				FACE			
		Generic [5]	Generic [23]	PlugBM3D	PlugGMM	Generic [5]	Generic [23]	PlugBM3D	PlugGMM
Gaussian	30	0.78	1.62	7.23	8.88	1.78	0.33	5.88	6.01
Linear motion	30	0.86	0.12	8.68	8.99	0.47	− 0.75	5.36	5.47
Out-of-focus	30	0.46	− 0.65	8.19	9.40	1.51	− 1.96	4.98	5.32
Uniform	30	0.79	− 0.45	8.94	11.48	0.11	− 0.20	3.66	3.56
Nonlinear motion	30	0.59	0.94	13.08	16.44	1.09	0.11	4.44	4.65
Gaussian	40	1.04	1.70	10.22	10.34	4.31	0.55	6.64	7.10
Linear motion	40	1.37	0.77	10.36	13.07	1.81	0.12	4.86	7.75
Out-of-focus	40	0.71	− 0.10	14.64	17.75	2.86	− 1.75	6.78	8.95
Uniform	40	0.98	− 0.50	15.63	12.99	0.85	− 1.06	8.50	7.07
Nonlinear motion	40	0.84	1.05	15.87	23.83	4.43	0.37	5.94	7.33

Bold values represent the best results for every category

(five of each) using one clean image of text or one clean face image, and 11×11 synthetic kernels that represent, respectively, Gaussian, linear motion, out-of-focus, uniform, and nonlinear motion blur, as shown in Fig. 1, and noise levels corresponding to *blurred signal to noise ratio* (BSNR) of 30 dB and 40 dB (Table 1). We compared our results with two generic methods by Almeida et al. [5] and Krishnan et al. [23]. Here, we tested two versions of the proposed algorithm: PlugBM3D and PlugGMM. The results in Table 1 show that our method outperforms state-of-the-art methods for generic images, when tested on images that belong to a specific class (text and face). Additionally, slightly better results are achieved with a class-specific denoiser plugged into the algorithm (PlugGMM), instead of the generic denoiser (PlugBM3D). Note that the generic algorithm of Almeida et al. [5] is designed for a wide variety of blur filters, while Krishnan et al. [23] is designed mostly for motion blurs.

Furthermore, to show that the proposed method can handle text images corrupted with strong noise, we created three test images of text corrupted with motion blur number 2 from [25] and three noise levels (BSNR = 40, 20, and 10

dB). Our results, presented in Fig. 7, are compared with the state-of-the-art method designed for text images by Pan et al. [36], and, again, we use two versions of the proposed method (PlugBM3D and PlugGMM). In these experiments, we use the ℓ_1 prior on the blurring filter to promote sparsity (as explained in Sect. 3.2.2). The results show that both versions of the proposed method are able to handle text images corrupted with different levels of noise. Slightly better results, in terms of the ISNR, are achieved by the class-adapted algorithm PlugGMM. The method of Pan et al. was originally designed for noise-free images and does not perform well on test images even with weak noise (BSNR = 40 dB).

Robustness of the proposed method to noise is shown in Fig. 8 (upper row). We created several experiments with two types of text images (typed and handwritten), corrupted with four different kernels (kernels 1 to 4 from [25]) and six noise levels corresponding to BSNR = 50, 40, 30, 20, 15, and 10 dB. We measure ISNR after 50 iterations of the PlugGMM method. We see that the proposed method performs stably for noise levels above 20 dB, reasonable well for BSNR = 15 dB, and fails for very high noise level, BSNR = 10 dB.

Fig. 7 Text image blurred with the motion blur number 2 from [25] and corrupted with three noise levels: BSNR = 40 dB, 20 dB, and 10 dB (from top to bottom); methods: Pan et al. [36] (ISNR = -2.66, -2.72, -5.34), PlugBM3D (ISNR = 14.40, 10.79, 5.76), and

PlugGMM (ISNR = 17.50, 12.78, 5.84). *Note:* Black squares in the upper right corners show the ground truth kernel (first column) and the estimated kernels by each method (third to fifth columns)

Fig. 8 Typed and handwritten text: final ISNR and regularization parameter λ as a function of different noise levels

Additionally, the proposed method is tested in the challenging case of a blurred fingerprint image. We choose to show the performance of our method on images containing fingerprints due to two reasons: i) Although rare, fingerprints can be found in old documents as a means of identification; ii) an image containing fingerprints has specific statistics greatly different than natural images, and as such is highly interesting as a testing ground. We created the experiment by using the

simplest case of motion blur: linear motion blur and weak noise (BSNR = 40 dB). Results of two versions of our algorithm (PlugBM3D and PlugGMM) are compared with the methods of Almeida et al. [5] and Krishnan et al. [23], constructed for generic images (Fig. 9). The results show that, due to the specific structure of images containing fingerprints, algorithms designed for generic images perform poorly. PlugBM3D manages to estimate the blurring filter

Fig. 9 Results obtained on the image containing fingerprint, corrupted with a linear motion blur and weak noise (BSNR = 40 dB); methods: Almeida et al. [5], ISNR = 0.36, Krishnan et al. [23], ISNR = 0.64, PlugBM3D, ISNR = 0.56, PlugGMM, ISNR = 1.19 Note: Black squares in the upper right corner of the images show the ground truth kernel (first image) and the estimated kernels by each method (third to sixth image)

closely; however, the resulting image still contains blurred regions (upper left part), while the proposed algorithm with a class-adapted image prior (PlugGMM) produces the best result, both visually and in terms of ISNR.

Next, we tested the proposed algorithm with the dictionary-based prior (PlugDictionary) on synthetic blurred text image from [12] (Fig. 10). The results show that the method of Almeida et al. [5] for generic images performs well, introducing small artifacts, similar to the method Cho et al. [12], specially constructed for text images. Good performance of the former method is most likely due to fact that it uses (Sobel-type) edge detectors to preserve important edges in the image. The generic method Krishnan et al. [23] introduces strong artifacts in the reconstructed image, and the method Pan et al. [36], designed for text images only, performs equally well as the proposed method, PlugDictionary, constructed for different image classes.

Figure 11 shows a challenging case of a blurred text with narrow spaces between letters (the test image is introduced in [12]). Results are compared with four generic methods: Xu et al. [49], Almeida et al. [5], Krishnan et al. [23], and Pan et al. [37]. All of these methods show poor results on the tested image. Pan et al. [36] show good visual results with still slightly blurry letter edges. Our method, PlugGMM, gives the sharpest visual result.

Fig. 10 Experiments on synthetic blurred image from [12]; methods: Almeida et al. [5], Krishnan et al. [23], Cho et al. [12], Pan et al. [36], PlugDictionary (text dictionary) Note: Black squares in the upper right part of the images represent the estimated kernels by each method

Fig. 11 Typed text with narrow spaces between letters and a challenging blurring kernel; methods: Xu et al. [49], Almeida et al. [5], Krishnan et al. [23], Pan et al. [37], Pan et al. [36], and PlugGMM Note: Black squares in the upper right corner of the images show the estimated kernels by each method

Fig. 12 Text image corrupted with a standard size blurring kernel (upper row) and a large blurring kernel (bottom row); Methods: Pan et al. [36] and PlugBM3D Note: Black squares in the upper right corner of the images show the estimated kernels by each method

To show that the method, when used on text images, is able to estimate a blurring kernel that has a large support (69×69 pixels), we perform several experiments and compare our results with the state-of-the-art method Pan et al. [36]. Here, we used PlugBM3D to show that even with this generic denoiser, we can achieve good performance (Fig. 12).

4.2 Results: two or more classes

To test the proposed method on images that contain two classes, we created an image with a face and typed text, corrupted by motion blur and different noise levels. The main reasons to choose text and face images are that these two classes are very common in some applications (e.g., identification documents in document analysis and forensics) and, second, these two classes have completely different structure (namely, sharp transitions in text, versus smooth and textured areas in faces), so they are a natural test ground for the proposed technique. Figure 13 shows the results obtained with this synthetic document image, corrupted by motion blur number 1 from [25], and lower noise (BSNR = 40 dB). Methods tailored for natural images [23,48,49] produce strong ringing artifacts and lose details in the part of the image that contains the face (e.g., neck-lace). The method of Pan et al. [36], constructed only for text images, performs reasonably well in the part of the image that contains faces, but gives a slightly spread kernel estimate and introduce artifacts in the part of the image that contains text. We used PlugDictionary with a sparsity-based kernel prior (Sect. 3.2.2). PlugDictionary is used with

Fig. 13 Experiments on synthetic blurred image that contains two classes—text and face; methods: Krishnan et al. [23], Xu et al. [48], Xu et al. [49], Pan et al. [36], Pan et al. [37], and PlugDictionary with dictionary constructed using the direct approach and a sparsity prior

on the blurring filter. Note: Black squares in the upper right corner of the images show the ground truth kernel (first image) and the estimated kernels by each method (other images)

Table 2 Results in terms of ISNR obtained on images that contain two classes: (i) text + face (gray scale image), (ii) text + face (RGB)

Images	BSNR (dB)	Direct	kNN	SVM
text + face (grayscale)	40	22.70	23.43	23.48
	30	13.64	13.98	14.30
	20	3.34	3.45	6.50
text + face (RGB)	40	19.42	19.93	20.03
	30	11.02	11.75	11.93
	20	4.07	4.21	4.27

Bold values represent the best results for every category

Methods: Direct—without classification step; kNN—with classification step based on kNN classifier 3.3.2; SVM—with classification step based on SVM classifier 3.3.2

the direct approach explained in Sect. 3.3.1. The result shows that the proposed method outperforms state-of-the-art BID methods for generic images [23,37,48,49] or text images [36].

Furthermore, the proposed method is tested on images that contain two classes (e.g., text and face) with the inclusion of the classification step, instead of using the direct approach. Results in terms of ISNR are presented in Table 2. We used two images that contain text and face (gray scale and RGB), blurred with motion blur number 2 from [25] and corrupted with noise with three levels, corresponding to BSNR = 40, 30, and 20 dB. We compare results of the proposed method with the direct approach and the proposed method with the classification step using kNN or SVM. The results show that the classification step improves the performance on images containing two classes. Additionally, for images with higher noise levels (e.g., BSNR = 20 dB), in some cases, significant improvement can be achieved by proper selection of the classifier and by including noisy patches (images) in the training data.

As mentioned above, the main focus of this work is on the BID performance, not the accuracy of the patch classifier. Still, it is interesting to observe some examples of classifiers (kNN and SVM) as here we see the potential for improvement of the proposed framework. Figure 14 shows results of BID with a classification step on the color image with text and face. Before the classification step, the color image is converted to gray scale. In this example, the segmented image clearly shows significant classification error in the upper region (blue stripe): Patches that should be classified as text are classified as a face. There are at least two possible explanations for this: 1) Both classifiers used in this work are trained only on text image patches that contain black text on white background; 2) the patch size (6×6 pixels) is small compared with the image size (532×345 pixels). The reason behind gray areas in the segmented image is that for this experiment, instead of classifying patches into two

Fig. 14 BID with a classification step (with the kNN classifier) performed on color image

Fig. 15 Segmentation results after using two patch classifiers (kNN and SVM) for images corrupted with the blur kernel number 1 from [25] and different noise levels: BSNR = 40 dB (upper row) and BSNR = 10 dB (bottom row)

classes (text and face), we use three classes (text, face, and other) to achieve more realistic classification. Although, in this example, the classification errors do not significantly influence the final deblurring result, it shows that there is space for improvement of the patch classifier. Finally, we show how the noise level influences the segmentation result. Figure 15 shows results after a patch classification step performed with different classifiers (kNN and SVM) on two images corrupted with the same blur kernel and different noise levels, BSNR = 40 and 10 dB. These results show that the SVM classifier performs slightly better than the kNN classifier in both cases, but still, the same classification error is present in the upper part of the image. Also, we can see that classifiers do not perform much worse when strong noise is present (BSNR = 10 dB), probably due to the fact that both classifiers are trained with patches corrupted with different noise levels.

4.3 Results: real blurred images

To experiment with real data, we first use a 248×521 image, acquired with a handheld mobile phone, corrupted by motion blur and unknown noise (Fig. 16). Note that we

use term “unknown noise” when we do not know whether noise present in an image is Gaussian and we do not know a noise level. We choose to use an image acquired by a mobile phone in order to test the robustness of the method with regard to an unknown type of noise. Before the deblurring process, to obtain high contrast, we preprocessed the image by computing its luminance component and setting all pixels with values above 0.8 to 1. (The range of pixel intensity values is from 0 to 1.) The preprocessed image is used as an input for all the methods, except the one of Pan et al. [37], because that method uses properties of an image dark channel that can be disturbed after preprocessing. We compared our results with four state-of-the-art methods: three designed for natural images [5,37,48] and one for text images [36]. The results show that the method of Almeida et al. [5] is able to estimate a reasonable blurring kernel, but the estimated image contains strong artifacts and noise. The method of Xu et al. [48] gives reasonably good results, although the estimated image still contains noise. The method Pan et al. [37], when used on an image that is not preprocessed, gives good result with some regions that are still blurred. The method Pan et al. [36] is not able to deal with unknown noise. Our method, PlugBM3D, provides the best visual result.

To emphasize the influence of unknown noise on the deblurring process, Fig. 17 shows the results obtained with the zoomed real blurred image (an image is zoomed before deblurring, introducing new artifacts). As before, the meth-

Fig. 16 Experiments on a real blurred image; methods: Almeida et al. [5], Xu et al. [48], Pan et al. [37] (preprocessed and original input image), Pan et al. [36], and PlugBM3D Note: Black squares in the upper right corner of the images show the estimated kernels by each method

Fig. 17 Experiments on a zoomed real blurred image; methods: Pan et al. [36], Xu et al. [49], Pan et al. [37], Almeida et al. [5], and PlugBM3D Note: Black squares in the upper right corner of the images show the estimated kernels by each method

Fig. 18 Experiments on a real blurred image with rich background; methods: Xu et al. [48], Xu et al. [49], Pan et al. [36], Pan et al. [37], and PlugGMM Note: black squares in the upper right corner of the images show the estimated kernels by each method

ods of Pan et al. [36] and Xu et al. [49] are not able to deal with unknown noise. The methods of Almeida et al. [5] and Pan et al. [37] estimate the blurring kernel realistically, but leaving some parts of the image blurred. Again, our method gives the best visual results. In experiments presented in Figs. 16 and 17, we used PlugBM3D as BM3D denoiser seems to be more suitable for images corrupted with noise that is not necessarily Gaussian.

Figure 18 shows the performance of the proposed method on a real blurred text image from [12]. Here, we use PlugGMM with a sparsity-based prior on the blurring kernel. Results are compared with three generic methods by Xu et al. [48], Xu et al. [49], and Pan et al. [37], and the text deblurring method by Pan et al. [36]. All methods perform reasonably well, but some of them introduce ringing artifacts.

The performance on the real blurred document images that contain text and face is shown in Figs. 19 and 20. We use real blurred images corrupted with two types of blur, motion and out-of-focus, and compare our results with a method tailored to natural images [23] and a method tailored to text images [36]. In addition to motion blur, the image in Fig. 19 contains some saturated pixels, which influences the performance of

Fig. 19 Experiments on a real blurred image corrupted by motion blur that contains text and face with rich background; Methods: Krishnan et al. [23], Pan et al. [35,36], and PlugDictionary with classification step (SVM classifier)

the evaluated methods: Krishnan et al. [23], Pan et al. [36], and PlugDictionary. Here, we use PlugDictionary with the classification step performed by an SVM, as explained in Sect. 3.3.2, and a weak prior on the blurring filter. In the case of motion blur, the method from Krishnan et al., tailored to natural images, introduces strong blocking artifacts, especially visible in the face region. Our method manages to better recover the part of the image with the numbers. For out-of-focus blur (Fig. 20), we can see slightly sharper details in the face region when we use PlugDictionary. All three methods fail to deblur small letters. Note that the images used in these experiments are very challenging due to several reasons: rich background, parts with very different statistics, and saturated pixels. Therefore, we choose to compare our results only with one alternative method tailored to natural images and one focused on text images.

Finally, Fig. 21 shows experiments performed on real blurred document (magazine) images that contain text and face. The image is acquired by the handheld mobile phone camera. As before, we used PlugDictionary with a classification step based on an SVM and a weak prior on the blurring kernel. The results are compared with four methods tailored to natural images, Krishnan et al. [23], Xu et al. [48,49] and Pan et al. [37], and the BID method for text images Pan et al. [36]. All methods perform reasonably well in the part of

Fig. 20 Experiments on a real blurred image corrupted by out-of-focus blur that contains text and face with rich background; methods: Krishnan et al. [23], Pan et al. [35,36], and PlugDictionary with classification step (SVM classifier)

the image containing a face, except that methods from Pan et al. [36,37] over-smooth it. In the part of the image containing text, all methods introduce, more or less, ringing artifacts taking into consideration that the proposed method arguably least affects its readability. We recommend zooming into the figure in order to clearly appreciate the differences.

Note that in Figs. 19, 20, and 21, we do not present the estimated blurring kernels. This is due to the images being mildly blurred and the obtained resulting blurring kernels only differ slightly.

4.4 OCR results

One of the main reason to do text deblurring is to improve OCR (optical character recognition) accuracy. As OCR software typically uses a language model to improve recognition, it needs continuous text as input. To evaluate OCR accuracy, we use five blurred text images from [22] (images 1–5 in Table 3), two that contain text and face corrupted with different noise levels (images 6–7) and a real blurred text image (image 8). We assess the quality of OCR with three measures: *average word confidence* (AWC), *word error rate* (WER), and *character error rate* (CER); all three measure are in the range from 0 to 1, with 1 as the best value for AWC and 0 as the best value for WER and CER. Table 3 shows the results of

Fig. 21 Experiments on a real blurred image corrupted by motion blur that contains text and face; methods: Krishnan et al. [23], Xu et al. [48], Xu et al. [49], Pan et al. [35–37], and PlugDictionary with classification

step (SVM classifier). *Note:* In the brackets, we stated a type of images for which the specific method is tailored

the OCR tests on clean, blurred, and estimated images of the method by Pan et al. [36] and the proposed method. OCR is performed on a clean image (when available) as a reference.

Furthermore, to give more insight into the performance of OCR, Figs. 22 and 23 show the results obtained on two images blurred with different intensities. Figure 22, which corresponds to Image 1 from Table 3, shows the results obtained on the slightly blurred text image. The results of three error measures used to assess the quality of OCR for the method by Pan et al. [36] and the proposed method are comparable. As this image is corrupted by very low intensity blur, results show that it is the best to perform OCR on the blurred image itself. Figure 23, which corresponds to Image 2 from Table 3, tells different story. Here, we have a situation where, if OCR takes the blurred image as an input, it will not give any “result.” Furthermore, it will give a very bad result if the input image is one estimated by the method from Pan et al. [36] and a reasonable result if the input image is estimated by the proposed method.

The results show that, in most of the experiments, the image estimated by the proposed method is able to improve OCR accuracy. When OCR is not possible on the blurred image (images 2, 4, and 5), the proposed method slightly improves the results, and in some cases (images 6, 7, and 8), the improvement is significant.

4.5 Regularization parameters

One of the main challenges of the proposed framework is setting the regularization parameter associated with the image

prior. This parameter is influenced by the type of image, blurring kernel, and noise level. The bottom row of Fig. 8 shows the chosen regularization parameter λ as a function of different noise levels for text images. From here, we can see that for a very high noise level (BSNR = 10 dB), we should use higher values of λ , but also that it depends on the blurring kernel (shape and size).

Figure 24 shows the behavior of the final ISNR (after 50 iterations of the PlugGMM algorithm) as a function of the parameter λ for text image corrupted with four different blurring kernels (first four kernels from [25]). We can see how the choice of the regularization parameter influences the result. For some kernels (kernels 1 and 2), it is relatively safe to choose big enough parameter, but it is not the case for, for example, kernel 4 where only one value of the parameter λ gives the maximum result.

5 Conclusion

In this work, we have proposed an approach for using the so-called plug-and-play framework for class-adapted blind image deblurring (BID). We tested three state-of-the-art denoisers: two adapted to specific image classes (GMM and dictionary-based) and one generic denoiser (BM3D). Additionally, we used two priors on the blurring filter: a weak one (positivity and limited support) and a sparsity-inducing one. Experiments show that the proposed approach yields state-of-the-art results, when applied to images that belong to a specific class (e.g., text, faces, and fingerprints), compared

Table 3 OCR results obtained on clean, blurred, and estimated images from the method of Pan et al. [36] and the proposed method, in terms of three error measurements: AWC (averaged word confidence), WER (word error rate), CER (character error rate)

Images		Clean image	Blurred image	Pan et al. [36]	Proposed
Image 1	AWC	0.744	0.726	0.681	0.696
	WER	0.23	0.576	0.73	0.807
	CER	0.131	0.274	0.668	0.717
Image 2	AWC	0.788	—	0.698	0.714
	WER	0.666	—	1	0.666
	CER	0.396	—	0.577	0.448
Image 3	AWC	0.795	0.786	—	0.808
	WER	0.111	0.222	—	0.055
	CER	0.018	0.092	—	0.009
Image 4	AWC	0.686	—	—	0.4818
	WER	0.692	—	—	1
	CER	0.401	—	—	1
Image 5	AWC	0.806	—	0.654	0.809
	WER	0.538	—	0.576	0.346
	CER	0.568	—	0.75	0.327
Image 6	AWC	0.89	0.594	—	0.893
	WER	0	1	—	0
	CER	0	0.927	—	0
Image 7	AWC	0.89	0.607	—	0.846
	WER	0	1	—	0.071
	CER	0	0.958	—	0.01
Image 8	AWC	—	0.659	—	0.56
	WER	—	1	—	0.5
	CER	—	1	—	0.259

Bold values represent the best results for every category

Results of the experiments when we do not have access to the ground truth (Clean image 8) or when the OCR method is returning zero are marked with a dash

Fig. 22 OCR results on slightly blurred image: left—tested image; right—OCR result

with several generic BID methods [5,23], and can be used for a variety of blurring filters. In addition, the method is able to handle strong noise in the case of text images, outperforming

the state-of-the-art method for BID of text images [36]. For text images corrupted with motion blur, we are able to outperform several generic BID methods [5,23,37,48,49], and some methods tailored for text images [12], and to perform on par with the state-of-the-art method designed only for text images [36]. When dealing with images that contain two classes with

Fig. 24 Final ISNR for different values of λ for text images corrupted with four different blurring kernels

completely different structures (namely, text and face), our method outperforms several state-of-the-art methods tailored for generic images [23,37,48,49] and a BID method specially designed for text images [36]. We show that slightly better results are achieved with a class-aware denoiser, compared with a generic denoiser, when tested on images that do belong to a specific class. Additionally, we show that for images that contain two (or more) classes, better results are achieved when we use an image prior which is appropriate for all classes present in the image. With the inclusion of a classification step, the results can be further improved. Arguably, the biggest drawback of the proposed method is its computational cost, which strongly depends on the choice (and settings) of the denoiser, and which increases with the inclusion of the classification step. Other potential limitations of the proposed method are the need to set the regularization parameters (that have to be hand-tuned) and stopping criteria for the inner ADMM algorithms. Ongoing work includes developing an approach for automatic (or semiautomatic) setting of these parameters. Future work will be focused on the extension of the proposed method for BID of generic images and other image restoration tasks (e.g., image inpainting).

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

1. Afonso, M.V., Bioucas-Dias, J.M., Figueiredo, M.A.T.: Fast image recovery using variable splitting and constrained optimization. *IEEE Trans. Image Process.* **19**, 2345–2356 (2010)
2. Afonso, M.V., Bioucas-Dias, J.M., Figueiredo, M.A.T.: An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems. *IEEE Trans. Image Process.* **20**, 681–695 (2011)
3. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process.* **54**, 4311–4322 (2006)
4. Almeida, M.S.C., Almeida, L.B.: Blind and semi-blind deblurring of natural images. *IEEE Trans. Image Process.* **19**, 36–52 (2010)
5. Almeida, M.S.C., Figueiredo, M.A.T.: Blind image deblurring with unknown boundaries using the alternating direction method of multipliers. In: *IEEE International Conference on Image Processing* (2013)
6. Altman, N.S.: An introduction to kernel and nearest-neighbor non-parametric regression. *Am. Stat.* **46**, 175–185 (1992)
7. Anwar, S., Huynh, C.P., Porikli, F.: Class-specific image deblurring. *IEEE International Conference on Computer Vision*, 495–503 (2015)
8. Anwar, S., Porikli, F., Huynh, C.P.: Category-specific object image denoising. *IEEE Trans. Image Process.* **26**(11), 5506–5518 (2017)
9. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.* **3**, 1–122 (2011)
10. Cai, J.F., Ji, H., Liu, C., Shen, Z.: Framelet-based blind motion deblurring from a single image. *IEEE Trans. Image Process.* **21**, 562–572 (2012)
11. Chen, X., He, X., Yang, J., Wu, Q.: An effective document image deblurring algorithm. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 369–376 (2011)
12. Cho, H., Wang, J., Lee S.: Text image deblurring using text-specific properties. In: *Proceedings of the 12th European Conference on Computer Vision* (2012)
13. Cho, S., Lee S.: Fast motion deblurring. *ACM Trans. Graph.* **28**, Article no. 145 (2009)
14. Combettes, P.L., Wajs, V.R.: Signal recovery by proximal forward-backward splitting. *SIAM J. Multiscale Model. Simul.* **4**, 1164–1200 (2005)
15. Cortes, C., Vapnik, V.: Support-vector networks. *Machine Learning* **20**, 273–297 (1995)
16. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Trans. Image Process.* **16**, 2080–2095 (2007)
17. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. Image Process.* **15**, 3736–3745 (2006)
18. Fergus, R., Singh, B., Hertzmann, A., Roweis, S.T., Freeman, W.T.: Removing camera shake from a single photograph. *ACM Trans. Graph.* **25**, 787–794 (2006)
19. Field, D.J.: What is the goal of sensory coding? *Neural Comput.* **6**, 559–601 (1994)
20. Gupta, A., Joshi, N., Zitnick, C.L., Cohen, M., Curless, B.: Single image deblurring using motion density functions. In: *European Conference on Computer Vision*, pp. 171–184 (2010)
21. HaCohen, Y., Shechtman, E., Lischinski, D.: Deblurring by example using dense correspondence. In: *IEEE International Conference on Computer Vision* (2013)
22. Hradis, M., Kotera J., Zemcik P., Sroubek, F.: Convolutional neural networks for direct text deblurring. In: *The British Machine Vision Conference* (2015)
23. Krishnan, D., Tay, T., Fergus, R.: Blind deconvolution using a normalized sparsity measure. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 233–240 (2011)
24. Lai, W.S., Huang, J.B., Hu, Z., Ahuja, N., Yang, M.H.: A comparative study for single image blind deblurring. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1709 (2016)

25. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Understanding and evaluating blind deconvolution algorithms. In: IEEE Conference on Computer Vision and Pattern Recognition (2009)
26. Li, T.-H., Lii, K.-S.: A joint estimation approach for two-tone image deblurring by blind deconvolution. *IEEE Trans. Image Process.* **11**, 847–858 (2002)
27. Ljubenovic, M., Figueiredo, M.A.T.: Blind image deblurring using class-adapted image priors. In: IEEE International Conference on Image Processing (2017)
28. Ljubenovic, M., Zhuang, L., Figueiredo, M.A.T.: Class-adapted blind deblurring of document images. In: 14th IAPR International Conference on Document Analysis and Recognition (2017)
29. Luo, E., Chan, S.H., Nguyen, T.Q.: Adaptive image denoising by targeted databases. *IEEE Trans. Image Process.* **24**, 2167–2181 (2015)
30. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: Proceedings of the 26th Annual International Conference on Machine Learning (2009)
31. Michaeli, T., Irani, M.: Blind deblurring using internal patch recurrence. In: European Conference on Computer Vision, pp. 783–798 (2014)
32. Nishiyama, M., Hadid, A., Takeshima, H., Shotton, J., Kozakaya, T., Yamaguchi, O.: Facial deblur inference using subspace analysis for recognition of blurred faces. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**, 838–845 (2011)
33. Niknejad, M., Bioucas-Dias, J., Figueiredo, M.A.T.: Class-specific image denoising using importance sampling. In: IEEE International Conference on Image Processing (2017)
34. Niknejad, M., Bioucas-Dias, J., Figueiredo, M.A.T.: Class-specific Poisson denoising by patch-based importance sampling. In: IEEE International Conference on Image Processing (2017)
35. Pan, J., Hu, Z., Su, Z., Yang, M.-H.: Deblurring face images with exemplars. In: European Conference on Computer Vision (2014)
36. Pan, J., Hu, Z., Su, Z., Yang, M.-H.: Deblurring text images via l_0 -regularized intensity and gradient prior. In: IEEE Conference on Computer Vision and Pattern Recognition (2014)
37. Pan, J., Sun, D., Pfister, H., Yang, M.-H.: Blind image deblurring using dark channel prior. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
38. Pan, J., Hu, Z., Su, Z., Yang, M.-H.: L_0 -regularized intensity and gradient prior for deblurring text images and beyond. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**, 342–355 (2017)
39. Remez, T., Litany, O., Giryes, R., Bronstein, A.M.: Deep class-aware image denoising. In: International Conference on Sampling Theory and Applications, pp. 138–142 (2017)
40. Rubinstein, R., Bruckstein, A.M., Elad, M.: Dictionaries for sparse representation modeling. *Proc. IEEE* **98**, 1045–1057 (2010)
41. Shan, Q., Jia, J., Agarwala, A.: High-quality motion deblurring from a single image. *ACM Trans. Graph.* **27**, 73:1–73:10 (2008)
42. Sun, L., Cho S., Wang, J., Hays J.: Edge-based blur kernel estimation using patch priors. In: IEEE International Conference on Computational Photography (2013)
43. Teodoro, A.M., Almeida, M., Figueiredo, M.A.T.: Single-frame image denoising and inpainting using Gaussian mixtures. In: International Conference on Pattern Recognition Applications and Methods, pp. 283–288 (2015)
44. Teodoro, A.M., Bioucas-Dias, J.M., Figueiredo, M.A.T.: Image restoration with locally selected class-adapted models. In: IEEE International Workshop on Machine Learning for Signal Processing (2016)
45. Teodoro, A.M., Bioucas-Dias, J.M., Figueiredo, M.A.T.: Image restoration and reconstruction using variable splitting and class-adapted image priors. In: IEEE International Conference on Image Processing, pp. 3518–3522 (2016)
46. Venkatakrishnan, S.V., Bouman, C.A., Wohlberg, B.: Plug-and-play priors for model based reconstruction. In: IEEE Global Conference on Signal and Information Processing (2013)
47. Wang, C., Sun, L.F., Chen, Z.Y., Zhang, J.W., Yang, S.Q.: Multi-scale blind motion deblurring using local minimum. *Inverse Probl.* **26**, 015003 (2010)
48. Xu, L., Jia, J.: Two-phase kernel estimation for robust motion deblurring. In: Proceedings of the 11th European Conference on Computer Vision: Part I (2010)
49. Xu, L., Zheng, S., Jia, J.: Unnatural l_0 sparse representation for natural image deblurring. In: IEEE Conference on Computer Vision and Pattern Recognition (2013)
50. Yu, G., Sapiro, G., Mallat, S.: Solving inverse problems with piecewise linear estimators: from Gaussian mixture models to structured sparsity. *IEEE Trans. Image Process.* **21**, 2481–2499 (2012)
51. Zhang, H., Yang, J., Zhang, Y., Nasrabadi, N.M., Huang, T.S.: Close the loop: joint blind image restoration and recognition with sparse representation prior. In: IEEE International Conference on Computer Vision, pp. 770–777 (2011)
52. Zoran, D., Weiss, Y.: From learning models of natural image patches to whole image restoration. In: International Conference on Computer Vision, pp. 479–486 (2011)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.