Blur kernel estimation using sparse representation and cross-scale self-similarity

Jing Yu¹ · Zhenchun Chang² · Chuangbai Xiao¹

Received: 10 April 2018 / Revised: 10 January 2019 / Accepted: 15 January 2019 /

Published online: 29 January 2019

© The Author(s) 2019

Abstract

Blind image deconvolution, i.e., estimating both the latent image and the blur kernel from the only observed blurry image, is a severely ill-posed inverse problem. In this paper, we propose a blur kernel estimation method for blind motion deblurring using sparse representation and cross-scale self-similarity of image patches as priors to recover the latent sharp image from a single blurry image. Sparse representation indicates that image patches can always be represented well as a sparse linear combination of atoms in an appropriate dictionary. Cross-scale self-similarity results in that any image patch can in some way be well approximated by a number of other similar patches across different image scales. Our method is based on the observations that almost any image patch in a natural image has multiple similar patches in down-sampled versions of the image, and down-sampling produces image patches that are sharper than those in the blurry image itself. In our method, the dictionary for sparse representation is trained adaptively from sharper patches sampled from the down-sampled latent image estimate to make the similar patches of the latent sharp image well represented sparsely, and meanwhile, all patches from the latent image estimate are optimized to be as close to the sharper similar patches searched from the down-sampled version to enforce the sharp recovery of the latent image by constructing a non-local regularization. Experimental results on both simulated and real blurry images demonstrate that our method outperforms state-of-the-art blind deblurring methods.

Keywords Blind deconvolution · Deblurring · Sparse representation · Self-similarity · Cross-scale

National Natural Science Foundation of China (61501008) and Beijing Municipal Natural Science Foundation (4172002).

✉ Jing Yu
jing.yu@bjut.edu.cn

Zhenchun Chang
txchangchun@163.com

Chuangbai Xiao
cbxiao@bjut.edu.cn

¹ Faculty of Information Technology, Beijing University of Technology, Beijing, 100124, China

² Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China

1 Introduction

Motion blur caused by camera shake has been one of the most common artifacts in digital imaging. Blind image deconvolution aims to recover the latent (unblurred) image from the only observed blurry image when the blur kernel is unknown. Despite over three decades of research in the field, blind deconvolution still remains a challenge for real-world photographs with unknown blurs. Recently, blind deconvolution has received renewed attention since Fergus et al.'s work [6] that removes the motion blur from a single image.

If a motion blur is shift-invariant, the degradation process can be modeled as the 2-D convolution of the latent image with the motion blur kernel:

$$\mathbf{y} = \mathbf{h} * \mathbf{x} + \mathbf{n} \quad (1)$$

where $*$ stands for the 2-D convolution operator, $\mathbf{y}$ is the observed blurry image, $\mathbf{h}$ is the blur kernel (or point spread function), $\mathbf{x}$ is the latent image and $\mathbf{n}$ is noise. Then, when the blur kernel is unknown, removing the motion blur from the observed blurry image becomes the so-called blind deconvolution operation, and the recovery of the latent image is a severely ill-posed inverse problem. The key to solve the ill-posed inverse problem is proper incorporation of various image priors about the latent image into the blind deconvolution process.

In recent years, impressive progress has been made in removing motion blur only given a single blurry image. Some methods explicitly or implicitly exploit edges for kernel estimation [3, 8, 10, 31]. This idea was introduced by Jia [8], who used an alpha matte to estimate the transparency of blurred object boundaries and performed the kernel estimation only using transparency. Joshi et al. [10] predicts sharp edges using edge profiles and estimates the blur kernel from the predicted edges. However, their goal is to remove small blurs, for it is not trivial to directly restore sharp edges from a severely blurred image. In [3, 31], strong edges are predicted from the latent image estimate using a shock filter and gradient thresholding, and then used for kernel estimation. Unfortunately, the shock filter could over-sharpen image edges, and is sensitive to noise, leading to an unstable estimate.

Another family of methods exploit various sparse priors for either the latent image $\mathbf{x}$ or the motion blur kernel $\mathbf{h}$, or both, and formulate the blind deconvolution as a joint optimization problem with some regularizations on both $\mathbf{x}$ and $\mathbf{h}$ [6, 14, 15, 22, 23, 26]:

$$(\hat{\mathbf{x}}, \hat{\mathbf{h}}) = \underset{\mathbf{x}, \mathbf{h}}{\operatorname{argmin}} \left\{ \sum_{*} \omega_{*} \|\partial_{*} \mathbf{y} - \mathbf{h} * \partial_{*} \mathbf{x}\|_2^2 + \lambda_x \rho(\mathbf{x}) + \lambda_h \rho(\mathbf{h}) \right\} \quad (2)$$

where $\partial_{*} \in \{\partial_0, \partial_x, \partial_y, \partial_{xx}, \partial_{xy}, \partial_{yy}, \dots\}$ denotes the partial derivative operator in different directions and orders, ω_{*} is a weight for each partial derivative, $\rho(\mathbf{x})$ is a regularization term on the latent sharp image $\mathbf{x}$, $\rho(\mathbf{h})$ is a regularization term on the blur kernel $\mathbf{h}$, and λ_x and λ_h are regularization weights. The first term in the objective function uses image derivatives for reducing ringing artifacts. Many techniques based on sparsity priors of image gradients have been proposed to deal with motion blur. Most previous methods assume that gradient magnitudes of natural images follow a heavy-tailed distribution. Fergus et al. [6] represent the heavy-tailed distribution over gradient magnitudes with a zero-mean mixture of Gaussian based on natural image statistics. Levin et al. [13] propose a hyper-Laplacian prior to fit the heavy-tailed distribution of natural image gradients. Shan et al. [26] construct a natural gradient prior for the latent image by concatenating two piece-wise continuous convex functions. However, sparse gradient priors always prefer trivial solutions, that is, the delta kernel and exactly the blurry image as the latent image estimate because the blur reduces

the overall gradient magnitude. To tackle this problem, there are mainly two streams of research works for blind deconvolution. They use maximum marginal probability estimation of $\mathbf{h}$ alone (marginalizing over $\mathbf{x}$) to recover the true kernel [6, 14, 15] or optimize directly the joint posterior probability of both $\mathbf{x}$ and $\mathbf{h}$ by performing some empirical strategies or heuristics to avoid the trivial solution during the minimization [22, 26]. Levin et al. [14, 15] suggest that a maximum-a-posterior (MAP) estimation of $\mathbf{h}$ alone is well conditioned and recovers an accurate kernel, while a simultaneous MAP estimation for solving blind deconvolution by jointly optimizing $\mathbf{x}$ and $\mathbf{h}$ would fail because it favors the trivial solution. Perrone and Favaro [22, 23] confirm the analysis of Levin et al. [14, 15] and on the other hand also declare that total variation-based blind deconvolution methods can work well by performing specific implementation. In their work, the total variation regularization parameter is initialized with a large value to help avoiding the trivial solution and iteratively reduced to allow for the recovery of more details. Blind deconvolution is in general achieved through an alternating optimization scheme. In [22, 23], the projected alternating minimization (PAM) algorithm of total variation blind deconvolution can successfully converge to the desired solution.

More present-day works often involve priors over larger neighborhoods or image patches, such as image super resolution [34], image denoising [30], non-blind image deblurring [9] and more. Gradient priors often consider two or three neighboring pixels, which are not sufficient for modeling larger image structures. Patch priors that consider larger neighborhoods (e.g., 5×5 or 7×7 image patches) model more complex structures and dependencies in larger neighborhoods. Sun et al. [27] use a patch prior learned from an external collection of sharp natural images to restore sharp edges. Michaeli and Irani [17] construct a cross-scale patch recurrence prior for the estimation of the blur kernel. Lai et al. [12] obtain two color centers for every image patch and build a normalized color-line prior for blur kernel estimation. More recently, Pan et al. [21] introduce the dark channel prior based on statistics of image patches to kernel estimation, while Yan et al. [32] propose a patch-based bright channel prior for kernel estimation.

Recent work suggests that image patches can always be well represented sparsely with respect to an appropriate dictionary and the sparsity of image patches over the dictionary can be used as a prior to regularize the ill-posed inverse problem. Zhang et al. [36] use sparse representation of image patches as a prior and train the dictionary from an external collection of natural images or the blurry image itself via the K-SVD algorithm [1]. Li et al. [16] combine the dictionary pair and the sparse gradient prior with assumption that the blurry image and the sharp image have the same sparse coefficients under the blurry dictionary and the sharp dictionary respectively to restore the sharp image via sparse reconstruction using the blurry image sparse coefficients on the sharp dictionary. The key issue of sparse representation is to identify a specific dictionary that represents latent image patches in a sparse manner. Most methods use an external collection consisting of enormous images to learn a universal dictionary as training samples. To make all latent image patches represented sparsely over such a universal dictionary, the collection need provides massive training samples, and thus this may lead to an inefficient learning and a potentially unstable dictionary. Meanwhile, the collection needs to provide patches similar to the patches from the latent image, which cannot hold all the time. Alternatively, the blurry image itself is used as training samples, which cannot constantly guarantee the sparsity of sharp image patches over the learned dictionary.

In this paper, we focus on the regularization approach using patch priors for blind image deblurring. In our previous work, sparse representation and self-similarity are combined to work for image super resolution (SR) [21, 19]. Super resolution algorithms typically assume

that the blur kernel is known (either the point spread function of the camera, or some default low-pass filter, *e.g.* a Gaussian), while blind deblurring refers to the task of estimating the unknown blur kernel. Michaeli and Irani [17] have showed super resolution algorithms cannot be applied directly to blind deblurring. We propose a blur kernel estimation method for blind motion deblurring using sparse representation and cross-scale self-similarity of image patches as priors to guide the recovery of the latent image. Our method is based on the observations that almost any image patch in a natural image has multiple similar patches in down-sampled versions of the image, and down-sampling produces image patches that are sharper than those in the blurry image itself. The additional information is thoroughly explored from abundant patch repetitions of cross-scale self-similar structures of the same image for the blind deconvolution problem. On the one hand, we incorporate cross-scale self-similarity into sparse representation via cross-scale dictionary learning that uses sharper patches sampled from the down-sampled version as training samples for better representing the similar patches of the latent sharp image over the learned dictionary. On the other hand, we construct a cross-scale non-local regularization to optimize all patches from the latent image estimate to be as close to the sharper similar patches searched from the down-sampled version for sharpening edges and details of the latent image estimate as possible. Finally, we take an approximate iterative approach to solve the resulting minimization problem by alternately optimizing the blur kernel and the latent image in a coarse-to-fine framework.

The remainder of this paper is organized as follows. Section 2 describes the background on sparse representation and multi-scale self-similarity. Section 3 makes detailed description on the proposed method, including our blind deconvolution model and the solution to our model. Section 4 presents experimental results on both simulated and real blurry images. Section 5 draws the conclusion.

2 Sparse representation and multi-scale self-similarity

2.1 Sparse representation

Image patches can always be represented well as a sparse linear combination of atoms (i.e. columns) in an appropriate dictionary. Suppose that an image patch can be represented as $\mathbf{Q}_j \mathbf{X}$, here $\mathbf{Q}_j$ is a matrix extracting the j th patch from an image $\mathbf{X}$ ordered lexicographically by stacking either the rows or the columns of $\mathbf{x}$ into a vector, and the image patch $\mathbf{Q}_j \mathbf{X} \in \mathbb{R}^n$ can be represented sparsely over $\mathbf{D} \in \mathbb{R}^{n \times t}$, that is:

$$\mathbf{Q}_j \mathbf{X} = \mathbf{D} \boldsymbol{\alpha}_j, \quad \|\boldsymbol{\alpha}_j\|_0 \ll n \quad (3)$$

where $\mathbf{D} = [\mathbf{d}_1, \dots, \mathbf{d}_t] \in \mathbb{R}^{n \times t}$ is referred to as the dictionary, each column $\mathbf{d}_j \in \mathbb{R}^n$ for $j = 1, \dots, t$ represents the atom of the dictionary $\mathbf{D}$, $\boldsymbol{\alpha}_j = [\alpha_1, \dots, \alpha_t]^T \in \mathbb{R}^t$ is the sparse representation coefficient of $\mathbf{Q}_j \mathbf{X}$ and $\|\boldsymbol{\alpha}_j\|_0$ counts the nonzero entries in $\boldsymbol{\alpha}_j$.

Given a set of training samples $\mathbf{s}_i \in \mathbb{R}^n$, $i = 1, \dots, m$, here m is the number of training samples, dictionary learning attempts to find a dictionary $\mathbf{D}$ that forms sparse representation coefficients $\boldsymbol{\alpha}_i$, $i = 1, \dots, m$ for the training samples by jointly optimizing $\mathbf{D}$ and $\boldsymbol{\alpha}_i$, $i = 1, \dots, m$ as follows:

$$\min_{\mathbf{D}, \boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_m} \sum_{i=1}^m \|\mathbf{s}_i - \mathbf{D} \boldsymbol{\alpha}_i\|_2^2 \quad \text{s.t.} \quad \forall i \quad \|\boldsymbol{\alpha}_i\|_0 \leq T \quad (4)$$

where $T \ll n$ controls the sparsity of $\boldsymbol{\alpha}_i$ for $i = 1, \dots, m$. The K-SVD algorithm [1] is an effective dictionary learning method which solves (4) by alternately optimizing $\mathbf{D}$ and $\boldsymbol{\alpha}_i$, i

$= 1, \dots, m$. As a matter of fact, the precision of the K-SVD algorithm can be controlled either by constraining the representation error or by constraining the number of nonzero entries in α_i . We use the latter formulated in (4), because it is required in the orthogonal matching pursuit (OMP) algorithm [29] which obtains an approximation solution for (3).

We firstly use the K-SVD algorithm [1] to obtain the dictionary $\mathbf{D}$. Then, for the patch $\mathbf{Q}_j\mathbf{X}$, we have to derive the sparse representation coefficient. Equation (3) can be formulated as the following ℓ_0 -norm minimization problem:

$$\min_{\alpha_j} \|\mathbf{Q}_j\mathbf{X} - \mathbf{D}\alpha_j\|_2^2 \quad \text{s.t.} \quad \|\alpha_j\|_0 \leq T \quad (5)$$

where T is the sparsity constraint parameter. In our method, we obtain an approximation solution $\hat{\alpha}_j$ for (5) by using the OMP algorithm [29]. The OMP algorithm is a greedy iterative algorithm for approximately solving the above ℓ_0 -minimization problem. It works by gradually finding the locations of the non-zeros in α_j one at a time. After $\hat{\alpha}_j$ is derived, the reconstructed image patch $\mathbf{Q}_j\hat{\mathbf{X}}$ can be represented sparsely over $\mathbf{D}$ through $\mathbf{Q}_j\hat{\mathbf{X}} = \mathbf{D}\hat{\alpha}_j$.

2.2 Multi-scale self-similarity and non-local means

Most natural images have properties of self-similarity, where structures from image fragments tend to repeat themselves, *e.g.* one part of road, building or natural landscape resembles another part of the object itself. Multi-scale self-similarity refers to explicit or implicit repetitions of structures at various sizes in the same scene. It can be observed that there are many multi-scale similar structures in a natural image. Figure 1 schematically illustrates patch repetitions of multi-scale self-similar structures both within the same scale and across different scales of a single image. For a patch marked with a red box in Fig. 1a, we search for its 5 most similar patches marked with blue boxes in this image. Figure 1b shows close-ups of the similar patches within the same scale. In this example, the image is down-sampled by a factor of $a = 2$, as shown in Fig. 1c. For the patch marked by a red box in Fig. 1a at the original scale, we also search for its 5 most similar patches in the down-sampled image, marked by blue boxes. Figure 1d shows close-ups of the similar patches in the down-sampled image, *i.e.* cross-scale similar patches. When small image patches are used, *e.g.*, 5×5 or 7×7 image patches, patch repetitions occur abundantly both within the same scale and across different scales of a natural image, even when we do not visually perceive any obvious repetitive structure. This is due to the fact that very small patches often contain only an edge, a corner, *etc.*, and thus such patch repetitions are found abundantly in multiple image scales of almost any natural image [7]. Glasner et al. [7] perform a test to find out the amount of multi-scale similar patches in natural images, and come to the conclusion that there are plenty of multi-scale similar patches both within the same scale and across different scales in a single image.

The non-local means was firstly introduced for image denoising based on this self-similarity property of natural images in the seminal work of Buades [2], and since then, the non-local means is extended successfully to other inverse problems such as image super resolution and non-blind image deblurring [5, 25]. The non-local means is based on the observation that similar image patches within the same scale are likely to be appeared in a single image, and these same-scale similar patches can provide additional information. For any patch $\mathbf{Q}_j\mathbf{X}$ in the sharp image, its similar patches can be obtained using block matching that the similarity is measured by the distance between $\mathbf{Q}_j\mathbf{X}$ and any other patch of this image. The p most similar patches $\mathbf{Q}_i\mathbf{X}$, $i = 1, \dots, p$ of $\mathbf{Q}_j\mathbf{X}$ are used to estimate $\mathbf{Q}_j\mathbf{X}$, and

Fig. 1 Patches repeat both within the same scale and across different scales of a single image

the difference between $\mathbf{Q}_j\mathbf{X}$ and its estimation is the non-local regularization. In our previous work [21], we use cross-scale similar patches as well as same-scale similar patches to construct the multi-scale non-local regularization for super resolution reconstruction.

3 Blind deconvolution

3.1 Use of cross-scale self-similarity

In our blind deblurring model, we exploit effectively the additional information provided by cross-scale similar patches at down-sampled scales by employing the cross-scale non-local regularization and the cross-scale dictionary learning. In the cross-scale non-local regularization, all patches from the latent image estimate are optimized to be as close to their sharper similar patches searched from the down-sampled version to enforce the sharp recovery of the latent image as possible. The cross-scale dictionary learning, meanwhile, uses the down-sampled version of the latent image estimate as training samples to make the similar patches of the latent sharp image have sparse representations over the learned dictionary.

Since almost any image patch in a natural image has multiple similar patches in down-sampled versions of the image [7], we search for similar patches from the down-sampled image and use these cross-scale patches to construct a cross-scale non-local regularization by exploiting the correspondence between these cross-scale similar patches. Suppose that $\mathbf{X} \in \mathbb{R}^N$ and $\mathbf{X}^a \in \mathbb{R}^{N/a^2}$ represent the latent image and its down-sampled version

respectively, where N is the size of the latent image, and a is the down-scaling factor. The latent image patch and its down-sampled version can be represented as $\mathbf{Q}_j\mathbf{X}$ and $\mathbf{R}_i\mathbf{X}^a$, here $\mathbf{Q}_j \in \mathbb{R}^{n \times N}$ and $\mathbf{R}_i \in \mathbb{R}^{n \times N/a^2}$ are matrices extracting the j th and i th patch from $\mathbf{X}$ and $\mathbf{X}^a$ respectively, and n is the size of the image patch. For each patch $\mathbf{Q}_j\mathbf{X}$ in the latent image $\mathbf{X}$, we can search for its p most similar patches $\mathbf{R}_i\mathbf{X}^a$, for $i = 1, \dots, p$ in $\mathbf{X}^a$ using block matching. The linear combination of the p most similar patches of $\mathbf{Q}_j\mathbf{X}$ (put into the set $\mathcal{S}_j$) is used to predict $\mathbf{Q}_j\mathbf{X}$, that is, the prediction can be represented as,

$$\mathbf{Q}_j\mathbf{X} \approx \sum_{i \in \mathcal{S}_j} w_i^j \mathbf{R}_i\mathbf{X}^a \quad (6)$$

where

$$w_i^j = \frac{\exp(-\|\mathbf{Q}_j\mathbf{X} - \mathbf{R}_i\mathbf{X}^a\|_2^2/h)}{\sum_{s \in \mathcal{S}_j} \exp(-\|\mathbf{Q}_j\mathbf{X} - \mathbf{R}_s\mathbf{X}^a\|_2^2/h)} \quad (7)$$

is the weight and h is the control parameter of the weight. The prediction error should be small and can be used as the regularization in our blind deblurring model [25].

The choice of training samples is very important for the dictionary learning problem. Ideally the sharp image should be used as training samples for dictionary learning. Unfortunately, the sharp image is an unknown quantity to recover. In our single-image super resolution work, the low-resolution image itself is used to learn an adaptive over-complete dictionary as training samples. However, it is not a good choice for blind deblurring to use the input blurry image itself as training samples, because these patches from the blurry image cannot guarantee the sparsity of sharp image patches over the learned dictionary. Since down-sampling the blurry image can provide sharper patches that are more similar to patches from the latent sharp image, we use the down-sampled version of the blurry image as training samples to obtain the dictionary $\mathbf{D}$ for sparse representation in our previous work [35]. In the proposed method, we present an improvement to the dictionary learning (see Section 3.4 for detail). Because of the use of cross-scale (i.e. down-sampled) similar patches, we call it *cross-scale dictionary learning*.

We now provide illustration to account for the use of cross-scale self-similarity. Although patches repeat at the same or different scales in the sharp image, as illustrated in Fig. 1, the similarity diminishes significantly between the sharp image and its blurry counterpart [17]. For the sharp patch marked by a red box in Fig. 1a, we still search for its 5 most similar patches from the blurry image (Fig. 2a and b) and its down-sampled version (Fig. 2c and d), respectively. Figure 2 illustrates that the patches searched from the blurry image are less similar to the given sharp patch than those searched from the down-sampled blurry image. This is because the blur effect tends to attenuate at coarser scales of the image despite the strong blur at the original scale.

Figure 3 illustrates the reason why similar patches across different scales are available for providing a prior for restoration. Suppose that $f(\xi)$ and $f(\xi/a)$ are cross-scale similar patches and $f(\xi/a)$ is an a -times larger patch in the sharp image, here ξ denotes the spatial coordinate. Accordingly, their blurry counterparts $q(\xi)$ and $r(\xi)$ are similar across image scales, and the size of $r(\xi)$ is a times as large as that of $q(\xi)$ in the blurry image. In Fig. 3, the blurry image is a times the size of its down-sampled version. Down-scaling the blurry patch $r(\xi)$ by a factor of a generates an a -times smaller patch $r^a(\xi)$. Then, $q(\xi)$ and $r^a(\xi)$ are of the same size and the patch $r^a(\xi)$ from the down-sampled image is exactly an a -times sharper version of the patch $q(\xi)$ in the blurry image. In such a case, $r^a(\xi)$ can offer much exact prior information for the recovery of $q(\xi)$. Figure 3 schematically demonstrates that the patches at coarser image scales can serve as a good prior, although it is an ideal case.

Fig. 2 Blurry patches are less similar to the sharp patch than down-scaled blurry patches

Ignoring sample issues, we refer to [17] and give a simple proof that $r^a(\xi)$ is a -times sharper than $q(\xi)$. Consider a small patch $f(\xi)$ in the sharp image and the blur kernel $h(\xi)$, and then we have

$$q(\xi) = h(\xi) * f(\xi) \quad (8)$$

Fig. 3 Similar patches across image scales are available for providing a prior for restoration

where $q(\xi)$ is the blurry counterpart of $f(\xi)$. Since there are abundant cross-scale similar patches in a single image, we assume there is a similar patch with $f(\xi)$ elsewhere, and its size is a times as large as that of $f(\xi)$, denoted by $f(\xi/a)$. This a -times larger patch $f(\xi/a)$ is convolved with the blur $h(\xi)$, and then we have

$$r(\xi) = h(\xi) * f(\xi/a) \quad (9)$$

where $r(\xi)$ is the blurry counterpart of $f(\xi/a)$. Now, if we down-scale the blurry image by a factor of a , then this patch $r(\xi)$ becomes:

$$r^a(\xi) = r(a\xi) = h(a\xi) * f(\xi) \quad (10)$$

In other words, $r^a(\xi)$ corresponds to the same patch $f(\xi)$, but convolved with the a -times narrower kernel $h(a\xi)$, rather than with $h(\xi)$. It implies that the patch $r^a(\xi)$ in the down-scaled image is an a -times sharper version of the patch $q(\xi)$ in the blurry image, as visualized in Fig. 3. The above proof shows that down-scaling an image by a factor of a produces a -times sharper patches of the same size that are more similar to patches from the latent sharp image.

3.2 Our model

We incorporate both sparse representation and cross-scale self-similarity as priors into our blind deconvolution model to guide the recovery of the latent image. With these priors as regularization terms, we get the following joint minimization problem of both the latent image $\mathbf{x}$ and the blur kernel $\mathbf{h}$:

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{h}} \quad & \left\{ \|\nabla \mathbf{y} - \mathbf{h} * \nabla \mathbf{x}\|_2^2 + \lambda_c \sum_j \|\mathbf{Q}_j \mathbf{X} - \mathbf{D} \boldsymbol{\alpha}_j\|_2^2 \right. \\ & \left. + \lambda_s \sum_j \|\mathbf{Q}_j \mathbf{X} - \sum_{i \in \mathcal{S}_j} w_i^j \mathbf{R}_i \mathbf{X}^a\|_2^2 + \lambda_g \|\nabla \mathbf{x}\|_2^2 + \lambda_h \|\mathbf{h}\|_2^2 \right\} \quad (11) \\ \text{s.t. } \forall j \quad & \|\boldsymbol{\alpha}_j\|_0 \leq T \end{aligned}$$

where $\nabla = \{\partial_x, \partial_y\}$ denotes the spatial derivative operator in two directions, $\mathbf{D}$ is the learned dictionary for sparse representation, $\mathbf{X}$ is the vector notation of the latent image $\mathbf{x}$, $\mathbf{X}^a$ is the down-sampled version of $\mathbf{X}$ by a factor of a , and λ_c , λ_s , λ_g and λ_h are regularization weights. Our blind deconvolution method is formulated as a constrained optimization problem that the objective could be minimized by constraining the number of nonzero entries in the sparse representation coefficients. In (11), the first term is the constraint of the observation model (i.e. data fidelity term), the second term is the sparsity prior, the third term is the cross-scale self-similarity prior (i.e. cross-scale non-local regularization), the fourth term is the smoothness constraint of the latent image, and the last term is the constraint of the blur kernel.

Blind deblurring in general involves two stages. The motion blur kernel $\mathbf{h}$ is firstly estimated by solving (11), which takes an iterative process that alternately optimizes the motion blur kernel $\mathbf{h}$ and the latent image $\mathbf{x}$. Then, the final deblurring result $\hat{\mathbf{x}}$ is recovered from the given blurry image $\mathbf{y}$ with the blur kernel estimate $\hat{\mathbf{h}}$ by performing various non-blind deconvolution methods, such as fast TV- ℓ_1 deconvolution [31], sparse deconvolution [14] and EPLL [37] etc..

3.3 Optimization

Equation (11) is a non-convex minimization problem, and cannot be solved in closed form. Instead it is solved by an approximate iterative optimization procedure, which alternates between optimizing the kernel $\mathbf{h}$ and the latent image $\mathbf{x}$. We will discuss these two steps separately.

3.3.1 Optimizing $\mathbf{h}$

In this step, we fix $\hat{\mathbf{x}}_k$ and update $\hat{\mathbf{h}}_{k+1}$. The objective function is simplified to:

$$\hat{\mathbf{h}}_{k+1} = \underset{\mathbf{h}}{\operatorname{argmin}} \{ \| \nabla \mathbf{y} - \mathbf{h} * \nabla \hat{\mathbf{x}}_k \|_2^2 + \lambda_h \| \mathbf{h} \|_2^2 \} \quad (12)$$

Equation (12) is a quadratic function of unknown $\mathbf{h}$, which has a closed-form solution for $\hat{\mathbf{h}}_{k+1}$:

$$\hat{\mathbf{h}}_{k+1} = \mathcal{F}^{-1} \left(\frac{\overline{\mathcal{F}(\partial_x \hat{\mathbf{x}}_k)} \mathcal{F}(\partial_x \mathbf{y}) + \overline{\mathcal{F}(\partial_y \hat{\mathbf{x}}_k)} \mathcal{F}(\partial_y \mathbf{y})}{\overline{\mathcal{F}(\partial_x \hat{\mathbf{x}}_k)} \mathcal{F}(\partial_x \hat{\mathbf{x}}_k) + \overline{\mathcal{F}(\partial_y \hat{\mathbf{x}}_k)} \mathcal{F}(\partial_y \hat{\mathbf{x}}_k) + \lambda_h} \right) \quad (13)$$

where $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ denote the fast Fourier transform and inverse Fourier transform respectively, and $\overline{\mathcal{F}(\cdot)}$ denotes the complex conjugate operator.

3.3.2 Optimizing $\mathbf{x}$

In this step, we fix $\hat{\mathbf{h}}_{k+1}$, and given $\hat{\mathbf{x}}_k$ update $\hat{\mathbf{x}}_{k+1}$. The objective function reduces to:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1} = \underset{\mathbf{x}}{\operatorname{argmin}} \left\{ \| \nabla \mathbf{y} - \hat{\mathbf{h}}_{k+1} * \nabla \mathbf{x} \|_2^2 + \lambda_c \sum_j \| \mathbf{Q}_j \mathbf{X} - \mathbf{D} \boldsymbol{\alpha}_j \|_2^2 \right. \\ \left. + \lambda_s \sum_j \| \mathbf{Q}_j \mathbf{X} - \sum_{i \in \mathcal{S}_j} w_i^j \mathbf{R}_i \mathbf{X}^a \|_2^2 + \lambda_g \| \nabla \mathbf{x} \|_2^2 \right\} \\ \text{s.t. } \forall j \quad \| \boldsymbol{\alpha}_j \|_0 \leq T \end{aligned} \quad (14)$$

Rearranging $\mathbf{y}$ in vector form, denoted by $\mathbf{Y} \in \mathbb{R}^N$, and rewriting the convolution of the blur kernel and the latent image in matrix-vector form, (14) can be expressed as

$$\begin{aligned} \hat{\mathbf{x}}_{k+1} = \underset{\mathbf{X}}{\operatorname{argmin}} \left\{ \| \mathbf{G}_x \mathbf{Y} - \mathbf{H}_{k+1} \mathbf{G}_x \mathbf{X} \|_2^2 + \| \mathbf{G}_y \mathbf{Y} - \mathbf{H}_{k+1} \mathbf{G}_y \mathbf{X} \|_2^2 \right. \\ \left. + \lambda_c \sum_j \| \mathbf{Q}_j \mathbf{X} - \mathbf{D} \boldsymbol{\alpha}_j \|_2^2 + \lambda_s \sum_j \| \mathbf{Q}_j \mathbf{X} - \sum_{i \in \mathcal{S}_j} w_i^j \mathbf{R}_i \mathbf{X}^a \|_2^2 \right. \\ \left. + \lambda_g (\| \mathbf{G}_x \mathbf{X} \|_2^2 + \| \mathbf{G}_y \mathbf{X} \|_2^2) \right\} \text{s.t. } \forall j \quad \| \boldsymbol{\alpha}_j \|_0 \leq T \end{aligned} \quad (15)$$

where $\mathbf{G}_x$ and $\mathbf{G}_y \in \mathbb{R}^{N \times N}$ are the matrix forms of the partial derivative operators ∂_x and ∂_y in two directions respectively, and $\mathbf{H}_{k+1} \in \mathbb{R}^{N \times N}$ is the blur matrix. Setting the derivative of (15) with respect to $\mathbf{X}$ to zero and letting $\mathbf{G} = \mathbf{G}_x^T \mathbf{G}_x + \mathbf{G}_y^T \mathbf{G}_y$, we derive

$$\begin{aligned} [(\mathbf{H}_{k+1}^T \mathbf{H}_{k+1} + \lambda_g) \mathbf{G} + (\lambda_c + \lambda_s) \sum_j \mathbf{Q}_j^T \mathbf{Q}_j] \hat{\mathbf{x}}_{k+1} = \\ \mathbf{H}_{k+1}^T \mathbf{G} \mathbf{Y} + \lambda_c \sum_j \mathbf{Q}_j^T \mathbf{D} \boldsymbol{\alpha}_j + \lambda_s \sum_j \mathbf{Q}_j^T \sum_{i \in \mathcal{S}_j} w_i^j \mathbf{R}_i \hat{\mathbf{x}}_{k+1}^a \end{aligned} \quad (16)$$

Since both sparse representation coefficients α_j and the down-sampled image $\hat{X}_{k+1}^a$ on the right-hand side of (16) depend on unknown $\hat{X}_{k+1}$, there is no closed-form solution available for solving (16). We solve approximately (16) with the following procedure:

1) Reconstruct Z_c through sparse reconstruction

The K-SVD algorithm [1] is used to obtain the dictionary by approximately solving (4). For each patch $\mathbf{Q}_j \hat{\mathbf{X}}_k$ in $\hat{\mathbf{X}}_k$, the OMP algorithm [29] is used here to derive the sparse representation coefficient α_j over the dictionary $\mathbf{D}$ by approximately solving (5). Another algorithms solve a convex relaxed version of the problem by replacing the ℓ_0 by an ℓ_1 -norm, called the ℓ_1 -minimization algorithm. Yang et al. have showed through experiments that the OMP algorithm outperforms all ℓ_1 -minimization algorithms in terms of success rate in the ideal scenario where the data noise is low, and is still effective for signals with high sparsity when the data are noisy [33]. In our method, we consider three reasons for directly solving the ℓ_0 minimization: first, the sparse representation problem is separately solved as formulated in (17); second, the sparse constraint parameter is extremely low relative to the size of the dictionary; and third, the OMP algorithm has simple, fast implementations [28].

Because the sparse representation coefficient α_j on the right-hand side of (16) depends on unknown $\hat{X}_{k+1}$, we approximate $\hat{X}_{k+1}$ using $\hat{X}_k$ to solve the the sparse representation coefficient $\hat{\alpha}_j$ over the dictionary $\mathbf{D}$, as follows:

$$\hat{\alpha}_j = \underset{\alpha_j}{\operatorname{argmin}} \|\mathbf{Q}_j \hat{\mathbf{X}}_k - \mathbf{D} \alpha_j\|_2^2 \quad \text{s.t.} \quad \|\alpha_j\|_0 \leq T \quad (17)$$

The reconstructed image patch $\mathbf{Q}_j \hat{\mathbf{X}}_k$ can be represented sparsely over $\mathbf{D}$, and the representation coefficient is $\hat{\alpha}_j$, that is, $\mathbf{Q}_j \hat{\mathbf{X}}_k = \mathbf{D} \hat{\alpha}_j$. Then the whole image can be reconstructed by averaging all reconstructed image patches $\mathbf{D} \hat{\alpha}_j$, such that

$$\mathbf{Z}_c = \left(\sum_j \mathbf{Q}_j^T \mathbf{Q}_j \right)^{-1} \sum_j \mathbf{Q}_j^T \mathbf{D} \hat{\alpha}_j \quad (18)$$

where $\mathbf{Z}_c$ is the reconstructed latent image individually through sparse reconstruction.

2) Reconstruct Z_s through cross-scale non-local regularization

For the same reason, since $\hat{X}_{k+1}$ is unknown, we approximate $\hat{X}_{k+1}$ and $\hat{X}_{k+1}^a$ using $\hat{X}_k$ and $\hat{X}_k^a$ respectively. For each patch $\mathbf{Q}_j \hat{\mathbf{X}}_k$ in $\hat{\mathbf{X}}_k$, we search for its similar patches $\mathbf{R}_i \hat{\mathbf{X}}_k^a$, $i \in \hat{\mathcal{S}}_j$ in the down-sampled image $\hat{\mathbf{X}}_k^a$ of $\hat{\mathbf{X}}_k$, and use the linear combination of these similar patches $\sum_{i \in \hat{\mathcal{S}}_j} \hat{w}_i^j \mathbf{R}_i \hat{\mathbf{X}}_k^a$ to predict the patch $\mathbf{Q}_j \hat{\mathbf{X}}_k$, that is,

$$\mathbf{Q}_j \hat{\mathbf{X}}_k \approx \sum_{i \in \hat{\mathcal{S}}_j} \hat{w}_i^j \mathbf{R}_i \hat{\mathbf{X}}_k^a \quad (19)$$

where $\hat{\mathcal{S}}_j$ and $\hat{w}_i^j$ are updated according to $\hat{\mathbf{X}}_k$ and $\hat{\mathbf{X}}_k^a$. Then the whole image can be reconstructed by averaging all reconstructed image patches $\sum_{i \in \hat{\mathcal{S}}_j} \hat{w}_i^j \mathbf{R}_i \hat{\mathbf{X}}_k^a$ such that

$$\mathbf{Z}_s = \left(\sum_j \mathbf{Q}_j^T \mathbf{Q}_j \right)^{-1} \sum_j \mathbf{Q}_j^T \sum_{i \in \hat{\mathcal{S}}_j} \hat{w}_i^j \mathbf{R}_i \hat{\mathbf{X}}_k^a \quad (20)$$

where $\mathbf{Z}_s$ is the reconstructed latent image individually through cross-scale non-local regularization.

3) Given $\mathbf{Z}_c$ and $\mathbf{Z}_s$, solve $\hat{\mathbf{X}}_{k+1}$

Substituting $\sum_j \mathbf{Q}_j^{\text{TD}} \boldsymbol{\alpha}_j$ with $n\mathbf{Z}_c$ and $\sum_j \mathbf{Q}_j^{\text{T}} \sum_{i \in S_j} w_i^j \mathbf{R}_i \hat{\mathbf{x}}_{k+1}^a$ with $n\mathbf{Z}_s$ as an approximation, (16) can be rewritten as:

$$[(\mathbf{H}_{k+1}^{\text{T}} \mathbf{H}_{k+1} + \lambda_g) \mathbf{G} + (\lambda_c + \lambda_s) n \mathbf{I}] \hat{\mathbf{x}}_{k+1} = \mathbf{H}_{k+1}^{\text{T}} \mathbf{G} \mathbf{Y} + \lambda_c n \mathbf{Z}_c + \lambda_s n \mathbf{Z}_s \quad (21)$$

It is easy to verify that $\sum_j \mathbf{Q}_j^{\text{T}} \mathbf{Q}_j = n \mathbf{I}$ [17], here n is the size of image patch, and $\mathbf{I}$ is the identity matrix of size N . Substituting (18), (20) and this term into (16) leads to (21). Since it is a linear equation with respect to $\hat{\mathbf{x}}_{k+1}$, (21) can be solved by direct matrix inversion or the conjugate gradient method. We solve it in the frequency domain and the closed-form solution is given by:

$$\hat{\mathbf{x}}_{k+1} = \mathcal{F}^{-1} \left(\frac{\overline{\mathcal{F}(\hat{\mathbf{h}}_{k+1})} (\overline{\mathcal{F}(\partial_x)} \mathcal{F}(\partial_x) + \overline{\mathcal{F}(\partial_y)} \mathcal{F}(\partial_y)) \mathcal{F}(\mathbf{y}) + \lambda_c n \mathcal{F}(\mathbf{z}_c) + \lambda_s n \mathcal{F}(\mathbf{z}_s)}{(\overline{\mathcal{F}(\hat{\mathbf{h}}_{k+1})} \mathcal{F}(\hat{\mathbf{h}}_{k+1}) + \lambda_g) (\overline{\mathcal{F}(\partial_x)} \mathcal{F}(\partial_x) + \overline{\mathcal{F}(\partial_y)} \mathcal{F}(\partial_y)) + \lambda_c n + \lambda_s n} \right) \quad (22)$$

where $\mathbf{z}_c$ and $\mathbf{z}_s$ represent $\mathbf{Z}_c$ and $\mathbf{Z}_s$ in 2-D image form, respectively.

3.4 Implementation

To speed up the convergence and handle of large blurs, following most existing methods, we estimate the blur kernel in a coarse-to-fine framework. That is, we apply our blind deconvolution model as solved in Section 3.3 using an approximate alternating iterative optimization procedure to each of the levels of the image pyramid constructed from the blurry image $\mathbf{y}$. At the coarsest scale level, the latent image estimate is initialized with the observed blurry image. The intermediate latent image estimated at each coarser level is interpolated and then propagated to the next finer level as an initial estimate of the latent image to progressively refine the blur kernel estimate in higher resolutions. The intermediate latent images estimated during the iterations have no direct influence on the final deblurring result, and only affect this result indirectly by contributing to the refinement of the blur kernel estimate $\hat{\mathbf{h}}$.

At the coarsest scale level, the dictionary learning uses the down-sampled blurry image as training samples. To better represent the latent image over the learned dictionary, we update the learned dictionary using the down-sampled intermediate latent image estimate as training samples. In the implementation of our coarse-to-fine iterative framework for estimating the blur kernel, the intermediate latent image estimated at the coarser scale is directly used for training the dictionary and the dictionary is iteratively updated once for each image scale during the solution.

We estimate the blur kernel $\mathbf{h}$ by the implementation of the pseudo-code outlined in Algorithm 1. We construct an image pyramid with L levels from an input blurry image $\mathbf{y}$. The number of pyramid levels is chosen such that, at the coarsest scale level, the size of the blur is smaller than that of the patch used in the blur kernel estimation stage. Let us use the notation $\hat{\mathbf{x}}_k^l$ for the intermediate latent image estimate, where the superscript l indicates the l th level in the image pyramid, while the subscript k indicates the k th iteration at each scale level. The blur kernel estimation starts from the coarsest scale level $l = 1$ of the image pyramid with the latent image initialized as $\hat{\mathbf{x}}_0^1 = \mathbf{y}$. At each scale level $l \in \{1, \dots, L\}$, we take the iterative procedure that alternately optimizes the motion blur kernel $\mathbf{h}$ and the latent image $\mathbf{x}$, which is implemented repeatedly until the convergence or for a fixed number of iterations. Then the outcome of updating the latent image at the l th level is upsampled by interpolation and then used as an initial estimate of the latent image for the next finer level $l + 1$ to progressively refine the motion blur kernel estimate $\hat{\mathbf{h}}$, which is repeated to achieve the final refinement of the blur kernel estimate $\hat{\mathbf{h}}$ for the finest level.

Algorithm 1 Blur kernel estimation.**Input:** Blurry image y **Output:** Blur kernel estimate $\hat{h}$

Set down-scaling factor a , regularization weights $\lambda_g, \lambda_c, \lambda_s, \lambda_h$, size of patch n , size of dictionary t , sparsity constraint parameter T , number of similar patches p , convergence tolerance ϵ and maximum number of iterations maxIters ;

Build an image pyramid with L levels;

Initialize $\hat{x}_0^1 = y$;

Train dictionary $\mathbf{D}$ using $\hat{x}_0^1$;

Outer loop: for $l = 1$ to $l = L$ do // for each level of image pyramid

Initialize $k = 0$;

Inner loop: repeat // for each iteration

Compute blur kernel $\hat{h}_{k+1}^l$ using (13);

Given $\hat{x}_k^l$, compute $\mathbf{Z}_c$ using (18) and $\mathbf{Z}_s$ using (20);

Update latent image $\hat{x}_{k+1}^l$ using (22);

$k = k + 1$

until $k > \text{maxIters}$ or $\|\hat{x}_k^l - \hat{x}_{k-1}^l\|_2^2 \leq \epsilon$;

Update dictionary $\mathbf{D}$ using $\hat{x}_k^l$;

Upscale image $\hat{x}_k^l$ to initialize $\hat{x}_0^{l+1}$ for the next finer level;

$\hat{h} = \hat{h}_k^L$; $\hat{x} = \hat{x}_k^L$.

In the blur kernel estimation process, we use the gray-scale versions of the blurry image y and the intermediate latent image estimate $\hat{x}$. Once the blur kernel estimate $\hat{h}$ has been obtained with the original image scale, we perform the final non-blind deconvolution with $\hat{h}$ on each color channel of y to obtain the deblurring result.

Finally, our method need perform deconvolution in the Fourier domain. To avoid ringing artifacts at the image boundaries, we process the image near the boundaries using the simple *edgetaper* command in Matlab.

4 Experiments

Several experiments are conducted to demonstrate the performance of our method. We first test our method on the widely used dataset introduced in [14] and [27], and make qualitative and quantitative comparisons with the state-of-the-art blind deblurring methods. Then we show visual comparisons on real blurry photographs with unknown blurs. The relevant parameters of our method are set as follows: the dictionary $\mathbf{D}$ is of size $t = 100$, and the sparsity constraint parameter $T = 4$, designed to handle image patches of size $n = 5 \times 5$, the maximum number of iterations maxIters is fixed as 14 for the inner loop, and the regularization weights are empirically set to $\lambda_c = 0.15/n$, $\lambda_s = 0.15/n$, $\lambda_g = 0.001$ and $\lambda_h = 0.0015N$. As the down-scaling factor increases, the patches at the down-sampled scale get sharper, but there exist less similar patches at the down-sampled scale. Following the setting of [17], the image pyramid is constructed with scale-gaps of $a = 4/3$ using down-scaling with a sinc function. Additional speed up is obtained by using the fast approximate nearest neighbor (NN) search of [18] in the blur kernel estimation stage, working with a single NN for every patch.

Fig. 4 Cumulative distributions of error ratios with various methods on the dataset of [14]

An additional parameter is the size of the blur kernel. Small blurs are hard to solve if it is initialized with a very large kernel. Conversely, large blurs will be truncated if too small a kernel is used [6]. Following the setting of [27], we do not assume that the size of the kernel is known and initialize that the size of the kernel is 51×51 . Experiment results on both simulated and real blurry images show the size of the blur kernel is generally no larger than 51×51 for most of blurry images. Despite an input blurry image with a small blur kernel, our method is still able to obtain a good deblurring result, relatively insensitive to the initial setting of the kernel size.

4.1 Quantitative evaluation on synthetic datasets

We test our method on two publicly available datasets. One dataset, which is provided by Levin et al. [14], contains 32 images of size 255×255 blurred with 8 different kernels. The kernels range in size from 13×13 to 27×27 . The blurred images with spatially invariant blur and ground-truth kernels were captured simultaneously by locking the Z-axis rotation handle but loosening the X and Y handles of the tripod. The other dataset provided by Sun

Table 1 Quantitative comparison of various methods over the dataset of [14]

	Success rate%	Mean error ratio
Ours	96.88	1.4653
Yu et al. [35]	93.75	1.7406
Perrone et al. [24]	93.75	1.2024
Xu & Jia [31]	93.75	2.1365
Perrone & Favaro [22]	87.50	2.0263
Levin et al. [15]	87.50	2.0583
Fergus et al. [6]	75.00	13.5268
Cho & Lee [3]	68.75	2.6688

Fig. 5 Cumulative distributions of error ratios with various methods on the dataset of [27]

et al. [27] comprises 640 large natural images of diverse scenes, which were obtained by synthetically blurring 80 high-quality images with the 8 blur kernels from [14] and adding 1% white Gaussian noise to the blurred images. We present qualitative and quantitative comparisons with the state-of-the-art blind deblurring methods [3, 4, 6, 11, 15, 17, 24, 22, 27, 31].

We measure the quality of the blur kernel estimate $\hat{h}$ using the error ratio measure ER [17]:

$$ER = \frac{\|x - \hat{x}_{\hat{h}}\|_2^2}{\|x - \hat{x}_h\|_2^2} \quad (23)$$

where $\hat{x}_{\hat{h}}$ represents the deblurring result with the recovered kernel $\hat{h}$, and $\hat{x}_h$ represents the deblurring result with the ground-truth kernel h . The smaller ER corresponds to the better quality. In principle, if $ER = 1$, the recovered kernel yields a deblurring result as good as the ground-truth kernel.

Table 2 Quantitative comparison of various methods over the dataset of [27]

	Success rate%	Mean error ratio
Ours	96.25	2.2047
Yu et al. [35]	96.88	2.2181
Michaeli & Irani [17]	95.94	2.5662
Sun et al. [27]	93.44	2.3764
Xu & Jia [31]	85.63	3.6293
Levin et al. [15]	46.72	6.5577
Cho & Lee [3]	65.47	8.6901
Krishnan et al. [11]	24.49	11.5212
Cho et al. [4]	11.74	24.7020

(a) Blurry image (b) Ground truth (c) Cho and Lee [4] (d) Xu & Jia [5] (e) Krishnan et al. [36]

(f) Cho et al. [35] (g) Levin et al. [7] (h) Sun et al. [15] (i) Michaeli & Irani [16] (j) Our method

Fig. 6 Qualitative comparison of various methods on a cropped image from the synthetic dataset provided by Sun et al. [27]

On the dataset provided by Levin et al. [14], we compare our error ratios with those of Fergus et al. [6], Cho and Lee [3], Xu and Jia [31], Perrone and Favaro [22], Levin et al. [15] and Perrone et al. [24]. Figure 4 shows the cumulative distribution of the error ratio of our method compared with the other methods over the dataset of [14]. Levin et al. [15]

(a) Blurry image (b) Ground truth (c) Cho and Lee [4] (d) Xu & Jia [5] (e) Krishnan et al. [36]

(f) Cho et al. [35] (g) Levin et al. [7] (h) Sun et al. [15] (i) Michaeli & Irani [16] (j) Our method

Fig. 7 Qualitative comparison of various methods on another cropped image from the synthetic dataset of [27]

(a) Blurry image

(b) Xu & Jia [5]

(c) Levin et al. [7]

(d) Krishnan et al. [36]

(e) Sun et al. [15]

(f) Michaeli et al. [16]

(g) Perrone & Favaro [9]

(h) Perrone et al. [37]

(i) Pan et al. [18]

(j) Yan et al. [19]

(k) Yu et al. [30]

(l) Our method

Fig. 8 Visual comparison between our method and some state-of-the-art methods on another real blurry image with unknown blur

use sparse deconvolution [14] to generate the final results, and observe that deconvolution results are usually visually plausible when their error ratios are below 3. Therefore, we standardize the final non-blind deconvolution by using sparse deconvolution [14] to obtain the results, for fair comparison. Table 1 lists the success rate and the average error ratio over 32 images for each method. The success rate is the percentage of images which obtain good deblurring results, that is, the percentage of images that have an error ratio below a certain

Fig. 9 Visual comparison between our method and some state-of-the-art methods on a real blurry image with unknown blur

threshold. On this dataset, the success rate is the percentage of the results under the error ratio of 3. Table 1 shows our method takes the lead with a success rate of 96.88%. Levin et al. [15], Perrone and Favaro [22] and Perrone et al. [24] initialize the size of the blur kernel with ground truth. However it is an unknown parameter for real applications. Notwithstanding, our method still achieves a much higher success rate than the other methods over the dataset of [14].

On the dataset provided by Sun et al. [27], we compare our error ratios with those of Cho and Lee [3], Xu and Jia [31], Levin et al. [15], Sun et al. [27], Michaeli and Irani [17], Cho et al. [4] and Krishnan et al. [11]. Figure 5 shows the cumulative distribution of error ratios over the entire dataset for each method. We apply the blur kernel estimated by each method to perform deblurring with the non-blind deblurring method of [37] to recover latent images. It is empirically observed by Michaeli and Irani [17] that the deblurring results are still visually acceptable for error ratios $ER \leq 5$, when using the non-blind deconvolution of [37]. Table 2 lists the success rate (i.e., an error ratio below 5) and the average error ratio over 640 images with different methods. Table 2 shows our method achieves the highest success rate and the lowest average error ratio followed by Michaeli and Irani [17] and Sun et al. [27].

Figures 6 and 7 show qualitative comparisons of cropped results by different methods from the synthetic dataset of [27]. Compared with the other methods, our method usually

obtains more accurate blur kernels, suffers from much less ringing artifact and restores sharp edges and fine details better.

4.2 Qualitative comparison on real images

We also experiment with real blurry images which are blurred with unknown kernels to demonstrate the robustness of our method. In this part, we also use the non-blind deconvolution of [37] to recover latent images in the deblurring stage once the blur kernel has been estimated. Figures 8 and 9 show visual comparison examples with the state-of-the-art blind deconvolution methods [11, 15, 17, 21, 24, 27, 31, 32], at the bottom of which are close-ups of different parts of these images. We observe from Fig. 8 that the deblurred images by Pan et al. [21] and Yan et al. [32] respectively suffer from noise and ringing artifacts and tend to be smooth. Experimental results on real blurry photographs with unknown blurs validate that our method can obtain robust blur kernels and restore sharp details with negligible artifacts.

5 Conclusion

In this paper, we have presented a blur kernel estimation method for blind motion deblurring using sparse representation and cross-scale self-similarity of image patches as priors to regularize the inverse problem of recovering the latent image. Since patches repeat across scales in a single image, our priors thoroughly exploit the additional information provided by cross-scale similar patches at down-sampled scales of the intermediate latent image that are sharper and more similar to patches from the latent sharp image by employing the cross-scale dictionary learning and the cross-scale non-local regularization. On the one hand, the cross-scale dictionary learning uses patches from the intermediate latent image estimated at the coarser level of the image pyramid as training samples and updates the dictionary once for each image scale to ensure the sparsity of the latent image over this dictionary. On the other hand, the cross-scale non-local regularization optimizes all patches from the intermediate latent image estimate to be as close to the similar patches searched from down-sampled version to enforce sharp recovery of the latent image as possible. We have extensively validated the performance of our method through experiments on both simulated and real blurry images, and demonstrated that our method can remove effectively complex motion blurs from nature images and obtain satisfactory deblurring results, thanks to the use of cross-scale similar patches.

Funding Information This study was funded by National Natural Science Foundation of China (61501008) and Beijing Municipal Natural Science Foundation (4172002).

Compliance with Ethical Standards

Conflict of interests The authors declare that they have no conflicts of interest.

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.

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

References

- Aharon M, Elad M, Bruckstein A (2006) K-svd: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans Signal Process* 54(11):4311–4322
- Buades A, Coll B, Morel J-M (2005) A non-local algorithm for image denoising. In: 2005 IEEE Computer society conference on computer vision and pattern recognition (CVPR), 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR), IEEE, San Diego, pp 60–6
- Cho S, Lee S (2009) Fast motion deblurring. *ACM Trans Graph* 28(5):89–97
- Cho TS, Paris S, Horn BKP, Freeman WT (2011) Blur kernel estimation using the radon transform. In: IEEE conference on computer vision and pattern recognition (CVPR), vol 42 of IEEE conference on computer vision and pattern recognition (CVPR), Providence, pp 241–248
- Dong W, Zhang L, Shi G, Wu X (2011) Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *IEEE Trans Image Process* 20(7):1838–1857
- Fergus R, Singh B, Hertzmann A, Roweis ST, Freeman WT (2006) Removing camera shake from a single photograph. *ACM Trans Graph* 25(3):787–794
- Glasner D, Bagon S, Irani M (2009) Super-resolution from a single image. In: International conference on computer vision, ICCV 2009, international conference on computer vision, ICCV 2009, IEEE, Kyoto, pp 349–356
- Jia J (2007) Single image motion deblurring using transparency. In: IEEE conference on computer vision and pattern recognition (CVPR), IEEE conference on computer vision and pattern recognition (CVPR), IEEE, Minneapolis, pp 1–8
- Jia C, Evans BL (2011) Patch-based image deconvolution via joint modeling of sparse priors. In: IEEE international conference on image processing (ICIP), IEEE international conference on image processing (ICIP), IEEE, Brussels, pp 681–684
- Joshi N, Szeliski R, Kriegman D (2008) Psf estimation using sharp edge prediction. In: IEEE conference on computer vision and pattern recognition (CVPR), IEEE conference on computer vision and pattern recognition (CVPR), IEEE, Anchorage, pp 1–8
- Krishnan D, Tay T, Fergus R (2011) Blind deconvolution using a normalized sparsity measure. In: IEEE conference on computer vision and pattern recognition (CVPR), IEEE conference on computer vision and pattern recognition (CVPR), IEEE, Providence, pp 233–240
- Lai WS, Ding JJ, Lin YY, Chuang YY (2015) Blur kernel estimation using normalized color-line priors. In: IEEE conference on computer vision and pattern recognition (CVPR), IEEE conference on computer vision and pattern recognition (CVPR), IEEE computer society, Boston, pp 64–72
- Levin A, Fergus R, Durand FED, Freeman WT Image and depth from a conventional camera with a coded aperture. *ACM Transactions on Graphics (TOG)* 26(3)
- Levin A, Weiss Y, Durand F, Freeman WT (2009) Understanding and evaluating blind deconvolution algorithms. In: IEEE conference on computer vision and pattern recognition, IEEE conference on computer vision and pattern recognition, IEEE, Miami, pp 1964–1971
- Levin A, Weiss Y, Durand F, Freeman WT (2011) Efficient marginal likelihood optimization in blind deconvolution. In: IEEE conference on computer vision and pattern recognition (CVPR), IEEE conference on computer vision and pattern recognition (CVPR), IEEE, Providence, pp 2657–2664
- Li H, Zhang Y, Zhang H, Zhu Y, Sun J (2012) Blind image deblurring based on sparse prior of dictionary pair. In: International conference on pattern recognition (ICPR), international conference on pattern recognition (ICPR), IEEE, Tsukuba, pp 3054–3057
- Michaeli T, Irani M (2014) Blind deblurring using internal patch recurrence. In: European conference on computer vision (ECCV), European conference on computer vision (ECCV), Springer International Publishing, Zurich, pp 783–798
- Olonetsky I, Avidan S, Treccani K-d (2012) Tree coherence approximate nearest neighbor algorithm. In: European conference on computer vision, European conference on computer vision, Springer, Berlin, pp 602–615
- Pan Z, Yu J, Huang H, Hu S, Zhang A, Ma H, Sun W (2013) Super-resolution based on compressive sensing and structural self-similarity for remote sensing images. *IEEE Trans Geosci Remote Sens* 51(9):4864–4876
- Pan Z, Yu J, Hu S, Sun W (2014) Single image super resolution based on multi-scale structural self-similarity. *Acta Automatica Sinica* 40(4):594–603
- Pan J, Sun D, Pfister H, Yang MH (2016) Blind image deblurring using dark channel prior. pp 1628–1636

22. Perrone D, Favaro P (2014) Total variation blind deconvolution: the devil is in the details. In: IEEE conference on computer vision and pattern recognition (CVPR), IEEE conference on computer vision and pattern recognition (CVPR), IEEE, Columbus, pp 2909–2916
23. Perrone D, Favaro P (2016) A clearer picture of total variation blind deconvolution. *IEEE Trans Pattern Anal Mach Intell* 38(6):1041–1055
24. Perrone D, Diethelm R, Favaro P (2015) Blind deconvolution via lower-bounded logarithmic image priors. In: International conference on energy minimization methods in computer vision and pattern recognition (EMMCVPR), international conference on energy minimization methods in computer vision and pattern recognition (EMMCVPR), Springer International Publishing, Hong Kong
25. Protter M, Elad M, Takeda H, Milanfar P (2009) Generalizing the nonlocal-means to super-resolution reconstruction. *IEEE Trans Image Process* 18(1):36–51
26. Shan Q, Jia J, Agarwala A (2008) High-quality motion deblurring from a single image. *ACM Trans Graph* 27(3):15–19
27. Sun L, Cho S, Wang J, Hays J (2013) Edge-based blur kernel estimation using patch priors. In: IEEE international conference on computational photography (ICCP), IEEE international conference on computational photography (ICCP), IEEE, Cambridge, pp 1–8
28. Tropp JA (2004) Greed is good: algorithmic results for sparse approximation. *IEEE Trans Inf Theory* 50(10):2231–2242. <https://doi.org/10.1109/TIT.2004.834793>
29. Tropp JA, Gilbert AC (2007) Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans Inf Theory* 53(12):4655–4666
30. Wang M, Yu J, Sun W (2015) Group-based hyperspectral image denoising using low rank representation. In: 2015 IEEE International conference on image processing (ICIP), 2015 IEEE international conference on image processing (ICIP), pp 1623–1627
31. Xu L, Jia J (2010) Two-phase kernel estimation for robust motion deblurring. In: European conference on computer vision: Part I, European conference on computer vision: Part I, Springer, Berlin Heidelberg, pp 157–170
32. Yan Y, Ren W, Guo Y, Wang R, Cao X (2017) Image deblurring via extreme channels prior. pp 6978–6986
33. Yang A, Ganesh A, Sastry S, Ma Y (2010) Fast ℓ_1 -minimization Algorithms and an Application in Robust Face Recognition: A Review, Tech. Rep. UCB/EECS-2010-13, EECS Department. University of California, Berkeley. <http://www2.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-13.html>
34. Yang J, Wright J, Huang TS, Ma Y (2010) Image super-resolution via sparse representation. *IEEE Trans Image Process* 19(11):2861–2873
35. Yu J, Chang Z, Xiao C, Sun W (2017) Blind image deblurring based on sparse representation and structural self-similarity. In: 2017 IEEE International conference on acoustics, speech and signal processing (ICASSP), 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, New Orleans, pp 1328–1332
36. Zhang H, Yang J, Zhang Y, Huang TS (2011) Sparse representation based blind image deblurring. In: IEEE international conference on multimedia and expo (ICME), IEEE international conference on multimedia and expo (ICME), IEEE, Barcelona, pp 1–6
37. Zoran D, Weiss Y (2011) From learning models of natural image patches to whole image restoration. In: IEEE international conference on computer vision (ICCV), IEEE international conference on computer vision (ICCV), IEEE, Barcelona, pp 479–486

Jing Yu received the Ph.D. degree from the Department of Electronic Engineering, Tsinghua University in 2011. She is currently a lecturer at the Faculty of Information Technology, Beijing University of Technology. Her current research interests are in inverse problems (blind deconvolution, super resolution, deblurring and denoising) and sparse representations.

Zhenchun Chang received his bachelor degree from Tsinghua University in 2014. He is currently master candidate at the Department of Electronic Engineering, Tsinghua University. His research interests are in image deblurring and sampling.

Chuangbai Xiao received the Ph.D. degree from the Department of Automation, Tsinghua University in 1995. He is currently a professor at the Faculty of Information Technology, Beijing University of Technology. His research interest are in digital image processing, audio and video signal processing, and pattern recognition