Blind image deblurring via coupled sparse representation

Ming Yin a, Junbin Gao b,⇑, David Tien b, Shuting Cai a

a School of Automation, Guangdong University of Technology, Guangzhou, China
b School of Computing and Mathematics, Charles Sturt University, Bathurst, NSW 2795, Australia

ABSTRACT

The problem of blind image deblurring is more challenging than that of non-blind image deblurring, due to the lack of knowledge about the point spread function in the imaging process. In this paper, a learning-based method of estimating blur kernel under the $\ell_0$ regularization sparsity constraint is proposed for blind image deblurring. Specifically, we model the patch-based matching between the blurred image and its sharp counterpart via a coupled sparse representation. Once the blur kernel is obtained, a non-blind deblurring algorithm can be applied to the final recovery of the sharp image. Our experimental results show that the visual quality of restored sharp images is competitive with the state-of-the-art algorithms for both synthetic and real images.

1. Introduction

Blurred images occur when the image acquiring process is influenced by the relative movement between the camera and objects during the exposure time [20]. A clean image from a blurred image can be obtained through solving the problem of image deblurring, which is one of challenging problems in image restoration. Image deblurring has been extensively studied in recent years [4,21].

In general, the degradation process of the blurred images is often modelled as a convolution followed by a noising process,

$$ Y = k \otimes X + n $$

where $Y$ is a blurred image, i.e. observed image, $X$ is a sharp image, i.e. unknown latent (clean/sharp) image, $k$ is a spatial-invariant kernel, i.e. point spread function (PSF) and $n$ is independent white Gaussian noise added to the convoluted image. The problem of image deblurring is to recover the latent image $X$ from the observed blurred one $Y$, which is also regarded as a deconvolution processing.

The process of image deblurring falls into two categories. If the blur kernel $k$ is known or well estimated, then the restoration of $X$ from $Y$ is considered as a non-blind deconvolution problem. If, on the other hand, there is very little or no information about the blur kernel $k$, the problem is regarded as blind deblurring.

Unfortunately, in most of the real-world cases, we have no knowledge of the exact kernel $k$ for a blurred image. Thus, most of the time, we have to face the challenging blind deconvolution problems [2,24,26]. This results in well-known, but ill-posed and difficult problems. In this paper, we focus on the issue of blind image deblurring.

Recently, many methods have been proposed to solve this kind of problems [5]. These methods can be roughly divided into two classes based on the ways of how blur kernels are identified or learned. In the first class, a blur kernel is identified independently from a single blurred image [14]. In the second class, both the blur kernel and latent (clean or sharp) image are estimated simultaneously via a learning process [5,6]. In most of the methods of the second class, a prior knowledge [9,14,17] about both latent image and blur kernel is exploited, for example in total variation and Bayesian paradigms [2,24]. Once the blur kernel is well estimated, the problem can be simplified into a non-blind deconvolution issue, which can be effectively solved by the popular sparse representation methods [3,29,31].

However, many hidden mappings between the blurred images and the latent images are not well exploited during estimating blur kernels [8,10]. For example, a motion blurred image [10] usually retains information about motion which gives us clues to recover motion from this blurred image by parameterizing the blurring model. The sparse representation of image is more helpful in estimating an appropriate blur kernel and its relevant latent image [8]. The advancements in sparse representation of signals [7,12]
and learning their sparsifying dictionaries [1] make it more effective in solving the image restoration problems.

In this paper, we focus on the spatially invariant blurring issue. Our goal is to recover the true blur kernel based on the assumption that there exists a coupled sparse representation between the blurred and sharp (latent) image patch pairs under their own dictionaries, respectively. This type of hidden mappings is helpful in estimating the unknown blur kernel. Then, with the estimated blur kernel, we can recover the latent image by a variety of non-blind deconvolution methods. The main contribution of this paper to the literature is summarized as the integration of the sparse representation of blurred image and its corresponding latent patch into a unified framework for optimization. Specifically, in addition to requiring that the coding coefficients of each local patch are sparse, we also enforce the compatibility of sparse representations between the blurred and latent image patch pair with respect to their own dictionaries. Once the blur kernel is learned, we restore the latent clean image by imposing a sparse prior regularization over the image derivatives as done in [18]. This regularization allows a robust recovery even for a possibly lower accurate kernel estimate and is an effective way to solve the non-blind deconvolution issue. This paper is organized as follows. In Section 2, we briefly review the related works on the approaches of blind image delurring. The blind image deblurring based sparsity prior is proposed in Section 3. The extensive experimental results are provided in Section 4. The conclusion is drawn in Section 5.

2. Related works

Due to its inherently ill-posed problem, a blind image deblurring process needs to be regularized by image priors for better solution. We briefly review a variety of approaches on blind image delurring in the following.

Generally, natural image statistics can be used as appropriate image priors, e.g., in constraining the output of local derivative operators. Among many priors, the Gaussian smoothness penalty is the simplest one, which has been widely used in blind image deconvolution [9,14,17]. Fergus et al. [14] first estimated a blur kernel using a prior on gradient distribution of natural images in a variational Bayes framework [2,24]. In [9], a computationally efficient Gaussian prior is applied to estimate the latent image. However, since images may be highly non-Gaussian, these Gaussian prior based approaches may favor noisy or dense kernel estimates.

Therefore, given that the distribution of image derivatives is well modeled by a hyper-Laplacian, a non-blind deconvolution approach proposed in [18] provides competitive performance with several orders of fast speed.

Instead of using statistical priors, Joshi et al. [17] proposed an algorithm to estimate blur functions at sub-pixel resolution, which is fulfilled by estimating regions of a sharp image from a single blurred image. For spatially non-uniform blur, Gupta et al. [15] proposed a spatially invariant deconvolution method, where the blurring was represented as motion density function (MDF) to compute initial estimates of the latent image.

Recently, the advancements in sparse representation of signals [7,12] and in learning their sparsifying dictionaries [1] have made it easier to solve the image deblurring problem [14,16,26]. Shan et al. [28] applied the sparse priors for both the latent image and blur kernel under an alternating-minimization scheme. Cai et al. [8] exploited a framelet and curvelet system to sparsely represent both blur kernel and sharp image. Finally, a sparse representation based on incremental iterative method was established for blurred image restoration [30]. Although the above methods have achieved impressive progress on blind deblurring, the quality of the recovered sharp image is far from perfect. In [22], Lin et al. proposed a framework, called Coupled Space Learning, to learn the relations between the image patches lying in different spaces. In statistical learning, the relationship between two image patch pairs can be regarded as the mapping between two vector spaces associated with two image styles. Inspired by this observation, we integrate the couple concept into the sparse representation framework and propose a novel method to learn the unknown blur kernel for image blind deblurring.

3. Learning-based blind image deblurring method

3.1. Conceptual framework

Consider a patch \( x \) extracted from a latent image \( X \) and assume that \( x \) can be well represented as a sparse linear combination over an appropriate dictionary \( D \). This dictionary \( D \) may be trained from those true sample patches extracted from training images. In general, we assume

\[
\mathbf{x} \approx \mathbf{D} \mathbf{z}, \quad \|\mathbf{z}\|_0 \ll m \quad \text{with} \quad \mathbf{z} \in \mathbb{R}^m. \tag{2}
\]

Such a sparse representation can be helpful in image processing tasks [7,12,30]. For example, the sparse representation approach has been widely used in image denoising [13]. The idea is based on the assumption that all the image patches can be adequately approximated by a sparse linear combination of a learned patch dictionary. Let \( R_i \) denote the extraction operator for a patch \( x \), from the given image \( X \) at location \( i \), i.e., \( x_i = R_i X \). Assume that all the patches \( \{x_i\} \) of the given image \( X \) permit a sparse representation with a set of sparse coefficients \( \Lambda = \{z_i\} \) under a known dictionary \( D \), then the reconstruction of \( X \) from the sparse coefficients can be formulated as an over-determined system and its straightforward least-square solution is given as follows [13],

\[
X^* = D \odot \Lambda = \left( \sum_i R_i^T R_i \right)^{-1} \sum_i (R_i^T D z_i).
\]

This idea can be generalized to our deblurring problem, as proposed below. Let us consider the blurring model in (1). Suppose the sharp latent image patches admit a well defined sparse representation under a dictionary as defined in (2), then based on the linearity of convolution process one can expect a similar sparse representation of the corresponding blurred image patches as defined in the following model,

\[
\mathbf{y} \approx \mathbf{k} \odot \mathbf{D} \mathbf{z}, \quad \mathbf{z} \in \mathbb{R}^m, \quad \|\mathbf{z}\|_0 \ll m. \tag{3}
\]

where \( \mathbf{D} = \mathbf{k} \odot \mathbf{D} \) is regarded as the blurred dictionary under the deblurring process \( k \).

Actually Eqs. (2) and (3) implicitly suggest a hidden relation [22] between the blurred image and the corresponding latent one as a set of the same sparse coefficients is utilized in the model. In real applications, it is reasonable for the sparse representation between blurred image and latent image to be equal by enforcing such coherence of the sparse coefficients, similar to the basic idea in [23,28]. Hence our hypothesis is that similar patch pairs will demonstrate similar sparse decomposition. Thus our approach for the blind deblurring problem is to formulate a coupled sparse representation framework, and then to obtain the blur kernel by using a learning-based minimization strategy and recover the latent image via non-blind deconvolution method.

One of significant difference from [23] is that we offer a learning procedure for blur kernel with certain prior knowledge through the addition of a regularizer term. In summary, our method consists of two stages. The first stage is to estimate the blur kernel from the blurred input \( y \) where the kernel estimation is performed on the high frequency part of the image. This is reasonable since blurring
will reliably attenuate the high frequency part of image [19]. Once acquiring the blur kernel, non-blind deblurring algorithm is applied for the final recovery of the sharp image in the second stage.

3.2. Coupled sparse representation for blurred image

The above discussion requires that the two models (2) and (3) are combined in a consistent way. Given both dictionaries \( \mathbf{D} \) and \( \mathbf{D} \) are known, then a combined formulation of coherent sparse representation for blurred image can be formulated as follows, for each pair of sharp image patch \( \mathbf{x} \) and the corresponding blurred patch \( \mathbf{y} \),

\[
\min_{\mathbf{a}} \| \mathbf{a} \|_0 \quad \text{s.t.} \quad \| \mathbf{y} - \mathbf{D} \mathbf{a} \|_2^2 \leq \varepsilon_1, \quad \| \mathbf{x} - \mathbf{D} \mathbf{a} \|_2^2 \leq \varepsilon_2
\]

(4)

where \( \varepsilon_1 \) and \( \varepsilon_2 \) are two pre-specified error tolerances. It is simpler to deal with the optimization problem by converting it into its equivalent unconstrained version, defined as follows,

\[
\min_{\mathbf{a}} \frac{1}{2} \| \mathbf{y} - \mathbf{D} \mathbf{a} \|_2^2
\]

where \( \mathbf{y} = \frac{\mathbf{H} \mathbf{x}}{\mathbf{f}} \) and \( \mathbf{D} = \left[ \mathbf{k} \otimes \mathbf{D} \right] \).

Since blurring usually affects the high-frequency band of an image [19], the objective function will not change too much if the sparse prior regularizes the whole image, i.e., the entire frequency band. Thus, we apply the sparsity measure to only the high-frequency components of images in this paper. The derivative filters \( f_1 = [1, -1, 1]^T \) and \( f_2 = [1, -1]^T \) will be used to generate a high-frequency version of blurred image patch, for example, the gradient image is obtained by \( \nabla f_1 \mathbf{y} \). Given high-frequency patches \( \mathbf{H}_k \) and \( \mathbf{H}_s \) corresponding to \( \mathbf{y} \) and \( \mathbf{x} \) respectively, these patches are obtained by concatenating two gradient parts extracted by \( f_1 \) and \( f_2 \). Therefore, the cost function for spatially invariant blurring is finally defined by,

\[
\min_{\mathbf{a}} \frac{1}{2} \| \mathbf{H} - \mathbf{D} \mathbf{a} \|_2^2 + \lambda_k \| \mathbf{k} \|_1 \quad \text{s.t.} \quad \| \mathbf{d}_j \|_2 = 1, k > 0
\]

(5)

where \( \mathbf{d}_j \) is the \( j \)-th column of \( \mathbf{D} \) and \( \mathbf{H} = \left[ \mathbf{H}_k \mathbf{H}_s \right] \). To attenuate noise in the kernel, we regularize \( \mathbf{k} \) by \( \ell_1 \) norm and its sum of each column elements is to be normalized to be one in computational procedure. Moreover, the constraint of non-negativity is further imposed by following the physical principles of blur formation. The scalar weights \( \lambda \) and \( \lambda_k \) are exploited to control the relative strength of the kernel and image sparsity regularization terms. Note that we here adopt \( \ell_1 \) norm regularization on sparse-coding for efficiently taking advantage of the greedy K-SVD algorithm [1] in our proposed procedure, however we use \( \ell_1 \) instead of \( \ell_0 \) norm on the blur kernel in favor of relaxed sparsity.

3.3. Iteratively estimate the blur kernel

In blind deblurring, using an appropriate blur kernel is a key factor on restoring latent images. However, generally speaking, it is difficult to accurately estimate a blur kernel by solving a highly non-convex optimization problem like (5). The traditional method is to start with an initialization on all unknowns, and then alternatively update among these variables. In this part, we propose an iterative algorithm by alternately optimizing one variable with others being fixed and dividing the optimization problem into several subproblems. To keep a consistent progress along each of the unknowns and avoid local optimal as far as possible, only a few iterations are performed in each update.

A. Estimating sparse coefficients of latent image. Firstly, we try to obtain the sparse coefficients by solving the following problem,

\[
\min_{\mathbf{a}} \frac{1}{2} \| \mathbf{y} - \mathbf{k} \otimes \mathbf{x} \|_2^2 + \lambda \| \mathbf{a} \|_1 \quad \text{s.t.} \quad \| \mathbf{D} \mathbf{a} \|_2 \leq 1
\]

(6)

This is a typical \( \ell_0 \) sparse optimization problem, and the Batch-OMP (Orthogonal Matching Pursuit) method [25] is efficient and effective in finding an optimal solution \( \mathbf{x} \). For initialization, we set \( \mathbf{H}_k = \mathbf{H}_s \), \( \mathbf{D} \) as an overcomplete discrete cosine transform (DCT) dictionary and \( \mathbf{k} \) is chosen as the Gaussian shaped kernel.

B. Estimating blur kernel. Secondly, once having obtained the sparse coefficients \( \mathbf{x} \), we can update the blur kernel \( \mathbf{k} \) with high-frequency patches \( \mathbf{H}_k \) blurred image. The estimate of the blur kernel can be found via solving the following constrained quadratic least-square regression subproblem,

\[
\min_{\mathbf{k}} \| \mathbf{k} \|_1 + \| \mathbf{k} \otimes \mathbf{D} \mathbf{x} - \mathbf{H}_k \|_2^2 \quad \text{s.t.} \quad k > 0
\]

(7)

Consider the vector–matrix form of image blurring model \( \mathbf{b} = \mathbf{H}_k \mathbf{k} \), where \( k \) and \( \mathbf{b} \) are the vectorial forms corresponding to \( \mathbf{k} \) and \( \mathbf{H}_k \) respectively. The above optimization problem could be transferred into a compressed sensing problem. To seek the estimator of the blur kernel, we then use iterative re-weighted least squares (IRLS) [11] algorithm to solve the subProblem 7. We pose the optimization as a sequence of least squares problems while the weight of each derivative is updated based on the previous iteration solution. Different from the unconstrained problem referred in [11], we need to further project the result \( k \) onto the constraints, such as setting negative elements to 0 and renormalizing. To attenuate the noise and make the algorithm robust, we threshold small elements of the kernel to zero.

C. Estimating the sharp dictionary.

By fixing the kernel, we can update the sharp dictionary \( \mathbf{D} \). However, the information extracted from the blurred part cannot be directly used for estimating the sharp dictionary. Here, for simplicity, we apply Richardson-Lucy (RL) deconvolution method to roughly recover \( \mathbf{H}_k \) from \( \mathbf{H}_s \) jointly with the kernel obtained in the above step. Then, the vectors \{ \( \mathbf{H}_k \) \} extracted from the recovered \( \mathbf{H}_k \), which shares the same dimension as the dictionary basis, is exploited to update the sharp dictionary \( \mathbf{D} \) of latent image. The subproblem of estimating a sharp dictionary is given by,

\[
\min_{\mathbf{H}_k} \sum |\mathbf{H}_k|_0 + \| \mathbf{D} \mathbf{x} - \mathbf{H}_k \|_2^2 \quad \text{s.t.} \quad \| \mathbf{d}_j \|_2 = 1
\]

(8)

Then, K-SVD algorithm [1] is applied to efficiently optimize the above problem. To warm up the K-SVD algorithm, we initialize the dictionary with an over-complete DCT basis. The detailed setting of parameters, such as the size of dictionary, is discussed in Section 4.

In the main loop of iterative processes A to C, we monitor the difference between the two successive latent image parts to decide whether the entire iteration still needs to continue. If the difference is small enough, the iterative processing will be terminated so that the estimated blur kernel can be determined from the algorithm. Once it is finished, we start a non-blind deconvolution procedure as described below to recover the latent image.

3.4. Recover the latent image

Once the kernel has been achieved through Section 3.3, we can recover the latent image via a variety of non-blind deconvolution methods. Instead of simply using Richardson-Lucy deconvolution algorithm, we solve the following optimization problem to get the sharp image. In the formulation, \( \beta > 0 \) is a scale parameter which balances the trade-off between the regularity and the fidelity of the blurred image \( \mathbf{y} \).

\[
\min_{\mathbf{x}} \| \mathbf{y} - \mathbf{k} \otimes \mathbf{x} \|_2^2 + \| \nabla_1 \mathbf{x} \|_1 + \| \nabla_2 \mathbf{x} \|_1
\]

(9)

We regularize the cost function using two penalty functions \( \| \cdot \|_1 \) that act on the output of a set of first-order filters applied to \( \mathbf{x} \), like those used in Section 3.3. By using this derivative filter, the recovery of
latent image will be more robust to possible wrong kernel estimate, which gives rise to ringing artifacts in $x$. This optimization problem can be efficiently solved by the split Bregman algorithm in $[31]$. 

3.5. Complexity analysis

Here we analyze the computational cost of the proposed learning-based algorithm for blind image deblurring. The cost constitutes the estimation of the sparse coefficients, blur kernel and sharp dictionary, and the recovery of the latent image using the trained blur kernel. The estimation by using Batch-OMP is conducted iteratively till a stop condition is satisfied. In general, each Batch-OMP operation needs $O(S^3 + S^2L + 4NL)$ time complexity, where $S$ is the target sparsity, $N$ is the dimension of the data to be sparsely coded and $L$ is the size of the dictionary. The K-SVD operation costs $O(n \cdot (S^2L + 2NL))$, where $n$ is the number of training signals. As for IRLS in recovery, it has time complexity $O(M^3)$, where $M$ is the size of blur kernel. When applying $t$ iterations to estimate the blur kernel, the algorithm has time complexity $O(t \cdot (S^3 + S^2L + 4NL + nS^2L + 2nNL + M^3))$.

4. Experimental results

In this section, we compare our proposed method to other blind deblurring algorithms for which there is public code available, such as Fergus et al.’s method $[14]$, Shan et al.’s method $[26]$ and Hu et al.’s method $[16]$.

First, we use several synthesized blurred images to verify the efficiency and performance of the method proposed in Section 3. Then, several real blurred images without the ground truth are used to assess the proposed method. To evaluate the performance objectively, we adopt the blurred signal-to-noise ratio (BSNR), the improvement in signal-to-noise ratio (ISNR), and the peak signal to noise ratio (PSNR) for assessing image restoration quality. BSNR represents the signal-to-noise ratio of the blurred and noisy input image. ISNR specifies the improvement in signal-to-noise ratio after image deblurring. In order to better measure the visual quality of the recovered image, we also apply SSIM (Structural Similarity Index).
are small constants given by \(2\sqrt{\pi}\), respectively, as default settings. In our experiments, \(C_1\) and \(C_2\) are small constants given by 2.35\(^2\) and 7.65\(^2\), respectively, as default settings.

### Table 1

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Barbara</td>
<td>19.93</td>
<td>23.82</td>
<td>19.53</td>
<td>24.09</td>
</tr>
<tr>
<td>Lena</td>
<td>19.077</td>
<td>26.131</td>
<td>22.734</td>
<td>26.168</td>
</tr>
<tr>
<td>Hat</td>
<td>19.11</td>
<td>26.765</td>
<td>26.931</td>
<td>26.87</td>
</tr>
<tr>
<td>Peppers</td>
<td>15.64</td>
<td>32.371</td>
<td>29.209</td>
<td>32.52</td>
</tr>
<tr>
<td>Straw</td>
<td>16.156</td>
<td>27.869</td>
<td>27.96</td>
<td>27.94</td>
</tr>
<tr>
<td>Boats</td>
<td>16.592</td>
<td>31.996</td>
<td>29.22</td>
<td>32.04</td>
</tr>
<tr>
<td>Baboon</td>
<td>17.564</td>
<td>29.364</td>
<td>29.476</td>
<td>29.33</td>
</tr>
</tbody>
</table>

**Fig. 3.** Deblurred image *Barbara* using different methods: (a) zoomed-in part and kernel by Fergus et al.’s method [14]; (b) zoomed-in part and kernel by Shan et al.’s method [26]; (c) zoomed-in part and kernel by Hu et al.’s method [16]; (d) zoomed-in part and kernel by the proposed method.

**Fig. 4.** Deblurred image *Lena* using different methods: (a) zoomed-in part and kernel by Fergus et al.’s method [14]; (b) zoomed-in part and kernel by Shan et al.’s method [26]; (c) zoomed-in part and kernel by Hu et al.’s method [16]; (d) zoomed-in part and kernel by the proposed method.

<table>
<thead>
<tr>
<th>PSNR comparison, BSNR = 40 dB.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Images</strong></td>
</tr>
<tr>
<td>Barboum</td>
</tr>
<tr>
<td>Lena</td>
</tr>
<tr>
<td>Hat</td>
</tr>
<tr>
<td>Peppers</td>
</tr>
<tr>
<td>Straw</td>
</tr>
<tr>
<td>Boats</td>
</tr>
<tr>
<td>Baboon</td>
</tr>
</tbody>
</table>

### 4.1. Determination of patch size and dictionary size

For effectively representing blurred image and achieving improved deblurring performance, both appropriate dictionary size and patch size are very important. Moreover, the size of each patch should also be chosen appropriately. A small patch will result in high computational expense to process the entire image, whereas larger patches will lack enough redundancy for good sparse representation of image patches. Here, we empirically set the best patch size to be 12 \(\times\) 12. In order to choose a suitable dictionary size, we carry out the experiments on the test image *Barbara* (Fig. 1(a)). The blurred image is synthesized by convolving with a \(9 \times 9\) Gaussian kernel of \(\sigma = 3\) and adding with additive noise whose standard deviation is 0.45. Then the test blurred image has BSNR = 40 dB. Fig. 1 shows the PSNR values and running time of the deblurring results using dictionaries with different sizes. We can observe that the larger dictionary size 2048 does not encourage a better (higher) PSNR value. Actually, PSNR achieves its highest value when the dictionary size is 1600. Another reason why we may favour a medium dictionary size is the running time consumed for deblurring. Fig. 2(a–g) show the deblurred images corresponding to different dictionary size \(d = 128, 256, 512, 1024, 1440, 1600, 2048\). Visually we cannot distinguish quality among those deblurred images with dictionary size \(d = 1440, 1600\) and 2048. According to the experimental results in Fig. 1 and Fig. 2, it is recommended that
a reasonable dictionary size would be around 1600 to keep a good balance between performance and time consumed. Therefore, we choose a dictionary size of 1600 for our experiments.

4.2. Comparison with other blind deconvolution methods

Due to the consideration of saving computational cost, we only empirically chose a proper size of $12 \times 12$ for patches extracted from the blurred image during estimation of the blur kernel. The following criteria,

$$\left\| H_k^o - H_k^{-1} \right\|_2^2 / \left\| H_k^{-1} \right\|_2^2 \leq 10^{-4}$$

was used to terminate the estimating iteration process, in which $H_k^o$ denotes the observed blurred image.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Barbara</td>
<td>0.474</td>
<td>0.676</td>
<td>0.578</td>
<td>0.684</td>
</tr>
<tr>
<td>Lena</td>
<td>0.536</td>
<td>0.817</td>
<td>0.811</td>
<td>0.7588</td>
</tr>
<tr>
<td>Hat</td>
<td>0.57</td>
<td>0.773</td>
<td>0.284</td>
<td>0.782</td>
</tr>
<tr>
<td>Peppers</td>
<td>0.561</td>
<td>0.763</td>
<td>0.333</td>
<td>0.771</td>
</tr>
<tr>
<td>Straw</td>
<td>0.375</td>
<td>0.776</td>
<td>0.364</td>
<td>0.774</td>
</tr>
<tr>
<td>Boats</td>
<td>0.0047</td>
<td>0.2498</td>
<td>0.2492</td>
<td>0.253</td>
</tr>
<tr>
<td>Baboon</td>
<td>0.272</td>
<td>0.771</td>
<td>0.353</td>
<td>0.778</td>
</tr>
</tbody>
</table>

Table 2
SSIM comparison, BSNR = 40 dB.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Barbara</td>
<td>-0.6684</td>
<td>0.168</td>
<td>-0.146</td>
<td>0.741</td>
</tr>
<tr>
<td>Lena</td>
<td>-1.51</td>
<td>0.7</td>
<td>0.429</td>
<td>0.3949</td>
</tr>
<tr>
<td>Hat</td>
<td>0.236</td>
<td>-0.22</td>
<td>-0.055</td>
<td>0.13</td>
</tr>
<tr>
<td>Peppers</td>
<td>0.1448</td>
<td>0.048</td>
<td>-0.4</td>
<td>0.264</td>
</tr>
<tr>
<td>Straw</td>
<td>-1.456</td>
<td>0.838</td>
<td>-2.32</td>
<td>0.88</td>
</tr>
<tr>
<td>Boats</td>
<td>-0.184</td>
<td>-0.0986</td>
<td>-0.0079</td>
<td>-0.007</td>
</tr>
<tr>
<td>Baboon</td>
<td>-1.68</td>
<td>1.051</td>
<td>-1.726</td>
<td>1.066</td>
</tr>
</tbody>
</table>

Table 3
ISNR comparison, BSNR = 40 dB.

Fig. 5. Zoom-in deblurred images using different methods for Koala: (a) Blurred image; (b) Fergus et al.’s method [14]; (c) Shan et al.’s method [26]; (d) Hu et al.’s method [16]; (e) The proposed method.
Here, we first demonstrate the performance of the proposed method with the synthesized blurred images. For the first experimental set, the eight test images (size: 256 × 256) are blurred with a Gaussian-shaped function with variance 9, and a white Gaussian noise is added to obtain the degraded images with a BSNR of 40 dB. For the sake of comparison we list the parameter settings for the compared methods as follows. In Hu et al.'s method [16] the blur kernel was initialized to be a Gaussian kernel with \( \sigma = 1 \) and the penalty parameter was set to 1. In Fergus et al.'s method [14] 4 scales of kernel estimates, starting with 3 × 3, were conducted. They both used the Richardson-Lucy deconvolution algorithm to finally recover the latent image. In Shan et al.'s method [26] a 9 × 9 kernel size was adopted and other default settings were used as the authors'.

Due to the page limitation, Figs. 3 and 4 only show the zoomed-in part of the reconstructed images of Barbara and Lena, respectively. To further demonstrate a quantitative comparison, we provide the evaluative values for all the recovered images given by different algorithms, illustrated in Tables 1–3. Besides PSNR and ISNR, we also report SSIM results to compare the visual quality of the recovered images. According to the results in the tables and figures, both visual and quantitative comparisons show that our proposed method is able to recover sharp images effectively in smooth and textured areas without many ringing artifacts for the synthesized blurred images.

In the second set of experiments, we evaluate the performance of the proposed method on real blurred images without the ground truth version, Koala and Castle. Figs. 5 and 6 provide a visual quality comparison of deblurred images using different methods. Fig. 5 is the zoomed-in part of Koala, which shows more details and better contrast. In terms of these results, we can conclude that the proposed method achieves a comparable performance against the benchmark methods.

In summary, the proposed method is competitive with the state-of-the-art methods and shows the validity of using the coupled sparse priors on natural images.

5. Conclusion

In this paper, a novel deblurring method based on coupled sparse representation is proposed for blind image deblurring. Except for the use of sparse representation of image patches, we also ensure that the sparse coefficients comply with the strict compatibility between the latent image and blurred one. After obtaining the estimated blur kernel, we further recover the sharp image by imposing a sparsity inducing constraint on images so that the recovered sharp image is more robust to ringing artifacts. Moreover, to facilitate a better kernel estimate, we apply non-negative and normalization constraints on blur kernels so that more accurate blur kernels can be obtained for reconstructing the latent image. Experimental results show that the proposed method achieves comparable performance against the state-of-the-art algorithms and even outperforms them in both synthetic and real image experiments.

Acknowledgments

This work is supported under the Grant DP130100364 from the Australian Research Council (ARC). The work of the first and fourth authors is supported by NSF China under Grants No. 61201392.

References


