J Math Imaging Vis (2011) 39: 1–12
DOI 10.1007/s10851-010-0220-8

Direct Sparse Deblurring

Yifei Lou · Andrea L. Bertozzi · Stefano Soatto

Published online: 14 August 2010
© The Author(s) 2010. This article is published with open access at Springerlink.com

Abstract We propose a deblurring algorithm that explicitly takes into account the sparse characteristics of natural images and does not entail solving a numerically ill-conditioned backward-diffusion. The key observation is that the sparse coefficients that encode a given image with respect to an over-complete basis are the same that encode a blurred version of the image with respect to a modified basis. Following an "analysis-by-synthesis" approach, an explicit generative model is used to compute a sparse representation of the blurred image, and its coefficients are used to combine elements of the original basis to yield a restored image.

Keywords Deblurring · Sparse coding · Over-complete dictionary

1 Introduction

Deblurring refers to the task of "undoing" the effects of convolving[1] the data with a known kernel. A common instance occurs when an image is taken with a finite-aperture system that is not well focused, so the measured image is a blurred version of the "ideal image," convolved with the point-spread function of the lens. Ideally, one would like to recover, or "restore," the image as would be captured by a well-focused lens. Unfortunately, deblurring is well-known to be an ill-posed inverse problem, so small perturbations in the data (for instance noise or quantization errors in the measured "blurred" image) lead to large errors in the reconstruction. These artifacts are usually kept at bay by means of regularization, following the classical work of Tikhonov [30]. Several choices of such *generic regularizers* have been proposed, mostly made for mathematical convenience, some based on empirical observations on the quality of the reconstruction. Recently, there has been some convergence towards regularizers that enforce the statistics of natural images, which are well-known to possess highly kurtotic behavior [18] due to the presence of large homogeneous regions bounded by sharp discontinuities at visibility boundaries such as occlusions and cast shadows. Some classic regularizers, such as the Total Variation, implicitly favor these kind of solutions, and remain among the most competitive deblurring algorithms to this day. Nevertheless, deblurring in this context involves solving an inverse diffusion partial differential equation (PDE). In [11], Favaro et al. have approached the problem of deblurring using a "direct" method: Rather than deconvolving the measured image and the noise that goes with it, they convolve the "model image," which is noiseless by definition, with the known kernel. This yields a simple diffusion PDE whose (space-varying) stopping time encodes the value of the kernel. They do not, however, exploit the statistics of natural images in their solution.

A related literature stream encodes the image as a discrete array of positive numbers, approximated by linear combinations of local overcomplete bases, where the natural statistics are captured by the fact that the vector of coefficients is *sparse*, so at any location only few bases contribute to the approximation [19]. In this case, there is no

[1] Although usually deblurring refers to non-shift-invariant kernels, we use deblurring and deconvolution interchangeably in this manuscript.

Y. Lou (✉) · A.L. Bertozzi
UCLA, Department of Mathematics, Los Angeles, USA
e-mail: yflou@math.ucla.edu

A.L. Bertozzi
e-mail: bertozzi@math.ucla.edu

S. Soatto
UCLA, Computer Science Department, Los Angeles, USA
e-mail: soatto@cs.ucla.edu

need for an explicit regularizer, due to the finite dimensionality of the representation, but there is still a trade-off between fidelity of the approximation and complexity of the model. While the measured image is undoubtedly a discrete object, with quantization of both the domain and the range, the object of inference, or the "ideal image," is best represented in the continuum, with the final discretization left only for the numerical implementation of the optimization scheme. We therefore take the continuum approach, and explicitly write cost functionals that have the ideal image as an infinite-dimensional unknown.

In this manuscript, we devise a deblurring algorithm that (a) explicitly takes into account the "sparse" natural statistics of the image as a regularizer, and (b) does not suffer from the numerical conditioning issues associated with solving an inverse diffusion PDE. We also extend the method to blind deconvolution by simply augmenting our dictionary to include several different blurring kernels. We discuss some caveats in Sect. 4.1.

1.1 The Basic Idea

The idea of direct sparse deblurring is simple and can be illustrated in three steps.

First, we assume that the image is square-integrable and sparse in some basis defined on the entire real plane. This is a common assumption underlying most image compression algorithms, in particular those based on over-complete bases, or "dictionaries" $\{d_k\} \in \mathcal{L}^2(\mathbb{R}^2 \to \mathbb{R}), k = 1, \dots, K$ where K is the number of atoms in the dictionary. A dictionary can be used to approximate the original image u to an arbitrary degree by a sparse linear combination $u \doteq \sum_{k=1}^K d_k \alpha_k + n \doteq \mathrm{D}\alpha + n$ where $\|\alpha\|_0$ is small[2] and so is $\|n\|_2$ [10]. For the sake of illustration, let us pretend that this representation is *exact*, that is $\|n\| = 0$, and $u = \mathrm{D}\alpha$ with $\|\alpha\|_0 \leq L$, where L is the bound of $\mathcal{L}^0$ norm. We will discuss the role of n later.

Now, convolving an image with a shift-invariant kernel h yields a *blurred image* $f = h * u = h * \mathrm{D}\alpha$; this shows that *the coefficients* α *that represent the* sharp image u *relative to the basis* $\{d_k\}$ *are the same that represent the blurred image* f *relative to the* blurred basis $\{b_k\} \doteq \{h * d_k\}$.

But while we do not have access to the sharp image u, we *do* have access to the original basis elements $\{d_k\}$. Therefore, all we need to recover the encoding of the sharp image are the coefficients of the encoding of the blurred image relative to the blurred basis. The ensuing algorithm is as follows:

1. Take a dictionary $\{d_k\}$, either from a generic over-complete basis or learned form the image using any of a variety of sparse coding algorithms, for instance [1]. Convolve the basis with the kernel h to obtain a "blurred basis" $\{b_k\} \doteq \{h * d_k\}$.
2. Perform sparse coding of the blurred image f relative to the blurred basis $\{b_k\}$ to obtain $\hat{\alpha}$ with $\|\hat{\alpha}\|_0$ small such that $\|f - \sum_{k=1}^K b_k \hat{\alpha}_k\|$ is also small.
3. Reconstruct the original (deblurred) image *directly* via $\hat{u} = \sum_{k=1}^K d_k \hat{\alpha}_k$.

Note that this algorithm performs deblurring without solving a backward diffusion or other numerically ill-conditioned procedure. Instead, it solves the inverse problem by "direct methods", an approach sometimes referred to as "analysis by synthesis" [16] whereby an explicit generative model is used to match the statistics of the measured data, and the model itself provides both the necessary regularization and the solution to the desired inverse problem.

Now a few caveats. In practice, no image fits the model $u = \mathrm{D}\alpha$ exactly, sparsely or otherwise. Therefore, one typically looks for the optimal representation $\hat{\alpha}$, defined as the solution of the following optimization problem

$$\hat{\alpha} = \arg\min_{\alpha} \left\{ \int \|n\|_2^2 \mathrm{d}x \mid u = \mathrm{D}\alpha + n,\ \|\alpha\|_0 \leq L \right\}, \tag{1}$$

where the integral is on all of $\mathbb{R}^2$. Following the prescription outlined above to extend the algorithm to blurred images, one would get $f = h * u + h * n = h * \mathrm{D}\alpha + h * n$. Therefore, the algorithm we have suggested does *not* solve the problem

$$\hat{\alpha} = \arg\min_{\alpha} \left\{ \int \|n\|_2^2 \mathrm{d}x \mid f = h * \mathrm{D}\alpha + n,\ \|\alpha\|_0 \leq L \right\}, \tag{2}$$

where the constraint $f = h * u + n$ describes the image formation model. Instead, we solve the modified problem

$$\tilde{\alpha} = \arg\min_{\alpha} \left\{ \int \|h * n\|_2^2 \mathrm{d}x \mid f = h * \mathrm{D}\alpha + h * n,\ \|\alpha\|_0 \leq L \right\}. \tag{3}$$

Note that, in principle, $\tilde{\alpha} \neq \hat{\alpha}$. However, the blurring kernel h is zero-mean (lest images would get brighter and dimmer as the focus changes), and therefore

$$\int \|n\|_2^2 \mathrm{d}x \geq \int \|h * n\|_2^2 \mathrm{d}x. \tag{4}$$

So, at least to first approximation, we indeed have that $\tilde{\alpha} = \hat{\alpha}$. The benefit of this approach is a considerably simpler "direct" algorithm, and the cost is having changed the terms of the problem, or equivalently the model and the underlying assumption, from minimizing the residua error n, to minimizing a blurred version of it.

Of course, the devil is in the details, as real images and dictionaries are not defined on the entire real plane, and if we wish to keep the complexity of the coding step manageable

[2] $\forall \epsilon \exists \bar{K} = K(\epsilon)$ such that if $M > \bar{K}$, then $\|f - \sum_{k=1}^M d_k \alpha_k\| < \epsilon$.

we will have to break down the image into patches, which raises the issue of boundary effects and scale, which causes $\tilde{\alpha} \neq \hat{\alpha}$. However, there is no reason why $\hat{\alpha}$ should be "better" than $\tilde{\alpha}$; they just represent different modeling assumptions. In the words of Box, "all models are wrong, some are useful." Ours is useful in the sense of yielding a particularly simple, direct algorithm, which we now derive for a partition of the image, explicitly taking the issues of boundaries and scale into account.

1.2 State of the Art

The concept of sparsity has recently re-gained popularity in image processing and computer vision; similarly, there is a sizable literature on the problems of deblurring, deconvolution and shape from defocus or motion blur.

The literature on "exemplars" in computer vision, although not explicitly about sparsity, goes in this direction, including the early work of non-local super-resolution [9, 15, 26]. The idea of sparse representation of local patches is widely used in the many applications of image processing, such as denoising [10], color denoising and inpainting [21] and super-resolution [6, 32]. Deconvolution, on the other hand, is mainly solved by regularization. For example, the Wiener filter [2] uses the H^1 semi-norm of the solution, which favors smooth reconstructions. Total Variation (TV) [27], as already mentioned, favors piece-wise constant solutions, whereas some wavelet-based deconvolution does not explicitly enforce a regularizer, except for [8], but exercises regularization through complexity bounds [7, 12–14, 24]. Segmentation-based regularization is discussed in [23]. Deconvolution is also directly extended from denoising algorithms, such as BLS-GSM [17, 25], kernel regression [28, 29] and BM3D [4, 5].

There are relatively fewer works addressing the use of sparse priors for deblurring, which is the goal in this paper. Although super-resolution is a close relative to deblurring (the point-spread function corresponds to block-averaging of neighboring pixels), the latter has not been addressed directly in a sparse setting. There are a few papers on image deblurring using global sparse transforms [3, 20], while we focus on local sparsity.

Since we use the K-SVD algorithm [10] as a building block, we will briefly review it here to make the manuscript self-contained. We will then extend it to the continuum and apply it to deblurring in Sect. 2. More details on the implementation are presented in Sect. 2.1, and Sect. 3 contains numerical experiments.

1.3 Sparse Representations for Denoising

If we consider discrete image patches, i.e. positive-valued matrices of size $\sqrt{n} \times \sqrt{n}$ pixels, ordered lexicographically as column vectors $x \in \mathbb{R}^n$, then the sparsity assumption corresponds to assuming the existence of a matrix $\mathrm{D} \in \mathbb{R}^{n\times K}$, the "dictionary," such that every image patch x can be represented as a linear combination of its columns with a vector of coefficients with small $\mathcal{L}^0$ norm. If we measure y, a version of x corrupted by additive Gaussian noise that is spatially white (independent and identically distributed) with standard deviation σ, then the maximum a-posteriori (MAP) estimator of the "denoised" patch x is given by $\mathrm{D}\hat{\alpha}$, where

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_0 \quad \text{s.t.} \quad \|\mathrm{D}\alpha - y\|_2^2 \leqslant T, \tag{5}$$

where T is dictated by σ. If one wishes to encode a larger image X of size $\sqrt{N} \times \sqrt{N}$ $(N \gg n)$, with a combination of columns of the low-dimensional dictionary D, a natural approach is to use a block-coordinate relaxation.

$$\hat{\mathrm{X}} = \arg\min_{\mathrm{X},\alpha_{ij},\mathrm{D}} \|\mathrm{X}-\mathrm{Y}\|_2^2 + \lambda \sum_{i,j} \|\alpha_{ij}\|_0 + \mu \sum_{i,j} \|\mathrm{D}\alpha_{ij} - R_{ij}\mathrm{X}\|_2^2 . \tag{6}$$

The first term measures the fidelity between the measured image Y and its denoised (and unknown) version X. The second term enforces sparsity of each patch; the $n \times N$ matrix R_{ij} extracts the (i, j)th block from the image. A simple denoising algorithm [10] based on sparse coding goes as follows,

1. Initialization: Set $\mathrm{X} = \mathrm{Y}$, $\mathrm{D} =$ an overcomplete discrete cosine transform (DCT) dictionary.
2. Repeat until it converges:
   - *Sparse Coding:* fix X and D, compute the representation vectors α_{ij} for each patch $R_{ij}\mathrm{X}$

     $$\hat{\alpha}_{ij} = \arg\min_{\alpha} \|\alpha\|_0 \quad \text{s.t.} \quad \|\mathrm{D}\alpha - R_{ij}\mathrm{X}\|_2^2 \leqslant T \, . \tag{7}$$

   - *Dictionary Update:* fix X and $\{\alpha_{ij}\}$, compute D via K-SVD [1] one column at a time.
3. Set:

   $$\mathrm{X} = \frac{\mathrm{Y} + \mu \sum_{ij} R_{ij}^T \mathrm{D}\alpha_{ij}}{Id + \mu \sum_{ij} R_{ij}^T R_{ij}} , \tag{8}$$

   which is a simple averaging of shifted patches.

One could also fix the dictionary and only perform sparse coding in the iteration. Alternatively, the dictionary can be learned from a large number of patches in natural images via K-SVD [1] so that it is tailored to the data.

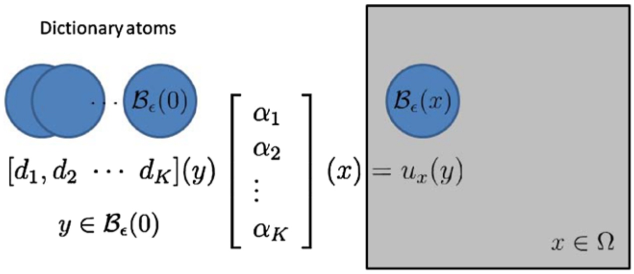

Dictionary atoms
$\mathcal{B}_\epsilon(0)$
$[d_1, d_2 \cdots d_K](y)$
$y \in \mathcal{B}_\epsilon(0)$
$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_K \end{bmatrix}(x) = u_x(y)$
$\mathcal{B}_\epsilon(x)$
$x \in \Omega$

Fig. 1 Diagram of the continuum formulation

2 Direct Sparse Deblurring

In this section we formalize the problem of direct sparse deblurring. We find the formalization to be clearer when written in the continuum, so one knows on what domain each function is calculated. The previous claim, that the blurred image can be sparsely represented in the blurred basis by the same coefficients that the "ideal" image would have on the original basis, will become clear.

Let $u : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^+; x \mapsto u(x)$ be the "ideal image", corresponding to X in the discrete model. The procedure of extracting a small patch from an epsilon ball centered at x can be represented by

$$u_x(y) = \{u(y) : y \in \mathcal{B}_\epsilon(x)\} = \{u(x+y) : y \in \mathcal{B}_\epsilon(0)\}. \tag{9}$$

The function $u_x(\cdot)$ describes a mapping from an epsilon ball to a patch centered at x, which can be expressed by an indicator function $\chi_\epsilon(x-y)$ acting on the image $u(x)$.

Let $d_k : \mathcal{B}_\epsilon(0) \subset \mathbb{R}^2 \to \mathbb{R},\ k = 1, \ldots, K$ be a given overcomplete basis of $\mathcal{L}_{loc}(\mathcal{B}_\epsilon(0) \to \mathbb{R})$, and $\alpha_k(x)$ be the kth sparse coefficient of the patch centered at x. Then the sparse representation of one image patch is given by

$$\begin{aligned} u_x(y) &= \chi_\epsilon(y)u(x+y) \\ &= \sum_{k=1}^{K} d_k(y)\alpha_k(x) \doteq \boldsymbol{d}(y)\boldsymbol{\alpha}(x)\ , \end{aligned} \tag{10}$$

where $\boldsymbol{d}(y) = [d_1, d_2, \ldots, d_K](y),\ y \in \mathcal{B}_\epsilon(0)$ and $\boldsymbol{\alpha}(x) = [\alpha_1, \alpha_2, \ldots, \alpha_K](x),\ x \in \Omega$. Figure 1 illustrates the dictionary elements and how they match to local patches.

We want to use this local sparsity to enforce a global reconstruction prior in the sense that $u(x)$ is the minimizer of the sparse representation error for all the local patches.

$$\hat{u}(x) = \arg\min J(u), \tag{11}$$

where $J(u)$ is defined to be

$$\begin{aligned} &\int_{x\in\Omega}\int_{y\in\mathcal{B}_\epsilon(0)} \|\chi_\epsilon(y)u(x+y) - \boldsymbol{d}(y)\boldsymbol{\alpha}(x)\|^2 \mathrm{d}y\mathrm{d}x \\ &\quad = \iint_{\Omega\times\Omega} \chi_\epsilon(y)\|u(x+y) - \boldsymbol{d}(y)\boldsymbol{\alpha}(x)\|^2 \mathrm{d}y\mathrm{d}x \\ &\quad = \iint_{\Omega\times\bar{\Omega}} \chi_\epsilon(z-x)\|u(z) - \boldsymbol{d}(z-x)\boldsymbol{\alpha}(x)\|^2 \mathrm{d}z\mathrm{d}x, \end{aligned} \tag{12}$$

where $z = x + y$ and $\bar{\Omega} = \Omega + \mathcal{B}_\epsilon(0)$.

To solve for $u(x)$, we compute its Euler-Lagrange equation

$$\begin{aligned} &\partial_u J(u)(z) \\ &\quad = \int \chi_\epsilon(z-x)(u(z) - \boldsymbol{d}(z-x)\boldsymbol{\alpha}(x))\mathrm{d}x \\ &\quad = \left[\int \chi_\epsilon(z-x)\mathrm{d}x\right]u(z) - \int \chi_\epsilon(z-x)\boldsymbol{d}(z-x)\boldsymbol{\alpha}(x)\mathrm{d}x. \end{aligned} \tag{13}$$

There is a closed-form solution for $u(x)$ w.r.t $\boldsymbol{\alpha}(x)$ that minimizes the objective function $J(u)$. It is obtained by setting the Euler-Lagrange equation to zero:

$$\hat{u}(x) = \frac{1}{\omega}\int \chi_\epsilon(x-y)\boldsymbol{d}(x-y)\boldsymbol{\alpha}(y)\mathrm{d}y, \tag{14}$$

where $\omega = \int \chi_\epsilon(x-y)\mathrm{d}y$ is the area of the ϵ-ball. Now, the measured image is, by assumption

$$\begin{aligned} &f(x) \\ &\quad \doteq \int h(x-\bar{x})u(\bar{x})\mathrm{d}\bar{x} + n(x) \\ &\quad = \frac{1}{\omega}\int h(x-\bar{x})\int \chi_\epsilon(\bar{x}-y)\boldsymbol{d}(\bar{x}-y)\boldsymbol{\alpha}(y)\mathrm{d}\bar{x}\mathrm{d}y + n(x) \\ &\quad = \frac{1}{\omega}\iint [h(x-\bar{x})\chi_\epsilon(\bar{x}-y)\boldsymbol{d}(\bar{x}-y)\mathrm{d}\bar{x}]\boldsymbol{\alpha}(y)\mathrm{d}y + n(x), \end{aligned} \tag{15}$$

where h is a space-invariant blurring kernel and $n(x)$ is the additive noise, whose variance is σ^2. The blurred basis is easily defined as

$$b_k(z) \doteq \int h(z-\bar{x})\chi_\epsilon(\bar{x})d_k(\bar{x})\mathrm{d}\bar{x}. \tag{16}$$

The characteristic function χ_ϵ implies that the boundary condition for the convolution is zero-padding. Denote with r the support of the blurred basis, in particular $r = \epsilon + \mathrm{supp}(h)$. Therefore the measured image is a sparse representation under this blurred basis:

$$f(x) = \frac{1}{\omega}\int \chi_r(x-y)\boldsymbol{b}(x-y)\boldsymbol{\alpha}(y)\mathrm{d}y + n(x). \tag{17}$$

We solve for the sparse coefficients in the following,

$$\begin{aligned} &\hat{\boldsymbol{\alpha}}(x) = \arg\min \int \|\boldsymbol{\alpha}(x)\|_0 \mathrm{d}x, \\ &\text{s.t.}\quad \int \left\| f(x) - \omega^{-1}\int \chi_r(x-y)\boldsymbol{b}(x-y)\boldsymbol{\alpha}(y)\mathrm{d}y \right\|^2 \mathrm{d}x \\ &\qquad \leqslant T. \end{aligned} \tag{18}$$

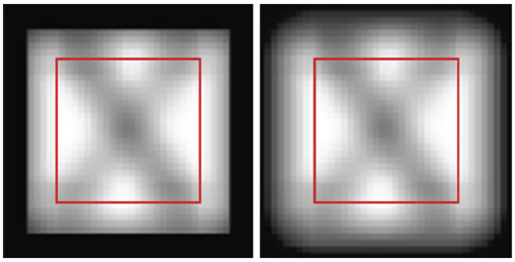


Fig. 2 *Left*: the blurry patch with zero-padding. *Right*: the blur basis. The *red square* indicates the region for our refined dictionary

This optimization problem (a) is finite-dimensional (the only unknown is α) and (b) *does not involve de-blurring*. All that is required is to find the finite-dimensional sparse set of coefficient that best approximates the given image. Note that this is accomplished by *blurring the base*. In other words, one is only required to solve a *direct* problem, rather than the inverse problem of deblurring. Once the coefficients $\hat{\alpha}$ are obtained, we can compute the “deblurred” image $\hat{u}$ via (14). Note that the deblurred image is a sparse combination of the (original, non-blurred) basis, and therefore—by construction—one should expect the reconstruction to exhibit the same spatial frequencies of the original (unblurred) data from which the overcomplete basis has been learned.

2.1 Boundary Issues

In practice, solving for $\boldsymbol{\alpha}$ from (18) is not an easy task, since $\boldsymbol{\alpha}$ at different y contribute to one value. Instead we minimize an upper bound.

$$\begin{aligned}\int \bigg\| \int \chi_{\epsilon}(x-y) f(x)\mathrm{d}y \\ - \int \chi_r(x-y)\boldsymbol{b}(x-y)\boldsymbol{\alpha}(y)\mathrm{d}y \bigg\|^2 \mathrm{d}x \\ \leqslant \iint \chi_{\epsilon}(z)\|f(z+y)-\boldsymbol{b}(z)\boldsymbol{\alpha}(y)\|^2\mathrm{d}z\mathrm{d}y. \end{aligned} \tag{19}$$

The above equation suggests coding the blurry patch centered at y in terms of the blurred basis $\{b_i\}$. Furthermore, the characteristic function indicates that the blurry patch has to be zero-padded in order to be consistent with the dimension of the blurred basis. However, these two terms cannot match due to the boundary issue of convolution, as shown in Fig. 2. Instead we match them in the region where the convolution is computed without the zero-padded edges. In particular, we refine our blurred basis to be

$$\tilde{b}_k(z) = \begin{cases} b_k(z) & |z| < \epsilon - \operatorname{supp}(h) \doteq \epsilon_0 \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

Experimentally we find that it is better to tailor the clear basis to have the same domain size as the blurred one. Therefore we have a two step algorithm for sparse deblurring

1. Solve the coefficients from the measured image

$$\begin{aligned} &\hat{\boldsymbol{\alpha}}(x) = \arg\min \|\boldsymbol{\alpha}\|_0, \\ &\text{s.t.} \quad \int \chi_{\epsilon_0}(z)\|f(z+y)-\boldsymbol{b}(z)\boldsymbol{\alpha}\|^2\mathrm{d}y \leqslant T. \end{aligned} \tag{21}$$

2. Stitch all the patches by averaging

$$\hat{u}(x) = \frac{\int \chi_{\epsilon_0}(x-y)\boldsymbol{d}(x-y)\hat{\boldsymbol{\alpha}}(x)\mathrm{d}y}{\int \chi_{\epsilon_0}(x-y)\mathrm{d}y}. \tag{22}$$

2.2 From Continuum to Discrete

In the discrete case, we assume the clear basis $\{d_k\}$ to be of size $a \times a$ and the blur kernel h be of size $c \times c$. It follows from (20) that the blurred basis $\{\tilde{b}_k\}$, which is the inner part of $\{b_k\}$, is of size $a_0 \times a_0$, where $a_0 = a - c + 1$. We crop the clear basis to be $a_0 \times a_0$ as well, denoted as $\{\tilde{d}_k\}$.

The first step (21) amounts to sparse coding for every x, or pixel (i, j) in the discrete sense. We use the Orthogonal Matching Pursuit (OMP) algorithm [31] to solve $\hat{\alpha}$ from (5) with y being the patch centered at (i, j) of size $a_0 \times a_0$ and dictionary D being comprised of $\{\tilde{b}_k\}$. The second step is to replace the blurred patch with the clear basis $\{\tilde{d}_k\}$ multiplying the sparse coefficient $\hat{\alpha}$. Finally, since each pixel is covered by different patches, the restored value is chosen to be the mean.

We use the L^0 solver OMP over a many of methods on L^1 minimization for two reasons. First, there is no additional parameter for L^0 and it has a natural stopping criterion, i.e. stops when the residual is smaller than the standard deviation of the addition noise. Second, it takes more iterations for L^1 to get a reasonable sparse coefficients.

2.3 Weighted Averaging

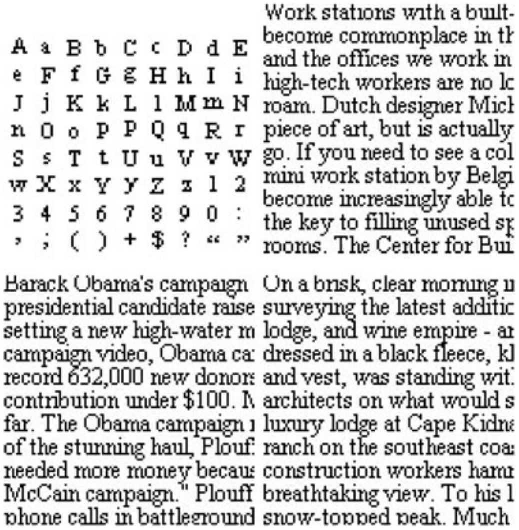
In addition to mentioned artifacts, the averaging of overlapping patches also degrades the reconstruction. As an alternative, one could use a weighted average or median to combine the results from local sparse coding. Accordingly, we perform sparse coding as described to get the coefficients $\alpha(y)$, then use weighted averaging to combine the results

$$\hat{u}(x) = \frac{\int_{y \in \mathcal{B}_\epsilon(x)} \boldsymbol{d}(x-y)\boldsymbol{\alpha}(y)w(y)\mathrm{d}y}{\int_{y \in \mathcal{B}_\epsilon(x)} w(y)\mathrm{d}y}. \tag{23}$$

When the dictionary is rich enough, each patch can be ideally represented by single atom. It is reasonable to assume that the smaller the L^0 norm of the sparse coefficients, the better representation of this patch. Therefore, the weight is chosen to penalize large $\mathcal{L}^0$ norm of the coefficients, for example,

$$w(y) = \exp\left\{\frac{-|\alpha(y)|_0}{s}\right\}, \tag{24}$$

where s is a control parameter.



A a B b C c D d E
e F f G g H h I i
J j K k L l M m N
n O o P p Q q R r
S s T t U u V v W
w X x Y y Z z 1 2
3 4 5 6 7 8 9 0 :
, ; ( ) + $? “ ”

Work stations with a built-
become commonplace in tl
and the offices we work in
high-tech workers are no lc
roam. Dutch designer Micl
piece of art, but is actually
go. If you need to see a col
mini work station by Belgi
become increasingly able tc
the key to filling unused sp
rooms. The Center for Bui

Barack Obama's campaign
presidential candidate raise
setting a new high-water m
campaign video, Obama ca
record 632,000 new donor
contribution under $100. N
far. The Obama campaign 1
of the stunning haul, Plouf
needed more money becau
McCain campaign." Plouff
phone calls in battleground

On a brisk, clear morning u
surveying the latest additic
lodge, and wine empire - ar
dressed in a black fleece, kl
and vest, was standing wit
architects on what would s
luxury lodge at Cape Kidna
ranch on the southeast coa
construction workers hamı
breathtaking view. To his l
snow-topped peak. Much

Fig. 3 The training data. The dictionary is obtained by randomly sampling raw 10 × 10 patches from the text images as well as the template shown on the *top left*. All the text images are from different categories of CNN news

3 Experiments

In this section we compare our algorithm to alternate methods such as ROF [27], the wavelet-based “ForWaRD” approach [24], regularized kernel regression-based deblurring (AKTV) [29] and BM3D-based image restoration [5]. The optimal method parameters for both ROF and ForWaRD are chosen from a series of values with wide range. Publicly available code was used for comparison with AKTV and BM3D, including the suggested parameter values. The parameters in our algorithm are determined by the data: the size of the dictionary is proportional to the width of the blurring kernel and the stopping criterion for the sparse coding stage is when the residual is below the variance of the noise.

We use root-mean-square (RMS) as a means of judging performance, $RMS(u, I) = \sqrt{\int_{x\in\Omega}(u(x) - I(x))^2 \mathrm{d}x}$, where $I(x)$ is the original image and $u(x)$ is the recovery.

3.1 Binary Text Images

We synthesize a template with all the alphanumeric characters and common punctuation as well as 5 text images from different categories of CNN news. The dictionary is comprised of 10 × 10 image patches randomly sampled from the three images and the template, all shown in Fig. 3. The template contains all individual characters, while the training images serve to represent meaningful pairs. We test the deblurring on the other two text images. The data are corrupted by convolution with a 5 × 5-pixel Gaussian kernel with $\sigma = 1$ and additive noise whose standard deviation is 5. For direct sparse deblurring, the visual quality is for the most part satisfactory except for the smoothing effects around some letters. This is mostly attributed to the limitations of the dictionary.

We also measure the effect of the number of the elements in the dictionary on the deblurring performance. Figure 5 shows the results averaged from ten different experiments of randomly sampled elements in the dictionary. In general, increasing the number of elements in the dictionary improves the results, but with a diminishing return.

3.2 Blind Deconvolution of the Text Images

For blind deconvolution, we convolve the clear basis with 3 different Gaussian kernels (same size, different σ_{dict}). Now the blurred basis has 3 times more elements than in the non-blind case. There still exists correspondence between blurred basis and clear basis. Therefore, sparse deblurring follows the same procedure as the non-blind case.

The blurry noisy data is obtained by convolving the image with 5 × 5 Gaussian kernel with $\sigma_{\text{data}} = 1$ plus noise. The clear dictionary is comprised of randomly sampling 10,000 patches from the training set. There are two cases for the blurring kernels to construct the blurred basis.

Case A $\sigma_{\text{dict}} = 0.5, 1, 1.5$: one of them happens to be the exactly same as σ_{data}.
Case B $\sigma_{\text{dict}} = 0.6, 0.9, 1.2$.

The results for both cases are presented in Fig. 6, along with the non-blind deconvolution. Case A is almost as good as non-blind with slightly worse RMS.

3.3 General Case

As in the case of super-resolution [32], the dictionary consists of random samples from the training images, which have statistics similar to the test image. Here we consider three images: “Rose,” “Koala,” and “Castle.” The training images are taken from the image datasets of flowers, animals and architecture respectively, while excluding the test ones. Figure 7 shows several examples in each training set. Flower images are from the Internet, while the other images are from the Berkeley Segmentation Dataset [22]. For each category we randomly sample 20,000 patches of size 16 × 16 to form the dictionary.

The input data are corrupted by convolving with a 9 × 9 Gaussian kernel of $\sigma = 1$ with additive noise whose standard deviation is 5. As shown in Fig. 8, ROF returns piecewise constant images, while ForWaRD produces noticeable artifacts in the reconstruction.

3.4 Trained Dictionary

We conduct a deblurring experiment of a trained dictionary. We cut a texture image into half, one as training and the

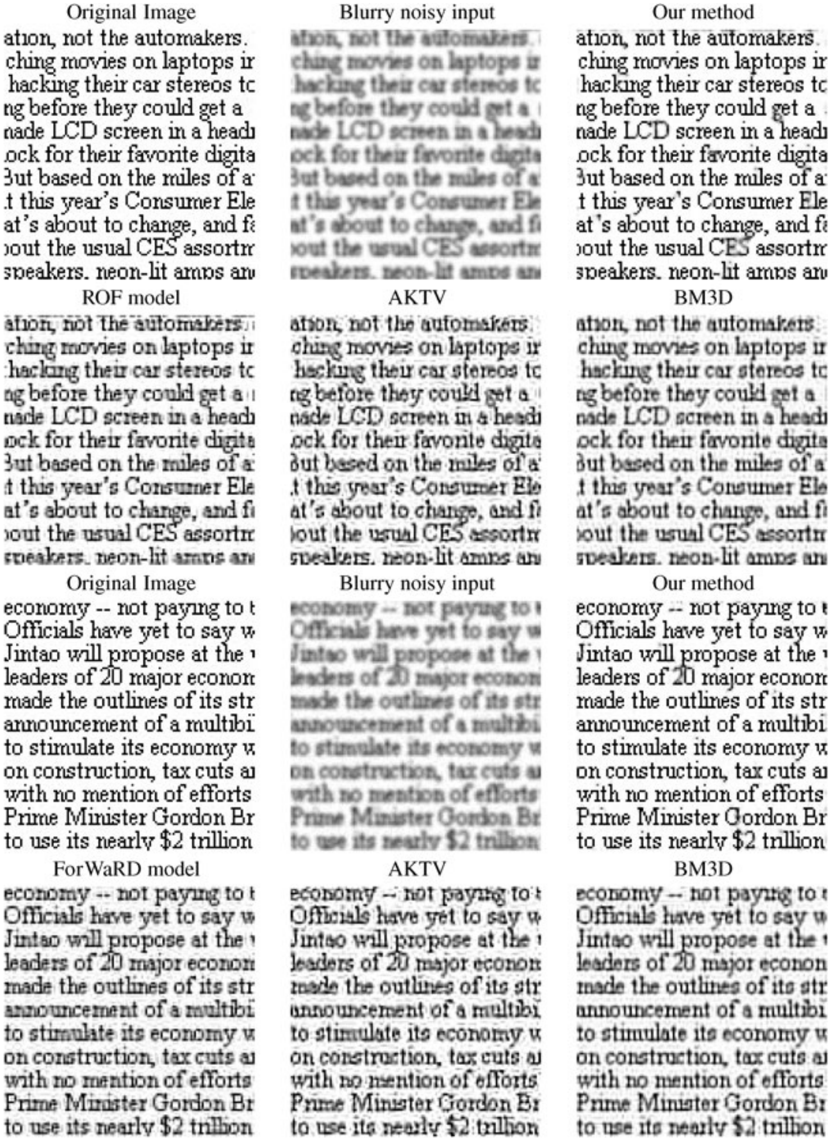


Fig. 4 Text Deblurring with 20,000 dictionary elements

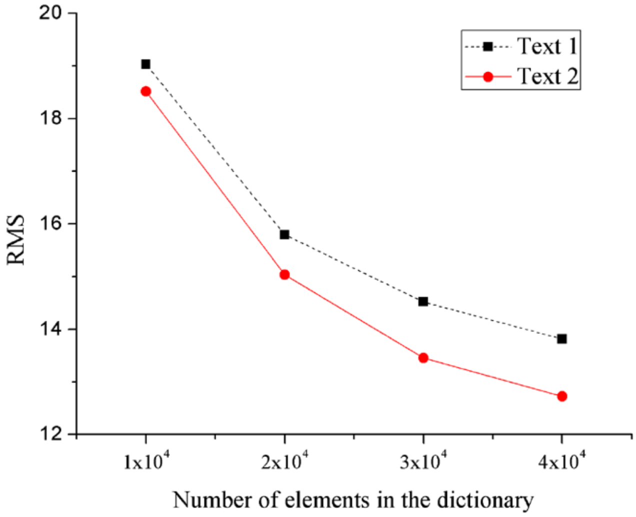


Fig. 5 Influence of the number of the elements in the dictionary on the deblurring performance. The average of 10 experiments for each column is reported

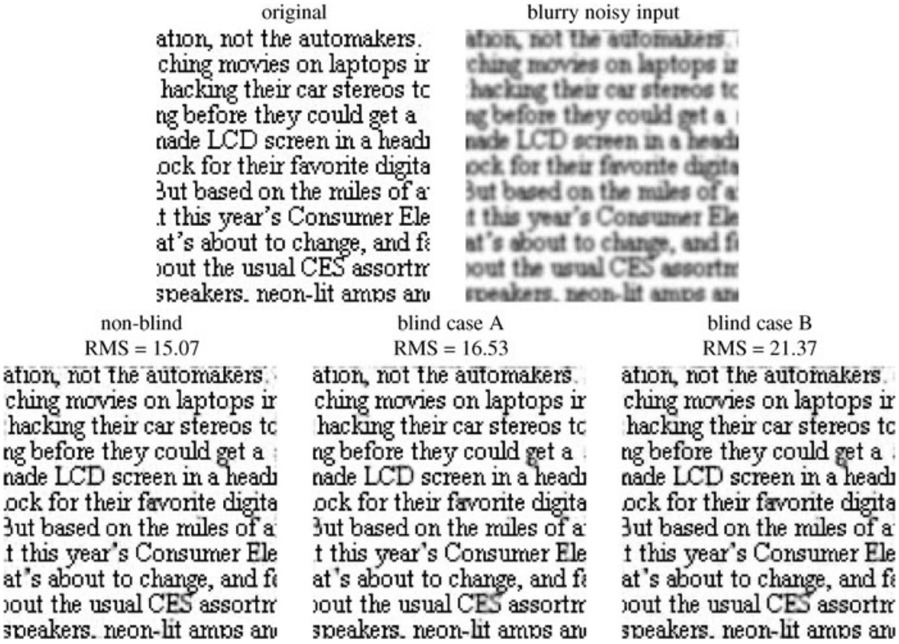


Fig. 6 Blind deconvolution with comparison to non-blind case

Fig. 7 Examples in the training images

other as testing. We take all the overlapping 16 × 16 patches in the training image to train a dictionary that has 1024 atoms. The dictionary, as shown in Fig. 9, is trained via KSVD [1]. We blur the test image with a Gaussian kernel of $\sigma = 2$ plus additive noise. As a comparison, we also construct a generic dictionary, which is comprised of 20,000 random samples from the training image. Figure 10 summarizes the results.

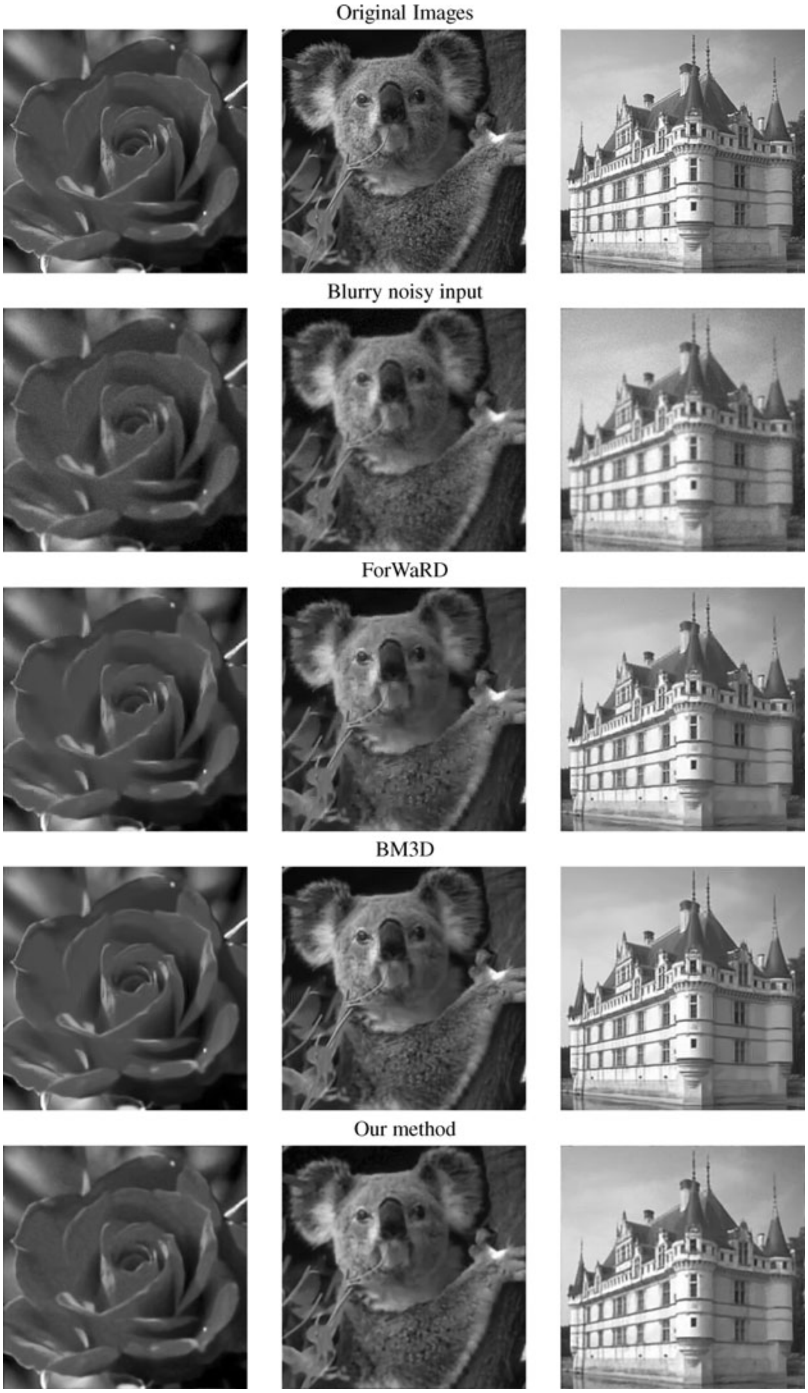
A quantitative comparison is provided in Table 1. In some case, BM3D is marginally better than our method (by about 10%), while in other cases our method fares significantly better (three times better in the Text examples and 50% better in the Texture).

4 Discussion

4.1 Coherence

The precision and stability of our "direct" approach depends on the smoothness of the blurring kernel and the geometry of the dictionary. The latter is roughly measured by the concept of *Coherence*, which is defined to be the maximum absolute inner product between two distinct vectors in the dictionary [31]. If the coherence of a dictionary is large, it is difficult for the sparse coding algorithms to choose the best atoms.

The coherence of a blurred dictionary is usually larger than the one of the clear dictionary. For example, it is very



Original Images

Blurry noisy input

ForWaRD

BM3D

Our method

Fig. 8 Grayscale image deblurring with 20,000 dictionary elements

likely that two blurred atoms b_i and b_j are similar or even identical, but their clean versions d_i and d_j are completely different. In this case, the algorithm may confuse d_i with d_j, leading to a large deblurring error. As a result, the coefficients $\hat{\alpha}$ recovered by sparse coding the blurred image f relative to the blurred basis $\{b_k\}$ could be very different from the true coefficients α of the clean image, and the deblurring estimation error can be thus inaccurate.

The amount of blur we can handle is limited by how distinctive the dictionary atoms are. For example, deblurring text images and texture yields very good results, since the dictionaries of these two cases are distinctive and the coherence of the blurred dictionary is more or less the same to the one of the clear dictionary. On the other hand, the results of "Rose," "Koala," and "Castle" imply that the associated dictionaries can only deal with smaller smoothing kernels.

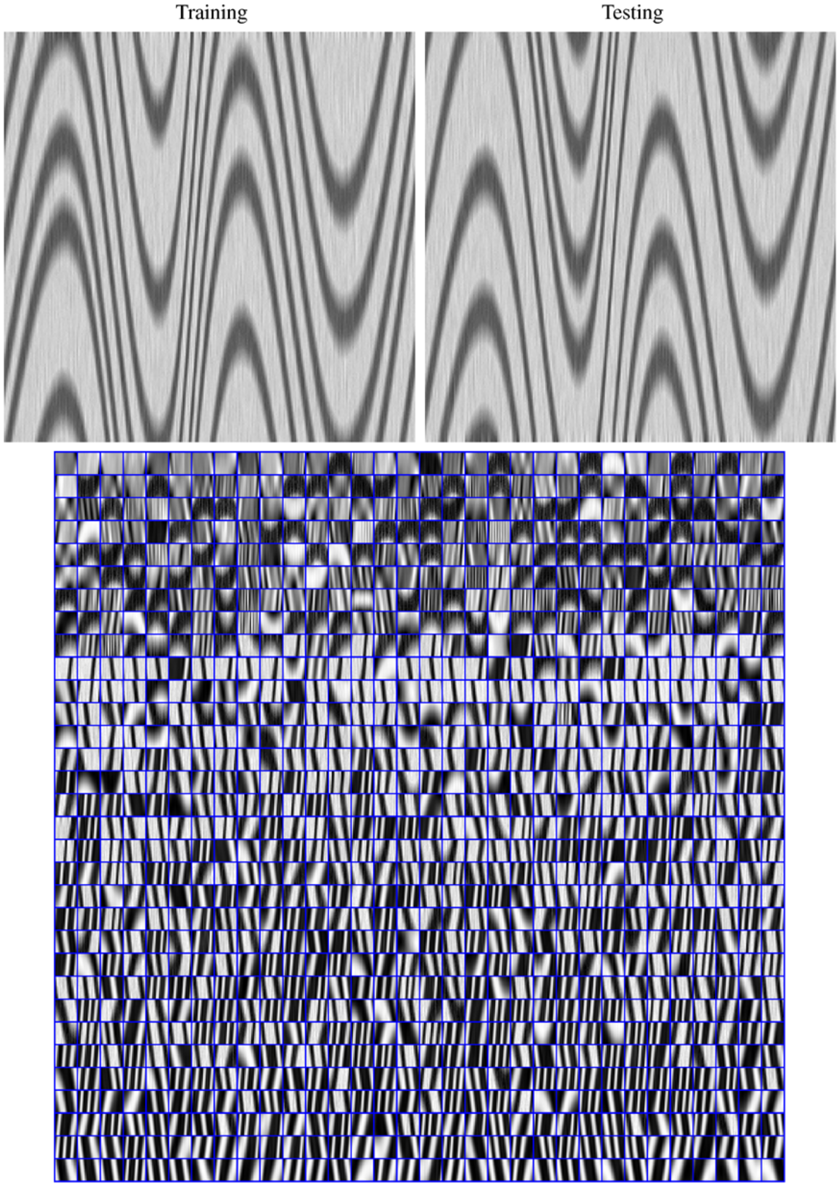


Fig. 9 The dictionary is trained using all the 16 × 16 patches in the training image (*top left*), which has a similar structure to the test one

Table 1 RMS errors for different methods. (G) and (T) indicate what kind of dictionary to use for our method with (G) for generic and (T) for trained. In some case, BM3D is marginally better than our method (by about 10%), while in other cases our method fares significantly better (three times better in the Text examples and 50% better in the Texture)

Image	ROF	ForWaRD	AKTV	BM3D	Our method
Text 1	65.43	62.75	60.68	69.87	**14.60**
Text 2	63.59	61.78	57.93	70.33	**13.81**
Rose	6.68	5.75	4.90	**4.83**	5.62
Koala	9.80	8.98	8.45	**7.97**	8.64
Castle	14.23	12.82	12.51	**11.77**	13.62
Texture	14.46	14.54	12.91	12.13	**8.72** (G) 10.37 (T)

4.2 Domain Overlapping

We want to point out the problem in minimizing the upper bound to the original formulation. Ideally, pixels that overlap with many blobs should be jointly coded, but it is computationally expensive. Instead, we adapt the procedure in [1, 21] to code the pixels multiple times and average each encoding. However, it is the averaging that in turn degrades

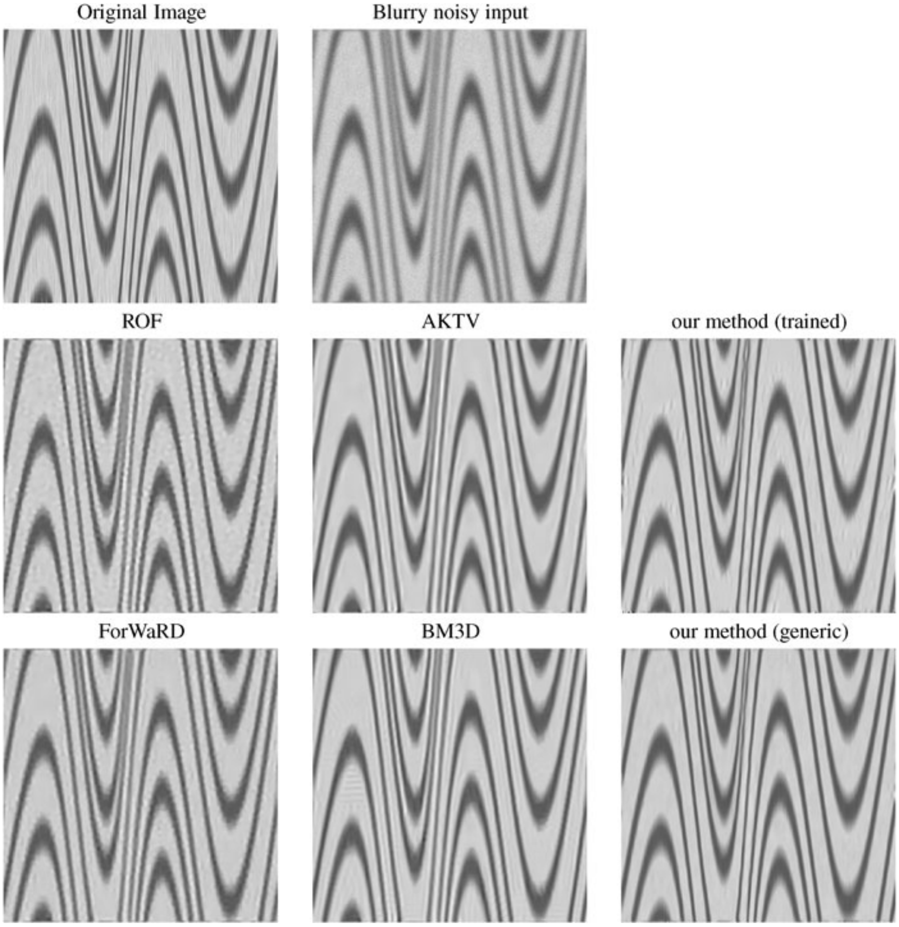


Fig. 10 The dictionary is either trained or generic (comprised of random samples from the training image). Our method using either dictionary improves upon traditional methods, with the generic dictionary providing further improvement over the trained one

the image reconstruction. One could code non-overlapping patches independently, but there is no guarantee for the smooth transition between neighboring patches. Yang et al. [32] process the patches in a raster-scan order with one additional constraint in the sparse coding step, that is, to enforce the overlap between the current target patch to match with previously reconstructed ones. This amounts to adding a linear equation into the optimization, thus easy to solve. However, it is not satisfactory since the results depend on the order of the scan. A better approach would consist in using a partition of unity of the domain of the image to trade off boundary artifacts while avoiding multiple encoding of the same pixel. We intend to pursue this approach as part of our future work.

Acknowledgements This work is supported by ONR grant N000140810363, NSF grant ACI-0321917, ONR N00014-08-1-0414 and AFOSR FA9550-09-1-0427. We wish to thank Guillermo Sapiro for detailed comments and suggestions on an earlier draft of the manuscript.

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

1. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans. Signal Process. **54**(11), 4311–4322 (2006)
2. Andrews, H.C., Hunt, B.R.: Digital Image Restoration. Prentice-Hall, Englewood Cliffs (1977)
3. Cai, J., Ji, H., Liu, C., Shen, Z.: Blind motion deblurring from a single image using sparse approximation. In: Intl. Conf. on Comp. Vis. and Patt. Recog. (CVPR), Miami, Florida, USA, 2009
4. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3D transform-domain collaborative filtering. IEEE Trans. Image Process. **16**(8) (2007)
5. Dabov, K., Foi, A., Egiazarian, K.: Image restoration by sparse 3D transform-domain collaborative filtering. In: Proc. SPIE Electronic Imaging, vol. 6812-07, San Jose, California, USA, January 2008
6. Datsenko, D., Elad, M.: Example-based single document image super-resolution: a global map approach with outlier rejection. Multidimens. Syst. Signal Process. **18**(2–3), 103–121 (2007)
7. Dias, J.M.B.: Fast GEM wavelet-based image deconvolution algorithm. In: Intl. Conf. on Image Process. (ICIP), vol. 3, pp. 961–964 (2003)
8. Dobrosotskaya, J.A., Bertozzi, A.L.: A wavelet-Laplace variational technique for image deconvolution and inpainting. IEEE Trans. Image Process. **17**(5), 657–663 (2003)
9. Efros, A.A., Leung, T.K.: Texture synthesis by non-parameteric sampling. In: Intl. Conf. Computer Vision (ICCV), vol. 2, pp. 1033–1038 (1999)

10. Elad, M., Aharon, M.: Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans. Image Process. **15**(12), 3736–3745 (2006)
11. Favaro, P., Soatto, S., Vese, L., Osher, S.J.: Shape from anisotropic diffusion. In: Intl. Conf. on Comp. Vis. and Patt. Recog. (CVPR), pp. 179–186 (June 2003)
12. Figueiredo, M., Nowak, R.: An EM algorithm for wavelet-based image restoration. IEEE Trans. Image Process. **12**, 906–916 (2003)
13. Figueiredo, M., Nowak, R.: A bound optimization approach to wavelet-based image deconvolution. In: Intl. Conf. on Image Process. (ICIP), vol. 2, pp. 782–785 (2005)
14. Foi, A., Dabov, K., Katkovnik, V., Egiazarian, K.: Shape-adaptive DCT for denoising and image reconstruction. In: Proc. of SPIE Electronic Imaging, Image Processing: Algorithms and Systems V. 6064, 6064N (2006)
15. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-based super-resolution. In: IEEE Computer Graphics and Applications, 2002
16. Grenander, U.: General Pattern Theory: A Mathematical Study of Regular Structures. Cambridge University Press, Cambridge (1993)
17. Guerrero-Colon, J.A., Mancera, L., Portilla, J.: Image restoration using space-variant Gaussian scale mixtures in overcomplete pyramids. IEEE Trans. Image Process. **17**(1), 27–41 (2008)
18. Huang, J., Mumford, D.: Statistics of natural images and models. In: Intl. Conf. on Comp. Vis. and Patt. Recog. (CVPR), pp. 541–547 (1999)
19. Lewicki, M.S., Olshausen, B.A.: Inferring sparse, overcomplete image codes using an efficient coding framework. In: Advances in Neural Information Processing 10 (Proc. NIPS '97), pp. 815–821 (1998)
20. Ma, J., Le Dimet, F.-X.: Deblurring from highly incomplete measurements for remote sensing. IEEE Trans. Geosci. Remote Sens. **47**(3), 792–802 (2009)
21. Mairal, J., Elad, M., Sapiro, G.: Learning multiscale sparse representations for image and video restoration. IEEE Trans. Image Process. **17**(1), 53–69 (2008)
22. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Intl. Conf. Computer Vision (ICCV), vol. 2, pp. 416–423 (2001)
23. Mignotte, M.: A segmentation-based regularization term for image deconvolution. IEEE Trans. Image Process. **15**(7), 1973–1984 (2006)
24. Neelamani, R., Choi, H., Baraniuk, R.G.: ForWaRD: Fourier-wavelet regularized deconvolution for ill-conditioned systems. IEEE Trans. Signal Process. **52**(2), 418–433 (2004)
25. Portilla, J., Strela, V., Wainwright, M., Simoncelli, E.P.: Image denoising using scale mixtures of Gaussians in the wavelet domain. IEEE Trans. Image Process. **12**(11), 1338–1351 (2003)
26. Protter, M., Elad, M., Takeda, H., Milanfar, P.: Generalizing the non-local-means to super-resolution reconstruction. IEEE Trans. Image Process 36 (2009)
27. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60**, 259–268 (1992)
28. Takeda, H., Farsiu, S., Milanfar, P.: Kernel regression for image processing and reconstruction. IEEE Trans. Image Process. **16**(2), 347–364 (2007)
29. Takeda, H., Farsiu, S., Milanfar, P.: Deblurring using regularized locally adaptive kernel regression. IEEE Trans. Image Process. **17**(4), 550–563 (2008)
30. Tikhonov, A., Arsenin, V.: Solution of Ill-Posed Problems. Wiley, New York (1977)
31. Tropp, J.A.: Greed is good: Algorithmic results for sparse approximation. IEEE Trans. Inf. Theory **50**, 2231–2242 (2004)
32. Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution as sparse representation of raw image patches. In: Intl. Conf. on Comp. Vis. and Patt. Recog. (CVPR), 2008