Optics & Laser Technology 54 (2013) 128–136

Contents lists available at ScienceDirect

Optics & Laser Technology

journal homepage: www.elsevier.com/locate/optlastec

Edge-preserving image deconvolution with nonlocal domain transform

Hang Yang [a,*], Zhongbo Zhang [b], Ming Zhu [a], Heyan Huang [b]

[a] Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Science, Changchun 130033, China
[b] School of Mathematics, Jilin University, Changchun 130012, PR China

ARTICLE INFO

Article history:
Received 31 March 2013
Received in revised form
9 May 2013
Accepted 17 May 2013
Available online 11 June 2013

Keywords:
Image deconvolution
Joint nonlocal domain transform
Regularization parameter

ABSTRACT

In this paper, we propose a new approach for performing efficient edge-preserving image deconvolution algorithm based on a nonlocal domain transform (NLDT). We present the geodesic distance-preserving transforming procedure of a 1D signal embedded in 2D space into a new 1D domain via a transformation for simplicity. The nonlocal domain transform derives from the (1D) nonlocal means filter kernel and iteratively and separably applies 1D edge-aware operations. In order to solve the main issue with noisy images that is finding robust estimates for their derivatives, we develop an efficient joint nonlocal domain transform filter in the deblurring process. Furthermore, we derive the discrepancy principle to automatically adjust the regularization parameter at each iteration. We compare our deconvolution algorithm with many competitive deconvolution techniques in terms of ISNR and visual quality.

1. Introduction

Image deconvolution is a classical inverse problem existing in a wide variety of image processing fields, including physical, optical, medical, and astronomical applications. For example, practical satellite images are often blurred due to limitations such as aperture effects of the camera, camera motion, or atmospheric turbulence [1]. Deconvolution becomes necessary when one wishes a crisp deblurred image for viewing or further processing. The degradation procedure is often modeled as the result of a convolution with a low-pass filter

$$y(n_1,n_2)=\mathcal{H}u_{orig}(n_1,n_2)+\gamma(n_1,n_2)$$
$$=(h*u_{orig})(n_1,n_2)+\gamma(n_1,n_2) \quad (1)$$

where u_{orig} and y are the original image and the observed image, respectively. γ is the noise introduced in the procedure of image acquisition, and it is generally assumed to be independent and identically distributed (i.i.d.) zero-mean additive white Gaussian noise (AWGN) with variance σ^2. "$*$" denotes convolution, and h denotes the point spread function (PSF) of a linear shift-invariant (LTI) system $\mathcal{H}$.

In the discrete Fourier transform (DFT) domain, Eq. (1) can be written as

$$Y(k_1,k_2)=H(k_1,k_2)\cdot U_{orig}(k_1,k_2)+\Gamma(k_1,k_2) \quad (2)$$

where Y, H, U_{orig} and Γ are the discrete Fourier transform of y, h, u_{orig}, and γ, respectively. Given y and h, we seek to estimate u_{orig}. The process of deconvolution is known to be an ill-posed problem, which can be interpreted as an inverting lowpass filtering, backward diffusion, or entropy decreasing. Thus, to obtain a reasonable image estimate, a method of reducing/controlling noise needs to be utilized.

To find a unique and stable solution, a number of deconvolution algorithms have been proposed. In these methods, the Wiener filter [2,3] and the constrained least squares algorithm [2] can solve this problem in the frequency domain in a fast speed. However, they often obtain a noisy result with ringing effects. As a result, the visual quality of the recovered image often degrades. Increased performance of deconvolution methods can be attributed to the inclusion of the wavelet-based estimators. One such technique called the Fourier-Wavelet Regularized Deconvolution (ForWaRD) was proposed in [4], which works with any ill-conditioned convolution system. This method can obtain good results via tandem scalar shrinkages in both the Fourier and wavelet domain. Recently, considerable effort has been spent on designing alternative sparsity constraints which preserve such features. Methods based on these sparsity constraints and incomplete measurements [5] have been successfully used for image deconvolution. Transformations such as wavelets, curvelets [6], shearlets [7] and wave atoms [8] are popular for image representation and are often used for image restoration. It has been shown that learning representation from examples instead of using prespecified ones usually leads to improved results [9,10].

Another popular deconvolution method is based on total variation [11,12]. The TV deconvolution method finds approximate solutions to differential equations in the space of bounded variation (BV) functions. Variations of this method have also been proposed in [13] (FTVd), [14] (TVMM), [15] (TVS). These methods

* Corresponding author. Tel.: +86 15044389995; fax: +86 04313218515.
E-mail addresses: yhang3109@163.com, yanghang09@mails.jlu.edu.cn (H. Yang).

0030-3992/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.optlastec.2013.05.020

are well known for its edge-preserving, can generally achieve state-of-the-art results. In [16], iterative shrinkage/thresholding (IST) algorithms were placed on solid mathematical grounds, and FISTA (Fast IST algorithm) [17], TwIST (Two-step IST) [18] have improved the IST algorithm.

An inhomogeneous deconvolution model under the Bayesian framework exploiting a non-parametric adaptive prior distribution derived from non-local means method (NLRS) was proposed in [19]. In particular, the SV-GSM [20] which employs Gaussian scale mixtures in overcomplete directional and multiresolution pyramids, and the BM3D (Block Matching 3D) [21,22] which employs a non-local modeling of images by collecting similar image patches in 3D arrays, are among the current best image deconvolution methods. There are many useful algorithms and additional techniques which may be found within the references [23,24].

In this paper, we adopt a different approach to the problem of image restoration by exploiting a novel nonlocal domain transform (NLDT) to regularize the inverse problem. Recently, Gastal and Oliveira [25] propose a O(N) time filter known as the Domain Transform (DT) filter. The key idea is to iteratively and separably apply 1D edge-aware filters. In our work, we replace the point similarity in DT [25] with the self-similarities of neighborhoods to find a new transformation (nonlocal domain transform) that maintains the edge-preserving property of the filter. Our deconvolution algorithm is based on the decoupling of deblurring and denoising steps in the restoration process. In the deblurring step, an efficient deblurring method using fast Fourier transforms can be employed. In the denoising step, we present a new approach based on the nonlocal domain transform for efficiently performing edge-preserving filtering. The idea is originally from geodesic distance map, which keeps the edge characteristics in image processing. The transformation enables the aggregation of 2D cost data to be accomplished using a sequence of 1D filters. Furthermore, regularization parameter λ plays an important role in our work. By adjusting λ, a compromise is achieved to suppress the noise and preserve the nature of the original image. In this paper, we apply the discrepancy principle to automatically determine regularization parameter in each iteration.

The contributions of our work include: (1) a nonlocal domain transform for efficiently performing edge-preserving filtering of images is proposed; (2) we present an efficient edge-aware image deconvolution method based on the joint nonlocal domain transform strategy; (3) we apply the discrepancy principle to automatically find a value of the regularizer λ.

In Section 2, we discuss the fundamentals of the domain transform and propose the nonlocal domain transform. In Section 3, we show how the nonlocal domain transform is used for regularizing the deconvolution problem and how to compute the regularizer λ. Simulation results are presented in Section 4 and concluding remarks are presented in Section 5.

2. Nonlocal domain transform

Our algorithm is inspired by the interpretation of edge-preserving filters [26]. Let $I:\Omega\subset R^2\rightarrow R$ be a 2D image. Also, let $\hat{p}=(x_p,y_p,I(p))$ be a point with spatial coordinates $p=(x_p,y_p)$ and range coordinate $I(p)$.

Let $F(\hat{p},\hat{q})$ be an edge-preserving filter kernel. In [25], authors addressed the fundamental question of whether there exists a transformation T and a filter kernel K that, for any input image I, produce an equivalent result as the edge-preserving kernel F:

$$J(p)=\int_{\Omega} I(q)F(\hat{p},\hat{q})\,dq=\int_{\Omega} I(q)K(T(\hat{p}),T(\hat{q}))\,dq \tag{3}$$

This is a dimensionality reduction technique that replaces the evaluation of a computationally expensive filter F with a lower-dimensional linear filter K and a domain transformation T.

2.1. Domain transform

A new O(N) time edge-preserving filter known as Domain Transform filter was proposed in [25]. This very efficient filter is derived from the (1D) bilateral kernel. The key idea is to iteratively and separably apply 1D edge-aware filters. We introduce the domain transform (DT) as follows.

The domain transform [25] states that the L1 distance between two neighboring points in the original domain R^2 and the distance between two corresponding samples in the new dimensionality reduction domain R must be equal. Let $I(x)$ be a 1D signal for $x\in\Omega$, and $S=\{x_0,x_1,\ldots,x_n\}$ be a sampling of Ω, where $x_{i+1}=x_i+l$, for some sampling interval l. The goal is to seek a transform T that satisfies

$$|T(x_i,I(x_i))-T(x_j,I(x_j))|=|x_i-x_j|+\|I(x_i)-I(x_j)\|, \tag{4}$$

where $x_i,x_j\in S$, $|\cdot|$ is the absolute value operator, and $\|\cdot\|$ is some chosen metric.

For simplicity, in [25], authors use the nearest-neighbor l_1 norm; to be isometric, the desired transform must satisfy the following equality (in l_1 norm):

$$ct(x+l)-ct(x)=l+|I(x+l)-I(x)| \tag{5}$$

where $ct(x)=T(x,I(x))$ represents the transformation operator at point x. Following the derived strategy presented in [25], we divide both sides of Eq. (5) by l and take the limit as $l\rightarrow 0$. The obtained result is the derivative of $ct(x)$ with respect to x:

$$ct'(x)=1+|I'(x)| \tag{6}$$

Integrating (6) on both sides and letting $ct(0)=0$, one get

$$ct(\alpha)=\int_0^{\alpha}(1+|I'(x)|)\,dx \tag{7}$$

One call ct a domain transform (DT).

The analysis presented in [25] shows that we can also control the influence of spatial and intensity range information similar to the bilateral filter by embedding the values of σ_r and σ_s in the transformation:

$$ct(\alpha)=\int_0^{\alpha}\left(1+\frac{\sigma_s}{\sigma_r}|I'(t)|\right)dt\,dx \tag{8}$$

This is an excellent property since we do not lose the controllability of spatial and intensity information in spite of using lower dimensionality.

In [25], authors demonstrate realization of 1D domain transform filter by using a recursive edge-preserving filter. For a discrete signal $I[n]=I(x_n)$ and a given domain transform ct, this filter can be defined in the transformed domain as

$$J[n]=(1-\omega^d)I[n]+\omega^d J[n-1] \tag{9}$$

where $d=ct(I[n])-ct(I[n-1])$ is the distance between neighbor samples $I[n]$ and $I[n-1]$ in the transformed domain. While the two samples are different, such as at the edges of the signal, the value of d is sufficiently small for preventing the propagation chain, so that the edges can be preserved. We denote the result as $J=\mathbf{DT}(I)$.

The filter variance ω of Eq. (9) can be computed as $\omega=\exp(-\sqrt{2}/\sigma_s))$ (see more details in [25]). The filter is stable since $\omega\in[0,1]$, and its implementation in $O(N)$ time is straightforward.

Domain transform filter is a new approach for performing high-quality edge-preserving filtering of images and videos in real time. The realization for 1D filter based on recursion has very distinct impulse responses, making more appropriate for specific

applications. It manages to continuously smooth image regions while preserving strong edges.

2.2. Nonlocal domain transform

Recently, Buades et al. have proposed a natural and elegant extension of the image bilateral filtering paradigm, so-called Non-Local means (NL-means) filter [27]. For a given pixel p, its new intensity value is computed as a weighted average of grey level values within a search window. The weight of the pixel q in this weighted average is proportional to the similarity (according to the euclidean distance) between the neighborhood configurations of pixels p and q. In this procedure, the denoising process is due to the regularity assumption that self-similarities of neighborhoods exist in a real image and that one (or several) neighborhood configuration(s) can efficiently predict the central value of the pixel.

In this section, the idea proposed by [27] is herein used to derive an efficient nonlocal domain transform (NLDT). For 1D signals, we replace the point similarity in domain transform [25] with the self-similarities of neighborhoods to find a transformation T that maintains the edge-preserving property of the filter.

In our work, we use the weighted l_1 distance as the metric $\|\cdot\|$ in Eq. (4). The distance between two sampling x_i and x_j depends on the similarity of the intensity level vectors $I(L_i)$ and $I(L_j)$, where L_i denotes a line neighborhood of fixed size and centered at a point x_i. This similarity is measured as a decreasing function of the weighted l_1 distance, $\|I(x_i)-I(x_j)\| = \|I(L_i)-I(L_j)\|_{1,a} = G_a * |I(x_i+\cdot)-I(x_j+\cdot)|(0)$, where $a>0$ is the standard deviation of the Gaussian kernel. T only needs to preserve the weighted distances between neighboring samples x_i and x_{i+1}.

Finally, let $ct_{nl}(x) = T(x, I(x))$. To be weighted isometric, the desired transform must satisfy the following equality:

$$\begin{aligned} ct_{nl}(x+l)-ct_{nl}(x) &= l + G_a * |I(x+l+\cdot)-I(x+\cdot)|(0) \\ &= l + \int_{-s}^{s} G_a(p)|I(x+l-p)-I(x-p)|\,dp, \end{aligned} \tag{10}$$

which states that the weighted l_1 distance between neighboring samples in the new domain (R) must equal the distance between them in the original domain (R^2), and G_a is a Gaussian kernel, the line neighborhood of size is 2 s. To avoid the need for the absolute value operator on the left of (10), we constrain $ct(x)$ to be monotonically increasing ($ct_{nl}(x+l) \geq ct_{nl}(x)$). Dividing both sides of (10) by l and taking the limit as $l \to 0$ we obtain

$$\begin{aligned} ct'_{nl}(x) &= \lim_{l\to 0} 1 + \int_{-s}^{s} \frac{1}{l} G_a(p)|I(x+l-p)-I(x-p)|\,dp \\ &= 1 + \lim_{l\to 0} \int_{x-s}^{x+s} \frac{1}{l} (G_a(x-t)|I(t+l)-I(t)|)\,dt \\ &= 1 + \int_{x-s}^{x+s} G_a(x-t)|I'(t)|\,dt \end{aligned} \tag{11}$$

where $ct'_{nl}(x)$ denotes the derivative of $ct_{nl}(x)$ with respect to x. Integrating (11) on both sides and letting $ct_{nl}(0) = 0$, we get

$$ct_{nl}(\alpha) = \int_0^{\alpha} \left(1 + \int_{x-s}^{x+s} G_a(x-t)|I'(t)|\,dt\right) dx \tag{12}$$

The distance between any two points α and β in Ω, $\beta \geq \alpha$, in the new transformed domain, which corresponds to the arc length from α to β of the signal I, is also subsequently computed by

$$ct_{nl}(\beta)-ct_{nl}(\alpha) = \int_{\alpha}^{\beta} \left(1 + \int_{x-s}^{x+s} G_a(x-t)|I'(t)|\,dt\right) dx \tag{13}$$

As such, the transformation given by Eq. (12) preserves the weighted geodesic distance between all points on the signal. We call ct_{nl} a nonlocal domain transform.

Similar to the domain transform (8), we obtain the final nonlocal domain transform to encode the values of σ_r and σ_s in the transformation itself:

$$ct_{nl}(\alpha) = \int_0^{\alpha} \left(1 + \int_{x-s}^{x+s} \frac{\sigma_s}{\sigma_r} G_a(x-t)|I'(t)|\,dt\right) dx \tag{14}$$

The recursive edge-preserving filter (9), $J[n] = (1-\omega^{d_{nl}})I[n] + \omega^{d_{nl}} J[n-1]$, can be used for performing this filtering operation on digital signals. For the nonlocal domain transform, $d_{nl} = ct_{nl}(I[n]) - ct_{nl}(I[n-1])$.

As explained previously in Section 2.1, this recursive filter can preserve the signal's edges well. From Eq. (14), one can see that the regions with large gradient magnitude (such as the edges of the input signal) are preserved, whereas those where gradient magnitude is not significant are smoothed by using the same response of a linear smoothing filter. Note that in both cases, our nonlocal domain transform filter behaves as an edge-aware one. We denote the nonlocal domain transform as $J = \mathbf{NLDT}(I)$.

Fig. 1 gives an example of the defined nonlocal domain transform for an input signal. The input signal $I(x) = 10 \sin(2\pi/512)x$ ($x \in [0, 511]$)) is shown in the left of Fig. 1, and the right of Fig. 1 shows the associated nonlocal domain transform $ct_{nl}(\alpha)$ computed using Eq. (14).

The problem with domain transform (DT) filters is that comparing only grey level values in a single pixel is not so robust when these values are noisy. The NLDT compares not only the grey level in a single point but also the geometrical configuration in a whole neighborhood. This fact allows a more robust comparison than DT filter.

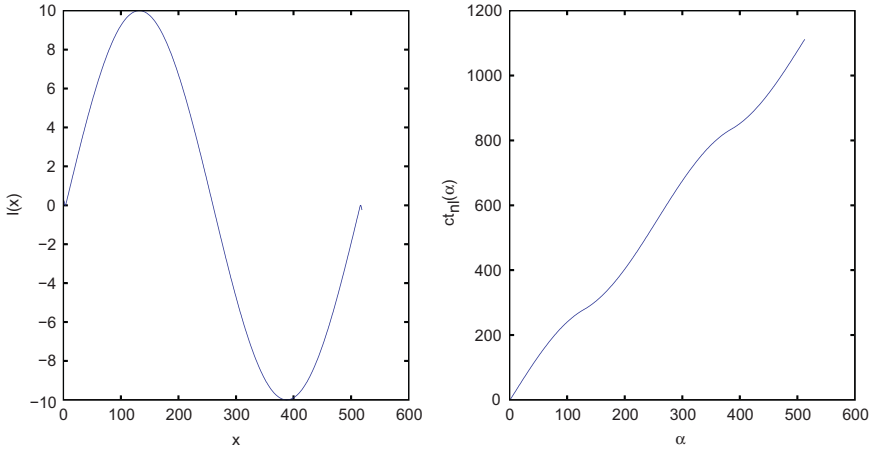

Fig. 1. Left: Input signal $I(x) = 10 \sin(2\pi/512)x (x \in [0, 511])$. Right: Its nonlocal domain transform computed using Eq. (14).

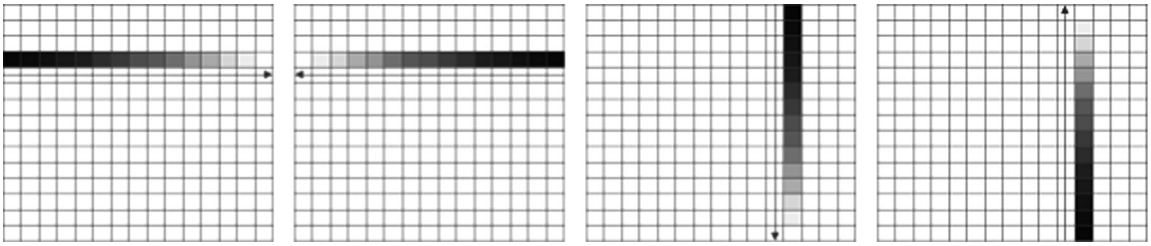

Fig. 2. Illustration of the process with separate horizontal and vertical filtering. From left to right: left-to-right, right-to-left, top-to-bottom, bottom-to-top.

2.3. Filtering 2D images

Eq. (14) defines a nonlocal domain transform for 1D signals. Gastal et al. [25] suggested that we can filter 2D signals using a chain of 1D operations by performing separate filtering. The horizontal and vertical passes are conducted for each row and column, respectively. More precisely, assuming the horizontal pass is first performed using the initial input signals, and the vertical pass is then applied to the result using the horizontal filtered output. This situation is illustrated in Fig. 2. The required number of horizontal and vertical filtering depends on (the geometry of) the image content, and, therefore, is hard to predict. In practice, four iterations usually suffice to achieve good results.

3. Nonlocal domain transform based image deconvolution algorithm

3.1. Fourier based regularized deconvolution

Having established a method for obtaining a good image estimate when the image is corrupted by colored noise $\mathcal{H}^{-1}\gamma$, let us focus on how we are to use this method as part of a deconvolution routine. The blurring model is described by Eq. (1), a suitable pseudo-inverse estimate which can be found by regularizing the convolution operator from a discrete Fourier basis. Using the regularized inverse operator

$$\mathbf{H}(k_1,k_2)=\frac{\overline{H}(k_1,k_2)}{|H(k_1,k_2)|^2+\lambda} \tag{15}$$

for some regularizing parameter $\lambda\in R^+$, $\overline{H}$ is the complex conjugate of H. An image estimate u_λ in the Fourier domain is given by

$$\begin{aligned} U_\lambda(k_1,k_2) &= Y(k_1,k_2)\cdot\mathbf{H}(k_1,k_2)\\ &= U(k_1,k_2)\frac{|H(k_1,k_2)|^2}{|H(k_1,k_2)|^2+\lambda}+\Gamma_\lambda(k_1,k_2) \end{aligned} \tag{16}$$

where Γ_λ denotes the DFTs of leaked noise γ_λ. This type of regularization applied is often referred to as FoRD. The strengths of FoRD can be found in [4].

3.2. Joint filtering

To suppress the amplified noise and artifacts introduced by FoRD method, we plan to apply the nonlocal domain transform to denoise the estimate image u_λ. But in the denoising process, the main issue with noisy images is finding robust estimates for their derivatives, which cannot be done using simple forward differences. One possible solution is pre-filtering the image with a low-pass filter to avoid large oscillations in the image gradient. So, we propose the joint filtering, where the content of one image is smoothed based on the edge information from a reference image.

The main problem of the nonlocal domain transform filter in image denoising is that the d_{nl} in Eq. (9) could not be estimated accurately based on the noisy image. If a reference contains a much better estimate of the true high-frequency information than the noisy image [28], we can present a joint nonlocal domain transform filter to compute the parameter d_{nl} as follows:

$$d_{nl}=ct_{nl}(I^{ref}[n])-ct_{nl}(I^{ref}[n-1]) \tag{17}$$

where I^{ref} is the reference image. The result can be written as $J=\mathbf{JNLDT}(I,I^{ref})$.

After the Fourier shrinkage step [see Eq. (16)], the deblurred image u_λ contains the leaked noise γ_λ. So, the d_{nl} could not be estimated accurately based on the noisy image u_λ. Considering that the pre-denoising image preserves most of the important image features, we can use it as a reference image [see Proposed Algorithm]. In this way, the d_{nl} could be estimated more accurately using Eq. (17).

3.3. Proposed deconvolution algorithm

Based on Morozov discrepancy principle, which selects λ by matching the norm of the residual to some upper bound, a good regularized solution u should lie in the set $\{u;\|h*u-y\|_2^2\le c^2\}$, where c is a constant that depends on the noise level [29,30]. In general, when the variance of the noise is available, the upper bound is given by $c^2=N^2\sigma^2$.

It was observed, however, in [30] that the choice of λ based on $c^2=N^2\sigma^2$ usually yields an oversmooth solution, which implies that λ is too large. In order to obtain a smaller λ, which could preserve the image edges or textures and also suppress the amplified noise, we use a smaller set $\mathcal{K}=\{u;\|h*u-y\|_2^2\le\rho N^2\sigma^2,\ \rho<1\}$ in this paper.

Let u_E is the pre-estimate image, we solve the following problem with u_E and $\mathcal{K}$:

$$\hat{u}=\arg\min_u\{\lambda\|u-u_E\|_2^2+\|h*u-y\|_2^2\} \tag{18}$$

and we choose the parameter λ to let $\hat{u}\in\mathcal{K}$.

This minimization problem (18) can be solved in the Fourier domain easily by

$$\hat{U}(k_1,k_2)=\frac{\overline{H}(k_1,k_2)\cdot Y(k_1,k_2)+\lambda U_E(k_1,k_2)}{|H(k_1,k_2)|^2+\lambda} \tag{19}$$

where $\hat{U}$, H, Y and U_E are the 2-D DFTs of $\hat{u}$, h, y, and u_E, respectively.

From Eq. (16), one can find that the Fourier based regularized method (FoRD) is the special condition of Eq. (19) with $u_E=0$. Furthermore, regularization parameters λ and ρ play the important roles in our work. In Section 3.4, we will discuss how to choose them.

We summarize the main steps of the proposed image deconvolution algorithm as follows:

Proposed Algorithm.

1. Set $k=0$, $u^k=0$, σ_s, σ_r and $u^{ref}=y$.
2. **Repeat**:
3. Use u^k and the set $\mathcal{K}=\{u;\|h*u-y\|_2^2\le\rho N^2\sigma^2\}$ to obtain the estimate v^k using the Eq. (19).

4. Apply our joint domain transform and edge-preserving filters to v^k, compute the parameter d using Eq. (17) with reference image u^{ref}, and obtain an estimate $u^{k+1} = JNLDT(v^k, u^{ref})$.
5. Set $k = k+1$, $u^{ref} = u^{k+1}$, and update σ_r.
6. **Until** stopping criterion is satisfied.

We adaptively set the spatial variance $\sigma_s = r_h/3$, where r_h is the radius of the blur kernel h. During the iterative restoration procedure, σ_r is then refined, at each step k of the iterative restoration process. We adaptively set the range variance σ_r to $0.04 \times |\max(u^{ref}) - \min(u^{ref})|^2$ by following [31].

3.4. Choose regularization parameter

Note that the Fourier-based regularized inverse operator in Eq. (15) and the solution in Eq. (18) depends on the choice of regularization parameter λ. The deblurred image depends greatly on the degree of regularization which is determined by the regularization parameter [32]. And the value of λ is related with the other parameter ρ. Now, we describe a simple and effective method to compute the parameters λ and ρ automatically. It depends only on the data and automatically adjusts the regularization parameter according to the data.

First, we show how to determine the parameter λ. If the pre-estimate image $u_E \in \mathcal{K}$, we set $\hat{u} = u_E$, and $\lambda = \infty$; else, the parameter λ in Eq. (19) can be chosen as follows: in the case where we know some statistics of additive noise, e.g., the standard deviation of noise, a proper parameter λ is chosen such that the restored image $\hat{u}$ in (19) satisfies

$$\|h * \hat{u} - y\|_2^2 = \rho N^2 \sigma^2 \tag{20}$$

By Parseval's theorem, this is equivalent to finding λ satisfying

$$\rho N^2 \sigma^2 = \|H \cdot \hat{U} - Y\|_2^2 = \| \frac{\lambda(H \cdot U_E - Y)}{|H|^2 + \lambda} \|_2^2 \tag{21}$$

Notice that the right-hand side is monotonically increasing function in λ, hence there exist a unique solution λ, which can be *determined via bisection*.

From Eq. (21), one can see that the λ increases with the increase of ρ. A typical choice is to set $\rho = 1$ [29,30]. But in practice, we find that the large $\lambda(\rho = 1)$ often causes a noisy result with ringing effects, though it can substantially reduce the noise variances. So, we should choose a smaller $\lambda(\rho < 1)$ which would obtain an edge preserving image with a few noise. Then, in the denoising step, our effective approach based on domain transform can be employed.

The parameter ρ should satisfy an important property: the ρ decreases with the increase of image variance. For example, a smooth image which contains a few high-frequency information will not produce the strong ringing effects with large ρ, and a large ρ could substantially suppress the noise. According to this property, we choose the ρ as follow:

$$\rho = \sqrt{1 - \frac{\|y - E(y)\|_2^2 - N^2\sigma^2}{\|h\|_1^2 \|y\|_2^2}} \tag{22}$$

where $E(y)$ denotes the mean of y.

Concerning the algorithm complexity we note that the algorithm is fast as it is based on fast domain transform. We start by analyzing the cost of each iteration. The Step (3) can be carried out with cost $O(N^2 \log N)$ using the FFT algorithm; in Step (4), recursive filtering is used to compute joint domain transform. The cost of this phase is $O(N^2)$. And the cost of computing the parameter λ is also low because it is to use FFT algorithm.

4. Experiments

First, we have performed an experiment to evaluate the effectiveness of the proposed NLDT filter compared with the DT filter. The results are shown in the Fig. 3. The proposed NLDT filter outperforms the DT filter in PSNR for the "Peppers" image.

Then, we present results of our proposed algorithm and compare them with some of the deconvolution methods such as ForWaRD [4], SV-GSM [20], TVS [15], NLRS [19], L0-AbS [23]. We also use the domain transform (DT) filter to replace the nonlocal domain transform (NLDT) in the Proposed Algorithm, and we show that the proposed NLDT-based filter can outperform DT-based filter when applied to deconvolution problem. In these experiments we will use the improvement in signal-to-noise-ratio (ISNR) to measure the performance. The ISNR is defined as $ISNR = 10 \log_{10}(\|u_{orig} - y\|_2^2 / \|u_{orig} - \hat{u}\|_2^2)$, where $\hat{u}$ is the corresponding estimated image.

We consider six benchmark deconvolution problems. In these experiments, original images are Cameraman (experiments 1, 2 and 3) of size 256×256, Lena of size 512×512 (experiments 4), House of size 256×256 (experiment 5) and Boat of size 512×512 (experiment 6). Table 1 summarizes the different degradation models used, which are defined by the blur type and the variance of the additive white Gaussian noise for each of the experiments. The original images are shown in Fig. 4. We have tested our method with different blur models at various noise levels. Good and consistent results have been achieved.

In the experiments, the proposed method outperforms the other techniques in terms of ISNR. We note that the results of the six standard experiments for different images are included in Table 2 as follows.

In this paper, we also used the Normalized cross-correlation (Ncc) to measure the "visual quality" performance. Table 3 summarizes the results of the six standard experiments in terms of Ncc. The Ncc is defined as $Ncc = \Sigma(u - E(u))(\hat{u} - E(\hat{u}))/\|u - E(u)\|\|\hat{u} - E(\hat{u})\|$, where $E(u)$ denotes the mean of u.

Fig. 3. 2D edge-preserving filtering. From left to right: noisy image (PSNR = 28.48 dB), DT-based denoising result (PSNR = 32.52 dB), NLDT-based denoising result (PSNR = 33.35 dB).

Table 1
Description of the observation parameters for the six experiments.

Scenario	Blur	σ^2
Exp 1	9×9 *uniform* (Cameraman)	0.308
Exp 2	$h_{i,j} = (1 + i^2 + j^2), i, j = -7, \ldots, 7$ (Cameraman)	2
Exp 3	25×25 Gaussian PSF with standard deviation 1.6 (Cameraman)	4
Exp 4	$[1, 4, 6, 4, 1]^T [1, 4, 6, 4, 1]/256$ (Lena)	49
Exp 5	25×25 Gaussian PSF with standard deviation 1.6 (House)	4
Exp 6	9×9 uniform (Boat)	0.308

Fig. 4. Images used in this paper for different experiments. (a) Cameraman image, (b) Lena image, (c) House image, (d) Boat image.

Table 2
ISNR for different experiments.

Methods	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6
NLDT	**9.18**	**8.01**	**3.88**	**4.42**	**5.43**	**8.15**
DT	8.47	7.36	3.47	3.93	4.84	7.39
ForWaRD	7.40	6.75	3.14	2.93	3.85	6.78
NLRS	7.81	7.14	3.29	3.84	4.03	7.33
TVS	8.41	7.37	3.39	3.53	4.54	7.85
L0-AbS	9.01	7.70	3.58	4.06	4.79	7.76
SV-GSM	7.33	7.45	3.25	3.61	4.11	7.89

Table 3
Normalized cross-correlation (Ncc) values for different experiments.

Methods	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6
NLDT	**0.9921**	**0.9922**	**0.9840**	**0.9932**	**0.9926**	**0.9896**
DT	0.9857	0.9903	0.9819	0.9919	0.9917	0.9869
ForWaRD	0.9871	0.9896	0.9813	0.9915	0.9897	0.9852
NLRS	0.9886	0.9899	0.9816	0.9911	0.9906	0.9877
TVS	0.9898	0.9907	0.9822	0.9913	0.9908	0.9883
L0-AbS	0.9911	0.9912	0.9829	0.9923	0.9916	0.9879
SV-GSM	0.9865	0.9911	0.9825	0.9907	0.9911	0.9890

All the experiments were performed using MATLAB, on a computer equipped with an Pentium(R) Dual-Core CPU E5300 @2.60HZ and 2G RAM, and running Windows XP. In Table 4, we compare the speed of the algorithms in Table 2.

In the first set of tests, we consider the setup of [4], where a Cameraman image is blurred by a 9×9 uniform box-car blur. The AWGN variance, $\sigma^2 = 0.308$. A comparison of different methods in terms of ISNR is shown in Table 1 under the Exp 1 column. The proposed method yields a value 9.18 dB which is better than the values obtained by any of the other methods. In Fig. 5, we show

Table 4
CPU time (s) for different experiments.

Methods	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6
NLDT	2.53	2.46	2.56	12.2	2.47	11.85
DT	2.41	2.38	2.50	11.92	2.35	11.66
ForWaRD	9.02	9.07	9.03	42.82	9.07	43.10
NLRS	130.17	101.34	120.59	500.42	93.77	438.62
TVS	54.41	21.73	17.85	55.39	14.85	186.74
L0-AbS	25.88	26.17	25.68	67.14	26.23	66.83
SV-GSM	16.49	16.72	16.58	132.67	16.52	133.75

details of the observations and the corresponding restored images for Experiment 1.

In the second set of experiments performed over the Cameraman image, we replicate the experimental setup of [20] (Exp 2 and Exp 3). The ISNR are summarized in Table 2 under the Exp 2 and Exp 3 columns. Again, our deconvolution algorithm outperforms the other methods in terms of ISNR.

In the fourth experiment, the original image of Lena is blurred by a 5×5 separable filter with weights $[1, 4, 6, 4, 1]/16$ in both the horizontal and vertical directions and then contaminated with AWGN by $\sigma = 7$. A portion of the image is zoomed in to reveal the

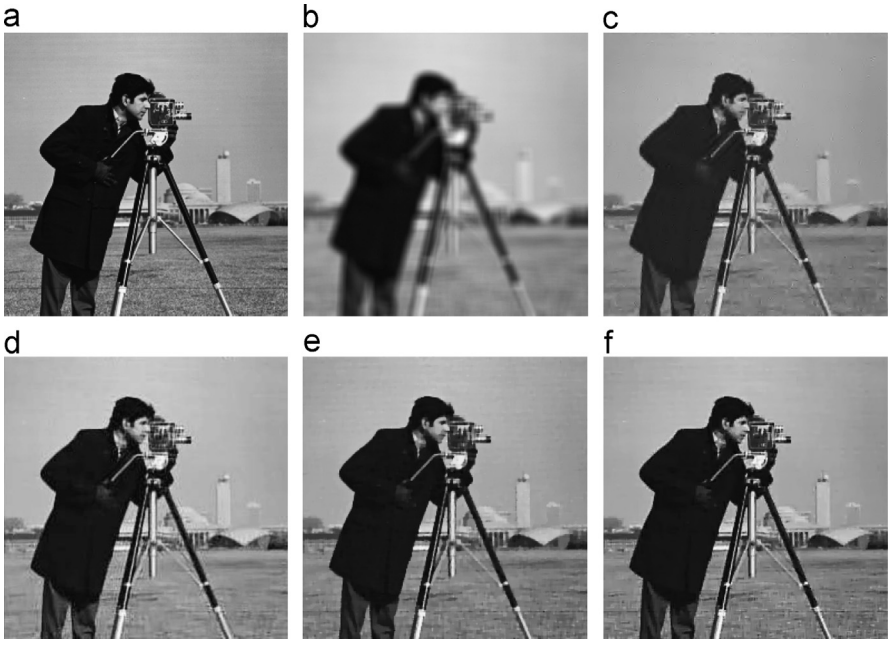

Fig. 5. The results of the first experiment with a 256×256 Cameraman image. (a) Original image, (b) blurred image, (c) ForWaRD result, ISNR=7.40 dB, (d) NLRS result, ISNR=7.81 dB, (e) TVS result, ISNR=8.41 dB, (f) our method result, ISNR=9.18 dB.

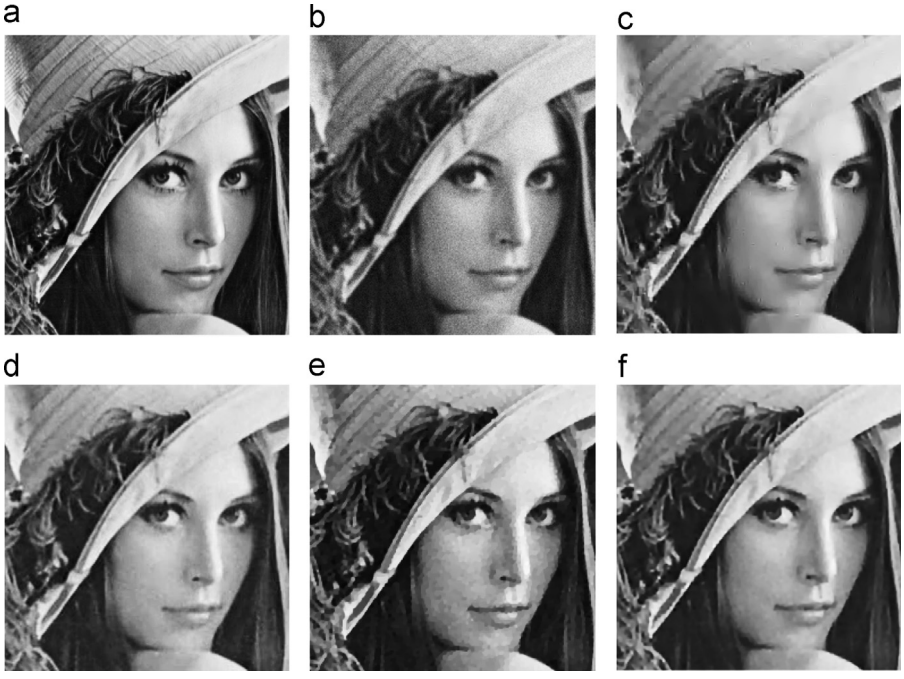

Fig. 6. Details of the image deconvolution experiment with a Lena image. (a) Original image, (b) blurred image, (c) ForWaRD result, ISNR=2.93 dB, (d) NLRS result, ISNR=3.84B, (e) TVS result, ISNR=3.53 dB, (f) our method result, ISNR=4.42 dB.

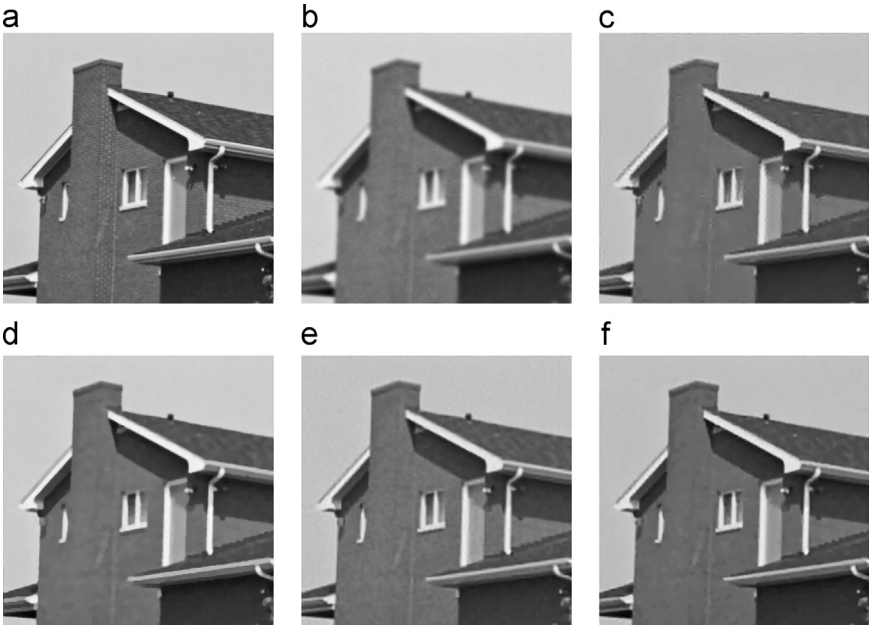

Fig. 7. Details of the image deconvolution experiment with a House image. (a) Original image, (b) blurred image, (c) ForWaRD result, ISNR=3.85 dB, (d) TVS result, ISNR=4.54 dB, (e) L0-AbS result, ISNR=4.79 dB, (f) Our method result, ISNR=5.43 dB.

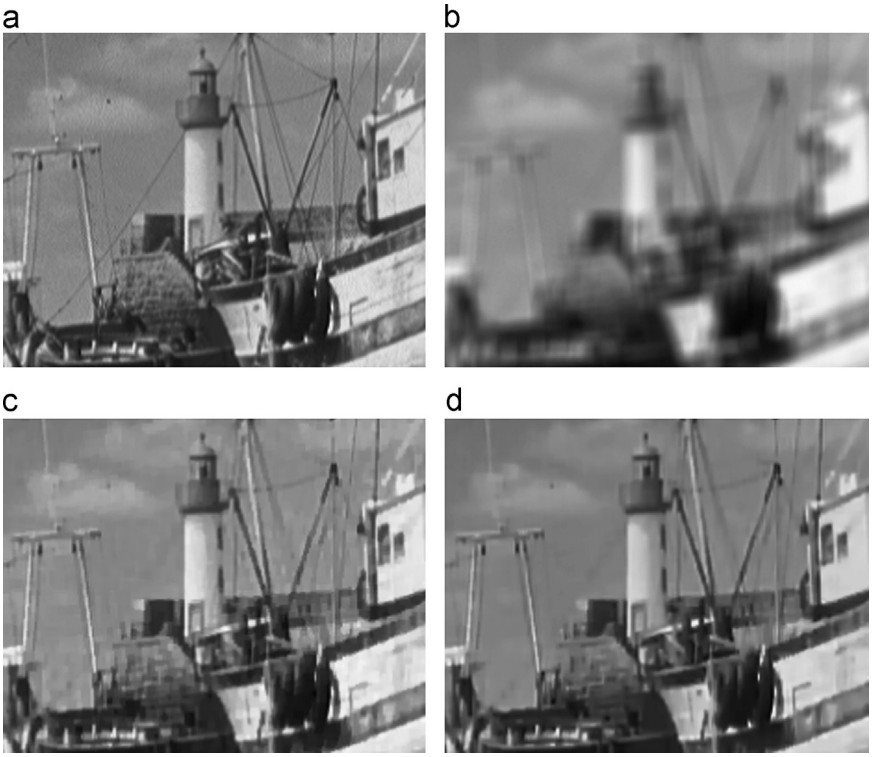

Fig. 8. Visual comparison of *Boat* image in Exp 6. (a) Crop from *Boat* image, (b) blurred image, (c) DT result, ISNR=7.38 dB, (d) NLDT result, ISNR=8.15 dB.

visual details of the results obtained by the different methods, and is shown in Fig. 6(a)–(f). As can be seen from the figure, our NLDT-based method recovers details better than the other methods.

In the fifth experiment, we apply a Gaussian PSF on the House image. The deconvolution results obtained by different methods are reported under the Exp 5 column of Table 2. Our nonlocal domain transform based method performs yielding ISNR values of 5.43 dB. This experiment shows that our proposed method can provide better reconstruction than some of the competitive deconvolution methods. The details of the images obtained by the different methods are shown in Fig. 7

In the sixth experiment, the original image of Boat is blurred by a 9×9 uniform box-car blur, the noise variances are $\sigma^2 = 0.308$. From Table 1, we can notice that our method performs the best in terms of ISNR. Fig. 8 shows a visual comparison between the proposed method and the DT-based method. Results have shown that the NLDT-based method obtains an edge-preserving deblurring result with better quantitative and visual performance.

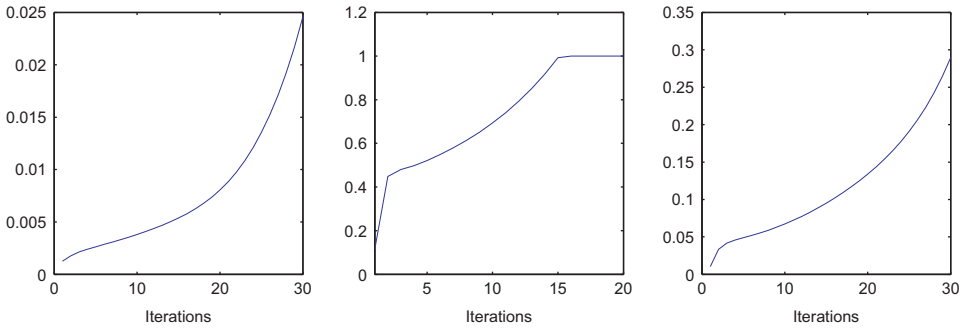

Fig. 9. λ Values obtained by the method described in Eq. (21). From left to right: Cameraman (Exp 1), Lena (Exp 4), House (Exp 5).

In Fig. 9, we display a few curves for different λ values obtained from Experiments 1, 4, and 5. These curves are chosen to be the regularization parameters in each experiment. Hence, unlike some of the other deconvolution algorithms such as that in [13], our method automatically determines the regularization parameter at each iteration.

5. Conclusion and future work

In this work, we have proposed an effective edge-preserving image deconvolution method. Our algorithm is based on a non-local domain transform that preserves the geodesic distance between points on the curve, adaptively warping the input signal so that 1D edge-preserving filtering can be efficiently performed in linear time. We demonstrated the realization for our 1D edge-preserving filters, based on recursion.

To suppress the amplified noise and artifacts, we present a joint nonlocal domain transform filter to denoise the estimate image obtained by FoRD. In addition, we have adapted a method of automatically determining the regularization parameter at each iteration. We have compared the performance of the proposed method against some state-of-the-art methods. Results have shown that the proposed method is attractive to obtain an edge-preserving deconvolution result with better visual and quantitative performance.

In this paper, we have assumed knowledge of the convolution operator. However, the convolution operator is unknown in many cases. In such "blind" deconvolution problems, the convolution system must be estimated from observations. It would also be of interest to apply the method developed here to blind deconvolution.

References

[1] Jain AK. Fundamental of digital image processing. Englewood Cliffs, NJ: Prentice-Hall; 1989.
[2] Katsaggelos AK, editor. Digital image restoration. New York: Springer-Verlag; 1991.
[3] Hansen PC. Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion. Philadelphia, PA: SIAM; 1998.
[4] Neelamani R, Choi H, Baraniuk RG. ForWaRD: Fourierwavelet regularized deconvolution for ill-conditioned systems. IEEE Transactions on Image Processing 2004;52(February (2)):418–33.
[5] Yang Hang, Zhang Zhongbo. Image deblurring based on ForIcM: Fourier shrinkage and incomplete measurements. Imaging Science Journal 2012;60 (6):344–51.
[6] Starck J, Nguyen MK, Murtagh F. Wavelets and curvelets for image deconvolution: a combined approach. Signal Processing 2003;83(10):2279–83.
[7] Patel Vishal M, Easley Glenn R, Healy Jr. Dennis M. Shearlet-based deconvolution. IEEE Transactions on Image Processing 2009;18(December (12)):2673–85.
[8] Yang Hang, Zhang Zhongbo. Fusion of wave atom-based wiener shrinkage filter and joint non-local means filter for texture-preserving image deconvolution. Optical Engineering 2012;51(June).
[9] Mairal J, Elad M, Sapiro G. Sparse representation for color deficient. Further, the computation can be made more efficient by image restoration. IEEE Transactions on Image Processing 2008;17(January (1)):53–69.
[10] Rubinstein R, Bruckstein AM, Elad M. Dictionaries for sparse representation modeling. Proceedings of IEEE 2010;98(June (6)):1045–57.
[11] Rudin L, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. Physica D 1992;60:259–68.
[12] Chan T, Esedoglu S, Park F, Yip A. Recent developments in total variation image restoration. In: Paragios N, Chen Y, Faugeras O, editors. Handbook of mathematical models in computer vision. New York: Springer-Verlag; 2005.
[13] Wang Y, Yang J, Yin W, Zhang Y. A new alternating minimization algorithm for total variation image reconstruction. SIAM Journal of Imaging Sciences 2008; 1(3):248–72.
[14] Oliveira J, Bioucas-Dias JM, Figueiredo MA. Adaptive total variation image deblurring: a majorization-minimization approach. Signal Processing 2009;89(9):1683–93.
[15] Michailovich Oleg V. An iterative shrinkage approach to total-variation image restoration. IEEE Transactions on Image Processing 2011;20(May (5)):1281–99.
[16] Daubechies I, Defrise M, De Mol C. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Communications on Pure and Applied Mathematics 2004;57(August (11)):1413–57.
[17] Beck A, Teboulle M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal of Imaging Sciences 2009;2:183–202.
[18] Bioucas-Dias J, Figueiredo M. A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration. IEEE Transactions on Image Processing 2007;16(December (12)):2992–3004.
[19] Mignotte M. A non-local regularization strategy for image deconvolution. Pattern Recognition Letters 2008;29(December (16)):2206–21.
[20] Guerrero-Colon JA, Mancera L, Portilla J. Image restoration using space-variant Gaussian scale mixtures in overcomplete pyramids. IEEE Transactions on Image Processing 2007;17(January (1)):27–41.
[21] Dabov K, Foi A, Katkovnik V, Egiazarian K. Image denoising by sparse 3D transform-domain collaborative filtering. IEEE Transactions on Image Processing 2007;16(August (8)):2080–95.
[22] Dabove Kostadin, Foi Alessandro, Katkovnik Vladimir, Egiazarian Karen. Image restoration by sparse 3D transform-domain collaborative filtering. In: Proceedings of SPIE electronic Image'08, vol. 6812, San Jose, 2008.
[23] Portilla J. Image restoration through l0 analysis-based sparse optimization in tight frames. In: Proceedings of 16th IEEE ICIP, Cairo, Egypt, 2009, pp. 3909–3912.
[24] Ni Jie, Turaga Pavan, Patel Vishal M, Chellappa Rama. Example-driven manifold priors for image deconvolution. IEEE Transactions on Image Processing 2011;20(November (11)):3086–96.
[25] Gastal E, Oliveira M. Domain transform for edge-aware image and video processing. ACM Transactions on Graphics 2011;30(4):655 Proceedings of SIGGRAPH 2011, Article 69.
[26] Barash D. A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation. IEEE Transactions on PAMI 2002;24:844–7.
[27] Buades A, Coll B, Morel JM. Nonlocal image and movie denoising. International Journal of Computer Vision 2008;76(2):123–39.
[28] Petschnigg G., et al. Digital photography with flash and no-flash image pairs. In: Proceedings of SIGGRAPH, 2004, p. 664–72.
[29] Ng M, Weiss P, Yuan X. Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods. SIAM Journal of Scientific Computing 2010;32(August (5)):2710–36.
[30] Weiss P, Blanc-Fraud L, Aubert G. Efficient schemes for total variation minimization under constraints in image processing. SIAM Journal of Scientific Computing 2009;31(3):2047–80.
[31] Yuan Lu, Sun Jian, Quan Long, Shum Heung-Yeung. Progressive inter-scale and intra-scale non-blind image deconvolution. ACM Transactions on Graphics 2007;27 Article 70.
[32] Hansen PC, Nagy JG, OLeary DP. Deblurring images: matrices, spectra, and filtering. Philadelphia, PA: SIAM; 2006.