REDUCING SPATIALLY VARYING OUT-OF-FOCUS BLUR
FROM NATURAL IMAGE

Faming Fang

Shanghai Key Laboratory of Multidimensional Information Processing
Department of Computer Science, East China Normal University
Shanghai, China

Fang Li

Department of Mathematics, East China Normal University
Shanghai, China

Tieyong Zeng

Department of Mathematics, Hong Kong Baptist University
Kowloon Tong, Hong Kong, China

(Communicated by Gabriele Steidl)

Abstract. In this paper, we focus on the challenging problem of removing the spatially varying out-of-focus blur from a single natural image. We first propose an effective method to estimate the blur map by the total variation refinement on Hölder coefficient, then discuss the properties of the corresponding kernel matrix. A tight-frame based energy functional, whose minimizer is related to the optimal defocus result, is thus built. For tackling functional more efficiently, we describe the numerical procedure based on an accelerated primal-dual scheme. To verify the effectiveness of our method, we compare it with some state-of-the-art schemes using both synthesized and natural images. Experimental results demonstrate that the proposed method performs better than the compared methods.

1. **Introduction.** In the past decades, with the increasing availability of portable digital imaging devices with high performance and low price, the digital images have attracted more and more attention in our life. Generally, digital images can be divided into many categories, such as natural images, remote sensing images, medical images, microscopic images, etc. All these images may suffer from a common distortion, which is blur. The factors that cause blur contain out-of-focus, shake, motion, etc [12]. Among them, the out-of-focus blur is the most common one. It is caused by the fact that there are some trade-offs between aperture-size, depth-of-field, and exposure-time while collecting images. It becomes the main drawback when using a large aperture to capture enough light and prevent the motion-blur [39]. How to measure and remove this kind of blur is a challenging task in many image applications such as computer vision and image processing.

From the pixel-level perspective, the out-of-focus blur is caused by a pixel capturing light from its surrounding points, and it is commonly characterized by a point
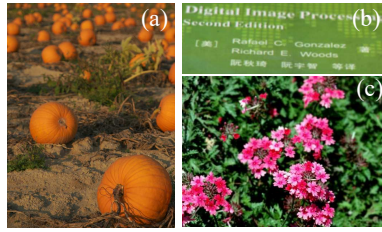
Figure 1. The spatially-varying blurring images.

spread function (PSF), or "kernel", i.e., the blurred image can be seen as a convolution of the clear one. Based on this idea, many outstanding methods have been developed for deblurring (or deconvolution) problems [8–10, 16, 22, 30, 33, 36, 44, 48]. Most of these methods assume that the blur is uniform on the whole image. Unfortunately, the blur typically varies over the image plane. Such spatially-varying blur is much more difficult to detect and analyze than the spatially-invariant one. Consequently, most methods need multiple images and fuse them to obtain a deblurred result.

The removal of spatially-varying blur using single image has also presented in recent years. The main challenges of this issue contain two aspects [15]: 1) It is a blind deconvolution problem since both clear image and blur kernel are unknown; 2) It is a very complex problem because the blur kernel differs form pixel to pixel (which cannot be modeled as a circulant matrix). Due to its complexity, many existing methods, such as [2, 15, 35, 41], first segmented the blurred image into several regions where each region has a uniform blur kernel, then handled the spatially-invariant deblurring in each region gradually to obtain a global deblurred result [32]. We collectively call these methods as segmentation-based deblurring methods. Segmentation-based deblurring methods are effective specific to the images whose blur kernel is regional different. However, since there are many natural images which can't be segmented reasonably (Figure 1 is a convincing example), the segmentation-based deblurring methods is limited in some practical applications.

The kernel of out-of-focus blur is usually approximated by a Gaussian function $\mathbf{g}(\boldsymbol{x}, \boldsymbol{\sigma}(\boldsymbol{x}))$ (or simply written as $\mathbf{g}(\boldsymbol{x})$), where the standard deviation $\boldsymbol{\sigma}(\boldsymbol{x})$ measures the blur level on the pixel $\boldsymbol{x}$ [52]. Therefore, intuitively, we can firstly estimate $\boldsymbol{\sigma}(\boldsymbol{x})$ based on some certain properties, and use $\boldsymbol{\sigma}(\boldsymbol{x})$ to generate kernel $\mathbf{g}(\boldsymbol{x}, \boldsymbol{\sigma}(\boldsymbol{x}))$ for every pixels, then remove blur using $\mathbf{g}(\boldsymbol{x}, \boldsymbol{\sigma}(\boldsymbol{x}))$ directly. Here, $\boldsymbol{\sigma}$, the same as the original blurred image in size, is generally termed as blur map. Based on the above ideas, in 2007, Bae and Durand extracted the blur scale of edges by calculating the first and second derivatives of the input image, and obtained a blur map by propagating the blur scale using interpolation method [1]. In 2009, Tai and Brown first built a coarse blur map by using their local contrast (LC) prior, and then refined it by the exploitation of Markov random field formulation [40]. In 2011, Zhuo and Sim re-blurred the input image using a Gaussian kernel, and calculated the blur amount at edges from the ratio between the gradients of input and re-blurred images. They then obtained the full blur map by propagating the blur amount to the entire image [52]. In 2013, Zhu et al. presented a method for coherent defocus blur estimating based on local probability estimation and coherent map labeling [51]. Moreover, Shen et al. generated a blur map by combining Tai and Zhuo's methods. They then dealt with deblurring based on the $L1$-$L2$ optimization

using different blur scales, and used the blur map to do scale selection to reconstruct an all-in-focus image [39]. This method is rather effective.

In the paper, we propose a new method for blind spatially-varying out-of-focus blur removal. Our contributions are as follows:

- Firstly, we analyze the property of the Hölder coefficient [20,29], and estimate a blur map based on it, then refine the blur map using a total variation (TV) based algorithm [38]. This can be regarded as an interesting complimentary for the previous approaches to estimate the blur map.
- Secondly, we discuss some properties and the bound of the corresponding kernel matrix which are important for our spatially variant blur removal task.
- Finally, we build an energy functional for deblurring based on the framelet system, and describe the numerical procedure for our energy based on an accelerated primal-dual (APD) scheme. To the best of our knowledge, this is the first study to introduce the APD algorithm into spatially-varying out-of-focus blur removal realm.

The rest of the paper is organized as follows. In Section 2, we estimate the blur map and discuss the properties of the kernel matrix. The proposed deblurring model and numerical scheme with its convergence property are addressed in Section 3. The various experiments on synthesized and real blurry images are then reported in Section 4 to illustrate the superior performance of our approach. Finally, some concluding remarks are given in Section 5.

2. Blur map estimation and kernel matrix construction.

As aforementioned, for spatially varying blur, let $\mathbf{b} : \Omega \to \mathbb{R}$ be a blurred image of size $M \times N$, which $\mathbf{b}$ can be seen as a convolution of its clear version $\mathbf{f}$ at each pixel $\boldsymbol{x}$. Formally, we have,

$$(1) \qquad \mathbf{b}(\boldsymbol{x}) = (\mathbf{g} \circledast \mathbf{f})(\boldsymbol{x}) + \mathbf{n}(\boldsymbol{x}) = \int_{\Omega} \mathbf{g}(\boldsymbol{y})\mathbf{f}(\boldsymbol{x} - \boldsymbol{y})d\boldsymbol{y} + \mathbf{n}(\boldsymbol{x}),$$

where $\circledast$ is the convolution operator, $\mathbf{n}$ is an additive noise. Since only $\mathbf{b}(\boldsymbol{x})$ is given, we need to estimate $\mathbf{g}(\boldsymbol{x})$ and $\mathbf{f}(\boldsymbol{x})$. That is, (1) is an under-constrained problem. So we should add some assumption to get reasonable result.

In what follows, we first estimate the blur map using the Hölder coefficient and a TV based algorithm. Then we discuss some properties of the corresponding blur kernel matrix.

2.1. Blur map estimation.

Generally, the blurred image is smoother than its clear version. That is, the more blurred the image, the smaller the value of the overall difference. Thus, we can use a variable, which describes the overall difference in $\omega_{\boldsymbol{x}}$, to measure the degree of blurring of $\boldsymbol{x}$, where $\omega_{\boldsymbol{x}}$ denotes a neighborhood of $\boldsymbol{x}$. Broadly speaking, the Hölder coefficient can roughly satisfy this requirement.

As a powerful regularity, the Hölder coefficient has been applied to many image processing tasks, such as image interpolation and denoising [28, 31, 34, 42, 43]. Formally, the Hölder coefficient of the image $\mathbf{b}$ in $\omega_{\boldsymbol{x}}$ is defined as [20, 29],

$$(2) \qquad [\mathbf{b}]_{\omega_{\boldsymbol{x}}}^{\beta} := \sup_{\substack{\boldsymbol{y}, \, \boldsymbol{z} \, \in \, \omega_{\boldsymbol{x}} \\ \boldsymbol{y} \, \neq \, \boldsymbol{z}}} \left\{ \frac{|b(\boldsymbol{y}) - b(\boldsymbol{z})|}{\|\boldsymbol{y} - \boldsymbol{z}\|^{\beta}} \right\},$$

where $\beta \geq 0$ is an adjustable parameter. When $\beta = 0$, (2) is actually equivalent to

$$(3) \qquad [\mathbf{b}]_{\omega_{\boldsymbol{x}}}^{0} = \max_{\boldsymbol{y} \in \omega_{\boldsymbol{x}}} \{b(\boldsymbol{y})\} - \min_{\boldsymbol{y} \in \omega_{\boldsymbol{x}}} \{b(\boldsymbol{y})\}.$$
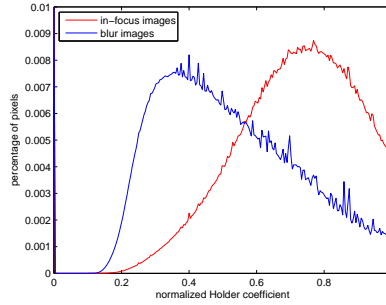
FIGURE 2.   The distributions of the NHC for in-focus (red) and blur images (blue).

Hence, $\frac{[\mathbf{b}]_{\omega_{\boldsymbol{x}}}^{\beta}}{[\mathbf{b}]_{\omega_{\boldsymbol{x}}}^{0}}$ can be regarded as the normalized Hölder coefficient (NHC) (remark that we set $|\omega_{\boldsymbol{x}}| = 7 \times 7$ and $\beta=2$ for NHC hereinafter, where $|\omega_{\boldsymbol{x}}|$ is the area of $\omega_{\boldsymbol{x}}$).

Figure 2 presents two histograms of the NHC for 1000 natural randomly selected in-focus images (red curve) and 1000 natural randomly selected blur images (blue curve), respectively. The x-axis is NHC scores and y-axis is percentage of pixels. It can be seen that, the in-focus images have larger NHC over the blur ones roughly, and the distribution peak of the in-focus image is around 0.75, which is much greater than that of the blur image (around 0.35).

On the other hand, numerous experiments we have done suggest that the NHC varies inversely to the level of the blur roughly. That is, the NHC can indicate the blur level, and the blur level is positively associated with the inverse NHC. For simplicity, we assume this positive correlation is a linear correlation. Then, the blur map $\boldsymbol{\sigma}(\boldsymbol{x})$ can be obtained by

$$(4) \qquad \tilde{\boldsymbol{\sigma}}(\boldsymbol{x}) = \frac{C}{\frac{[\mathbf{b}]_{\omega_{\boldsymbol{x}}}^{\beta}}{[\mathbf{b}]_{\omega_{\boldsymbol{x}}}^{0}}} = C \frac{[\mathbf{b}]_{\omega_{\boldsymbol{x}}}^{0}}{[\mathbf{b}]_{\omega_{\boldsymbol{x}}}^{\beta}},$$

where $\tilde{\boldsymbol{\sigma}}$ is a rough version of $\boldsymbol{\sigma}$, and C is a predefined constant. In our following experiments, we find that $C = \sqrt{\frac{|\omega_{\boldsymbol{x}}|}{2\pi}}$ is proper.

Figure 3(a) is the estimated blur map $\tilde{\boldsymbol{\sigma}}$ from an blurry image (Figure 1(a)). As we can see, it is reasonably good. However, the blur map $\tilde{\boldsymbol{\sigma}}$ is relatively rough due to the existence of noise and soft edges. So we should refine it. In practice, the blur map is somewhat smooth and piecewise constant. Thus, here we adopt a TV based algorithm to perform the blur map refining [38]. Formally, the blur map refining problem can be formulated as minimizing the following energy,

$$(5) \qquad E(\boldsymbol{\sigma}) = \nu\|\nabla\boldsymbol{\sigma}\| + \frac{1}{2}\|\Pi \odot (\boldsymbol{\sigma} - \tilde{\boldsymbol{\sigma}})\|^2,$$

where $\nu > 0$ is a balanced parameter, ∇ is gradient operator, $\odot$ is componentwise multiplication, and Π is an edge indicator which is defined as,

$$\Pi(\boldsymbol{x}) = \begin{cases} 1, & \|\nabla\mathbf{b}(\boldsymbol{x})\| > \xi \\ 0, & \text{otherwise} \end{cases}$$
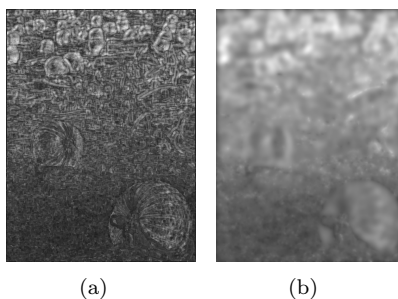
with positive parameter ξ.

(a)   (b)

FIGURE 3. The blur maps of Figure 1(a) (800×600 pixels): (a) the rough blur map $\tilde{\boldsymbol{\sigma}}$; (b) the refined blur map $\boldsymbol{\sigma}$.

Indeed, (5) can be efficiently solved by iterating the following equations based on Chambolle-Pock algorithm [13],

$$\begin{cases} \boldsymbol{p}^{n+\frac{1}{2}} = \boldsymbol{p}^n + \tau\lambda\nabla\boldsymbol{\sigma}^n, \\ \boldsymbol{p}^{n+1} = \dfrac{\boldsymbol{p}^{n+\frac{1}{2}}}{\max(1,|\boldsymbol{p}^{n+\frac{1}{2}}|)}, \\ \boldsymbol{\sigma}^{n+1} = \dfrac{\Pi\odot\tilde{\boldsymbol{\sigma}}+\boldsymbol{\sigma}^n}{\varsigma\Pi+1}. \end{cases}$$

where τ and ς are two positive parameters, $\boldsymbol{p}$ is an auxiliary variable which is initialized to the identity matrix $\boldsymbol{I}$.

Figure 3(b) shows the refining result using Figure 3(a) as the data fidelity term. We can see that the refined blur map $\boldsymbol{\sigma}$ contains less noise and is more reasonable.

2.2. **Kernel matrix analysis.** After calculating the blur map $\boldsymbol{\sigma}$, we can obtain the kernel **g** by the definition of Gaussian function, and then solve **f** through (1). However, since $\boldsymbol{\sigma}$ is different in each pixel, we should solve (1) pixel by pixel. Consequently, the time complexity is very high for obtaining the whole deblurred image. Fortunately, (1) can be rewritten as,

$$\tag{6} \boldsymbol{b} = \mathbf{A}\boldsymbol{f} + \boldsymbol{n},$$

where $\boldsymbol{b}, \boldsymbol{f}$ and $\boldsymbol{n}$ are the vectorized versions of $\mathbf{b}, \mathbf{f}$ and $\mathbf{n}$ after column concatenation, respectively. The kernel matrix $\mathbf{A}$ (size $MN \times MN$) is the matrix form of the kernel **g**.

The constructing of the kernel matrix $\mathbf{A}$ is rather straightforward [25,46]. One can easily generate it following the step of the spatially-invariant kernel matrix construction. The only difference is that, all rows of the kernel matrix of the spatially-invariant kernel share the same variance, while the matrix $\mathbf{A}$ may possess different variances row by row. Besides, we have the following two properties:

Proposition 2.1. *Let λ_m be the largest eigenvalue of $\mathbf{A}$, then the $\lambda_m \leq 1$ holds.*

Proof. See Appendix. $\square$

Proposition 2.2. *Let $\boldsymbol{x} = (x_1, x_2)$, $\boldsymbol{y} = (y_1, y_2)$, and let $\mathrm{diag}(\boldsymbol{\sigma})$ be a diagonal matrix expanded by $\boldsymbol{\sigma}$, and $\mathbf{A}^i$ be the matrix form of 1-D Gaussian kernel g^i toward i-axis, where*

$$g^i(\boldsymbol{x})(y_i) = \frac{1}{\sqrt{2\pi}\boldsymbol{\sigma}(\boldsymbol{x})} \exp\big(-\frac{|y_i - x_i|^2}{2\boldsymbol{\sigma}^2(\boldsymbol{x})}\big), \ i = 1, 2.$$

Then, we have

$$(7) \qquad \mathbf{A} = \sqrt{2\pi} \mathrm{diag}(\boldsymbol{\sigma}) \mathbf{A}^1 \mathbf{A}^2.$$

Proof. See Appendix. □

Given the induced norm of $\mathbf{A}$ defined as,

$$(8) \qquad \|\mathbf{A}\| = \max\{\|\mathbf{A}\boldsymbol{\varphi}\| : \ \|\boldsymbol{\varphi}\| \le 1\}.$$

The bound of $\|A\|$ can be guaranteed by the following property:

Proposition 2.3. *Let $\boldsymbol{\sigma}_m$ be the largest value in $|\boldsymbol{\sigma}|$, the norm of kernel matrix $\mathbf{A}$ is bounded by $\|\mathbf{A}\| \le L$, where $L := \sqrt{2\pi}(\frac{\pi^2}{2} - 2)\boldsymbol{\sigma}_m$.*

Proof. See Appendix. □

2.2.1. *Sparse kernel matrix analysis.* Theoretically, the $\mathbf{g}(\boldsymbol{x}, \boldsymbol{\sigma}(\boldsymbol{x}))$ will be non-zero at every point of the image, meaning that the $\mathcal{N}_{\boldsymbol{x}}$ should be the same size with the entire image. In practice, when calculating $\mathbf{g}(\boldsymbol{x}, \boldsymbol{\sigma}(\boldsymbol{x}))$, pixels at a distance that more than $3\boldsymbol{\sigma}(\boldsymbol{x})$ are small enough to be considered effectively zero [24]. Thus the contributions from pixels outside that range can be ignored. Therefore, we can set the radius of $\mathcal{N}_{\boldsymbol{x}}$ as $r_{\boldsymbol{x}} = 3\boldsymbol{\sigma}(\boldsymbol{x})$ in our convolution process.

In this setting, $\mathbf{A}$ is obviously a sparse matrix, and each row/column has at most $(2r + 1)^2$ non-zero elements, here $r = \max\{r_{\boldsymbol{x}}\}$. Then the induced norm of this sparse kernel has the following simple property,

Proposition 2.4. *The norm of sparse kernel matrix A is bounded by $\|A\| < (2r + 1)^2$.*

Proof. See Appendix. □

The time complexity of generating the sparse matrix $\mathbf{A}$ is $O(r^2 MN)$. Since $r \ll MN$, the generating of kernel matrix will be very fast. In practice, only about 0.75 seconds is needed in Matlab for an image sized 256×256.

3. **Deblurring formulation based on framelet system.** In this section, before presenting the deblurring formulation for spatially varying out-of-focus blur, we first give a brief introduction to the framelet system. For simplicity, we only show the framelets in the univariate setting, and the framelets in the bivariate setting can be obtained by tensor product of the univariate one. Those who are interested in the framelets can refer to [7, 8, 11, 14, 21] for more details.

3.1. **Framelets and image representation.** Firstly, we give the definition of the tight frame. A countable function subset $X \subset L^2(\mathbb{R})$ is called a tight frame of $L^2(\mathbb{R})$, if

$$(9) \qquad f = \sum_{h \in X} \langle f, h \rangle h, \forall f \in L^2(\mathbb{R}),$$

or equivalently,

$$(10) \qquad \|f\|^2 = \sum_{h \in X} |\langle f, h \rangle|^2, \forall f \in L^2(\mathbb{R}),$$

where $\langle \cdot, \cdot \rangle$ and $\| \cdot \| = \langle \cdot, \cdot \rangle^{\frac{1}{2}}$ are the inner product and the norm of $L^2(\mathbb{R})$, respectively.

An arbitrary orthogonal bases in $L^2(\mathbb{R})$ is a tight frame since it meets both the (9) and (10). In other words, tight frames are the generalization of the orthogonal basis, it relaxes the requirements of the orthogonality and linear independence to bring in redundancy which has been verified useful in many applications such as deblurring [8].

Given a finite set $\Phi := \{\phi^1, .., \phi^r\} \subset L^2(\mathbb{R})$, a wavelet system $X(\Phi)$ is defined as the collection of dilations and shifts of Φ, i.e.,

$$X(\Phi) := \{2^{k/2}\phi^j(2^k x - l) : 1 \le j \le r; k, l \in \mathbb{Z}\}.$$

When $X(\Phi)$ forms a tight frame, it is termed as a wavelet tight frame, and ϕ^j is termed as a (tight) framelet.

To construct compactly supported wavelet tight frames $X(\Phi)$, one generally first obtains a compactly supported refinable function $\psi \in L^2(\mathbb{R})$ with a refinement mask (low-pass filter) g_0 such that

$$\psi(x) = \sum_l g_0(l)\psi(2x - l).$$

Then for given ψ, the construction of a wavelet tight frame is actually to find an appropriate set of framelets $\Phi := \{\phi^1, .., \phi^r\} \subset L^2(\mathbb{R})$, which is defined as,

$$\phi^j = \sum_l g_j(l)\psi(2x - l), j = 1, .., r,$$

where g_j is a high-pass filter. The unitary extension principle (UEP) in [37] asserts that the system $X(\Phi)$ forms a tight frame in $L^2(\mathbb{R})$ provided that the filters $g_0, g_1, .., g_r$ satisfy,

$$\zeta_{g_0}(\omega)\overline{\zeta_{g_0}(\omega + \gamma\pi)} + \sum_{j=1}^r \zeta_{g_j}(\omega)\overline{\zeta_{g_j}(\omega + \gamma\pi)} = \delta(\gamma), \gamma = 0, 1,$$

for almost all $\omega \in \mathbb{R}$. Here $\zeta_g(\omega) = \sum_l g(l)e^{il\omega}$ and $\delta(\gamma)$ is a delta function. Thus, the construction of framelets Φ essentially is to design the filters $g_1, .., g_r$ for a given g_0 such that UEP holds.

As a simple application of the UEP, a piecewise linear B-splines can be used as the refinable function ψ [4, 37]. The corresponding filters are

$$g_0 = [\frac{1}{4}, \frac{1}{2}, \frac{1}{4}], g_1 = [-\frac{1}{4}, \frac{1}{2}, -\frac{1}{4}], g_2 = [\frac{\sqrt{2}}{4}, 0, -\frac{\sqrt{2}}{4}].$$

In the numerical scheme, the framelet transform can be represented by a matrix $\mathcal{W}$. The processes of generating such matrices have been detailed in [7, 11]. We omit them here for readability.

With matrix $\mathcal{W}$, the frame transformation (decomposition) process can be easily described. Let $\boldsymbol{f}$ be a vector, the frame coefficient vector $\boldsymbol{u}$ is given by

$$\boldsymbol{u} = \mathcal{W}\boldsymbol{f}.$$

Besides, the frame reconstruction process can be expressed as,

$$\boldsymbol{f} = \mathcal{W}^T\boldsymbol{u},$$

where $\mathcal{W}^T$ is the inverse framelet transform. More importantly, the tight frame $\mathcal{W}$ satisfies $\boldsymbol{f} = \mathcal{W}^T\mathcal{W}\boldsymbol{f}$. That is, $\mathcal{W}^T\mathcal{W} = \mathcal{I}$. Generally, $\mathcal{W}\mathcal{W}^T \ne \mathcal{I}$, unless in the orthogonal case.

Finally, it should be noted that we use the same decomposition and reconstruction algorithms as [8, 18] hereafter. And we work in the bivariate case for our

deblurring task. This corresponding transform matrix, still denote $\mathcal{W}$, can be readily obtained by using the Kronecker product of the matrix corresponding to the univariate frame transform.

3.2. **The proposed energy model.** To restore the clear image $\boldsymbol{f}$, one needs to solve the following minimization problem,

$$\min_{\boldsymbol{f}} R(\boldsymbol{f}) + F(\boldsymbol{f}),$$

where $R(\boldsymbol{f})$ is a regularization term, and $F(\boldsymbol{f})$ is a data fidelity term.

Generally, the regularization term is derived according to some prior assumptions (or properties) of the underlying solutions. One of the popular assumptions is the sparsity in some domain or some bases. Since the images usually have sparse representations or approximations in the framelet domain, and this assumption has proven to be efficient in many image deblurring works [8, 9], we use this prior and take $R(\boldsymbol{f}) = \|\mathcal{W}\boldsymbol{f}\|_1$, where $\mathcal{W}$ is the framelet decomposition operator.

The fidelity term $F(\boldsymbol{f})$ is determined by the type of noise distribution. For example, the ℓ_1- and ℓ_2-norm functions are usually used for the impulse and Gaussian noises, respectively. However, it is difficult to determine the type of noise in real applications. What's more, the noise in a real image seldom follows a single specific distribution. Thus, it is necessary to build a function that can remove the unknown type as well as the mixture type of noises. For this propose, we will use $F(\boldsymbol{f}) = \|\mathbf{A}\boldsymbol{f} - \boldsymbol{b}\|_p^p, \ p \in [1, 2]$. Here, the ℓ_p-norm function has been proved effective for unknown and mixture type of noises, and been successfully used in image processing realms [27].

Taking together, our total variational model is rather simple,

$$(11) \qquad \min_{\boldsymbol{f}} \mu\|\mathcal{W}\boldsymbol{f}\|_1 + \frac{1}{p}\|\mathbf{A}\boldsymbol{f} - \boldsymbol{b}\|_p^p, \ p \in [1, 2]$$

where μ is a positive parameters to balance the regularization and the fidelity terms.

Since $\mathcal{W}$ is a linear operator, it is obviously that the above energy is strictly convex. Therefore, the minimization of (11) admits a unique solution.

3.3. **Numerical algorithm.** In what follows, we will detail the numerical algorithm of our blind out-of-focus deblurring energy (11). Clearly, (11) is a ℓ_1-norm-based minimization problem. There are many efficient methods have been presented to solve this issue, such as augmented Lagrangian method (ALM) [47], alternating direction method of multipliers (ADMM) [5], and split Bregman iteration (SB) [23]. In this paper, we adopt a primal-dual algorithm for (11).

The saddle-point formulation of (11) can be derived as,

$$(12) \qquad \min_{\boldsymbol{f}} \max_{\boldsymbol{d}} \mu\langle\mathcal{W}\boldsymbol{f}, \boldsymbol{d}\rangle + \frac{1}{p}\|\mathbf{A}\boldsymbol{f} - \boldsymbol{b}\|_p^p - \delta_{\mathbf{D}}(\boldsymbol{d}),$$

where $\boldsymbol{d}$ is the dual variable. The convex set $\mathbf{D}$ is given by $\mathbf{D} = \{\boldsymbol{d}: \ \|\boldsymbol{d}\|_\infty \leq 1\}$, and the function $\delta_{\mathbf{D}}$ is the indicator function of the set $\mathbf{D}$ which is defined as

$$\delta_{\mathbf{D}}(\boldsymbol{d}) = \begin{cases} 0 & \boldsymbol{d} \in \mathbf{D} \\ +\infty & \boldsymbol{d} \notin \mathbf{D} \end{cases}$$

Since $\mathcal{W}$ is a linear operator, the induced norm of $\mathcal{W}$ has the following property.

Proposition 3.1. *The induced norm of $\mathcal{W}$ admits $L_{\mathcal{W}} := \|\mathcal{W}\| = 1$.*

Proof. Since $\mathcal{W}^T\mathcal{W} = \mathbf{I}$ and $\|\mathcal{W}^T\| = \|\mathcal{W}\|$ (see [8]), we can obtain $\|\mathcal{W}\| = 1$ directly. $\qquad\square$

Since both $\frac{1}{p}\|\mathbf{A}\boldsymbol{f} - \boldsymbol{b}\|_p^p$ and $\delta_{\mathbf{D}}(\boldsymbol{d})$ are proper, convex and lower semi-continuous, (12) is exactly the same problem in [13]. Therefore, the primal-dual algorithm is proper for solving our problem (11).

The regular primal-dual algorithm of (12) is as follows,

$$
(13) \quad \begin{cases} \boldsymbol{d}^{i+1} = \arg\max_{\boldsymbol{d}} \mu\langle \mathcal{W}\boldsymbol{f}^{\bar{i}+1}, \boldsymbol{d}\rangle - \delta_{\mathbf{D}}(\boldsymbol{d}) - \frac{1}{2\tau^i}\|\boldsymbol{d} - \boldsymbol{d}^i\|^2, \\[2mm]
(14) \quad \boldsymbol{f}^{i+1} = \arg\min_{\boldsymbol{f}} \mu\langle \mathcal{W}\boldsymbol{f}, \boldsymbol{d}^{i+1}\rangle + \frac{1}{p}\|\mathbf{A}\boldsymbol{f} - \boldsymbol{b}\|_p^p + \frac{1}{2\varsigma^i}\|\boldsymbol{f} - \boldsymbol{f}^i\|^2, \\[2mm]
(15) \quad \overline{\boldsymbol{f}^{i+1}} = \boldsymbol{f}^{i+1} + \theta^i(\boldsymbol{f}^{i+1} - \boldsymbol{f}^i), \end{cases}
$$

where τ^i, ς^j and θ^i are positive parameters.

From (13), the solution of $\boldsymbol{d}$ can be readily given by,

$$
(16) \quad \boldsymbol{d}^{i+1} = \frac{\mu\tau^i \mathcal{W}\bar{\boldsymbol{f}}^i + \boldsymbol{d}^i}{\max(1, |\mu\tau^i \mathcal{W}\bar{\boldsymbol{f}}^i + \boldsymbol{d}^i|)}.
$$

Due to the existence of the ℓ_p-norm, (14) is not easy to solve. In this case, we can replace $\frac{1}{p}\|\mathbf{A}\boldsymbol{f} - \boldsymbol{b}\|_p^p$ by its linear approximation using the first order Taylor series expansion at $\boldsymbol{f}^i$,

$$
\frac{1}{p}\|\mathbf{A}\boldsymbol{f}^i - \boldsymbol{b}\|_p^p + \langle \mathbf{A}^T(\mathbf{A}\boldsymbol{f}^i - \boldsymbol{b})^{p-1}, \boldsymbol{f} - \boldsymbol{f}^i\rangle.
$$

Then (14) becomes

$$
\boldsymbol{f}^{i+1} = \arg\min_{\boldsymbol{f}} \mu\langle \mathcal{W}\boldsymbol{f}, \boldsymbol{d}^{i+1}\rangle + \langle \mathbf{A}^T(\mathbf{A}\boldsymbol{f}^i - \boldsymbol{b})^{p-1}, \boldsymbol{f}\rangle + \frac{1}{2\varsigma^i}\|\boldsymbol{f} - \boldsymbol{f}^i\|^2.
$$

Thus, $\boldsymbol{f}$ can be given by the following iterative algorithm,

$$
(17) \quad \boldsymbol{f}^{i+1} = \boldsymbol{f}^i - \varsigma^i\left(\mu\mathcal{W}^T\boldsymbol{d}^{i+1} + \mathbf{A}^T(\mathbf{A}\boldsymbol{f}^i - \boldsymbol{b})^{p-1}\right).
$$

Taking all above into account, the overall numerical procedure of the proposed method can be summarized as

$$
(18) \quad \begin{cases} \boldsymbol{d}^{i+1} = \frac{\mu\tau^i \mathcal{W}\bar{\boldsymbol{f}}^i + \boldsymbol{d}^i}{\max(1, |\mu\tau^i \mathcal{W}\bar{\boldsymbol{f}}^i + \boldsymbol{d}^i|)}, \\[2mm]
\boldsymbol{f}^{i+1} = \boldsymbol{f}^i - \varsigma^i\left(\mu\mathcal{W}^T\boldsymbol{d}^{i+1} + \mathbf{A}^T(\mathbf{A}\boldsymbol{f}^i - \boldsymbol{b})^{p-1}\right), \\[2mm]
\overline{\boldsymbol{f}^{i+1}} = \boldsymbol{f}^{i+1} + \theta^i(\boldsymbol{f}^{i+1} - \boldsymbol{f}^i), \end{cases}
$$

It is clear that (18) can be considered as a linearized version of primal-dual algorithm. Here, we further use an accelerated primal-dual (APD) scheme to accelerate (18). The detail APD is summarized in Algorithm 1. As discussed in [17], the iteration cost for Algorithm 1 is about the same as that for the (18). However, by specifying a different selection of β^i, the convergence of Algorithm 1 can be significantly improved.

For the convergence assessment, we propose to use the relative error as our stopping criteria. Classically, the relative error for the proposed algorithm is defined as,

$$
(19) \quad \mathcal{R}(\boldsymbol{f}^i) := \frac{\|\boldsymbol{f}^i - \boldsymbol{f}^{i-1}\|}{\|\boldsymbol{f}^{i-1}\|}.
$$

Given a prescribed positive small parameter ρ, for instance 10^{-4}, when the relative error is less than ρ, our algorithm can be considered to reach a steady state.

The convergence of the Algorithm 1 can be guaranteed by the following Theorem:

Algorithm 1: An accelerated primal-dual scheme of the proposed model (11).

- Initialize: $\boldsymbol{f}^0 = \boldsymbol{b}, \bar{\boldsymbol{f}}^0 = \boldsymbol{f}^0_{ag} = \boldsymbol{f}^0, \boldsymbol{d}^0 = \boldsymbol{I}, d^0_{ag} = \boldsymbol{d}^0$,
- Repeat until $\mathcal{R}(\boldsymbol{f}^i) < \rho$:

$$
\begin{cases}
\boldsymbol{f}^i_{md} & = (1 - \frac{1}{\beta^i})\boldsymbol{f}^i_{ag} + \frac{1}{\beta^i}\boldsymbol{f}^i, \\
\boldsymbol{d}^{i+1} & = \frac{\mu\tau^i\mathcal{W}\bar{\boldsymbol{f}}^i + \boldsymbol{d}^i}{\max(1, |\mu\tau^i\mathcal{W}\boldsymbol{f}^i + \boldsymbol{d}^i|)}, \\
\boldsymbol{f}^{i+1} & = \boldsymbol{f}^i - \varsigma^i \left(\mu\mathcal{W}^T\boldsymbol{d}^{i+1} + \mathbf{A}^T(\mathbf{A}\boldsymbol{f}^i - \boldsymbol{b})^{p-1}\right), \\
\boldsymbol{f}^{i+1}_{ag} & = (1 - \frac{1}{\beta^i})\boldsymbol{f}^i_{ag} + \frac{1}{\beta^i}\boldsymbol{f}_{i+1}, \\
\boldsymbol{d}^{i+1}_{ag} & = (1 - \frac{1}{\beta^i})\boldsymbol{d}^i_{ag} + \frac{1}{\beta^i}\boldsymbol{d}_{i+1}, \\
\overline{\boldsymbol{f}^{i+1}} & = \boldsymbol{f}^{i+1} + \theta^i(\boldsymbol{f}^{i+1} - \boldsymbol{f}^i).
\end{cases}
$$

Theorem 3.1. *Choose the parameters $\beta_i, \tau_i, \varsigma_i$ and θ_i such that for all $i \geq 1$,*

(20)
$$
\begin{cases}
\beta^1 = 1, \ \beta^{i+1} = 1 + \beta^i\theta^{i+1} \\
0 \leq \theta^i \leq \min\{\frac{\varsigma_{i-1}}{\varsigma_i}, \frac{\tau_{i-1}}{\tau_i}\} \\
\frac{1}{\varsigma_i} - \frac{1}{\beta_i} - L^2_{\mathcal{W}}\tau_i \geq 0.
\end{cases}
$$

let $(\boldsymbol{f}^i, \boldsymbol{d}^i)$ be the sequence derived form Algorithm 1, then the following conclusion hold:

(a) *there exists a saddle-point $(\boldsymbol{f}^\star, \boldsymbol{d}^\star)$ such that*

$$\boldsymbol{f}^i \to \boldsymbol{f}^\star \text{ and } \boldsymbol{d}^i \to \boldsymbol{d}^\star.$$

(b) *the limitation $\boldsymbol{f}^\star$ is exactly the solution of (11).*

Proof. (a) can be immediately proved following the Theorem 2.1 in [17]. Moreover, according to Proposition 3.1 in [19], (b) is equivalent to (a). Thus Theorem 3.1 is proved. □

4. **Experimental results and analysis.** In this section, in order to examine the effectiveness of the proposed method, we present and analyze the experimental results on some synthesized and real blurry images. Note that all the following experiments are implemented in Matlab R2013a on an Intel(R) 3.33 GHz PC with 12 GB RAM. In what follows, we set the parameters of energy (5) and (11) as $\xi = 0.01$, and $\nu = 8$; set the stopping criteria of Algorithm 1 as $\rho = 10^{-4}$; and set the parameters of the APD algorithm according to (20), i.e., $\beta_i = \frac{i+1}{2}, \tau_i = 1, \varsigma_i = \frac{i}{2+i}$ and $\theta_i = \frac{i-1}{i}$.

4.1. **Effect of blur map.** In first part of the experiments, we will verify the effect of our blur map generation algorithm using some synthesised blurry images. For comparison, the results of two other outstanding blur map generation methods, i.e., Zhuo and Sim's method (ZS) [52] and Shen et al.'s method (SHP) [39], will also presented.

In Figure 4, we synthesise an image using four squares where each one processes a constant intensity (see (a)), and then blur 191th-210th and 271th-330th columns of the image by 1 and 4 blur levels respectively. The blurry image are shown in (b). Obviously, there are three edges in (b): the left edge is sharp, the middle edge has a little blur, while the right edge is much blurry.

The blur maps of Figure 4(b) generated by ZS [52], SHP [39] and proposed methods are shown in (c)-(e), respectively. From these results, it can be seen that the result of ZS is gradient-related (see Figure 4(c) and green line of (g)), which
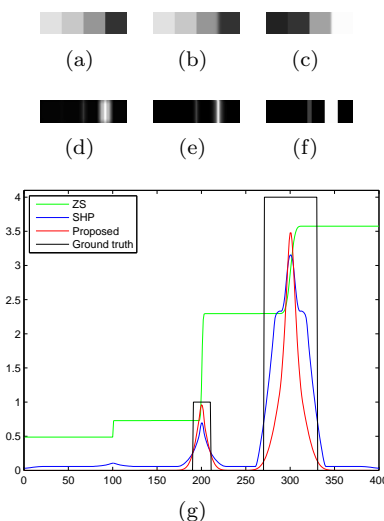
FIGURE 4. (a) a sharp image (size:100× 400); (b) a blur version of (a) (the blur levels are 1 and 4 for 171th-230th and 271th-330th columns, respectively); (c)-(e) the blur maps genrated by ZS [52], SHP [39] and proposed methods; (f) the ground truth of blur map; (g) corresponding profiles of 1-st line for (c)-(f).

TABLE 1. The MAE value of each blur map generated by ZS [52], SHP [39] and Proposed methods.

	ZS [52]	SHP [39]	Proposed
Figure 4(b)	1.3799	0.6495	**0.5308**
Figure 5(e)	0.4043	0.3753	**0.3751**
Figure 5(f)	0.3959	0.3950	**0.3684**
Figure 5(g)	0.4811	0.4455	**0.3064**
Figure 5(h)	0.4823	0.4756	**0.3418**

means that it will change according to the varying of image structures, rather than the real blur. SHP's result seems acceptable (see Figure 4(d) and blur line of (g)). However, it has a jump around the 100-th column, where actually has no blur. On the contrary, the proposed result is more approximate to the ground truth visually (see Figure 4(e) and red line of (g)).

To validate those observations, a well-known quantitative metric, termed mean absolute error (MAE), is adopted to evaluate the blur maps. Note that for better illustration, we also measure the estimated blur maps generated from other four synthesized blurry images. The blurry images are shown in Figure 5(e)-(h), where the first/last two images are the blurred versions of the original natural image (a)/(b) using (c) and (d) as blur maps, respectively. The quantitative results are given in Table 1. It can be observed from this table that the proposed method has the best MAE values. Therefore, our blur map generation method is acceptable.
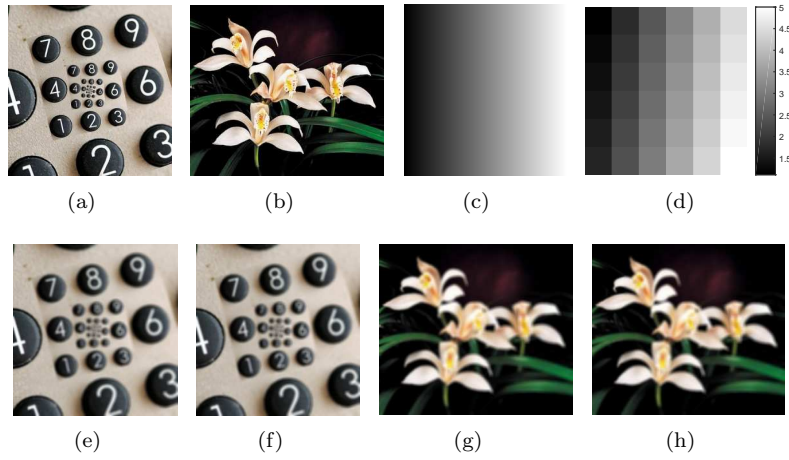
FIGURE 5. (a) and (b) are original images; (c) and (d) are two blur maps (range: 1-5); (e): the blurred version of (a) using (c) as blur map; (f): the blurred version of (a) using (d) as blur map; (g): the blurred version of (b) using (c) as blur map; (h): the blurred version of (b) using (d) as blur map.
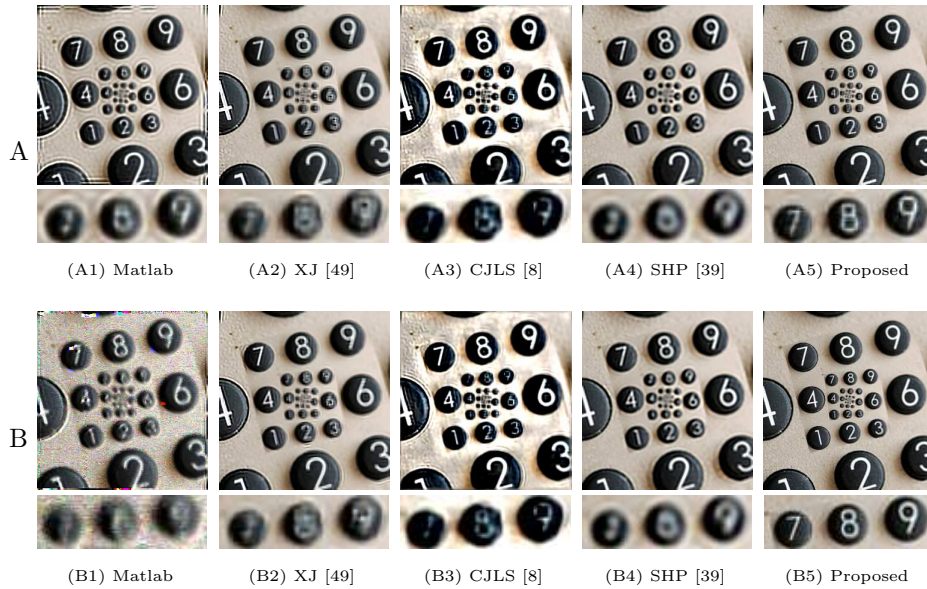

FIGURE 6. Synthesised experiments: (A1-A5)/(B1-B5) are the deblurred results (with zoomed regions) of Figure 5(e)/(f) by Matlab, XJ [49], CJLS [8], SHP [39], and proposed methods, respectively.

4.2. **Simulated images.** In second part of the experiments, we use the aforementioned four synthesized images (Figure 5(e)-(h)) to test the behavior of our method, and adopt peak signal to noise ratio (PSNR), structural similarity (SSIM) index [45], and sharpness index (SI) [3] to measure the quality of deblurred images. Notice that
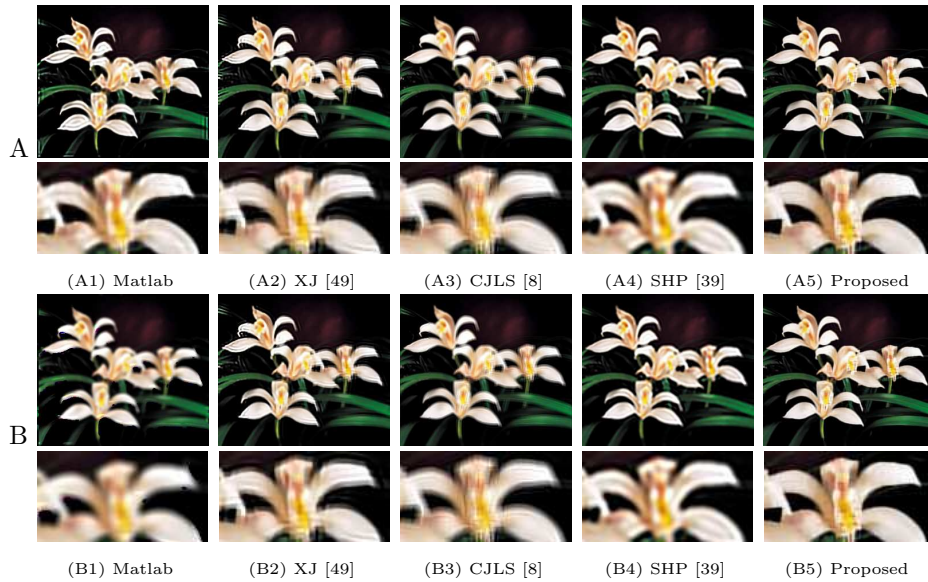
A

(A1) Matlab   (A2) XJ [49]   (A3) CJLS [8]   (A4) SHP [39]   (A5) Proposed

B

(B1) Matlab   (B2) XJ [49]   (B3) CJLS [8]   (B4) SHP [39]   (B5) Proposed

FIGURE 7. Synthesised experiments: (A1-A5)/(B1-B5) are the deblurred results (with zoomed regions) of Figure 5(g)/(h) by Matlab, XJ [49], CJLS [8], SHP [39], and proposed methods, respectively.

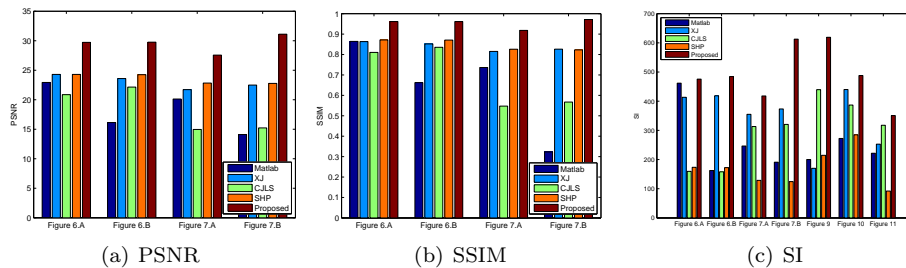(a) PSNR   (b) SSIM   (c) SI

FIGURE 8. PSNR, SSIM, and SI values corresponding to each figure.

to calculate PSNR and SSIM, both the deblur result and original reference image should be given, while SI can be computed with only the single deblur image. For each metric, higher value represents a better deblurring performance.

Besides, we compare our method against the other four state-of-the-art methods, i.e., the 'deconvblind' function in Matlab (hereinafter referred to as "Matlab"), Xu and Jia's method (XJ) [49], Cai et al.'s method (CJLS) [8] and SHP method [39], in which the first three methods are the spatially invariant blind deblurring methods, while the last one is the spatially variant blind deblurring method. We note that all parameters of the above algorithms are set according to the authors' recommendations, and we set the initial PSF of the 'deconvblind' as gaussian distribution with size of 4×4.

The corresponding deblurred results are shown in Figures 6 and 7, in which the first-fifth columns are respectively the results of Matlab, XJ, CJLS, SHP and
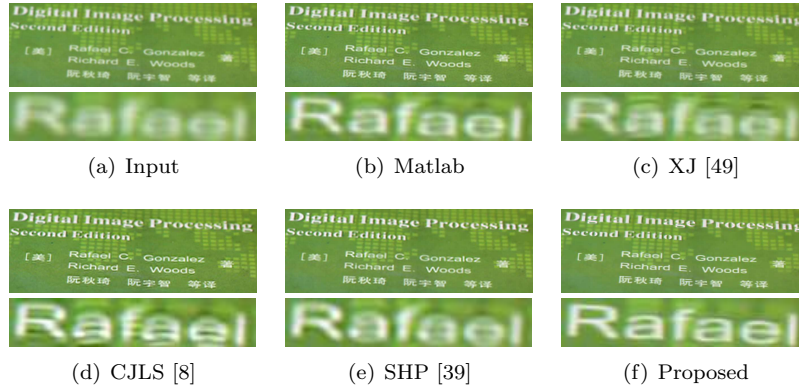
<table>
<tr><td>(a) Input</td><td>(b) Matlab</td><td>(c) XJ [49]</td></tr>
<tr><td>(d) CJLS [8]</td><td>(e) SHP [39]</td><td>(f) Proposed</td></tr>
</table>

FIGURE 9. Original image and its deblurred results with zoomed regions. (a) The original image (RGB,296×877 pixels); (b)-(f) the results of Matlab, XJ [49], CJLS [8], SHP [39], and the proposed methods.

the proposed methods. From these figures, it is clear that the proposed method is empirically approaches the ground truth of the original images. On the contrast, the results of Matlab, XJ and CJLS are either over-deblurred on the regions with little blur, or under-deblurred in the high blurry regions. Such results are not surprising as the spatially invariant methods just assume an invariant kernel overall the image. As an spatially variant method, SHP first dealt with spatially invariant non-blind deblurring based on the $L1$-$L2$ optimization using different blur scale, then used the blur map to do scale selection to reconstruct the deblurred images. These results (i.e., Figures 6(A4) and (B4), Figures 7(A4) and (B4)) are thus better than the the spatially invariant methods. However, since this method is not precisely spatially variant, it is reasonably worse than our proposed method.

As an example, we can see that Figures 6(A1)-(A2) are over-deblurred on the left-side, and are blurry on the right-side. Figure 6(A3) is over-deblurred overall the image, Figure 6(A4) is clearer than (A1)-(A3), while there are some artifacts on the right buttons. Figure 6(A5) is the best one among (A1)-(A5) since it has fewer artifacts and clearer than (A1)-(A4). For instance, as shown in the zooming area, the numbers '7', '8' and '9' in Figure 6(A5) are clear, while are ambiguous in (A1)-(A4).

The above analyses and observations are also confirmed by the image quality indicators PSNR, SSIM, and SI. As shown in Figure 8, our results outperform the others with respect to all metrics. E.g. the PSNR of our result (A5) is at least 5 larger than that of the other methods. As other three kind of results are also present the similar results. Those show that our method performed well over these images, and the results are of good quality with few artifacts.

4.3. **Real images.** In the third part, we will present the comparison studies on some real blurred images. Since the reference image is not available, here we only use the aforementioned SI as the quality metric. We also compare our method with aforementioned four methods, i.e., the 'deconvblind' function in Matlab, XJ method [49], CJLS method [8], and SHP method [39]. The comparison results are shown in Figures 9-11, and the corresponding quality results are given in Figure 8(c).
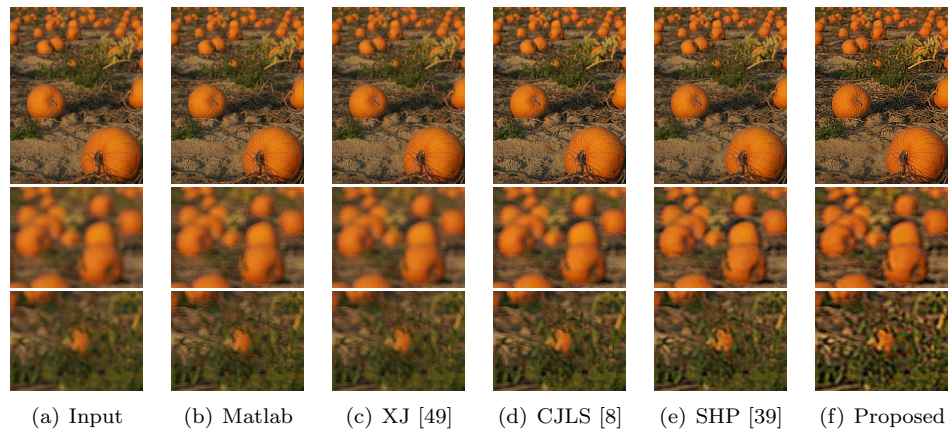
(a) Input   (b) Matlab   (c) XJ [49]   (d) CJLS [8]   (e) SHP [39]   (f) Proposed

FIGURE 10. Original image and its deblurred results with zoomed regions. (a) The original image (RGB,800×600 pixels); (b)-(f) the results of Matlab, XJ [49], CJLS [8], SHP [39], and the proposed methods.

(a) Input   (b) Matlab   (c) XJ [49]
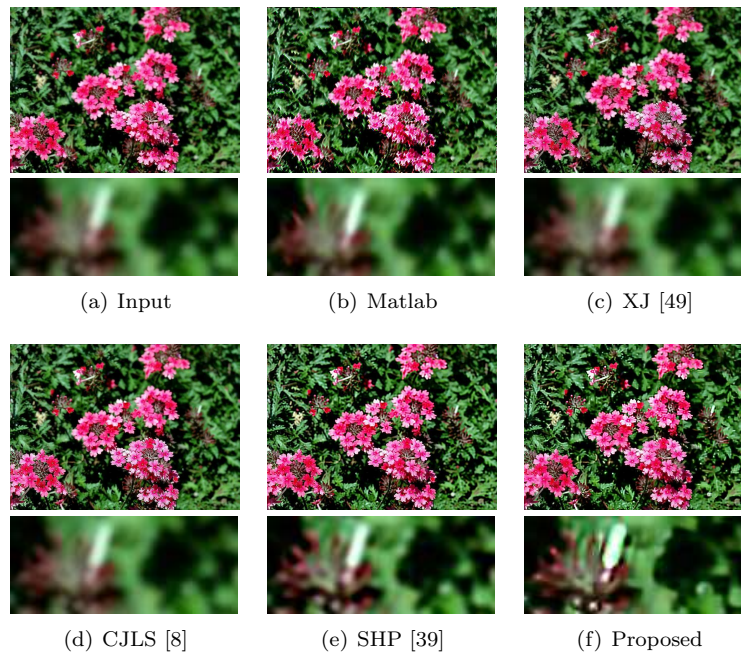
(d) CJLS [8]   (e) SHP [39]   (f) Proposed

FIGURE 11. Original image and its deblurred results with zoomed regions. (a) The original image (RGB,550×760 pixels); (b)-(f) the results of Matlab, XJ [49], CJLS [8], SHP [39], and the proposed methods.

Table 2. Time comparison with other methods (second).

	Figure 5(e)	Figure 5(f)	Figure 5(g)	Figure 5(h)	Figure 9(a)	Figure 10(a)	Figure 11(a)
image size	300×286	300×286	265×300	265×300	200×300	800×600	534×800
Matlab	1.18	1.22	1.00	1.17	2.03	6.70	6.00
XJ [49](C)	6.45	6.55	6.36	6.35	11.43	18.01	16.95
CJLS [8]	86.59	83.74	88.27	89.84	194.95	780.30	693.77
SHP [39]	70.21	69.40	72.26	71.55	126.37	187.01	168.68
Proposed	**38.33**	**37.60**	**34.98**	**36.46**	**93.45**	**137.93**	**141.32**

In these three figures, (a) shows the original input image, (b)-(f) illustrate the results generated by the Matlab, XJ, CJLS, SHP and proposed methods, respectively. The deblurred results of these methods may differ from each other with different type of image content and the blurring degree. However, from an overall perspective, the proposed method consistently works well, and can recover details with effective quality and few artifacts, while the results of other four compared methods seem inferior than the proposed outcomes.

In detail, the Matlab, XJ and CJLS methods perform not very well in all the figures. For example, as seen in Figure 9, compared with original image (a), (b) and (c) are somewhat clear on the close-shot while blurring on the long-shot. (d) is relatively clearer than (b) and (c) yet it contains much artifacts. The performance of the SHP method relies on how well the separability of blur map, it has blurry results in (e). Compared with (a)-(e), the content in (f) are more pleasing. As shown in the zooming of (f), the word 'Rafael' is much more clear and contains fewer artifacts.

Therefore, the results of the proposed method, which is efficient and visually pleasant with few noticeable artifacts, are consistent on these images and, in general, outperforms that of some other methods.

4.4. **Efficiency analysis.** In the last part, to overall evaluate the computational efficiency, we test all the compared methods and the proposed method on the aforementioned eight blurred images, i.e., Figure 5(e)-(h), Figures 9-11(a). Please note that XJ method is implemented by C program [50], while others is implemented by Matlab. The CPU time requirements are presented in Table 2.

According to Table 2, The Matlab and XJ methods are the fastest among the five compared methods. However, as shown in above two parts, the results of these two methods are not satisfactory in both performance and visual quality. By taking the Figure 6 as an example, we can see that the PSNRs of our method is nearly 7 lager than that of Matlab, and 5 larger than that of XJ method. As for other three methods, our method is the best one.

Moreover, the CPU time of the proposed method can still be reduced at least via two possibilities: (1): a computer with better configuration since ours is not good enough; (2): employing the parallel computing techniques since some time-consuming procedures, such as the calculating of NHC (see (4)), can be computed in paralleling way.

5. **Conclusions.** We have introduced a tight-frame based method for spatially-varying out-of-focus blur removal. The proposed method first generated the blur map using normalized Hölder coefficient and a TV based refining algorithm, then discussed the properties of corresponding kernel matrix. A tight-frame based energy functional was thus built. For tackling functional more efficiently, we used an

APD algorithm to obtain the minimizing solution. The performance of the proposed method has been compared with some state of the art algorithms for both synthesized and real images. The quantitative and qualitative results on test images demonstrated that our method can generate reasonable blur maps and can remove the blur efficiently with few artifacts.

Till now, the spatially-varying deblurring is still a high challenging task, and there are many open questions that should be investigated. Our method is effective, but it also can be improved. For example, in blur map estimation step, we cannot distinguish whether a blur texture is caused by defocus or blur texture. Further research will be tried to develop some more precise blur map estimation methods. Besides, the existing spatially-varying deblurring methods mainly estimate the blur kernel and deblurred image independently. We will extend our work to obtain them simultaneously for more excellent deblurring results in future.

Appendix.

Proof of property 2.1.

Proof. Let

$$R_j(\mathbf{A}) = \sum_{k=1, k\neq j}^{MN} |\mathbf{A}_{jk}|, \ 1 \leq j \leq MN,$$

where $\mathbf{A}_{jk}$ is the element of $\mathbf{A}$ at pixel (j, k). By Gershgorin circle theorem [26], all the eigenvalues of $\mathbf{A}$ are located in the union of MN discs:

$$\bigcup_{j=1}^{MN} \{|\mathcal{Z} - \mathbf{A}_{jj}| \leq R_j(\mathbf{A})\} \equiv G(\mathbf{A}).$$

We know that

$$R_j(\mathbf{A}) \leq R_j(\mathbf{A}) + \mathbf{A}_{jj} = \sum_{k=1}^{MN} |\mathbf{A}_{jk}| = 1,$$

which implies that $G(\mathbf{A}) \subset \mathbf{U}$, where $\mathbf{U}$ is the unit disc centered at origin. Thus, all the eigenvalues of $\mathbf{A}$ are smaller than the radius of $\mathbf{U}$, i.e., $\lambda_m < 1$. $\qquad\square$

Proof of property 2.2.

Proof. Recall that $\mathbf{g}(\boldsymbol{x}, \boldsymbol{\sigma}(\boldsymbol{x}))$ is a Gaussian function, i.e.,

$$(21) \qquad \mathbf{g}(\boldsymbol{x}, \boldsymbol{\sigma}(\boldsymbol{x}))(\boldsymbol{y}) = \frac{1}{\sqrt{2\pi}\boldsymbol{\sigma}(\boldsymbol{x})} \exp\big(-\frac{\|\boldsymbol{y} - \boldsymbol{x}\|^2}{2\boldsymbol{\sigma}^2(\boldsymbol{x})}\big), \text{ for } \boldsymbol{y} \in \mathcal{N}_{\boldsymbol{x}},$$

where $\mathcal{N}_{\boldsymbol{x}}$ denotes a neighborhood of $\boldsymbol{x}$.

Because of the separability of Gaussian function, (21) can be rewritten as:

(22)
$$\mathbf{g}(\boldsymbol{x}, \boldsymbol{\sigma}(\boldsymbol{x}))(\boldsymbol{y}) \quad = \frac{1}{\sqrt{2\pi}\boldsymbol{\sigma}(\boldsymbol{x})} \exp\left(-\frac{|y_1 - x_1|^2}{2\boldsymbol{\sigma}^2(\boldsymbol{x})}\right) \exp\left(-\frac{|y_2 - x_2|^2}{2\boldsymbol{\sigma}^2(\boldsymbol{x})}\right)$$
$$:= \sqrt{2\pi}\boldsymbol{\sigma}(\boldsymbol{x}) g^1(\boldsymbol{x})(y_1) g^2(\boldsymbol{x})(y_1).$$

Then $(\mathbf{g} \circledast \mathbf{f})(\boldsymbol{x})$ in (1) equals to:

$$(\mathbf{g} \circledast \mathbf{f})(\boldsymbol{x}) = \sqrt{2\pi}\boldsymbol{\sigma}(\boldsymbol{x}) g^1(\boldsymbol{x}) \circledast g^2(\boldsymbol{x}) \circledast \mathbf{f}(\boldsymbol{x}).$$

The matrix form of the above formula for whole image is exactly $\sqrt{2\pi}\text{diag}(\boldsymbol{\sigma})\mathbf{A}^1\mathbf{A}^2$. Thus Property 2.2 is proved. $\qquad\square$

Proof of property 2.3.

Proof. Using Property 2.2, we can readily obtain the following inequality according to the compatibility of matrix norm:

$$\|\mathbf{A}\| \le \sqrt{2\pi}\|\text{diag}(\boldsymbol{\sigma})\|\|\mathbf{A}^1\|\|\mathbf{A}^2\|.$$

For any matrix, the induced norm is essentially the largest singular value. Thus, the induced norm of the diagonal matrix $\text{diag}(\boldsymbol{\sigma})$ can be readily given as,

$$\|\text{diag}(\boldsymbol{\sigma})\| = \boldsymbol{\sigma}_m.$$

To obtain the bound of $\|\mathbf{A}^1\|$, we can first look for the bound of $\|\mathbf{A}^1\boldsymbol{\varphi}\|^2$:

(23)
$$\|\mathbf{A}^1\boldsymbol{\varphi}\|^2 = \sum_{j=1}^{MN}\left(\sum_{k=1}^{MN}\mathbf{A}^1_{jk}\boldsymbol{\varphi}_k\right)^2 \le 2\sum_{j=1}^{MN}\sum_{k=1}^{MN}(\mathbf{A}^1_{jk})^2\boldsymbol{\varphi}_k^2 = 2\sum_{k=1}^{MN}\left(\sum_{j=1}^{MN}(\mathbf{A}^1_{jk})^2\right)\boldsymbol{\varphi}_k^2,$$

where $\mathbf{A}^1_{jk}$ and $\boldsymbol{\varphi}_k$ are the elements of $\mathbf{A}^1$ and $\boldsymbol{\varphi}$ at pixels (j, k) and k, respectively.

We know that $\mathbf{A}^1$ is the matrix form of the g^1, and $g^1(\boldsymbol{x})$ is a 1-D Gaussian function. Let $j = (x_2 - 1)M + x_1$, the j-th row of $\mathbf{A}^1$ is exactly constructed from the discrete version of $g^1(\boldsymbol{x})$. Besides, given $g^1(\boldsymbol{x})$ with an arbitrary variance, when $|y_1 - x_1| = 0$, $g^1(\boldsymbol{x})(y_1) < 1$ always holds. Furthermore, due to the symmetry property of Gaussian function, when $|y_1 - x_1| = 1$, there are three values that greater than or equal to $g^1(\boldsymbol{x})(y_1)$. That is, $g^1(\boldsymbol{x})(y_1) < \frac{1}{3}$. Analogously, $g^1(\boldsymbol{x})(y_1) < \frac{1}{2k+1}$ holds when $|y_1 - x_1| = k$.

Thus, under the symmetric boundary conditions, the upper bound of $\mathbf{A}^1_{ij}$ is given by,

(24)
$$\mathbf{A}^1_{jk} < \frac{1}{2|j - k| + 1}, \quad \text{where } j, k \in [1, MN].$$

Accordingly, (23) can be further rewritten as:

(25)
$$\|\mathbf{A}^1\boldsymbol{\varphi}\|^2 \quad < 2\sum_{k=1}^{MN}\boldsymbol{\varphi}_k^2\sum_{j=1}^{MN}\left(\frac{1}{2|j-k|+1}\right)^2$$
$$\le 2\sum_{k=1}^{MN}\boldsymbol{\varphi}_k^2\left(2\sum_{j=0}\frac{1}{(2j+1)^2} - 1\right)$$
$$\le 2\left(2\sum_{j=0}\frac{1}{(2j+1)^2} - 1\right) = 2\left(\frac{3}{2}\sum_{j=1}\frac{1}{j^2} - 1\right) = \frac{\pi^2}{2} - 2.$$

Thus, we have $\|\mathbf{A}^1\| < \sqrt{\frac{\pi^2}{2} - 2}$.

Since the 1-D Gaussian function $g^2(\boldsymbol{x})$ is the same as $g^1(\boldsymbol{x})$ except the direction of convolution, the bound of $\|\mathbf{A}^2\|$ is the same as $\|\mathbf{A}^1\|$.

Therefore, taking all above into account, we have

$$\|\mathbf{A}\| \le \sqrt{2\pi}\|\mathrm{diag}(\boldsymbol{\sigma})\|\|\mathbf{A}^1\|\|\mathbf{A}^2\| \le \sqrt{2\pi}(\frac{\pi^2}{2} - 2)\boldsymbol{\sigma}_m = L.$$

$\square$

Proof of property 2.4.

Proof. As shown in Theorem 2.1 of [6], for any α in the interval $[0,\,2]$, no singular value of $\mathbf{A}$ exceeds $\max\limits_{k}\left\{\sum\limits_{j=1}^{MN}|\mathbf{A}_{jk}|^{\alpha}\right\} \cdot \max\limits_{j}\left\{\sum\limits_{k=1}^{MN}|\mathbf{A}_{jk}|^{2-\alpha}\right\}$. Let $\alpha = 1$, since $\sum\limits_{k=1}^{MN}|\mathbf{A}_{jk}| = 1$, and each column has at most $(2r+1)^2$ non-zero elements, we have,

$$\|\mathbf{A}\| \le \max_{k}\left\{\sum_{j=1}^{MN}|\mathbf{A}_{jk}|\right\} \cdot \max_{j}\left\{\sum_{k=1}^{MN}|\mathbf{A}_{jk}|\right\} \le \max_{k}\left\{\sum_{j=1}^{MN}|\mathbf{A}_{jk}|\right\} \le (2r+1)^2.$$

$\square$

REFERENCES

[1] S. Bae and F. Durand, Defocus magnification, in *Computer Graphics Forum*, **26** (2007), 571–579.

[2] L. Bar, N. Sochen and N. Kiryati, Restoration of images with piecewise space-variant blur, in *In Proceedings of the First International Conference on Scale Space Methods and Variational Methods in Computer Vision*, **4485** (2007), 533–544.

[3] G. Blanchet and L. Moisan, An explicit sharpness index related to global phase coherence, in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, 1065–1068.

[4] C. Boor, *A Practical Guide to Splines*, vol. 27, 1st edition, New York: Springer, 1987.

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato and J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, *Foundations and Trends in Machine Learning*, **3** (2011), 1–122.

[6] C. Byrne, Bounds on the largest singular value of a matrix and the convergence of simultaneous and block-iterative algorithms for sparse linear systems, *International Transactions in Operational Research*, **16** (2009), 465–479.

[7] J. Cai, R. Chan and Z. Shen, A framelet-based image inpainting algorithm, *Applied Computational Harmonic Analysis*, **24** (2008), 131–149.

[8] J. Cai, H. Ji, C. Liu and Z. Shen, Framelet-based blind motion deblurring from a single image, *IEEE Transactions on Image Processing*, **21** (2012), 562–572.

[9] J. Cai, S. Osher and Z. Shen, Linearized bregman iterations for frame-based image deblurring, *SIAM Journal on Imaging Sciences*, **2** (2009), 226–252.

[10] J. Cai, S. Osher and Z. Shen, Split bregman method and frame based image restoration, *Notices of the American Mathematical Society*, **8** (2009), 337–369.

[11] A. Chai and Z. Shen, Deconvolution: A wavelet frame approach, *Numerische Mathematik*, **106** (2007), 529–587.

[12] A. Chakrabarti, T. Zickler and W. Freeman, Analyzing spatially-varying blur, in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, 2512–2519.

[13] A. Chambolle and A. Pock, A first-order primal-dual algorithm for convex problems with applications to imaging, *Journal of Mathematical Imaging and Vision*, **40** (2011), 120–145.

[14] R. Chan, S. Riemenschneider, L. Shen and Z. Shen, Tight frame: an efficient way for high-resolution image reconstruction, *Applied Computational Harmonic Analysis*, **17** (2004), 91–115.

[15] S. Chan and T. Nguyen, Single image spatially variant out-of-focus blur removal, in *Image Processing (ICIP), 2011 18th IEEE International Conference on*, 2011, 677–680.

[16] T. Chan, *Theory and Computation of Variational Image Deblurring*, Lecture notes series, WSPC, 2005.

[17] Y. Chen, G. Lan and Y. Ouyang, Optimal primal-dual methods for a class of saddle point problems, *SIAM Journal on Optimization*, **24** (2014), 1779–1814.

[18] I. Daubechies, B. Han, A. Ron and Z. Shen, Framelets: Mra-based constructions of wavelet frames, *Applied and Computational Harmonic Analysis*, **14** (2003), 1–46.

[19] I. Ekeland and R. Témam, *Convex Analysis and Variational Problems*, Classics in Applied Mathematics (Book 28), Society for Industrial and Applied Mathematics, Philadelphia, 1999.

[20] L. Evans, *Partial Differential Equations*, vol. 19, chapter Holder-spaces, 456–461, American Mathematical Society, 1998.

[21] F. Fang, G. Zhang, F. Li and C. Shen, Framelet based pan-sharpening via a variational method, *Neurocomputing*, **129** (2014), 362–377.

[22] E. Faramarzi, D. Rajan and M. Christensen, Unified blind method for multi-image super-resolution and single/multi-image blur deconvolution, *IEEE Transactions on Image Processing*, **22** (2013), 2101–2114.

[23] T. Goldstein and S. Osher, The split bregman method for L1-regularized problems, *SIAM Journal on Imaging Sciences*, **2** (2009), 323–343.

[24] R. Gonzalez and R. Woods, *Digital Image Processing*, 3rd edition, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.

[25] A. Gupta, N. Joshi, L. Zitnick, M. Cohen and B. Curless, Single image deblurring using motion density functions, in *ECCV '10: Proceedings of the 10th European Conference on Computer Vision*, **6311** (2010), 171–184.

[26] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, 1985.

[27] Y. Huang, D. Lu and T. Zeng, A two-step approach for the restoration of images corrupted by multiplicative, *SIAM Journal on Scientific Computing*, **35** (2013), 2856–2873.

[28] S. Jaffard, Wavelet techniques in multifractal analysis, in *In Proceedings of symposia in pure mathematics*, **72** (2004), 91–152.

[29] S. Kim, I. Eom and Y. Kim, *Image Interpolation Based on Statistical Relationship Between Wavelet Subbands*, in Multimedia and Expo, 2007 IEEE International Conference on, 2007.

[30] S. Kindermann, S. Osher and P. Jones, Deblurring and denoising of images by nonlocal functionals, *SIAM Multiscale Modeling and Simulation*, **4** (2005), 1091–1115.

[31] P. Legrand and J. Vehel, Local regularity-based image denoising, in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, 3 (2003), 377–380.

[32] Y. Lou, E. Esser, H. Zhao and J. Xin, Partially blind deblurring of barcode from out-of-focus blur, *SIAM Journal on Imaging Sciences*, **7** (2014), 740–760.

[33] Y. Lou, X. Zhang, S. Osher and A. Bertozzi, Image recovery via nonlocal operators, *Journal of Scientific Computing*, **42** (2010), 185–197.

[34] J. Muzy, E. Bacry and A. Arneodo, The multifractal formalism revisited with wavelets, *International Journal of Bifurcation and Chaos*, **4** (1994), 245–302.

[35] J. Nagy and D. O'Leary, Restoring images degraded by spatially variant blur, *SIAM Journal on Scientific Computing*, **19** (1998), 1063–1082.

[36] J. Oliveira, M. Figueiredo and J. Bioucas-Dias, Parametric blur estimation for blind restoration of natural images: Linear motion and out-of-focus, *IEEE Transactions on Image Processing*, **23** (2014), 466–477.

[37] A. Ron and Z. Shen, Affine system in $l_2(\mathbb{R}^d)$: The analysis of the analysis operator, *Journal of Functional Analysis*, **148** (1997), 408–447.

[38] L. Rudin, S. Osher and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D*, **60** (1992), 259–268.

[39] C. Shen, W. Hwang and S. Pei, Spatially-varying out-of-focus image deblurring with l1-2 optimization and a guided blur map, in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, 2012, 1069–1072.

[40] Y. Tai and M. Brown, Single image defocus map estimation using local contrast prior, in *Proceedings of the 16th IEEE International Conference on Image Processing*, ICIP'09, 2009, 1777–1780.

[41] H. Trussell and S. Fogel, Identification and restoration of spatially variant motion blurs in sequential images, *IEEE Transactions on Image Processing*, **1** (1992), 123–126.

[42] M. Unser and T. Blu, Mathematical properties of the JPEG2000 wavelet filters, *IEEE Transactions on Image Processing*, **12** (2003), 1080–1090.

[43] J. Véhel, *Fractals in Multimedia*, vol. 132 of The IMA Volumes in Mathematics and its Application, chapter Signal Enhancement Based on Hölder Regularity Analysis, 197–209, Springer, New York, 2002.

[44] Y. Wang, W. Yin and Y. Zhang, *A Fast Algorithm for Image Deblurring with Total Variation Regularization*, CAAM technical reports, 2007.

[45] Z. Wang, A. Bovik, H. Sheikh and E. Simoncelli, Image quality assessment: from error visibility to structural similarity, *IEEE Transactions on Image Processing*, **13** (2004), 600–612.

[46] O. Whyte, J. Sivic, A. Zisserman and J. Ponce, Non-uniform deblurring for shaken images, *Int. J. Comput. Vis.,* **98** (2012), 168–186.

[47] C. Wu and X. Tai, Augmented lagrangian method, dual methods and split bregman iteration for ROF model, in *Scale Space and Variational Methods in Computer Vision*, 2009, 502–513.

[48] J. Xu, H. Chang and J. Qin, Domain decomposition method for image deblurring, *Journal of Computational and Applied Mathematics*, **271** (2014), 401–414.

[49] L. Xu and J. Jia, Two-phase kernel estimation for robust motion deblurring, *Computer Vision C ECCV 2010: The series Lecture Notes in Computer Science,* **6311** (2010), 157–170.

[50] L. Xu and J. Jia, Two-phase kernel estimation for robust motion deblurring, `http://www.cse.cuhk.edu.hk/leojia/projects/robust_deblur/`, 2010.

[51] X. Zhu, S. Cohen, S. Schiller and P. Milanfar, Estimating spatially varying defocus blur from a single image, *IEEE Transactions on Image Processing*, **22** (2013), 4879–4891.

[52] S. Zhuo and T. Sim, Defocus map estimation from a single image, *Pattern Recognition*, **44** (2011), 1852–1858.

Received December 2015; revised September 2016.

E-mail address: fmfang@cs.ecnu.edu.cn

E-mail address: fli@math.ecnu.edu.cn

E-mail address: zeng@hkbu.edu.hk