# Efficient Patch-Wise Non-Uniform Deblurring for a Single Image

Xin Yu, Feng Xu, Shunli Zhang, and Li Zhang

***Abstract*—In this paper, we address the problem of estimating a latent sharp image from a single spatially variant blurred image. Non-uniform deblurring methods based on projective motion path models formulate the blur as a linear combination of homographic projections of a clear image. But they are computationally expensive and require large memory due to the calculation and storage of a large number of the projections. Patch-wise non-uniform deblurring algorithms have been proposed to estimate each kernel locally by a uniform deblurring algorithm, which does not require to calculate and store the projections. The key issues of these methods are the accuracy of kernel estimation and the identification of erroneous kernels. To perform accurate kernel estimation, we employ the total variation (TV) regularization to recover a latent image, in which the edges are better enhanced and the ringing artifacts are reduced, rather than Tikhonov regularization that previous algorithms adopt. Thus blur kernels can be estimated more accurately from the latent image and estimated in a closed form while previous methods cannot estimate kernels in closed forms. To identify the erroneous kernels, we develop a novel metric, which is able to measure the similarity between the neighboring kernels. After replacing the erroneous kernels with the well-estimated ones, a clear image is obtained. The experiments show that our approach can achieve better results on the real-world blurry images while using less computation and memory.**

***Index Terms*—Blind deconvolution, non-uniform deblurring, total variation (TV) regularization.**

## I. Introduction

CAMERA shake during the exposure time leads to integrate a stream of photographs of the scene taken from slightly different viewpoints, known as motion blur. Removing blurriness from a single blurred image is an ill-posed problem as the number of unknowns exceeds the number of equations which can be observed from the data.

Significant progress has been recently made towards removing uniform blur from a single image. Uniform deblurring algorithms can achieve plausible results efficiently when camera motions only contain translations.

Manuscript received November 30, 2013; revised March 13, 2014; accepted April 28, 2014. Date of publication May 05, 2014; date of current version September 15, 2014. This work was supported by the National Natural Science Foundation of China under Grant 61132007. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Cees G. M. Snoek.

X. Yu, S. Zhang, and L. Zhang are with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: xin-yu09@mails.tsinghua.edu.cn; zslsdu@163.com; chinazhangli@tsinghua.edu.cn).

F. Xu is with Microsoft Research, Beijing 100084, China (e-mail: fengxu@microsoft.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2014.2321734

However, real camera shake, which also includes camera rotations, does not in general cause uniform blur but non-uniform blur. Non-uniform deblurring algorithms have been proposed, which model the blur as an integration of the clear scene under a sequence of planar projective transforms, known as Projective Motion Path (PMP) model [1], but they require to calculate and store a number of the projections of the intermediate estimated image in each iteration. Therefore, the bottlenecks of these algorithms are the expensive computation and the demand of large memory.

To achieve an efficient non-uniform deblurring algorithm while requiring less memory, patch-wise deblurring algorithms have been proposed, which do not need to calculate and store the projections in each loop, but divide an image into patches and estimate a blur kernel from each patch using a uniform deblurring algorithm. The estimated blur kernels, known as point spread functions (PSFs), are used to recover a clear image. The performance of the patch-wise deblurring algorithms is determined by two key factors: the accuracy of kernel estimation and the identification of erroneous kernels.

To obtain accurate blur kernels, we employ the TV regularization to recover a sharp image, in which the latent edges are better recovered while ringing artifacts are suppressed. The PSFs are not only better estimated from the latent image but also solved in a closed form. In the previous work, Harmeling *et al.* [2] employ Tikhonov regularization to recover latent images. As the recovered images may suffer from ringing artifacts, the kernel estimation may be degraded. Thus Harmeling *et al.* exploit a global constraint to guarantee the similarity between the PSFs in the kernel estimation. Because of the global constraint, the PSFs cannot be solved in a closed form. However, in our algorithm we do not require the global constraint, and the PSFs can be estimated correctly and efficiently.

In order to detect the erroneous kernels, previous works employ either the Euclidean distance to measure the similarity between the PSFs or the entropies of the kernels to evaluate the divergence [2]. The Euclidean distance of two vectors is not able to validate the similarity precisely, when rotations and deformations exist between the PSFs. For example, if there is a rotation between two similar kernels, the Euclidean distance may be large because the kernel supports are not overlapping. The entropy of a kernel is high only when the kernel diverges severely. But it may be low when the kernel does not diverge severely. To address this problem, we propose a novel metric to measure the similarity between neighboring PSFs. We integrate a kernel along horizontal and vertical directions via Radon transformation, and then assemble the integrations into a vector. Because integrations are able to eliminate slight differences, the

metric is robust to deformations between PSFs. Regarding the rotations between PSFs, the vector of the kernel is also calculated in slightly different degrees deviated from the horizontal and vertical directions. Thus the metric is also robust to rotations between PSFs.

After detecting the poorly estimated blur kernels, previous works [2], [3] replace the rejected kernels with the average of their neighboring PSFs. However, simply averaging the PSFs causes the substituted kernel to become denser than any other kernel and may lead to artifacts in the deblurred images. To maintain the sparsity of the blur kernel, we replace a rejected kernel with a well estimated neighboring one. We observe that a kernel can be estimated accurately in a patch which contains a number of informative edges. Thus we propose an edge-map to measure the number of informative edges in each patch, and replace the erroneous kernel with a neighboring one with the highest score in the edge-map.

In this paper, we mainly have two contributions. First, we take advantage of the TV regularization to recover latent images, which reduces ringing artifacts and enhances the edges. From the latent images, the smoothly varying PSFs are not only better estimated but also solved in a closed form without requiring extra constraints. Second, with the help of the proposed metric of the kernel similarity and the edge map, we detect and replace the erroneous PSFs, and obtain a clear image with less computation and memory cost.

## II. Related Work

Our work relates to the uniform and non-uniform deblurring algorithms, so we briefly review both of them. The uniform deblurring algorithms assume that a blurred image $B$ is the convolution of a latent sharp image $l$ and a space-invariant blur kernel $k$ along with additional random noise $n$, expressed as

$$B = k \otimes l + n. \tag{1}$$

Early works to recover the latent image $l$ are referred to [4]. But they are restricted to small blur kernels. Lately, impressive progress has been made, in which a more complicated kernel can be estimated from a single image. Many image priors, such as Tikhonov and the TV priors, have been exploited to regularize the model into a well-posed problem. You and Kaveh [5] employ the Gaussian prior on image gradients, known as Tikhonov regularization, to restore latent images. The drawback of Tikhonov regularization is that it enforces the smoothness of the latent images. Thus it tends to smooth out image edges [6]. Besides, the Tikhonov regularization imposes the Gaussian prior on the gradients, which is not the case for natural images. Therefore, using Tikhonov regularization as the prior of image gradients is improper and may produce ringing artifacts in the recovered images. Based on the fact that the image gradients and the blur kernels are sparse, the TV regularization and its variations [7], [8], [9], [10], [11] are developed to keep the sparsity of image gradients and blur kernels. The TV regularization is viewed as the $\ell_1$ regularization on the image gradients. Chan and Wong [7] exploit the TV regularization to recover both the latent images and blur kernels. Shan *et al.* [8] introduce a local smoothness prior to reduce ringing artifacts and employ the TV regularization to keep the sparsity of the image gradients. Krishnan *et al.* [9] propose a normalized TV regularization to favor sharp edges over blurry ones, and apply the $\ell_1$ regularization to recover blur kernels. Cai *et al.* [10], [11] exploit the $\ell_1$ regularization on the wavelet coefficients of the latent images and blur kernels to restore motion blur. Fergus *et al.* [12] exploit the heavy-tailed natural image prior to recover the latent images. Xu *et al.* [13] apply the unnatural $\ell_0$ sparse representation of the image derivatives as a prior. Goldstein and Fattal [14] and Hu *et al.* [15] exploit the spatial correlation of natural images to estimate blur kernels. Levin *et al.* [16] efficiently marginalize all the latent images to estimate a blur kernel in the framework of Bayesian Variation. To further improve the performance, sophisticated methods [17], [18], [19], [20] are developed to predict sharp images and select salient structures to estimate blur kernels. Lin *et al.* [21] apply directional filters to a noisy and blurry image, and then use inverse Radon transform to reconstruct the blur kernel from the filtered images. After obtaining the space-invariant blur kernel, restoring a sharp image from a blurry one is referred to non-blind deconvolution [22], [23], [24], [25]. These methods can also be utilized in the space-variant non-blind deconvolution with slight modification.

When the blurriness is non-uniform, a single spatially invariant kernel is not able to represent the blurriness in the whole image. Therefore, we rewrite Eqn. (1) into a matrix form:

$$B = Kl + n \tag{2}$$

where $K$ represents the matrix form of $k$. Gupta *et al.* [26] model the blur matrix $K$ as a motion density function and the blur image $B$ as the summation over the images taken from all camera poses. Eqn. (2) is rewritten as

$$B = \sum_i a_i K_i l + n$$

where $a_i$ denotes the density at the camera pose $i$, and $K_i$ is modeled by three degrees of freedom of camera motions in [26], [27], [28], including in-plane rotations and translations in the horizontal and vertical directions. Meanwhile, Whyte *et al.* [29] and Xu *et al.* [13] model each basis $K_i$ with 3D rotations. Tai *et al.* [1] propose a PMP blur model to solve the space-invariant non-blind deconvolution.

However, the computation of the algorithms based on PMP models is expensive and the memory requirement is quite large. To address these issues, patch-wise non-uniform deblurring algorithms have been proposed. Ji and Wang [30] divide a blurred image into overlapped patches to estimate the PSFs, and then construct a pixel-wise blur matrix interpolated from the PSFs. The clear image is recovered by the blur matrix. However, generating the pixel-wise blur matrix is still time consuming while storing the matrix requires large memory. Harmeling *et al.* [2] also divide a blurred image into overlapped patches, and employ the efficient flow filter (EFF) [31] to recover a latent sharp image with the estimated PSFs.

With the help of the advanced hardware, some algorithms are designed to recover the blurry images. Tai *et al.* [32], [33] and Ben-Ezra and Nayar [34] explore a hybrid camera to capture high resolution images while recording the path of the camera

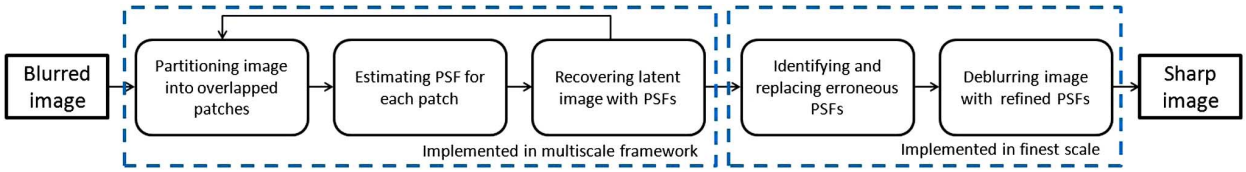



Fig. 1. Outline of the proposed blind non-uniform deblurring algorithm.

motions by a low resolution camera. Joshi *et al.* [35] exploit an inertial measurement unit to catch camera motions. Bando *et al.* [36] use a time-varying light field modulator at the lens aperture to capture a time-varying light field. However, the sophisticated hardware is impractical for amateurs.

## III. Methodology

In this section, we incorporate the non-blind deconvolution based on the TV regularization into the framework of the EFF to better recover a latent sharp image while previous works [2], [27], [31] use Tikhonov regularization. Then we estimate the smoothly varying PSFs from overlapping patches in a closed form. After obtaining the PSFs, we use three steps to determine whether a kernel is well estimated or not. Using the proposed edge map, we replace the erroneous kernels with the well estimated ones. Finally, we recover a sharp image with the refined PSFs. The pipeline of our algorithm is illustrated in Fig. 1.

### A. Framework of Efficient Filter Flow

The framework of the EFF proposed by Hirsch *et al.* [31] is exploited to handle smoothly space-variant convolutions. The basic idea of the EFF is that a spatially variant filtering can be implemented by chunking a signal into overlapping patches, and then filtering each patch with a spatially invariant PSF, finally assembling the filtered signal from the filtered patches using the overlap-add (OLA) method of Stockham [37].

Let $B$ be a blurred image of length $m$, $l$ be a sharp image of length $n$, and $k$ be a blur kernel of length $s$. (Here, an image is stacked into a long column vector.) Then the convolution is written as $B_i = \sum_{j=1}^{s} k_j l_{i-j}$, for $1 \le i \le m$. The framework of the EFF aims to extend a space-invariant filtering to a space-variant one and is defined as

$$B_i = \sum_{r=1}^{p} \sum_{j=1}^{s} k_j^{(r)} w_{i-j}^{(r)} l_{i-j}, \quad \text{for } 1 \le i \le m \tag{3}$$

where $p$ is the number of the overlapping patches, $k^{(r)}$ is the blur kernel of the $r$-th patch ($1 \le r \le p$), and $w^{(r)} \ge 0$ is a window function to fade the $r$-th patch in and masking the others out. The sum of the weights at each pixel should be equal to one, *i.e.*, $\sum_{r=1}^{p} w_i^{(r)} = 1$, for $1 \le i \le m$. Without the normalization, there will be artifacts in the overlapping areas.

As indicated in Eqn. (3), the EFF is linear in $l$ and in $\boldsymbol{k}$, where $\boldsymbol{k}$ is a vector stacked by $p$ PSFs $k^{(1)}, \ldots, k^{(p)}$. It implies there are matrices $K$ and $L$ such that $B = Kl = L\boldsymbol{k}$. According to [31], the matrices are expressed as

$$K = Z_b \sum_{r=1}^{p} C_r^T F^{-1} Diag(F Z_k k^{(r)}) F C_r Diag(w^{(r)}), \tag{4}$$

$$L = Z_b \sum_{r=1}^{p} C_r^T F^{-1} Diag(F C_r Diag(w^{(r)}) l) F Z_k B_r \tag{5}$$

where $Z_b$ is a matrix that appends zeros to the valid part of the space-variant convolution such that its size matches the full size of an input image $l$, $C_r$ is a matrix that crops the $r$-th patch from the input image, $F$ is the Discrete Fourier Transform matrix, Diag($v$) is a diagonal matrix with $v$ along its diagonal, $Z_k$ is a matrix that appends zeros to $k^{(r)}$ such that its size matches the patch size, and $B_r$ is a matrix that crops the $r$-th PSF from the vector $\boldsymbol{k}$. For the blind deconvolution, $K^T$ and $L^T$ are also required and expressed as

$$K^T = \sum_{r=1}^{p} Diag(w^{(r)}) C_r^T F^{-1} \overline{Diag(F Z_k k^{(r)})} F C_r Z_b, \tag{6}$$

$$L^T = \sum_{r=1}^{p} B_r^T Z_k^T F^{-1} \overline{Diag(F C_r Diag(w^{(r)}) l)} F C_r Z_b. \tag{7}$$

Eqn. (6) implies that the patches are firstly recovered locally and then assembled into a latent image. Eqn. (7) implies that each PSF can be estimated from the corresponding patch locally. The computational complexity of implementations for $K$, $K^T$, $L$ and $L^T$ is $O(n \log q)$, where $q$ is the patch size [31]. It indicates that space-variant convolutions can be implemented in the EFF as efficiently as space-invariant ones.

### B. TV non-blind deconvolution in EFF

Previous works [2], [17], [18] adopt the $\ell_2$ regularization, known as Tikhonov regularization, as the constraint on the image gradients in the non-blind deconvolution. Although the $\ell_2$ regularization can be solved in a closed form, it does not favor the sparsity of image gradients and will cause ringing artifacts in the estimated images, as shown in Fig. 2. To improve the sparsity of the image gradients, the TV regularization is employed. Therefore, we estimate a latent image by minimizing the following objective function

$$\hat{l} = \arg\min_{l} \frac{\lambda}{2} \|B - k \otimes l\|_2^2 + \|\nabla l\|_1,$$
$$\|\nabla l\|_1 = \sum_{x} |l_h| + |l_v| \tag{8}$$

where $\lambda$ is a trade-off weight between the data term and the regularization term, $\nabla l = \{l_h, l_v\}$ represents the image gradients, and $x$ denotes the pixel coordinate in the image. We assume there is i.i.d zero-mean Gaussian noise in the image.

We exploit the Split Bregman Iteration (SBI) method [38] to minimize Eqn. (8), which is one of the fastest methods to solve the $\ell_1$ minimization. Then Eqn. (8) is rewritten as

$$\min_{d,l} \|d\|_1 + \frac{\lambda}{2}\|B - k \otimes l\|_2^2, \text{subject to } d(x) = \nabla l(x) \tag{9}$$

where $d$ is a substitutive variable. We reformulate Eqn. (9) into an unconstrained optimization problem

$$\min_{d,l} \|d\|_1 + \frac{\lambda}{2}\|B - k \otimes l\|_2^2 + \frac{\gamma}{2}\|d - \nabla l - b\|_2^2 \tag{10}$$

where $b$ is a slack variable, and $\gamma$ is a weight for the slackness term. The Split Bregman method minimizes $d$ and $l$ alternatingly. It leads to two separate subproblems. The $d$ and $l$ subproblems are given as below

$$\begin{aligned} &d \text{ subproblem} : \arg\min_{d} \|d\|_1 + \frac{\gamma}{2}\|d - \nabla l - b\|_2^2, \\ &l \text{ subproblem} : \arg\min_{l} \frac{\lambda}{2}\|B - k \otimes l\|_2^2 + \frac{\gamma}{2}\|\nabla l - d + b\|_2^2. \end{aligned} \tag{11}$$

The $d$ subproblem can be solved by the iterative shrinkage threshold (IST) when fixing $l$

$$d(x) = \frac{\nabla l(x) + b(x)}{\|\nabla l(x) + b(x)\|_2} \max\left\{\|\nabla l(x) + b(x)\|_2 - \frac{1}{\gamma}, 0\right\}. \tag{12}$$

After obtaining $d$, the $l$ subproblem can be solved in the FFT domain,

$$l = \mathcal{F}^{-1}\left(\frac{\lambda\bar{\mathcal{F}}(k)\mathcal{F}(B) + \sum_{*\in h,v}\gamma\bar{\mathcal{F}}(g_*)\mathcal{F}((d-b)_*)}{\lambda\bar{\mathcal{F}}(k)\mathcal{F}(k) + \sum_{*\in h,v}\gamma\bar{\mathcal{F}}(g_*)\mathcal{F}(g_*)}\right) \tag{13}$$

where $\mathcal{F}$, $\mathcal{F}^{-1}$ and $\bar{\mathcal{F}}$ represent the FFT, the inverse FFT and the complex conjugation of the FFT, respectively. $g_h$ and $g_v$ indicate the horizontal and vertical derivative filters: $g_h = [-1, 1]$ and $g_v = [-1, 1]^T$ respectively. Finally, the auxiliary variable $b$ in the $t+1$th iteration is updated as

$$b^{t+1} = b^t + \nabla l - d. \tag{14}$$

Then we incorporate the TV non-blind deconvolution into the EFF, expressed as

$$l = \sum_{r=1}^{p} Diag(w^{(r)}) C_r^T \mathrm{TV}_{\mathrm{NBD}}\{k^{(r)}, C_r B\} \tag{15}$$

where $\mathrm{TV}_{\mathrm{NBD}}\{k^{(r)}, C_r B\}$ represents the TV non-blind deconvolution in Eqn. (8), *i.e.*, $k^{(r)}$ and $C_r B$ refer to the blur kernel and the blurry image of the $r$-th patch, respectively. Given the PSFs, each patch is recovered separately and then all the overlapping patches are assembled to the whole image. The window function $w^{(r)}$ is a Bartlett-Hanning window with 50% overlap to crop the $r$-th patch from the whole image for all the experiments.

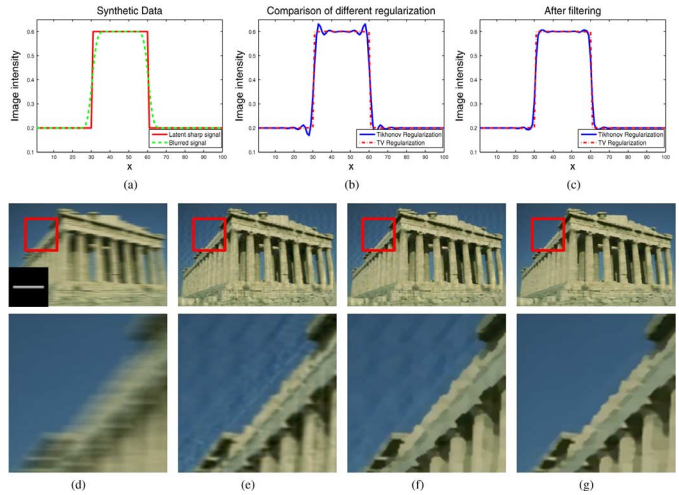



Fig. 2. Comparison between the TV regularization and Tikhonov regularization. (a) Synthetic 1D signal and its blurred version. (b) Deblurring results based on the two regularization models. (c) Deblurring results after bilateral and shock filtering. (d) Synthetic blurred image and its blur kernel. (e) Deblurred image by Tikhonov regularization. (f) Result of (e) after bilateral and shock filtering. (g) Deblurred image by the TV regularization.

As shown in Fig. 2, both on the 1D signal and the 2D image, the deblurring results recovered by Tikhonov regularization suffer from obvious ringing artifacts while the ringing artifacts are unnoticeable in the deblurring results recovered by the TV regularization. Thus the results recovered by the TV regularization are better than those recovered by Tikhonov regularization. The bilateral and shock filters are used to suppress the ringing artifacts and enhance the edges [17]. However, the filtered edges are not as sharp as those recovered by the TV regularization model and the ringing artifacts still exist shown in Fig. 2(c) and Fig. 2(e).

Although solving the $\ell_1$ regularization needs more iterations than solving the $\ell_2$ regularization, the computation is not expensive. Note that after the first iteration the SBI only needs two FFTs (*i.e.*, FFTs of $(d-b)_h$ and $(d-b)_v$) and one inverse FFT to solve Eqn. (13) while the FFTs of the other variables are only calculated once in the first iteration. In Fig. 3, We choose four images from the Berkeley database [39] and eight blur kernels from the dataset [40]. We blur each image with the eight kernels and add $0.1\%$ zero-mean Gaussian noise. The convergence of the SBI for the four images is shown in Fig. 3. The SBI converges quite fast. Notice that 20 iterations of the SBI is enough to achieve a good result.

### C. Blur Kernel Estimation

As implied in the EFF, the smoothly varying PSFs can be estimated locally. We exploit the uniform kernel estimation method to estimate each PSF in each overlapping patch. The spatially invariant blur kernel is estimated by minimizing the following energy function

$$E(k) = \sum_{i=1,2} \omega_i \|k \otimes \partial^{i-1} P - \partial^i B\|_2^2 + \alpha\|k\|_2^2 + \beta\|\nabla k\|_2^2 \tag{16}$$

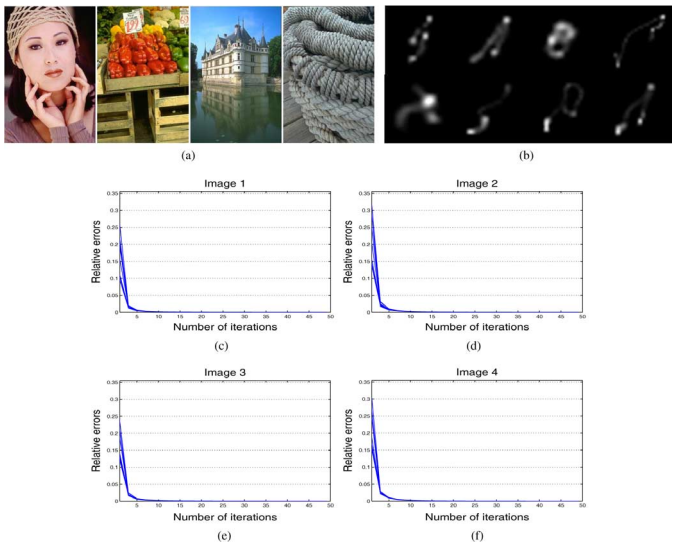



Fig. 3. (a) Four images used to generate the blurred images. (b) Blur kernels. (c)–(f) Convergence of the SBI for the four images.

where $\partial^0$, $\partial^1$ and $\partial^2$ represent the zeroth, the first and the second order partial derivative operators respectively, $\omega_1$ and $\omega_2$ indicate the corresponding weights, $P = \{P_h, P_v\}$ is the gradient map of the predicted image, in which R-map [18] is employed to select the informative edges while suppressing sparkle noise, $\alpha$ and $\beta$ are the weights for the regularization terms.

However, the blurriness in the image is spatially variant. Even though we divide a whole image into overlapping patches, a patch recovered by a space-invariant kernel may suffer ringing artifacts. Besides, when the blind deconvolution is implemented in a multi-scale framework, the PSFs are usually inaccurate in the top levels of the image pyramid, which may also lead to ringing artifacts. Therefore, we apply the bilateral and shocking filters to enhance the edges while reducing the ringing artifacts. Then we use a window function to fade the $r$-th patch in and mask the others out, and then estimate the $r$-th kernel locally. To estimate $p$ PSFs for an image, the energy function in Eqn. (16) is reformulated as

$$E(\boldsymbol{k}) = \sum_{r=1}^{p} \sum_{i=1,2} \omega_i \| k^{(r)} \otimes w^{(r)} \odot \partial^{i-1} P - w^{(r)} \odot \partial^i B \|_2^2 + \alpha \| k^{(r)} \|_2^2 + \beta \| \nabla k^{(r)} \|_2^2 \quad (17)$$

where $\odot$ denotes the Hadamard product. Notice that the PSFs in $\boldsymbol{k}$ are independent from each other, thus each kernel can be solved separately. The energy function of the kernel $k^{(r)}$ is expressed as

$$\begin{aligned} E(k^{(r)}) &= \sum_{i=1,2} \omega_i \| k^{(r)} \otimes P_i^r - B_i^r \|_2^2 + \alpha \| k^{(r)} \|_2^2 \\ &\quad + \beta \| \nabla k^{(r)} \|_2^2 \\ &= \sum_{i=1,2} \omega_i \| \mathbb{P}_i^r \mathbf{k}^{(r)} - \mathbf{B}_i^r \|_2^2 + \alpha \| \mathbf{k}^{(r)} \|_2^2 + \beta \| \nabla \mathbf{k}^{(r)} \|_2^2 \end{aligned}$$

where $P_i^r = w^{(r)} \odot \partial^{i-1} P$, $B_i^r = w^{(r)} \odot \partial^i B$, $\mathbb{P}_i^r$ is a matrix consisting of $P_i^r$, and $\mathbf{B}_i^r$ and $\mathbf{k}^{(r)}$ represent the vector forms of $B_i^r$ and $k^{(r)}$ respectively. To obtain the optimal $k^{(r)}$, let $\frac{\partial E(k^{(r)})}{\partial k^{(r)}} = 0$, and then we have

$$\left( \sum_{i=1,2} \omega_i (\mathbb{P}_i^r)^T \mathbb{P}_i^r + \alpha I + \beta (G_h^T G_h + G_v^T G_v) \right) \mathbf{k}^{(r)} = \sum_{i=1,2} \omega_i (\mathbb{P}_i^r)^T \mathbf{B}_i^r \quad (18)$$

where $I$ is an identity matrix, and $G_h$ and $G_v$ denote the Toeplitz matrices of the derivative filters $g_h$ and $g_v$ respectively. According to Parseval's theorem, Eqn. (18) can be solved by the FFT in a closed form, written as

$$k^{(r)} = \mathcal{F}^{-1} \left( \frac{\sum_{i=1,2} \omega_i \bar{\mathcal{F}}(P_i^r) \mathcal{F}(B_i^r)}{\sum_{i=1,2} \omega_i \bar{\mathcal{F}}(P_i^r) \mathcal{F}(P_i^r) + \sum_{*\in h,v} \beta \bar{\mathcal{F}}(g_*) \mathcal{F}(g_*) + \alpha} \right), \quad \text{for } 1 \le r \le p. \quad (19)$$

Finally, as kernel values are positive and sum to 1, each kernel $k^{(r)}$ is normalized accordingly. In our experiments, we adopt the weights of [8] for $\omega_1$ and $\omega_2$, and set $\alpha$ to 50 and $\beta$ to 5.

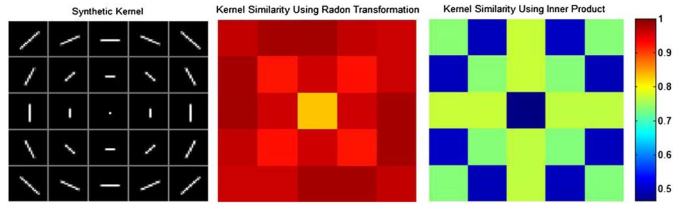



Fig. 4. Comparison of the similarity measurements between the proposed metric and the cosine similarity. Due to the rotation, the similar blur kernels (left) are judged to be dissimilar by the cosine similarity while our metric can identify their similarity.

### D. Removing Poorly Estimated Blur Kernels

When the patches are flat or lack salient structural information, our method may produce inaccurate kernels. In that case, the erroneous kernels will lead to ringing artifacts. Fig. 5 illustrates that the textureless patches lead to inaccurate blur kernels. There are mainly three types of erroneous kernels: a delta kernel, a kernel stuck in a local minimum and a divergent kernel. As shown in Fig. 5, a delta kernel (*e.g.*, the kernel in the top-right corner) estimated from a textureless patch does not lead to obvious ringing artifacts but remains the textureless patch unchanged. The ringing artifacts are mainly caused by the divergent kernels and the kernels stuck in local minima.

Determining whether a kernel is well estimated or not is another key problem in our patch-wise deblurring algorithm. Harmeling *et al.* [2] employ an entropy to determine which kernel is poorly estimated. However, when a blur kernel does not diverge much or is stuck in a local minimum, the kernel entropy may be low. It is difficult to distinguish erroneous kernels from well estimated ones by thresholding the kernel entropies. When the trajectory of the camera motion is long, the threshold is required to be tuned, because larger kernels lead to higher entropies. We observe that a well estimated kernel always satisfies two conditions: (i) a kernel should be sparse. If a kernel is dense, it must be divergent. (ii) a kernel should be similar to its neighboring kernels. If a kernel is different from its neighbors, it is probably a poorly estimated kernel. Based on this observation,

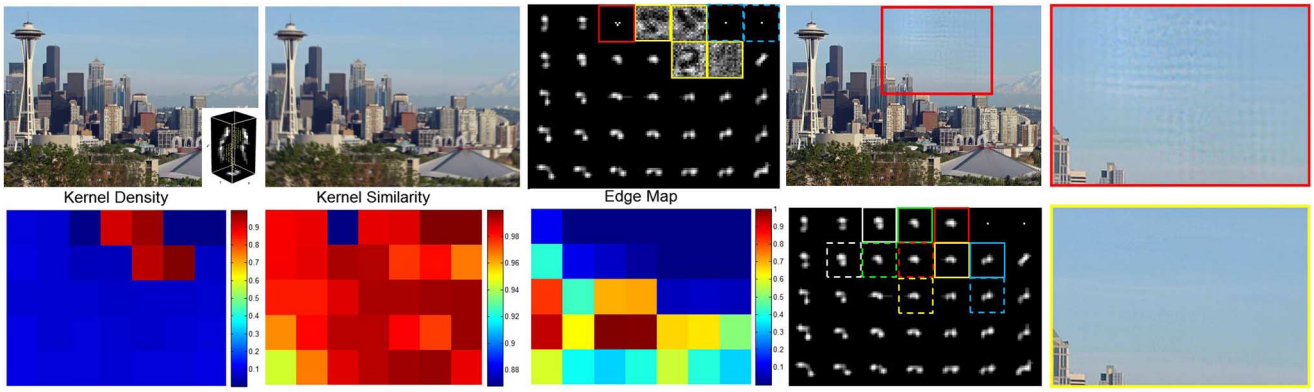



Fig. 5. An example for the identification and replacement of the erroneous kernels. The top row (from left to right): the original image and its synthetic kernel for generating the blurred image; the blurred image; the estimated PSFs from Eqn. (17); the deblurred image using the estimated PSFs; and the closed-up of the deblurred image. The bottom row: the kernel density; the kernel similarity; the edge map; the refined PSFs; and the closed-up recovered by the refined PSFs. (The poorly estimated kernels are plotted in the solid frames, and the kernels in the dash frames are used to replace the erroneous ones.)

we propose a simple but effective method to determine whether a kernel is poorly estimated or not. According to Condition (i), we exploit the kernel density to roughly determine severely divergent kernels. The density of a blur kernel is defined as

$$D = \frac{N_s}{k_s \times k_s} \tag{20}$$

where $k_s \times k_s$ denotes the kernel size and $N_s$ represents the number of kernel supports. The kernel density is more intuitive to measure the sparsity of kernels than the entropy. In our experiments, we set a threshold 0.5 on D to roughly determine a kernel is sparse or not. If the density is over the threshold, it indicates that the blur kernel is divergent.

After identifying the severely divergent kernels, we need to detect other poorly estimated kernels, which are stuck in local minima. Based on Condition (ii), if a kernel is not similar to its neighboring PSFs, it is probably poorly estimated. Since camera motions include rotations along the optical axis, there may be rotations between the space-variant PSFs. We propose a new metric to measure the similarity of the space-variant PSFs, which is robust to the rotations between the PSFs. The Radon transform of a blur kernel $k^{(r)}$ is defined as

$$R_\theta(k^{(r)}) = \int k^{(r)}(x\cos\theta - y\sin\theta, x\sin\theta + y\cos\theta)dy,$$
$$R_\phi(k^{(r)}) = \int k^{(r)}(x\cos\phi - y\sin\phi, x\sin\phi + y\cos\phi)dy$$

where $\phi = \theta + 90°$. We define $R_h(k^{(r)}) = R_\theta(k^{(r)})$ and $R_v(k^{(r)}) = R_\phi(k^{(r)})$ when $\theta = 0°$, and combine them into a vector $\kappa_r = [R_h(k^{(r)}); R_v(k^{(r)})]^T$. Regarding the rotations between a kernel $k^{(r)}$ and its neighboring kernels $k^{(j)}, j \in \mathcal{N}\{r\}$ (the severely divergent kernels are not taken into account), we need to calculate $\kappa_r$ in different degrees $\theta \in [-\varphi : \varphi]$, where $\varphi$ is the range of the camera rotations, and the vector $\kappa_r^\theta$ is represented as $[R_\theta(k^{(r)}); R_\phi(k^{(r)})]^T$. The similarity measurement between a kernel $k^{(r)}$ and its neighboring kernels $k^{(j)}, j \in \mathcal{N}\{r\}$ is defined as

$$c(k^{(r)}) = \max_{j\in\mathcal{N}\{r\},\theta\in[-\varphi:\varphi]} \frac{\kappa_r^\theta \kappa_j}{\|\kappa_r^\theta\|_2 \|\kappa_j\|_2}, \quad \text{for } 1 \le r \le p. \tag{21}$$

The highest score among the different degrees is regarded as the score of the similarity measurement. A higher score means a kernel is more similar to its neighbors. In our experiments, we set the largest rotation angle $\varphi$ to 5°, which gives good results in all our experiments. A kernel will be removed if its similarity measurement is below a threshold. The threshold is an empirical value, and we set it to 0.95 for all the experiments. As shown in Fig. 4, the cosine similarity is difficult to evaluate the similarity between the PSFs with rotations while our proposed metric evaluates the similarity of the PSFs well. In Fig. 5, the poorly estimated kernels are identified by our metric.

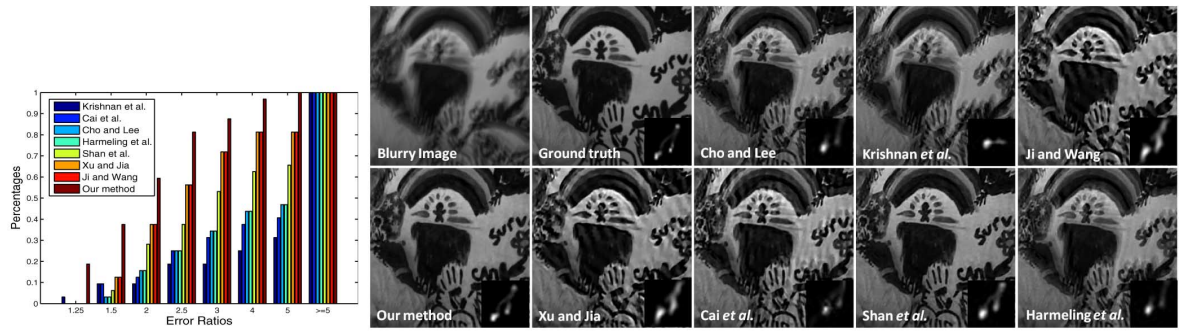
After detecting the poorly estimated kernels, we propose to find a well estimated neighboring kernel to replace the erroneous one. We observe that a kernel will be estimated more accurately from a patch containing many edges with different directions. Otherwise, it is difficult to estimate an accurate kernel when there are few edges in a patch. Therefore, we propose an edge map to measure the number of the informative edges in a patch. With the help of the edge map, we choose a kernel that is considered as a well estimated one to replace the erroneous one. To calculate the edge map, we divide the directions of the image gradients into $N$ bins evenly, and each bin accumulates the magnitudes of the image gradients for its corresponding direction. We set $N$ to 4 according to [17]. Then each patch is assigned a score based on the histogram

$$h_r = \prod_{j=1}^{N} \left(1 - \exp\left(-\frac{1}{\mu} f_j\right)\right), \quad \text{for } 1 \le r \le p \tag{22}$$

where $\mu$ is the mean of the magnitudes of the image gradients in R-map, and $f_j$ represents the accumulated magnitudes of the image gradients in the $j$-th bin. Notice that the mean of the exponential distribution is the reciprocal of the mean of the magnitudes of the image gradients. Eqn. (22) implies that a kernel can be estimated more accurately from the patch whose accumulated magnitudes are larger than the mean of the whole image gradient magnitudes. We normalize all the scores by the highest score

$$\hat{h}_r = \frac{h_r}{\max_r h_r}, \quad \text{for } 1 \le r \le p \tag{23}$$

and then the edge map can be measured by a value between 0 and 1. A higher score means that a kernel is more likely estimated well in the patch.




(a) (b)

Fig. 6. Quantitative evaluation across the dataset of Levin *et al.* [40]. (a) Cumulative histograms of the error ratios. (b) Visual comparison of the state-of-the-art methods.

Because we divide the images into highly overlapping patches, thus neighboring patches share a common region. In this situation, the PSFs of neighboring patches should be similar, even though the true blur kernels have spatially varying effects to a certain extent. Therefore, we believe it is reasonable to use the well estimated kernels in the neighboring patches to replace the poorly estimated ones. In Fig. 5, we show that after replacing the poorly estimated kernels with the well estimated ones, the ringing artifacts of the deblurred image are reduced.

Algorithm 1 illustrates our three steps to find and replace the erroneous PSFs. In our pipeline, we implement Algorithm 1 in the finest scale. There are two reasons that we only implement this algorithm in the finest scale but not in multi-scale. First, as the estimated kernel size is small in the coarse scales, *e.g.*, the kernel size is smaller than $5 \times 5$, the PSFs are inaccurate and denser. Thus Eqn. (20) cannot work well in identifying the severely divergent kernels. Besides, the PSFs are similar when the kernel size is small. As a consequence, it is difficult to distinguish the poorly estimated PSFs using Eqn. (21). Second, even though we have not removed the poorly estimated blur kernels in coarse scales, we remove them in the finest scale, which does not lower the accuracy of the kernel estimation but cuts the computational cost for removing the poorly estimated kernels in coarse scales.

**Algorithm 1** Identifying and replacing the erroneous PSFs

**Input:** PSFs estimated from Eqn. (17), the input image $B$

**Step 1:** calculate the density of each blur kernel using Eqn. (20) and identify the severely divergent PSFs.

**Step 2:** calculate the similarity between the PSFs using Eqn. (21) and find the poorly estimated PSFs.

**Step 3:** calculate the edge map using Eqn. (22) and Eqn. (23), and replace the erroneous PSFs with the well estimated ones based on the edge map.

**Output:** the refined PSFs.

### *E. Final Image Recovery*

There is a batch of non-blind deconvolution methods for uniform motion blur, which can be extended to deconvolute non-uniform motion blur in the framework of the EFF. We adopt the same non-blind deconvolution method proposed by Krishnan and Fergus [24]. To achieve a clear image, we minimize the following objective function

$$E(l) = \sum_{r=1}^{p} \|w^{(r)} \odot C_r B - k^{(r)} \otimes (w^{(r)} \odot C_r l)\|_2^2 + \eta \|\partial(w^{(r)} \odot C_r l)\|_\rho^\rho$$

where the $\ell_\rho$ norm represents a natural image prior with $0 \leq \rho \leq 1$. After introducing an auxiliary variable $v$, we minimize the objective function

$$E(l) = \sum_{r=1}^{p} \|w^{(r)} \odot C_r B - k^{(r)} \otimes (w^{(r)} \odot C_r l)\|_2^2 + 2^\tau \|\partial(w^{(r)} \odot C_r l) - C_r v\|_2^2 + \frac{1}{2000} \|C_r v\|_\rho^\rho \quad (24)$$

where $\tau$ increases from 0 to 8 during nine alternating updates in $l$ and $v$. Choosing $\rho = 2/3$ allows an analytical formula for the update in $v$ [24].

Taking all the steps into account, we derive the complete iteration of our algorithm in the multi-scale implementation (see Algorithm 2 for details).

**Algorithm 2** Overall procedure of the proposed algorithm

**Input:** Blurred image $B$, Kernel size $k_s \times k_s$, Patch number $p = N_h \times N_v$, Scale number $ns = 2\left\lceil \log(\frac{k_s}{3}) \right\rceil + 1$

**for** scale $= 1$ to $ns$ **do**

**while** PSFs do not converge**do**

Recover the latent image using Eqn. (15).

Estimate the PSFs using Eqn. (17).

**end while**

scale $=$ scale $+ 1$

**end for**

Identify the erroneous PSFs and replace them using Algorithm 1.

Deblur the blurry image $B$ with the refined PSFs using Eqn. (24).

**Output:** The deblurred image $l$.

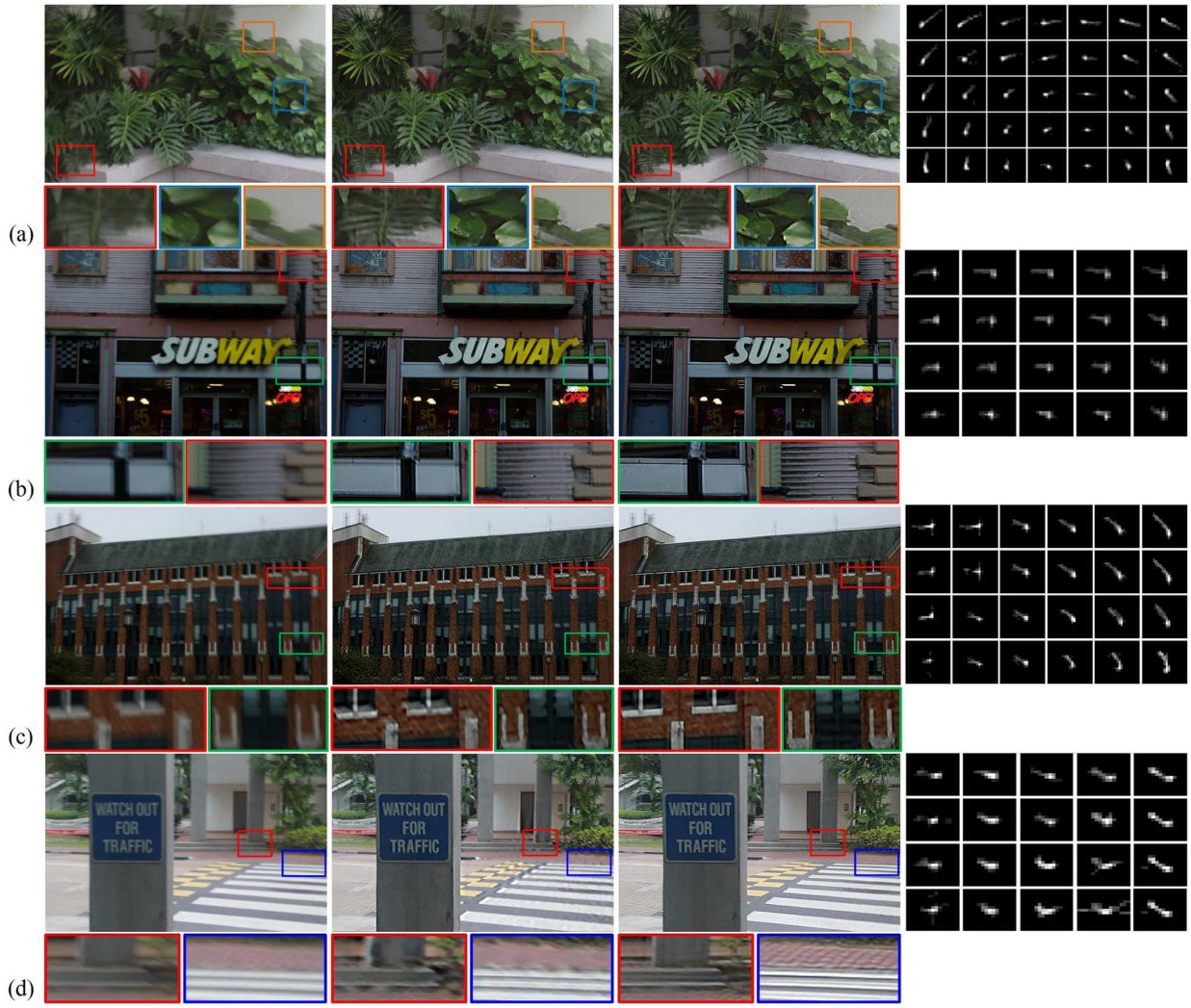



Fig. 7. Comparison with the approaches of Ji and Wang [30] and the proposed method. (From left to right: the blurry images; results of Ji and Wang; our results; and our refined PSFs after using the strategy of identifying and replacing the poorly estimated kernels.)

## IV. Experiments

In this section, we compare the image recovery on the synthetic dataset and comprehensive challenging images to demonstrate the effectiveness of our method, and then compare the speed with the state-of-the-art methods to show the efficiency of our method.

### A. Comparison of the image recovery

*Quantitative evaluation on the dataset of Levin* ***et al.[40]***: We compare our method with the blind image deconvolution algorithms proposed by Cai *et al.* [11], Shan *et al.* [8], Cho and Lee [17], Xu and Jia [18], Krishnan *et al.* [9], Harmeling *et al.* [2] as well as Ji and Wang *et al.* [30] on the dataset of [40]. These algorithms use different regularizations in non-blind deconvolution but similar kernel estimation methods. There are 32 blurred images synthesized from 4 images and 8 different spatially invariant kernels in the dataset. The ground truth $x$, the blur kernels $k_{gt}$ and the blurred data $y$ are provided. In order to evaluate the performance of kernel estimation, the same non-blind deconvolution [22] is performed to recover final images. Thus the quality of the recovered images only depends on the recovered kernels. The error ratio is defined as $\|I_k - x\|_2^2 / \|I_{k_{gt}} - x\|_2^2$, where $I_k$ and $I_{k_{gt}}$ represent the results of the non-blind deconvolution with the estimated kernel and the ground truth blur kernel respectively.

As illustrated in Fig. 6(a), our results are better than the results of the algorithms of [2] and [30], which employ the Tikhonov regularization to estimate latent images, judging from the error ratios of the deblurring results. Notice that in Fig. 6, the quantitative results of Ji and Wang's method [30] are identical to the results of Xu and Jia's method [18]. This is because when estimating the spatially invariant PSFs, Ji and Wang adopt the method of Xu and Jia to estimate the PSFs. Since here the blur kernel is spatially invariant, the interpolation of blur kernels proposed by Ji and Wang is not needed, thus the quantitative results of the two methods are identical. Also in Fig. 6, the quantitative results of Harmeling *et al.*'s method [2] are identical to the results of Cho and Lee's method [17]. This is because Harmeling *et al.* employ the identical method to Cho and Lee's method to estimate spatially invariant kernels.

The method of Krishnan *et al.* [9] are sometimes stuck in local minima because of its non-convexity. The method of Shan *et al.*

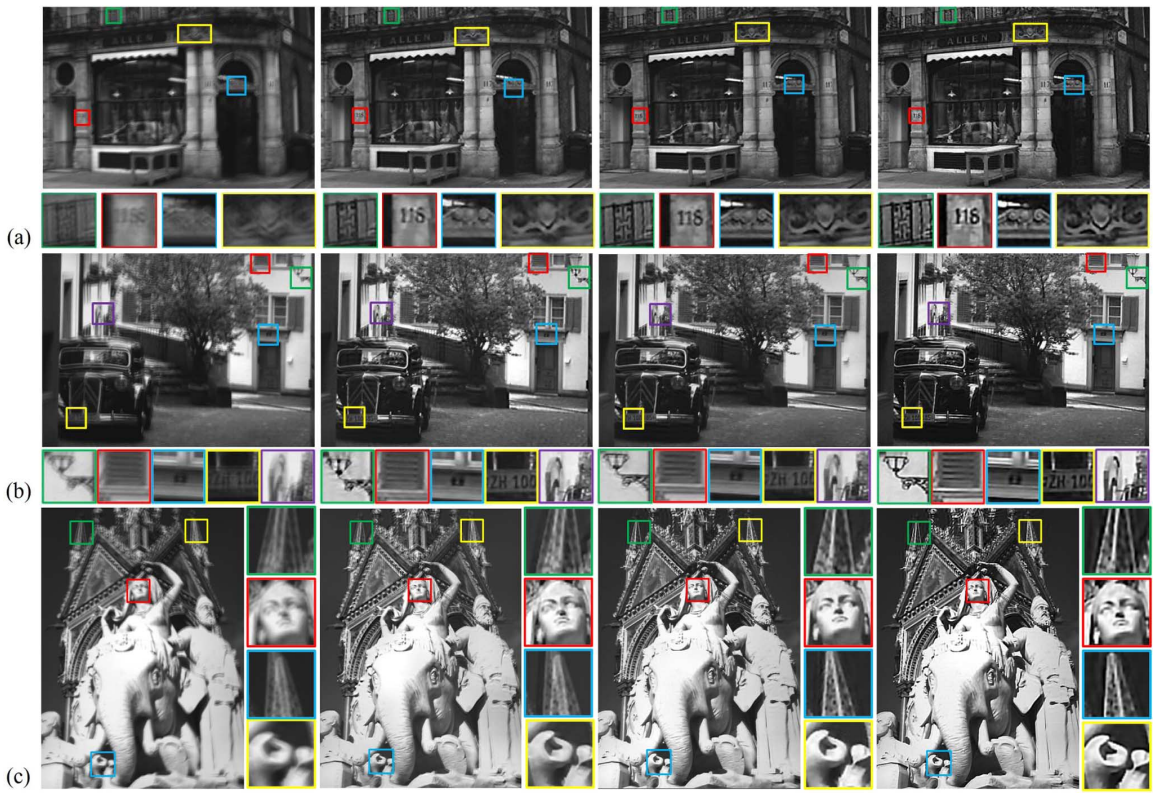



Fig. 8. Comparison with the approaches of Harmeling *et al.* [2] and Hirsch *et al.* [27] and the proposed method. (From left to right: the blurry images; results of Harmeling *et al.*; results of Hirsch *et al.*; and our results.)

[8] utilizes the TV regularization on both the images and kernels, but it uses the gradients which are not total variation preserving for the kernel estimation and produces erroneous kernels. Cai *et al.* [11] employ the SBI to solve the TV regularization, but they update a latent image once in each loop while we update a latent image till it converges. Since ringing artifacts can be better reduced when latent images converge in each loop, our method can achieve more accurate kernels from the latent images. From the experiment, we demonstrate that by using the TV regularization, our recovered latent images are better than the images recovered by using Tikhonov regularization, thus the blur kernels can be better estimated from the latent images. One visual comparison is shown in Fig. 6(b).

*Comparison of the real-world images*: As shown in Fig. 7, the edges restored by our method are sharper than the edges restored by Ji and Wang's method while the ringing artifacts of our results are less than those of Ji and Wang's results. In Fig. 8, our method also improves the recent work of Harmeling *et al.* [2]. Because our method can recover sharper edges, and then estimates more accurate PSFs from them, the images recovered by our method are more visually pleasant, *e.g.*, the top region in Fig. 8(c). As shown in Fig. 8(b), we can even achieve better results than Hirsch *et al.* [27], who model the motion blur as the PMP model, *e. g.*, the lamp recovered by our method is sharper with less ringing artifacts.

In order to evaluate the deblurring results of the real scenes quantitatively, we use the non-reference metric proposed by Liu *et al.* [42], which is learned based on a massive user study and incorporates features that capture common deblurring artifacts.

TABLE I
QUANTITATIVE COMPARISON WITH JI AND WANG [30]

| Images | Methods | | |
|---|---|---|---|
| | Blurry | [30] | Our method |
| Fig. 7(a) | −12.04 | −9.61 | −9.37 |
| Fig. 7(b) | −12.48 | −9.06 | −8.94 |
| Fig. 7(c) | −12.14 | −9.61 | −8.84 |
| Fig. 7(d) | −11.88 | −10.67 | −9.48 |

TABLE II
COMPARISON WITH HARMELING *ET AL.* [2] AND HIRSCH *ET AL.* [27]

| Images | Methods | | | |
|---|---|---|---|---|
| | Blurry | [2] | [27] | Our method |
| Fig. 8(a) | −14.80 | −9.88 | −9.05 | −8.76 |
| Fig. 8(b) | −11.87 | −9.44 | −9.38 | −8.92 |
| Fig. 8(c) | −12.20 | −9.38 | −8.39 | −8.23 |

To guarantee that the comparison is fair to the other methods, we directly use the results from their corresponding papers or websites. When evaluating the deblurring results by the metric of [42], the larger values mean higher quality of deburred results. The quantitative results of Fig. 7 are shown in Table I, and the quantitative comparisons of Fig. 8 are shown in Table II. From the comparisons, we demonstrate that our deblurring results outperform the results obtained by the other patch-based deblurring methods and the results of the method based on the PMP model [27].

In Fig. 9, Cho *et al.* [41] employ two blurry images, which undergo different motion blur, to recover a clear image. We only show one of the blurry images in Fig. 9. As shown in


Fig. 9. Comparison with the approaches of Cho and Lee [17], Cho *et al.* [41] and the proposed method. (From left to right: the blurry images; results of Cho and Lee; results of Cho *et al.*; and our results.)

Fig. 9, our method achieves sharper images, *e.g.*, the characters in the closeups. For comparison, we also show the results of [17] whose assumption of a invariant motion blur is too restrictive to yield good deblurring results.

Fig. 10 shows that our method improves the recent work of Gupta *et al.*, which models the motion blur as the PMP model. In Fig. 10, we also compare with the algorithm of Xu *et al.* [13], which models the motion blur as the PMP model caused by camera rotations only. The implementation code and the parameters of images are provided in their website [13]. The images recovered by our method are visually more pleasant and exhibit more details.

In Fig. 11, Joshi *et al.* [35] exploit additional inertial sensors to determine the PSFs. In contrast, we are able to estimate the PSFs without extra hardware and obtain comparable if not superior results. As shown in Fig. 11(c), the details are better recovered while the ringing artifacts are less than Joshi *et al.* For comparison, we also show the results of Shan *et al.* [8]. Since [8] assumes that the blur is invariant, it fails to find meaningful kernels.

In Fig. 12(a), Whyte *et al.* [29] exploit two images, of which one is blurry and the other is sharp but noisy, to restore a clear image, while we only use one image. Only the blurry one is shown in Fig. 12(a). In Fig. 12(b), Whyte *et al.* use a single blurry image to restore a clear image. We also compare the algorithm of Xu *et al.* [13] and try our best to tune the parameters of [13] for Fig. 12(a), but the result of [13] shows that the estimated kernel is stuck in a local minimum. As illustrated in Fig. 12, the images recovered by our method are sharper and have less ringing artifacts.

### B. Comparison of the Speed

For all the experiments, we implement our deblurring algorithm on a PC with an Intel Core i7 3.4 GHz CPU and 8 GB RAM. In order to compare with the runtime of the patch-wise deblurring algorithm of Harmeling *et al.* [2], we re-implement the algorithm of Harmeling *et al.*, the processing time of [2] for an image of size $441 \times 611$ is about 25 minutes while our method costs less than 3 minutes. As reported in [30], the processing time for the image of size $512 \times 768$, shown in Fig. 7(b), is about 30 minutes on a 3.0 GHz PC while our method only costs about 3 minutes on a 3.4 GHz PC. To demonstrate the efficiency of our algorithm, we also compare the processing time on different images with Xu *et al.* [13]. We run the Matlab code provided by Xu *et al.* on our PC for time comparisons. As shown in Table III, the run time of our method is less than the method of Xu *et al.*

## V. Discussions

*Parameter setting*: In the TV non-blind deconvolution, $\lambda$ balances the data term and the prior term, and $\gamma$ determines the strength of the constraint. Large $\lambda$ may overfit the blur model. Thus the result is sensitive to image noise and easy to produce ringing artifacts as small gradients are not penalized by the TV regularization. On the other hand, small $\lambda$ will not enforce the enough strength on the data term and leads to an inaccurate estimation where the recovered image will be overly smoothed. $\gamma$ is involved when reformulating the constrained optimization problem (Eqn. (9)) to the unconstrained problem (Eqn. (10)). As illustrated by [43], $\gamma$ should be positive and either large or small values of gamma will lead the iteration to converge slowly. Thus $\gamma$ should be set to an intermediate value. In all our experiments, we fix $\lambda = 1000$ and $\gamma = 5$ and achieve good results.

In our approach, the kernel size and the patch size are required to be determined by users. The kernel size is required to be a little larger than the largest blur in the blurry image, which can be easily estimated. Through two synthetic experiments, we illustrate how the patch size impacts on the estimation of each spatially invariant PSF. We use the different patch sizes which are proportional to the kernel size to estimate the PSFs. The patch number $N_h \times N_v$ is determined by the patch size. Then we divide the whole image into the $N_h \times N_v$ overlapping patches to recover the blurred images, and then evaluate the Peak Signal-to-Noise Ratio (PSNR) of the deblurring results. As shown in Fig. 13, when the patch size is too small, there may not be enough informative edges for estimating the blur kernel, thus the deblurring results are not good judging from the PSNRs. On

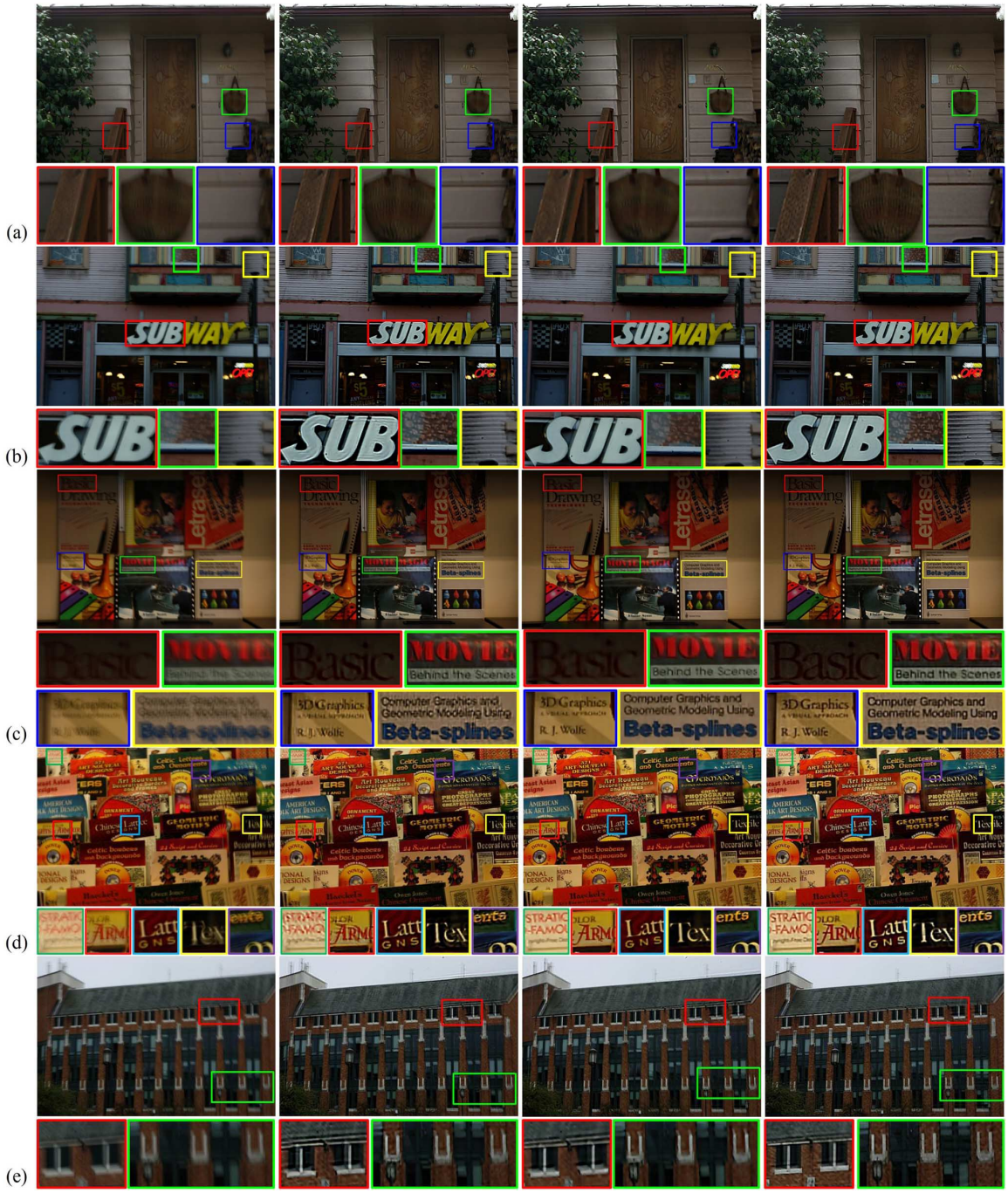



Fig. 10. Comparison with the approaches of Gupta *et al.* [26], Xu *et al.* [13] and the proposed method. (From left to right: the blurry images; results of Gupta *et al.*; results of Xu *et al.*; and our results.)

the other hand, when the patch size is too large, the assumption of the EFF, that a local blurry region shares an identical blur kernel, may not hold any more, thus the corresponding results are not good either. Therefore, our conclusion is that the spatially invariant PSFs will be better estimated when there are enough edges in the patches and the blur kernel is uniform in each patch. In all our experiments, these two criteria can always be satisfied by setting a proper patch size.

*Computation comparison*: In our method, the Split Bregman method is used to estimate the latent sharp images and the blur kernels are solved in the FFT domain. In the latent image estimation, the arithmetic operations of the SBI is $O(Cq\log(q))$, where $C$ is the maximum number of the SBI, and $q$ is the patch size. As shown in Fig. 3, $C$ is less than 30 for most of the cases. Thus we can regard $C$ as a constant and the complexity of the SBI is $O(q\log(q))$. In the blur kernel estimation, only the FFT is used to solve Eqn. (17). The complexity of this step is $O(q\log(q))$. For the PMP model based deblurring algorithm [27], Tikhonov regularization is employed to estimate latent images, which requires the arithmetic operations of $O(q\log(q))$. In the kernel estimation, calculating the projection of the predicted image onto the bases requires operations of $O(N_I \cdot N_d)$,

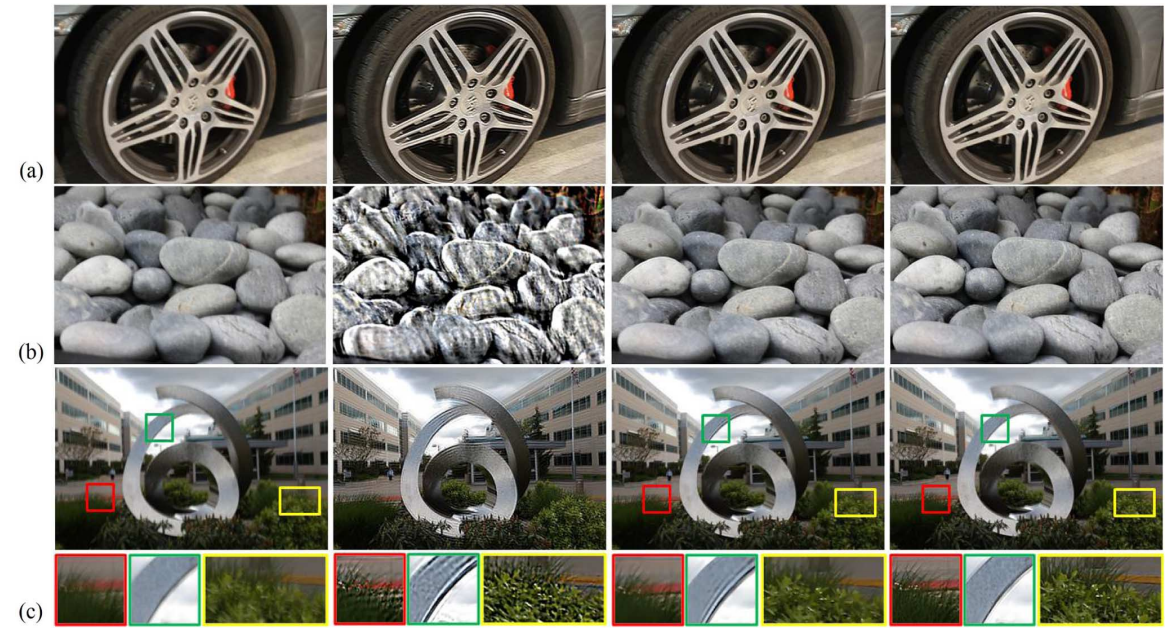



Fig. 11. Comparison with Shan *et al.* [8], Joshi *et al.* [35], and the proposed method. (From left to right: the blurry images; results of Shan *et al.*; results of Joshi *et al.*; and our results.)


Fig. 12. Comparison with Whyte *et al.* [29], Xu *et al.* [13] and the proposed method. (From left to right: the blurry images; results of Whyte *et al.*; results of Xu *et al.*; and our results.)

where $N_I$ is the image size and $N_d$ is the kernel space, *e.g.* $N_d$ is $21^3$ in [27]. To solve the blur kernel, iterative algorithms are required to estimate the blur kernel, such as Conjugate Gradient Method or Quasi-Newton Methods. Therefore, our method requires $O(q\log(q))$ operations while the methods based on PMP models at least require $O(q\log(q))+O(N_I\cdot N_d)$ at least. Therefore, our method can achieve much more efficiency.

*Memory requirement*: When calculating the blur kernel, the PMP model based deblurring algorithms require at least $2N_d\cdot M\cdot U$ Bytes to store the projections, where $M$ is the size of each projection, and $U$ is the unit of data storage. We multiply the factor 2 because of the horizontal and vertical gradients for each projection. This is the most memory-consuming part of these algorithms. However, our patch-wise deblurring algorithm only requires $2M\cdot U/(N_h\cdot N_v)$ Bytes to calculate FFTs for each patch, and is able to be run on memory limited machines.

TABLE III
RUN TIME OF MATLAB IMPLEMENTATION FOR SEVERAL EXAMPLES

| **Size in pixels** | | **Processing time in seconds** | |
|---|---|---|---|
| Image Size | Kernel Size | Xu *et al.* [13] | Our method |
| 273 × 366 | 15 × 15 | 215.9 | 100.7 |
| 441 × 611 | 19 × 19 | 776.2 | 167.4 |
| 512 × 768 | 17 × 17 | 1247.2 | 184.8 |
| 749 × 1123 | 21 × 21 | 2605.3 | 642.3 |

The patch-based deblurring algorithm of Ji and Wang [30] requires $N_I\cdot U_k$ Bytes to store the blur matrix, where $U_k$ is the memory requirement for storing a blur kernel. Our method only requires $N_h\cdot N_v\cdot U_k$ Bytes to store the PSFs. Because the number

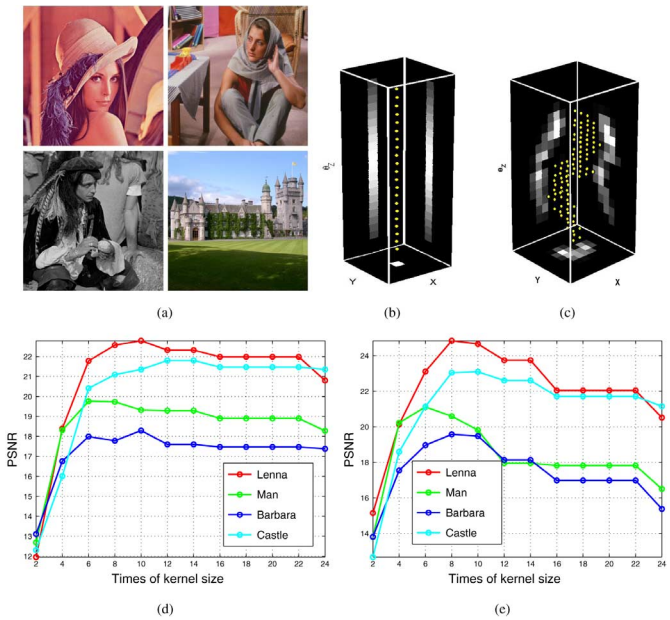



Fig. 13. Illustration of deblurring results with different patch sizes. (a) Four test images (Top left: Lenna; top right: Barbara; bottom left: man; bottom right: castle). (b)–(c) Two blur kernels used for generating the blurred images. (f) Deblurring results of the kernel (b). (g) Deblurring results of the kernel (c).

of the image pixels $N_I$ is much larger than the patch number $N_h \cdot N_v$, *i.e.*, $N_I \gg N_h \cdot N_v$, our method requires less memory than Ji and Wang's algorithm. Furthermore, manipulating the large blur matrix also requires extra large memory in [30]. For example, as shown in Fig. 7(a), Ji and Wang's method requires to store $768 \times 512$ kernels while our method only needs to store 35 kernels.

*Similarity constraint on the PSFs*: When estimating the blur kernels, Harmeling *et al.* [2] employ a similarity constraint on the PSFs. Because of the similarity constraint, the PSFs cannot be derived in a closed form. Harmeling *et al.* employ L-BFGS-B [44] to optimize the space-variant kernels. However, we employ the TV regularization to restore latent images. Thus the blur kernels can be better estimated from the latent images without the similarity constraint. Furthermore, the PSFs can be derived in a closed form.

*Limitations*: Our approach still shares the common limitations with other deblurring methods, such as the depth variations, object motions, noise, saturation regions and the non-linear effects of camera's response functions (CRFs), which may violate the linearity of the blur model and produce inaccurate kernels [45], [46], [47], [48]. We also show two cases of our limitations of our approach in Fig. 14. As shown in Fig. 14, when deblurring an image captured from a scene with depth variations, the PSFs will change abruptly at the boundaries of different depths. The assumption of the EFF that the PSFs vary smoothly is violated. Thus the ringing artifacts will appear at the boundaries of different depths. When the blurry image is contaminated by strong noise, the edges used for estimating blur kernels are damaged by the noise, which leads to erroneous kernels. Similar to the most deblurring methods handling spatially variant blur kernels, we have the

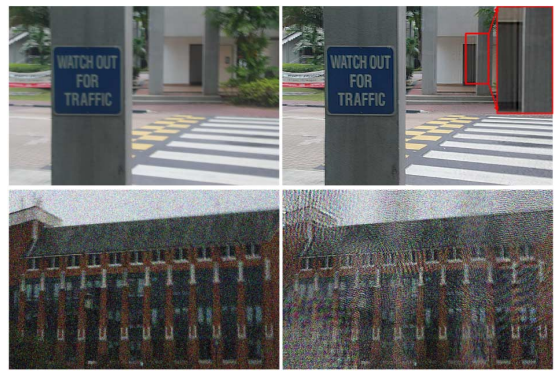



Fig. 14. Limitations. Left: Blurry image with significant depth variations and a noisy and blurry image. Right: Deblurring results.

basic assumption that the blur kernels vary smoothly in the spatial domain and the blur kernel in one region has a high probability to be similar to the PSFs of its neighboring regions. Thus when this assumption fails, *e.g.*, the spatially varying effect is large, the proposed method fails. Another limitation of our work is that the patch number is required to be determined by the users. Determining the patch sizes automatically to achieve better results will be our future work.

## VI. Conclusion

This paper proposes an efficient patch-wise non-uniform deblurring method. It is able to remove spatially variant blur while requiring less computation and memory. We incorporate the TV regularization into the framework of the EFF to better recover latent images while the previous patch-based algorithm [2] employs Tikhonov regularization. Benefited from the TV regularization, the latent images are better recovered, which enable us to estimate the blur kernel locally in a closed form without extra constraints, while the algorithm of [2] requires a global constraint to guarantee the similarity of the neighboring PSFs and the algorithm of [30] needs to interpolate a pixel-wise kernel matrix. Therefore, we require less computation and memory consumption compared to the PMP model-based deblurring algorithms and the patch-based deblurring algorithms. After obtaining the blur kernels, we propose a kernel similarity metric, which takes both the similarity of kernel values and the geometric similarity of neighboring kernels into account, to detect the erroneous kernels. With the help of our proposed metric, we can identify not only the severely divergent kernels but also the erroneous kernels which are stuck in local minima and dissimilar to their neighbors. We also propose a novel edge map to replace erroneous kernels with well estimate ones. The blurry images are better recovered by the refined kernels. Since our algorithm is implemented in the overlapping patches independently, it can be also parallelized easily. From the experiments, our method achieves better results while requiring less computation.

## Acknowledgment

The authors would like to thank the anonymous reviewers and the associate editor for their helpful suggestions and valuable comments.

## REFERENCES


[1] Y. W. Tai, P. Tan, and M. S. Brown, "Richardson-Lucy deblurring for scenes under a projective motion path," *IEEE Trans. Pattern. Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1603–1618, Aug. 2011.

[2] S. Harmeling, M. Hirsch, and B. Scholkopf, "Space-variant single-image blind deconvolution for removing camera shake," in *Proc. NIPS*, 2010, pp. 1–9.

[3] M. Šorel and F. Šroubek, "Space-variant deblurring using one blurred and one underexposed image," in *Proc. ICIP*, 2009, pp. 157–160.

[4] D. Kundur and D. Hatzinakos, "Blind image deconvolution," *IEEE Signal Process. Mag.*, vol. 13, no. 3, pp. 43–64, May 1996.

[5] Y. L. You and M. Kaveh, "A regularization approach to joint blur identification and image restoration," *IEEE Trans. Image Process.*, vol. 5, no. 3, pp. 416–428, Mar. 1996.

[6] A. N. Tikhonov and V. Y. Arsenin, *Solution of Ill-posed Problems*. Washington, DC, USA: V. H. Winston, 1977, vol. 2.

[7] T. F. Chan and C. K. Wong, "Total variation blind deconvolution," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 370–375, Mar. 1998.

[8] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," *ACM SIGGRAPH*, vol. 27, no. 3, pp. 73:1–73:10, 2008.

[9] D. Krishnan, T. Tay, and R. Fergus, "Blind deconvolution using a normalized sparsity measure," in *Proc. CVPR*, 2011, pp. 233–240.

[10] J. F. Cai, H. Ji, C. Q. Liu, and Z. W. Shen, "Blind motion deblurring from a single image using sparse approximation," in *Proc. CVPR*, 2009, pp. 104–111.

[11] J. F. Cai, H. Ji, C. Q. Liu, and Z. W. Shen, "Framelet-based blind motion deblurring from a single image," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 562–572, Feb. 2012.

[12] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," *ACM SIGGRAPH*, vol. 25, no. 3, pp. 787–794, 2006.

[13] L. Xu, S. Zheng, and J. Jia, "Unnatural $L_0$ sparse representation for natural image deblurring," in *Proc. CVPR*, 2013, pp. 1–8.

[14] A. Goldstein and R. Fattal, "Blur-kernel estimation from spectral irregularities," in *Proc. ECCV*, 2012, pp. 622–635.

[15] W. Hu, J. Xue, and N. Zheng, "PSF estimation via gradient domain correlation," *IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 386–392, Jan. 2012.

[16] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Efficient marginal likelihood optimization in blind deconvolution," in *Proc. CVPR*, 2011, pp. 2657–2664.

[17] S. Cho and S. Lee, "Fast motion deblurring," *ACM SIGGRAPH Asia*, vol. 28, no. 5, pp. 145:1–145:10, 2009.

[18] L. Xu and J. Jia, "Two-phase kernel estimation for robust motion deblurring," in *Proc. ECCV*, 2010, pp. 157–170.

[19] T. S. Cho, S. Paris, B. K. P. Horn, and W. T. Freeman, "Blur kernel estimation using the radon transform," in *Proc. CVPR*, 2011, pp. 241–248.

[20] T. S. Cho, C. L. Zitnick, N. Joshi, S. B. Kang, R. Szeliski, and W. T. Freeman, "Image restoration by matching gradient distributions," *IEEE Trans. Pattern. Anal. Mach. Intell.*, vol. 34, no. 4, pp. 683–694, Apr. 2012.

[21] L. Zhong, S. Cho, D. Metaxas, S. Paris, and J. Wang, "Handling noise in single image deblurring using directional filters," in *Proc. CVPR*, 2013, pp. 1–8.

[22] A. Levin, R. Fergus, F. Durand, and W. T. Freeman, "Image and depth from a conventional camera with a coded aperture," *ACM SIGGRAPH*, vol. 26, no. 3, pp. 70:1–70:9, 2007.

[23] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum, "Progressive inter-scale and intra-scale non-blind image deconvolution," *ACM SIGGRAPH*, vol. 27, no. 3, pp. 74:1–74:10, 2008.

[24] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-Laplacian priors," in *Proc. NIPS*, 2010.

[25] U. Schmidt, C. Rother, S. Nowozin, J. Jancsary, and S. Roth, "Discriminative non-blind deblurring," in *Proc. CVPR*, 2013, pp. 1–8.

[26] A. Gupta, N. Josh, Z. C. Lawrence, M. Cohen, and B. Curless, "Single image deblurring using motion density functions," in *Proc. ECCV*, 2010, pp. 171–184.

[27] M. Hirsch, C. J. Schuler, S. Harmeling, and B. Schölkopf, "Fast removal of non-uniform camera shake," in *Proc. ICCV*, 2011, pp. 463–470.

[28] Z. Hu and M. H. Yang, "Fast non-uniform deblurring using constrained camera pose subspace," in *Proc. Brit. Mach. Vision Conf.*, 2012.

[29] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce, "Non-uniform deblurring for shaken images," *Int. J. Comput. Vision*, vol. 98, no. 2, pp. 168–186, 2012.

[30] H. Ji and K. Wang, "A two-stage approach to blind spatially-varying motion deblurring," in *Proc. CVPR*, 2012, pp. 73–80.

[31] M. Hirsch, S. Sra, B. Schölkopf, and S. Harmeling, "Efficient filter flow for space-variant multiframe blind deconvolution," in *Proc. CVPR*, 2010, pp. 607–614.

[32] Y.-W. Tai, H. Du, M. S. Brown, and S. Lin, "Image/video deblurring using a hybrid camera," in *Proc. CVPR*, 2008, pp. 1–8.

[33] Y. W. Tai, H. Du, M. S. Brown, and S. Lin, "Correction of spatially varying image and video motion blur using a hybrid camera," *IEEE Trans. Pattern. Anal. Mach. Intell.*, vol. 32, no. 6, pp. 1012–1028, Jun. 2010.

[34] M. Ben-Ezra and S. K. Nayar, "Motion based motion deblurring," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 689–698, Jun. 2004.

[35] N. Joshi, S. B. Kang, C. L. Zitnick, and R. Szeliski, "Image deblurring using inertial measurement sensors," *ACM SIGGRAPH*, vol. 29, no. 4, pp. 30:1–30:9, 2010.

[36] Y. Bando, H. Holtzman, and R. Raskar, "Near-invariant blur for depth and 2D motion via time-varying light field analysis," *ACM SIGGRAPH*, vol. 32, no. 2, pp. 1–15, 2013.

[37] T. G. S. , Jr., "High-speed convolution and correlation," in *Proc. Joint Comput. Conf.*, 1966, pp. 229–233.

[38] T. Goldstein and S. Osher, "The split Bregman method for l1-regularized problems," *SIAM J. Imag. Sci.*, vol. 2, no. 2, pp. 323–343, 2009.

[39] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. ICCV*, Jul. 2001, vol. 2, pp. 416–423.

[40] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Proc. CVPR*, 2009, pp. 1964–1971.

[41] S. Cho, H. Cho, Y.-W. Tai, and S. Lee, "Registration based non-uniform motion deblurring," *Comput. Graphics Forum*, vol. 31, no. 7, pp. 2183–2193, 2012.

[42] Y. Liu, J. Wang, S. Cho, A. Finkelstein, and S. Rusinkiewicz, "A no-reference metric for evaluating the quality of motion deblurring," in *Proc. ACM SIGGRAPH Asia*, 2013, vol. 32, no. 175, pp. 1–12.

[43] G. Pascal, "Rudin-Osher-Fatemi total variation denoising using split Bregman," *Image Process. On Line*, vol. 2012, pp. 74–95, 2012.

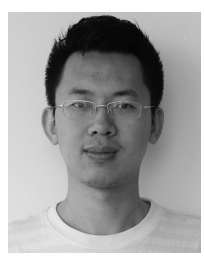
[44] R. H. Byrd, R. H. Byrd, P. Lu, P. Lu, J. Nocedal, J. Nocedal, C. Zhu, and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM J. Sci. Comput.*, vol. 16, pp. 1190–1208, 1994.

[45] L. Xu and J. Jia, "Depth-aware motion deblurring," in *Proc. ICCP*, 2012, pp. 1–8.

[46] Y. W. Tai and S. Lin, "Motion-aware noise filtering for deblurring of noisy and blurry images," in *Proc. CVPR*, 2012, pp. 17–24.

[47] S. Cho, J. Wang, and S. Lee, "Handling outliers in non-blind image deconvolution," in *Proc. ICCV*, 2011, pp. 495–502.

[48] S. Kim, Y. W. Tai, S. J. Kim, M. S. Brown, and Y. Matsushita, "Nonlinear camera response functions and image deblurring," in *Proc. CVPR*, 2012, pp. 25–32.

**Xin Yu** received the B.S. degree in electronic engineering from University of Electronic Science and Technology of China, Chengdu, China, in 2009. He is currently pursuing the Ph.D. degree from the Department of Electronic Engineering, Tsinghua University, Beijing, China. His interests include computer vision and image/video processing.

**Feng Xu** received the B.E. degree in physics from Tsinghua University, Beijing, China, in 2007, and the Ph.D. degree in automation from Tsinghua University, Beijing, China, in 2012. He is currently working as an associate researcher with Microsoft Research, Beijing, China. His research interests include image/video processing, computer vision, and computer graphics.

**Shunli Zhang** received the B.S. and M.S. degrees in electronics and information engineering from Shandong University, Jinan, China, in 2008 and 2011, respectively. He is currently pursuing the Ph.D. degree from the Department of Electronic Engineering of Tsinghua University, Beijing, China. His research interests include pattern recognition, computer vision, and image processing.

**Li Zhang** received the B.S., M.S., and Ph.D. degrees in signal and information processing from Tsinghua University, Beijing, China, in 1987, 1992, and 2008, respectively. From 1987 to 1989, he taught physics experiments in the Department of Physics at Tsinghua University, Beijing, China. In 1992, he joined the faculty of the Department of Electronic Engineering at Tsinghua University, Beijing, China, where he is currently a professor. His research interests include image processing, computer vision, pattern recognition, and computer graphics.