Article

Robust Image Restoration for Motion Blur of Image Sensors

Fasheng Yang [1,2,3]**, Yongmei Huang** [1,2,*]**, Yihan Luo** [1,2]**, Lixing Li** [1,2,3] **and Hongwei Li** [1,2,3]

[1] Institute of Optics and Electronics, Chinese Academy of Sciences, P.O. Box 350, Shuangliu, Chengdu 610209, China; yangfs_ioe@163.com (F.Y.); luo.yihan@foxmail.com (Y.L.); icecliff@126.com (L.L.); zkygds@126.com (H.L.)
[2] Key Laboratory of Optical Engineering, Chinese Academy of Sciences, Chengdu 610209, China
[3] University of Chinese Academy of Sciences, 19 A Yuquan Rd, Shijingshan District, Beijing 100039, China
[*] Correspondence: huangym@ioe.ac.cn; Tel.: +86-132-1908-5337

Abstract: Blind image restoration algorithms for motion blur have been deeply researched in the past years. Although great progress has been made, blurred images containing large blur and rich, small details still cannot be restored perfectly. To deal with these problems, we present a robust image restoration algorithm for motion blur of general image sensors in this paper. Firstly, we propose a self-adaptive structure extraction method based on the total variation (TV) to separate the reliable structures from textures and small details of a blurred image which may damage the kernel estimation and interim latent image restoration. Secondly, we combine the reliable structures with priors of the blur kernel, such as sparsity and continuity, by a two-step method with which noise can be removed during iterations of the estimation to improve the precision of the estimated blur kernel. Finally, we use a MR-based Wiener filter as the non-blind deconvolution algorithm to restore the final latent image. Experimental results demonstrate that our algorithm can restore large blur images with rich, small details effectively.

1. Introduction

Motion blur widely exists in digital photography and leads to disappointing blurry images with inevitable information loss. Due to the mechanism of image sensors that integrate incoming lights for an amount of time to produce images, if a relative motion happens between the subject and the image sensors during the integration time, a blurred image will be produced as shown in Figure 1.
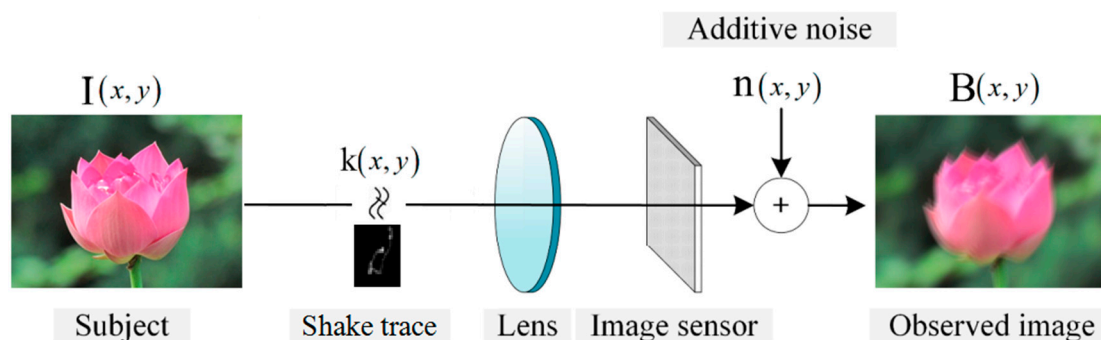

Figure 1. Image degradation model for motion blur of image sensor.

Motion deblurring has been deeply researched in computer vision and image processing due to its involvement of many challenges in problem formulation, regularization, and optimization. The motion blur produced by camera shake is generally modeled as:

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

where B, I, k, and n represent the blurred image, latent sharp image, blur kernel (also known as point spread function, PSF), and additive noise, respectively. $\otimes$ is the convolution operator. The blur kernel delineates motion trace between the subject and image sensors. Generally, the size of the blur kernel is relatively smaller than that of the latent sharp image and its value is non-negative. Then, estimating the latent sharp image I and blur kernel k from the blurred image B is a severely ill-conditioned inverse problem, which requires regularization to alleviate its ill-conditionness and stabilize the solution.

Single-image motion debluring can be roughly categorized into two groups: non-blind deconvolution and blind deconvolution. In non-blind deconvolution, both the blurred image and blur kernel are known and the latent sharp image is restored from the blurred image using the blur kernel. Early famous algorithms such as Richardson-Lucy (R.L.) [1] and Weiner filter [2] which are simple and efficient are still widely used. However, these algorithms may produce ringing artifacts near strong edges inevitably due to the drawback of being sensitive to noise. Recently, some robust motion deblurring algorithms have been proposed [3–6].

In blind deconvolution, the blur kernel is unknown and restoring the latent sharp image becomes more challenging. Most blind deconvolution algorithms commonly perform with some estimation criterions, such as maximum *a posteriori* (MAP) or Bayesian, and some priors of the latent sharp image or blur kernel to restore them from the blurred image. Previous algorithms, firstly, impose constraints on the motion blur kernel, such as sparse prior [3], Gaussian prior [7], or color prior [8]. After obtaining the blur kernel, these algorithms [3,7,8] restore the latent sharp image by non-blind deconvolution approaches.

In this paper, we introduce a robust blind motion restoration algorithm estimating the blur kernel and latent sharp image iteratively and hierarchically. As is reported [7], only reliable structures can contribute to kernel estimation, we firstly construct a pyramid representation of the blurred image by down-sampling with factor of $\sqrt{2}$ and extract the reliable structures according to the characteristics of the blurred image in each level. Then we estimate the blur kernel and latent sharp image in each level alternately. In order to eliminate the damage of noise to kernel estimation, which further conduce unreliable deblurred results, we combine some priors of the blur kernel with the reliable structures of the blurred image to estimate the blur kernel. For latent sharp image estimation, we add a spatial prior to suppress the noise and ringing artifacts. After obtaining the blur kernel and latent sharp image in a coarse level, we up-sample them to a finer level and reiterate the process above until reaching the finest level.

The contributions of this paper can be listed as follows:

1. In order to eliminate the effect of noise and ambiguous structures which may damage kernel estimation, we propose a novel structure extraction method with which the reliable structures can be selected adaptively and effectively;
2. As motion blur kernel is sparse and delineates the motion trace between the subject and image sensors, we introduce a two-step method for the kernel estimation process to eliminate the noise and guarantee the sparsity and continuity.

The rest of this paper is organized as follow: in Section 2 we review related works, the proposed kernel estimation and non-blind deconvolution algorithm are described in Section 3, experimental results are illustrated in Section 4 and, finally, we conclude our algorithm in Section 5.

2. Related Work

Blurred image restoration is a fundamental problem in enhancing images acquired by various types of image sensors [9–12]. Although various image sensors' signal processing techniques have been proposed, restoration of blurred images modeled in Equation (1) is still a challenging task because of the latent sharp image and blur kernel are highly unconstrained and there is no unique combination of them whose convolution is equal to the blurred image. Previous works generally impose constraints on the blur kernel and represent it in a simple parametric form. However, as is shown [3], the real blur kernels are too complicated to be formulated in a simple parametric form. Fergus *et al.* [3] used a zero-mean Mixture of Gaussian to approximate the gradients of nature image and proposed a variational Bayesian inference algorithm to deblur images. Shan *et al.* [4] used a series of optimization techniques to avoid trivial solutions and so be robust to noise. Krishnan *et al.* [13] introduced a new normalized sparsity as the regularization term into their MAP framework to estimate the blur kernel. Xu *et al.* [14] introduced an image decomposition scheme to make image deblurring process more robust. Levin *et al.* [15] showed the limitation of the naive MAP approach and suggested estimating the MAP of the blur kernel alone while marginalizing over the latent sharp image in [16]. Oh *et al.* [17] adopted a piecewise-linear model to approximate the curves for the blur kernel estimation. Shao *et al.* [18] impose a type of non-stationary Gaussian prior on the gradient fields of sharp images to estimate the blur kernel. Although extensive works have been done, the estimated kernels still contain some noise, and selecting a kernel with a hard threshold will destroy the intrinsic structures of the blur kernel.

Another group of algorithms [6–8,19] use predicted sharp edges to estimate the blur kernel. Money and Kang [19] used a shock filter to sharpen edges. Joshi *et al.* [8] used edge profiles instead of shock filtering, but those methods only work for small blurs. Cho and Lee [6] combined a bilateral filter with the shock filter to predict sharp edges from the blurred image iteratively. However, as described in the paper, that method could not predict sharp edges correctly for large blurs. Xu and Jia [7] introduced a mask to select useful gradients of the blurred image for kernel estimation. Although the performance improved greatly, the continuity and sparsity of the blur kernels still cannot be guaranteed and the estimated kernels still occasionally contain some noise.

With a known blur kernel, the blind deconvolution problems can be solved by the non-blind deconvolution approaches. However, as the latent sharp image restoration is very sensitive to noise and may produce some undesirable artifacts, the non-blind deconvolution is still ill-conditioned. Many works [4,20,21] have been conducted to overcome this problem. Levin *et al.* [20] used a heavy-tailed function to alleviate the ringing artifacts, which was based on the model of sparse image derivatives distribution. Yuan *et al.* [21] proposed a multi-scale bilateral Richardson-Lucy algorithm to reduce ringing artifacts. Shan *et al.* [4] adopted a local smoothness prior to suppress the artifacts in smooth regions.

3. Single Image Blind Deconvolution Using Reliable Structure

Many algorithms have been proposed to restore the latent sharp image using image structures and the deblurred results depend much on the reliability of the extractive structures. We propose a new method to extract the reliable structures, which will be discussed in Section 3.1. The blur kernel estimated by existing methods usually contain some noise which may damage the latent sharp image estimation. We introduce a two-step method which can guarantee the continuity and sparsity of the blur kernel to eliminate the noise in Section 3.2. The latent sharp image restoration method will be discussed in Sections 3.3 and 3.4. And we give the multi-scale implementation of our algorithm in Section 3.5. The overview framework of our approach is shown in Figure 2, and the deblurred result of a synthetic blurred image is shown in Figure 3.
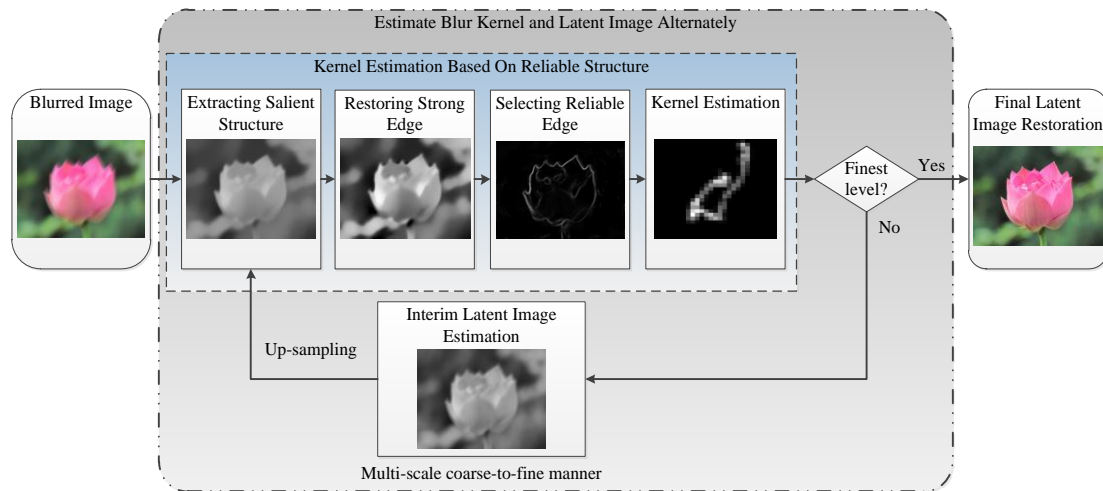
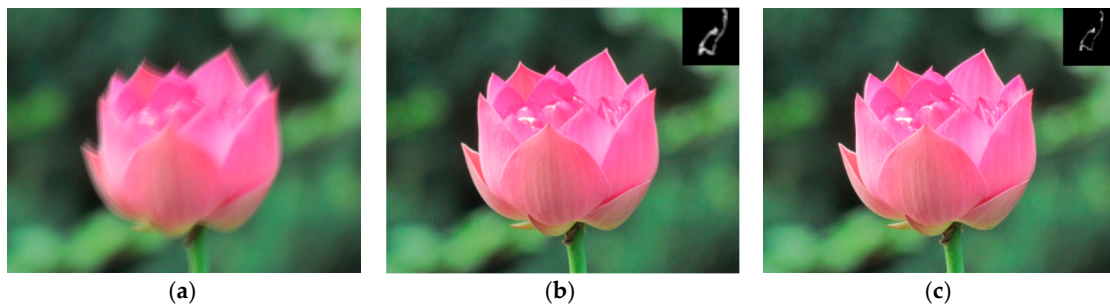**Figure 2.** The flowchart of the proposed blind deconvolution algorithm.

Figure 3. The restoration of a synthetic blurred image. (**a**) blurred image; (**b**) restored image and estimated blur kernel; and (**c**) ground truth image and blur kernel.

3.1. Reliable Structure Extraction from Blurred Image

As is discussed in [7], the structures of the blurred image do not always improve the blur kernel estimation. On the contrary, the edges whose size are smaller than that of the blur kernel will deteriorate the blur kernel estimation. Inspired by the total variation-based noise removal algorithm [22], we treat the textures and small details as "noise" and introduce a salient structure extraction method to separate the helpful structures of the blurred image for the blur kernel estimation from the detrimental textures and small details. The energy function is formulated as follow:

$$E\left(I_s\right) \;=\; \frac{1}{2\theta M(B)}\,||\,I_s\,-\,I\,||_2^2\,+\,||\,I_s\,||_{TV2} \tag{2}$$

where $||\,I_s\,||_{TV2}$ is the isotropic TV norm of salient structure I_s defined as Equation (3), which can preserve edges.

$$||\,I_s\,||_{TV2} = \int_{p\in I_s} \sqrt{\partial x I_{sp}{}^2\,+\,\partial y I_{sp}{}^2}\,dp \tag{3}$$

where M is the gradient attenuation function [23] defined as Equation (4), which is a self-adaptive weight for textures and small details attenuation:

$$M\left(L\right) \;=\; \frac{\alpha_s}{||\nabla L\left(x,y\right)||}\left(\frac{||\nabla L\left(x,y\right)||}{\alpha_s}\right)^\beta \tag{4}$$

where L is the image to be measured and the parameter, α_s controls the gradient magnitudes which remain unchanged, and β determines how much the lager magnitude will be attenuated (assuming β < 1), while gradients of magnitude smaller than α_s are slightly magnified. As $M(B)$ is the self-adaptive weight in Equation (2), the salient structures will be kept more while the flat areas and narrow strips will be smoothed, so the salient structure I_s can be obtained by minimizing Equation (2). We set the coarsest version of the blurred image B as the initial value of the latent image I for Equation (2).

We empirically set β between 0.8 and 0.9, and α_s to s times the average gradient magnitude of the blurred image in each level. Figure 4 show the self-adaptive weight $M(B)$ with different parameters and the corresponding extracted salient structures, a larger α_s will give a strong attenuation to the smooth regions while a smaller α_s will preserve more details.
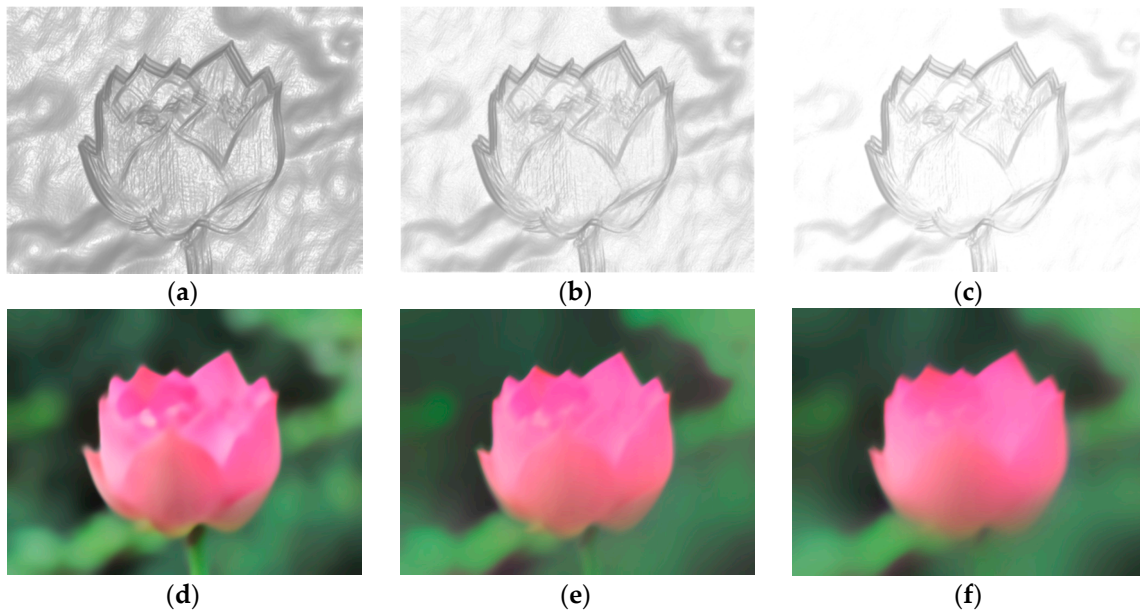

| (a) | (b) | (c) |
| (d) | (e) | (f) |

Figure 4. The influence of α_s on extracted salient structure. (**a**) $s = 0.2$; (**b**) $s = 0.5$; (**c**) $s = 0.8$; (**d**) extracted salient structure relevant to $s = 0.2$; (**e**) extracted salient structure relevant to $s = 0.5$; and (**f**) extracted salient structure relevant to $s = 0.8$.

After obtaining the salient structure I_s, we enhance it by the the shock filter [24] to recover strong edges:

$$\partial \widetilde{I_s}/\partial t \;=\; -sign\left(\Delta I_s\right) ||\nabla I_s||_2 \tag{5}$$

where t is the evolution time of the shock filter, ΔI_s and ∇I_s are the Laplacian and gradient of I_s, respectively. The enhanced structure $\widetilde{I_s}$ contains not only the sharp edges but also the enhanced noise. In order to remove the noise, we use a mask to select reliable structure S, which will be used to estimate the blur kernel:

$$\nabla S \;=\; \nabla \widetilde{I_s} H\left(\tau \,-\, M(I_s)\right) \tag{6}$$

where $H(\cdot)$ denotes the Heaviside step function which outputs ones for positive values and zeros otherwise, and M is defined as Equation (4). As $M(I_s)$ is large in smooth regions, while small near strong edges, and $M(\widetilde{I_s})$ is small near not only strong edges, but also enhanced noise, one can set an appropriate value of the threshold τ, referring to the value of $M(\widetilde{I_s})$ near strong edges and the differences between the value of $M(I_s)$ and $M(\widetilde{I_s})$ in smooth regions to eliminate the noise and obtain the reliable structures.

It is known that the less the salient edges are used in kernel estimation, the more unreliable the estimated blur kernel is. We take the following strategies to guarantee the reliability of the estimated blur kernel:

Firstly, as the initial value of τ is critical to kernel estimation [7], we take the method of [6] to set the value of τ adaptively at the beginning of the iterative restoration process. Four directions of the image gradients are taken into account to guarantee enough information of the salient edges are used to estimate the blur kernel. Additionally, the value of τ for later iterations is set to allow that at least $0.5\sqrt{P_I P_k}$ pixels take part in the kernel estimation in each group. P_k and P_I are the total number of pixels of the blur kernel and input image, respectively.

Secondly, as more edges are needed to estimate the blur kernel in higher level of the pyramid, the parameters θ and α_s are decreased to bring more edge information into the kernel estimation. Our strategies allow the recovery of fine structures during kernel refinement.

Figure 5d–f show some interim ∇S maps in different levels. It is obvious that the higher the level is, the more the sharp edges participate in kernel estimation.
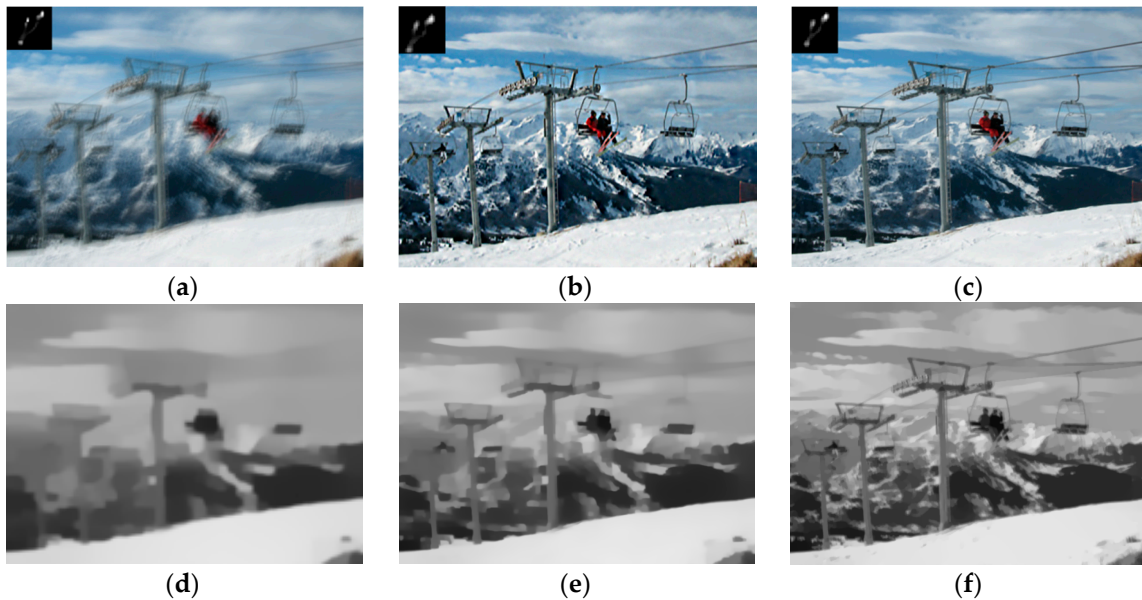

Figure 5. Results comparison of [7] and our method. (**a**) Blurred image and truth kernel; (**b**) result of [7]; (**c**) result of our method; and (**d**–**f**) interim ∇S maps with our method. Our result shown in (**c**) is better.

3.2. Kernel Estimation and Refinement

As motion blur ascribes to the relative motion between the subject and image sensors within the exposure time period, the blur kernel delineates the motion trace between them and should be continuous and sparse. We employ a two-step method to guarantee the sparsity and continuity, respectively.

Estimation. We combine the strictly-selected edges ∇S with a Hyper-Laplacian prior regularization term by the MAP estimation criterion to estimate the blur kernel with sparsity. The energy function is formulated as follow:

$$E(k) = \sum_{S_*, B_*} \omega_* ||S_* \otimes k - B_*||_2^2 + \zeta ||k||_\gamma^\gamma \tag{7}$$

$$\text{s.t.} \ k(x, y) \geqslant 0, \ \sum_{(x,y)} k(x, y) = 1 \tag{8}$$

where ω_* is the weight for each partial derivative, ζ is the weight for the Hyper-Laplacian regularization term with $0 < \gamma < 1$. S_* and B_* are the partial derivatives of the selected reliable structures and blurred image.

$$(S_*,\ B_*) \in \left\{ (\nabla S_x,\ \partial_x B),\ (\nabla S_y,\ \partial_y B),\ (\partial_x \nabla S_x,\ \partial_{xx} B),\ (\partial_y \nabla S_y,\ \partial_{yy} B) \right\} \tag{9}$$

We run the constrained iterative reweighted least square (IRLS) method [14] two iterations to minimize Equation (7) and use CG method for the inner IRLS system.

Refinement. As the estimated blur kernel may contain some noise which may deteriorate the following the estimated interim latent images and kernels (e.g., the deblurred image and estimated kernel shown in Figure 6), we eliminate the noise by checking the pixels' continuity of the estimated blur kernel. The energy function is defined as follow, which can be minimized using the alternative optimization method [25].

$$E\left(\widetilde{k}\right) = \sum_{p \in \widetilde{k}} \left(\widetilde{k}_p - k_p\right)^2 + \lambda C\left(\widetilde{k}\right) \tag{10}$$

where λ is the weight for the continuity constrained term which is defined as follow:

$$C\left(\widetilde{k}\right) = \left\{ p \left| \left|\partial_x \widetilde{k}_p\right| + \left|\partial_y \widetilde{k}_p\right| \neq 0 \right. \right\} \tag{11}$$

During the kernel estimation process, the two steps above are taken alternately three times (Itr = 3) with $\gamma = 0.5$, empirically, in each level of the pyramid. The value of λ can be set refer to the size of the blur kernels. Algorithm 1 illustrates the blur kernel estimation algorithm.

Algorithm 1. Blur Kernel Estimation.

Input: Blurred image, reliable structures and the initial value of *k* from previous iterations or previous level;

for *n* = 1 to Itr (Itr: number of iterations) **do**
 Estimate the blur kernel using the reliable structures. (Equation (7))
 Refine the blur kernel by solving Equations (10) and (11) alternately.
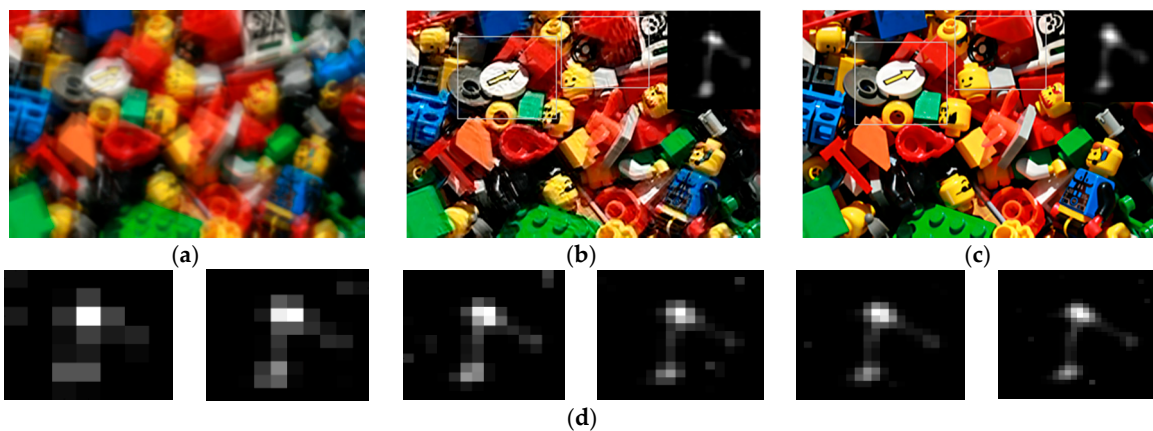 $k \leftarrow \widetilde{k}$;
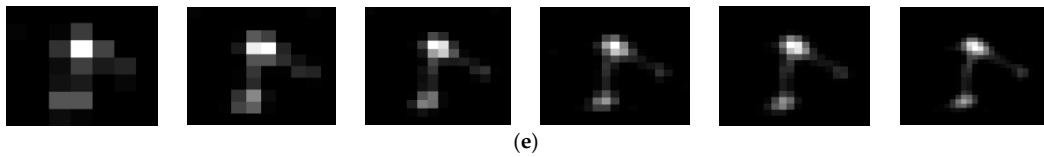 **end for**

Output: Blur kernel *k*.

(a)   (b)   (c)

(d)

Figure 6. *Cont.*

(**e**)

Figure 6. Comparison of results with and without the refinement step using same selected ∇S maps. (**a**) Blurred image; (**b**) estimated sharp image and blur kernel without the refinement step; (**c**) estimated sharp image and blur kernel with the refinement step; (**d**) show the iterations of the blur kernel estimation without the refinement step; and (**e**) show the iterations of the blur kernel estimation with the refinement step. Without the refinement step, the estimated blur kernel contains some noise which may deteriorate the deblurred image. Restored image in (c) and blur kernel in (e) outperform the results in (b,d).

We use the dataset of [15] to testify the effectiveness of our kernel estimation method, and the comparison of the estimated kernels between some previous works and our algorithm is illustrated in Figure 7. Benefiting from both the reliable structures and two-step estimation method, our kernel estimation results perform better. Then we adopt the SSDE (sum of squared differences error, which is defined as Equation (12)) to evaluate the accuracy of the estimated kernels shown in Figure 7 and give the results comparison in Figure 8.

$$SSDE = \sum_{(x,y)} \left[k_{GT}(x,y) - k(x,y) \right]^2 \tag{12}$$

where k and k_{GT} denote the estimated blur kernel and ground truth blur kernel respectively.
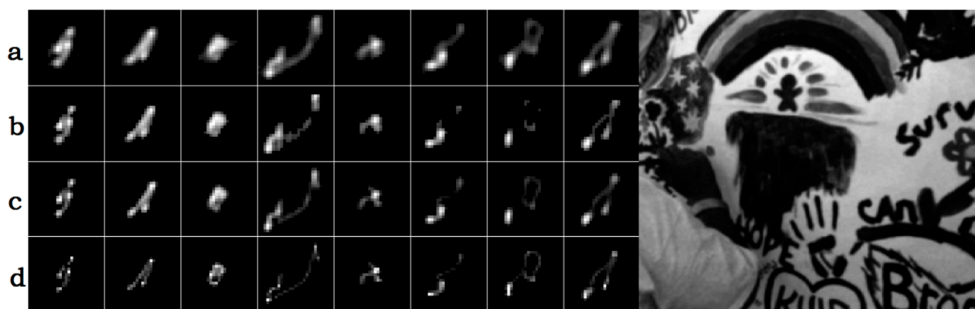

Figure 7. Comparison of the estimated kernels. (**a**) results of Fergus *et al.* [3]; (**b**) results of Xu and Jia [7]; (**c**) our results; and (**d**) the ground truth blur kernels.
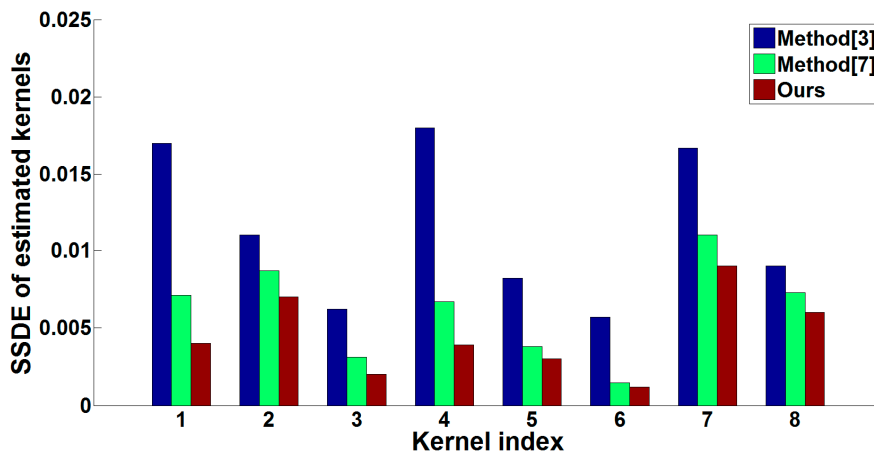

Figure 8. Comparison of the estimated kernels in terms of SSDE.

3.3. Interim Latent Image Restoration

As attention is paid on recovering the salient edges during the interim latent image restoration process, we use the strictly selected edges ∇S as a spatial prior to restore the coarse version of the latent image. We minimize the following energy function to obtain the latent image:

$$E(L) = ||L \otimes k - B||_2^2 + \kappa ||\nabla L - \nabla S||_2^2 \tag{13}$$

where κ is the weight for the spatial prior regularization term which can suppress the noise and avoid ringing artifacts. We employ FFTs on Equation (13) and take the partial derivative with respect to k to zero. The closed-form of the latent image is given by Equation (14):

$$L = F^{-1} \left(\frac{\overline{F(k)} \cdot F(B) + \kappa \left(\overline{F(\partial_x)} \cdot F(\nabla S_x) + \overline{F(\partial_y)} \cdot F(\nabla S_y) \right)}{F(k)^2 + \kappa \left(F(\partial_x)^2 + F(\partial_y)^2 \right)} \right) \tag{14}$$

where F and F^{-1} denote the forward and inverse Fourier transforms, respectively, and $\overline{F(*)}$ is the complex conjugate of $F(*)$.

3.4. Final Non-Blind Deconvolution

After obtaining the blur kernel, the final latent image will be restored from the full-scale blurred image, which contains more noise and the process is time consuming. In order to achieve high robustness and processing speed, we adopt the MR-based Wiener filter [26] to restore the final latent sharp image in frequency domain, whose transfer function is formulated as follow:

$$F(L) = \frac{\overline{F(k)} \cdot F(B)}{|F(k)|^2 + \Gamma} \tag{15}$$

where Γ is defined as:

$$\Gamma = |F(k)(u_{MH}, 0)|^{2\eta} \cdot \left[\max_{u \in D_T} |F(k)(u, 0)| \right]^{2(1-\eta)} - \left[\min_{u \in D_T} |F(k)(u, 0)| \right]^2 \tag{16}$$

The symbols D_T and u_{MH} denote the cut-off frequency and boundary between the high-frequency and mid-frequency regions of the blur kernel frequency spectrum $F(k)(u, v)$, respectively, and (u, v) is the index in frequency domain. The values of them are calculated with the method proposed in that paper. The parameter η controls the compromise between image details recovery and noise suppression, and we set $\eta = 0.8$ in our experiment.

3.5. Multi-Scale Implementation

In order to deal with large blur kernels and make the restoration algorithm effectively and efficiently, we build a coarse-to-fine pyramid of images with a down-sampling factor of $\sqrt{2}$ to estimate the blur kernel in multi-scale resolution. The number of pyramid levels is determined by the size of the blur kernel such that the width or height of the blur kernel at the coarsest level is about 3–7 pixels. The blur kernel and interim latent image are estimated alternately for a few iterations at each level. After obtaining the full scale blur kernel, the final latent sharp image is restored by using a MR-based Wiener filter. Algorithm 2 outlines our approach.

Algorithm 2. Overall Algorithm.

Input: Blurred image B, parameters θ, α_s and the size of blur kernel;
       Build an image pyramid $\{B_s\}$ and all-zero kernel pyramid $\{k_s\}$ with level index $\{1, 2, \ldots, n\}$ according
        to the size of blur kernel;

1. Blind estimation of blur kernel
   for = 1 to n **do**
     Compute adaptive weight $M(B_s)$ (Equation (4)).
     for i = 1 to m (m iterations) **do**
       Extract salient structure I_s (Equation (2)).
       Select reliable structure ∇S for kernel estimation (Equation (6))
       Estimate blur kernel according to Algorithm 1.
       Restore interim latent image L (Equation (14))
         $\theta \leftarrow \theta/1.1$, $\alpha_s \leftarrow \alpha_s/1.1$
     end for
     Up-sample latent image: $L_{l+1} \leftarrow L_l \uparrow$.
     Porject k_l onto the constraints (Equation (8)) and up-sample blur kernel: $k_{l+1} \leftarrow k_l \uparrow$.
   end for
2. Image restoration using MR-based Wiener Filter.
 -Recover I using k from B in full-scale resolution(Equation (15))

Output: Blur kernel k and latent sharp image I.

---

4. Experiments

In order to prove the effectiveness of our algorithm, we compare it with several state-of-art approaches on synthetic and real blurred images. Here, we give some implementation details. Before kernel estimation, we convert all color images to grayscale ones and experimentally set the initial value of θ to 1 based on numerous experiments. The initial value of α_s is set to 0.8 times the average gradient magnitude of the coarsest version of the blurred image. The value of θ and α_s in higher levels are obtained by dividing the value of them in the previous level by 1.1. During kernel estimation, before up-sampling the blur kernel estimated from the previous level, the negative elements of the blur kernel will be set to 0 and renormalized. In Algorithm 2, the iteration time m is set to 5, empirically. We deblur each color channel respectively in the final non-blind restoration. All experiments test on a PC running MS Windows 7 64 bit version with Intel Core i5 560 M CPU, 8 GB RAM and implementation platform is MATLAB 2014a.

4.1. Experimental Results and Evaluation

It is known that the more textures and small details in the blurred image, the harder the restoration. We, firstly, give a synthetic example shown in Figure 9 which contains rich textures and small details, such as flowers and leaves, to prove the robustness and effectiveness of our algorithm. The deblurred result of Fergus *et al.* [3] and Shan *et al.* [4] still contain some blur and ringing artifacts due to the inaccurate kernel estimation. The approach of Jia *et al.* [7] performs better, but the imperfect estimated kernel also leads to some artifacts in the restored result. In contrast, our algorithm gives the best performance on both the estimated blur kernel and final restored image shown in Figure 9e.

Table 1. Comparison of the results in Figure 8.

Algorithm	[3]	[4]	[7]	Ours
PSNR of images	7.6775	9.8314	12.2182	18.7644
SSDE of kernels	0.2859	0.0434	0.0286	0.0025

In Table 1, we adopt PSNR (Peak Signal to Noise Ratio, which is defined as Equation (17)) and SSDE to give a quantification comparison of the estimated blur kernels and deblurred images in Figure 8, respectively. As shown in Table 1, our algorithm gives a higher PSNR value for the deblurred image and lower SSDE value for the estimated blur kernel.

$$PSNR \;=\; 10\log_{10}\left\{ \frac{R^2 \;\times\; MN}{\sum\limits_{(x,y)} \left[I_{GT}(x,\,y)\;-\;I(x,\,y)\right]^2} \right\} \tag{17}$$

where I and I_{GT} are the restored latent sharp image and ground truth image, respectively, whose width is M and height is N. The symbol R denotes the maximum value of the input image data type, e.g., $R = 255$ when the images have 8-bit unsigned integer data type.
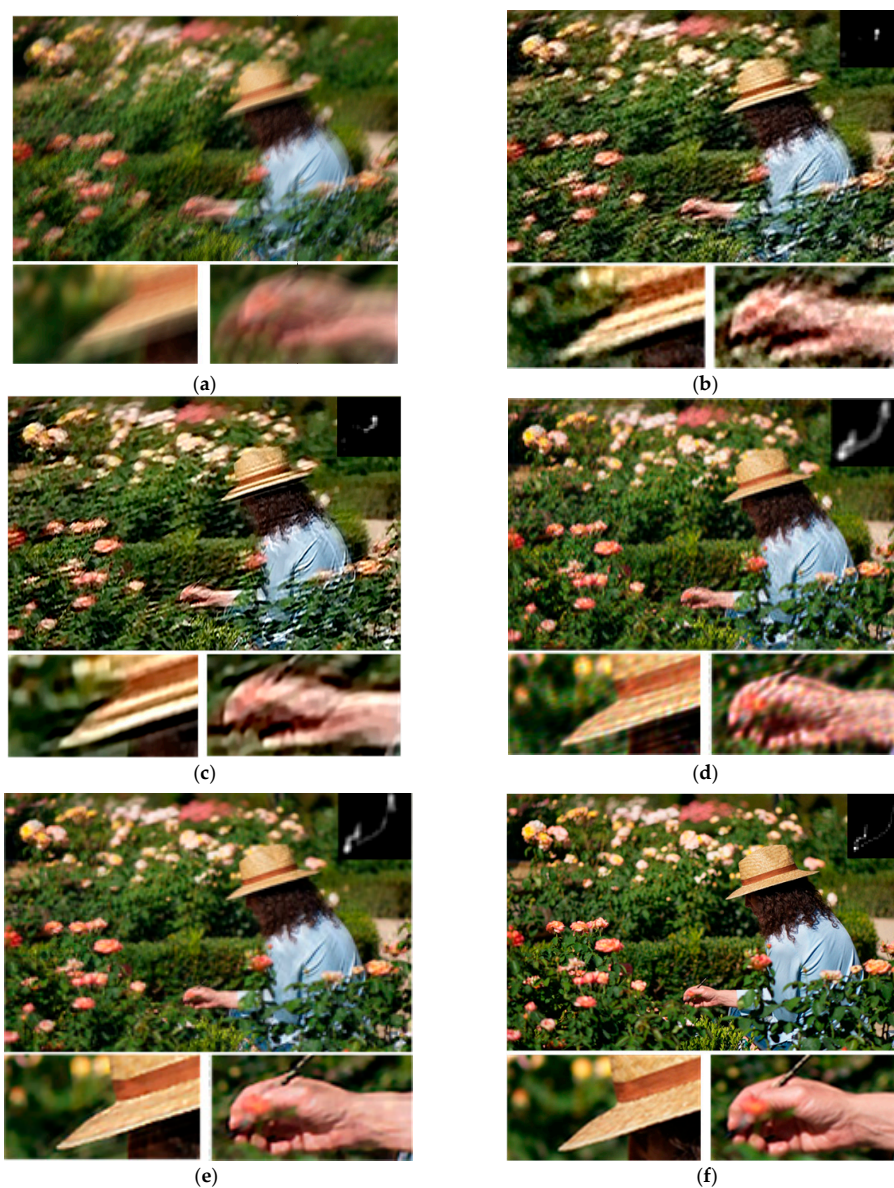
Figure 9. Comparison of different restoration algorithms on blur image with rich textures and small details. (**a**) Blurred image; (**b**) result of Fergus *et al.* [3]; (**c**) result of Shan *et al.* [4]; (**d**) result of Xu and Jia [7]; (**e**) our result; and (**f**) the ground truth image and blur kernel. The size of blur kernel is 27 × 27.

We next test our algorithm on real blurred images, as shown in Figures 10–12, which are presented in some previous works for comparison. Figure 10a is the real blurred image used in [3]. Due to the inaccurate blur kernel estimation, the results of [3,4,6] contain some noise. The method of [7] shows a better result, but there is still some noise in the estimated blur kernel. Our result shown in Figure 10f performs best in both kernel estimation and latent image restoration. Figure 11a is the blurred fish image used in [13]. The result of [13] contains same noise both in the estimated blur kernel and restored image. The result of [4] performs better in kernel estimation, but the restored image still contains some ringing artifacts. By contrast, our result shown in Figure 11d gives the best performance.
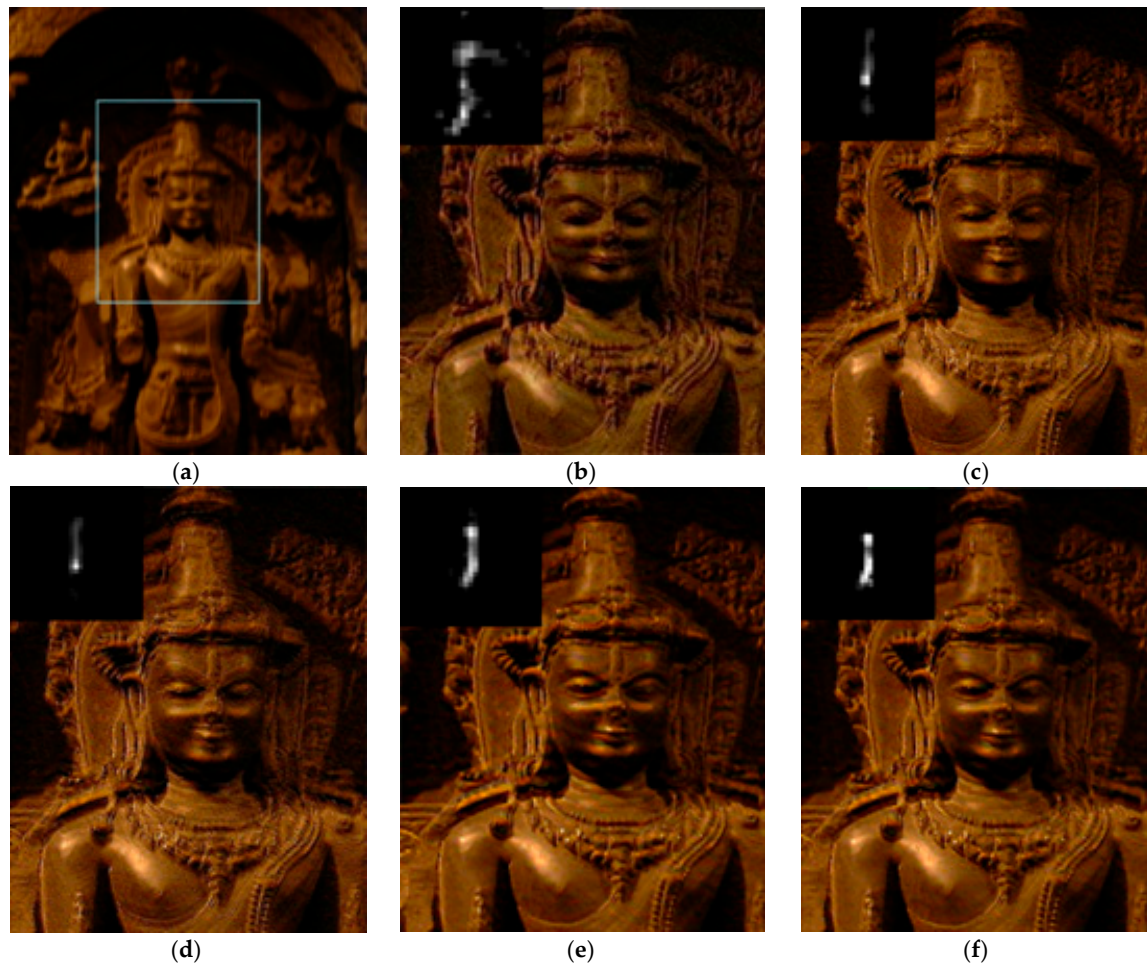

Figure 10. Comparison of different restoration algorithms on the Buddha image. (**a**) Real blurred image, (**b**-**f**) are the restored results of the region in the box in (**a**); (**b**) result of Fergus *et al.* [3]; (**c**) result of Shan *et al.* [4]; (**d**) result of Cho and Lee [6]; (**e**) result of Xu and Jia [7]; and (**f**) result of our algorithm.

In addition to the capability in dealing with the blurred image containing rich textures and small details, our algorithm also can deal with large blur kernels, as shown in Figure 12. Figure 12a is the blurred wall image which contains large motion blur and small details. Due to the large blur, the method of [3] cannot estimate the blur kernel accurately and the restored latent image still contains some blur. The restored result of [4] still contains some noise and ringing artifacts. The results of our algorithm give a clearer image with finer details.
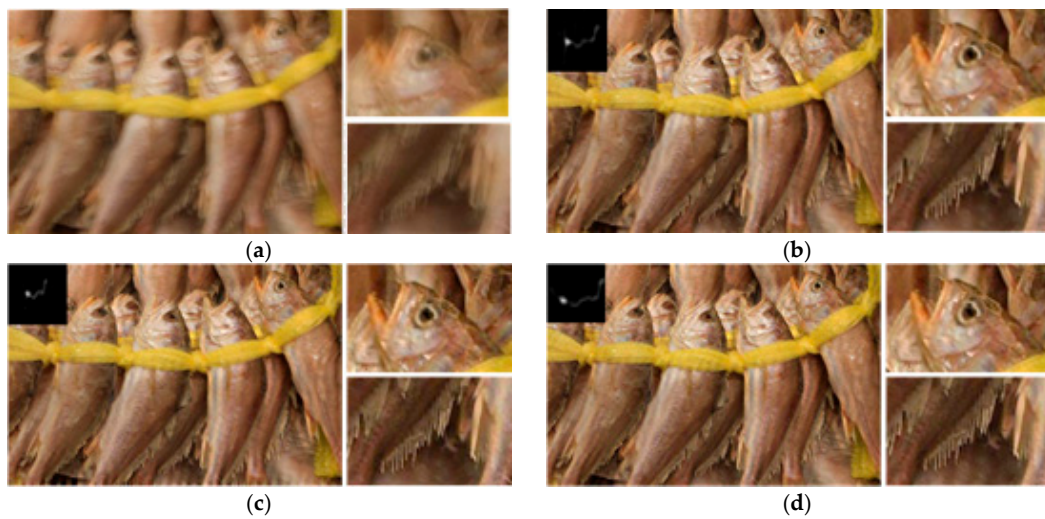
Figure 11. Comparison of different restoration algorithms on the fish image. (**a**) real burred image; (**b**) result of Krishnan *et al.* [13]; (**c**) result of Shan *et al.* [4]; and (**d**) our result.
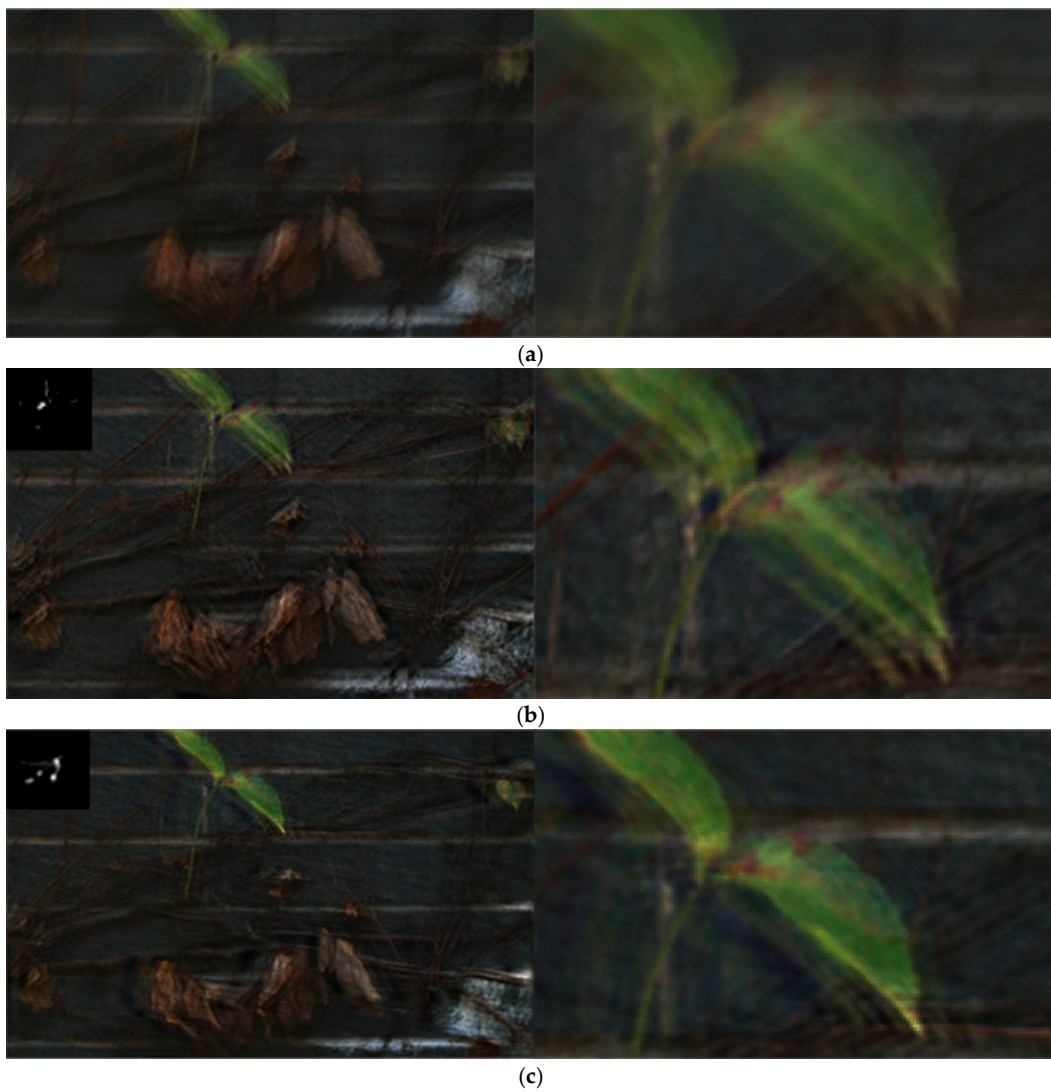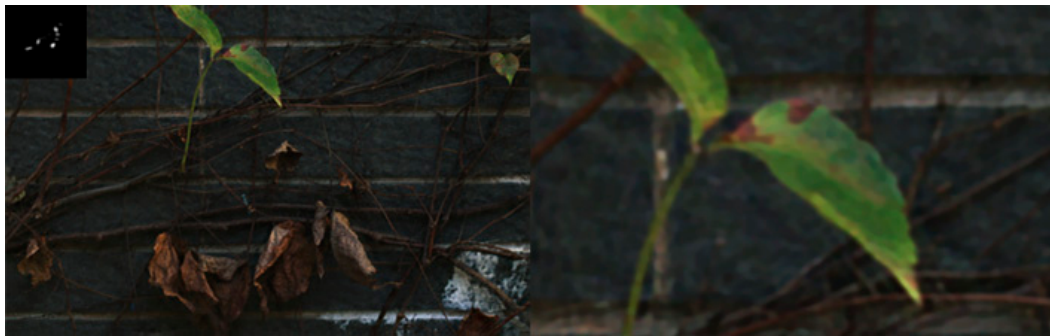

(**a**)

(**b**)

(**c**)

Figure 12. *Cont.*

(**d**)

Figure 12. Comparison of different restoration algorithms on the wall image with large blur. (**a**) Real burred image; (**b**) result of Fergus *et al.* [3]; (**c**) result of Shan *et al.* [4]; and (**d**) our result. The size of our estimated blur kernel is 95×95.

4.2. Operation Speed

During the restoration process, we estimated the blur kernel and latent sharp image alternately, which involved a few matrix-vector or convolution operations. For the operation speed, the MATLAB implementation of the proposed algorithm spends about 90 s to estimate a 27×27 blur kernel from a 255×255 blurred image with an Intel i5 560 M CPU@2.67 GHz and 8 GB RAM. In comparison, methods [3,13,16] need about 6 min, 3 min, and 4 min, respectively, which are computed by using the author's MATLAB source code. The algorithm [4] implemented in C++ spends about 50 s. As the proposed algorithm involves non-convex models in kernel estimation, it needs more computational time than [6,7]. As the proposed algorithm uses a CG method and FFTs for optimization, we believe that it is feasible to accelerate by using a GPU with the strategy in [6].

5. Conclusions

In this paper, we propose a robust image restoration algorithm for motion blur of image sensors. Our algorithm is composed of three steps: reliable structures extraction, blur kernel estimation, and non-blind deconvolution. We implement it in a multi-scale coarse-to-fine manner. Benefiting from the self-adaptive reliable structures extraction method, the structures which have adverse effect on kernel estimation will be removed. Then we use the reliable structures of the blur image and priors of the blur kernel, such as sparsity and continuity, to estimate and refine the blur kernel. After obtaining the blur kernel, we use the fast MR-based Wiener filter to restore the final latent image. Our algorithm can deal with large blur kernels even when the blurred images contain abundant textures and small details.

However, as saturated regions in blur images destroy the linearity of the blur model, our algorithm cannot estimate the blur kernel accurately and fails to restore the blur images which contain saturated regions. Furthermore, our algorithm also cannot deal with the spatially-varying blur. Our future work is to resolve these limitation.

Author Contributions: Fasheng Yang initiated the research, designed the experiments. Fasheng Yang and Lixing Li performed the experiments. Yongmei Huang and Yihan Luo gave some helpful discussions. Hongwei Li wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Richardson, W. Bayesian-based iterative method of image restoration. *J. Opt. Soc. Am.* **1972**, *62*, 55–59. [CrossRef]

2. Wiener, N. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*; The MIT Press: Cambridge, MA, USA, 1964.

3. Fergus, R.; Singh, B.; Hertzmann, A.; Roweis, S.; Freeman, W. Removing camera shake from a single photograph. *ACM Trans. Graph.* **2006**, *25*, 787–794. [CrossRef]

4. Shan, Q.; Jia, J.; Agarwala, A. High-quality motion deblurring from a single image. *ACM Trans. Graph.* **2008**, *27*, 73. [CrossRef]

5. Jia, J. Single image motion deblurring using transparency. In Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8.

6. Cho, S.; Lee, S. Fast motion deblurring. *ACM Trans. Graph.* **2009**, *28*. [CrossRef]

7. Xu, L.; Jia, J. Two-phase kernel estimation for robust motion deblurring. In Proceedings of the 11th European Conference on Computer Vision, Heraklion, Crete, Greece, 5–11 September 2010; pp. 157–170.

8. Joshi, N.; Szeliski, R.; Kriegman, D. PSF estimation using sharp edge prediction. In Proceedings of the CVPR, Anchorage, AK, USA, 23–28 June 2008.

9. Yang, J.; Zhang, B.; Shi, Y. Scattering Removal for Finger Vein Image Restoration. *Sensors* **2012**, *12*, 3627–3640. [CrossRef] [PubMed]

10. Zhang, W.; Quan, W.; Guo, L. Blurred Star Image Processing for Star Sensors under Dynamic Conditions. *Sensors* **2012**, *12*, 6712–6726. [CrossRef] [PubMed]

11. Manfredi, M.; Bearman, G.; Williamson, G.; Kronkright, D.; Doehne, E.; Jacobs, M.; Marengo, E. A New Quantitative Method for the Non-Invasive Documentation of Morphological Damage in Painting Using RTI Surface Normals. *Sensors* **2014**, *14*, 12271–12284. [CrossRef] [PubMed]

12. Cheong, H.; Chae, E.; Lee, E.; Jo, G.; Paik, J. Fast Image Restoration for Spatially Varying Defocus Blur of Imaging Sensor. *Sensors* **2015**, *15*, 880–898. [CrossRef] [PubMed]

13. Krishnan, D.; Tay, T.; Fergus, R. Blind deconvolution using a normalized sparsity measure. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011.

14. Xu, Y.; Hu, X.; Wang, L.; Peng, S. Single image blind deblurring with image decomposition. In Proceedings of the 2012 IEEE International Conference on Proceedings of Acoustics, Speech and Signal Processing (ICASSP), Kyoto, Japan, 25–30 March 2012.

15. Levin, A.; Weiss, Y.; Durand, F.; Freeman, W.T. Understanding and evaluating blind deconvolution algorithms. In Proceedings of the Computer Vision and Pattern Recognition, 2009, Miami, FL, USA, 20–25 June 2009; pp. 1964–1971.

16. Levin, A.; Weiss, Y.; Durand, F.; Freeman, W.T. Efficient marginal likelihood optimization in blind deconvolution. In Proceedings of the Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011; pp. 2657–2664.

17. Oh, S.; Kim, G. Robust Estimation of Motion Blur Kernel Using a Piecewise-Linear Model. *IEEE Trans. Image Process.* **2014**, *3*, 1394–1407.

18. Shao, W.; Ge, Q.; Deng, H.; Wei, Z.; Li, H. Motion Deblurring Using Non-Stationary Image Modeling. *J Math. Imaging. Vis.* **2015**, *52*, 234–248. [CrossRef]

19. Money, J.; Kang, S. Total variation minimizing blind deconvolution with shock filter reference. *Image Vis. Comput.* **2008**, *26*, 302–314. [CrossRef]

20. Levin, A.; Fergus, R.; Durand, F.; Freeman, W.T. Image and depth from a conventional camera with a coded aperture. *ACM Trans. Graph.* **2007**, *26*, 70–78. [CrossRef]

21. Yuan, L.; Sun, J.; Quan, L.; Shum, H.-Y. Image deblurring with blurred/noisy image pairs. *ACM Trans. Graph.* **2007**, *26*. [CrossRef]

22. Rudin, L.; Osher, S.; Fatemi, E. Nonlinear total variation based noise removal algorithms. *Phys. D Nonlinear Phenom.* **1992**, *60*, 259–268. [CrossRef]

23. Fattal, R.; Lischinski, D.; Werman, M. Gradient domain high dynamic range compression. *ACM Trans. Graph.* **2002**, *21*, 249–256. [CrossRef]

24. Osher, S.; Rudin, L. Feature-oriented image enhancement using shock filters. *SIAM J. Numer. Anal.* **1990**, *4*, 919–940. [CrossRef]

25. Xu, L; Lu, C.; Xu, Y.; Jia, J. Image smoothing via l_0 gradient minimization. *ACM Trans. Graph.* **2011**, *6*, 174. [CrossRef]

26. Luo, Y.; Fu, C. Midfrequency-based real-time blind image restoration via independent component analysis and genetic algorithms. *Opt. Eng.* **2011**, *4*, 047004. [CrossRef]