A Fast Algorithm for Single Motion Image Deblurring

1, 2 Yong Zhong Liao, 1 Cai Zi Xing, 1 Xiang Hua He
1 School of Information Science and Engineering, Central South University, Changsha, Hunan, 410002 China
2 Department of Information Science and Engineering, Hunan First Normal University, Changsha, Hunan, 410205 China
1 Tel.: 086073182841011
E-mail: lyz031608@126.com

Received: 15 April 2014 /Accepted: 15 May 2014 /Published: 31 May 2014

Abstract: The blurred image blind restoration is a difficult problem of image processing. The key is the estimation of the Point Spread Function and non-blind deconvolution algorithm. In this paper, we propose a fast robust algorithm based on radon transform-domain to determine the blur kernel function. Then the blurred images are restored by using a modified fast non-blind deconvolution method based on image prior. Compared with R. Fergus’ algorithm, the experiment results prove that the algorithm for a class of motion blurred image caused by the linear movement parallel to the lens has higher speed and good recovery effect.

Keywords: Blur kernel function, Motion blur, Radon transforms, Blind restoration.

1. Introduction

Camera shake or movement causes blurry photographs, especially when using telephoto lens or long shutter speed. Removing the effects from an image is one of the primary problems in the field of image processing, mathematically, a very general model for the motion blurring of images is usually considered as a convolution process in the space domain: expression:

\[ g(x, y) = f(x, y) \ast h(x, y) + n(x, y), \quad (1) \]

where \( g(x, y) \) is the blurred image, \( f(x, y) \) is the sharp image, \( h(x, y) \) is the blur kernel (or point spread function), \( \ast \) is the convolution operator, and \( n(x, y) \) the additive noise. It will also be useful to express the convolution in the frequency domain:

\[ G(u,v) = F(u,v)H(u,v) + N(u,v) \quad (2) \]

The capital letters denote the two-dimensional Discrete Fourier Transforms. If the blur kernel is unknown and the problem is called a blind deconvolution problem, otherwise a non-blind deconvolution problem. It is well known that the non-blind deconvolution problem is an ill-conditioned problem for its sensitivity to noise, but single-image blind deconvolution is even more ill-posed because we must estimate both the PSF and sharp image.

We know that the process of deconvolution is highly dependent on the estimation of the blur kernel, so exact estimation of blur kernel is crucial for the restoration of the blurred image.

In the past, many researchers have been working on recovering sharp images from motion-blurred images. Fergus et al. use ensemble learning...
to recover blur kernel while assuming a certain statistical distribution for natural image gradients [1]. A variational method is used to approximate the posterior distribution, and then Richardson-Lucy is used for deconvolution. Levin advocated Bayesian inference methods through which the image is marginalized from the optimization while estimating the unknown blur kernel [2]. Krishnan et al. take an MAP, approach but with $\lambda/2$ regularization to predict the sharp image in a coarse-to-fine iteration to deblur the images [3]. Michal Dobes et al. propose a fast method to estimate the blur kernel which is based on computation of the power spectrum of the image gradient in the frequency domain [4]. Zoran D, et al. have proposed a method to learn patch priors and perform patch restoration [5]. These methods can lead to tremendous computations.

This paper proposes an improved method to estimate the blur kernel. It is assumed that the real blurred image is caused by combined action of line motion and noise. The assumption of the blur kernel is based on the following considerations: if the distance between the imaging system and the object is constant and image is blurred by camera motion, then the blur kernel indicate the trajectory of camera shake during exposure. Exposure time is generally short in real photograph, so we think the trajectory of camera shake is approximately linear.

2. Motion Blur Model

Under the assumption above, the real motion blur kernel can be approximated by a linear blur kernel, so it can be represented only by the motion length and the direction, therefore, $h(x,y)$ is restricted to be a line segment as follows:

$$h(x,y) = \begin{cases} 0, & y \neq xtg\theta, -\infty \leq x \leq +\infty \\ 1/L, & y = xtg\theta, 0 \leq x \leq L \cos \theta \end{cases}$$

where $L$ is the motion length, and proportional to the relative velocity; $\theta$ is the motion direction. The Fourier transform of $h(x,y)$ is represented by [6-7].

$$H(u, v) = \frac{\sin(\sin(\frac{ucos\theta}{M} + \frac{v \sin \theta}{N}))}{\pi L(M \cos \theta + N \sin \theta)} \delta$$

The amplitude spectrum of $H(u,v)$ is characterized by periodic zeroes. A regular pattern of parallel stripes appears. We can estimate the parameters $L$ and $\theta$ from parallel stripes and be able to infer the blur kernel. However, in practice, the stripes are hardly visible or the stripes may be corrupted by other structures contained in the blurred image.

In this paper, we propose a robust algorithm based on Radon transform domain to determine the angle and the length from a single photograph when the stripes are hardly visible.

3. Blur Kernel Estimation

As mentioned previously, a regular structure is contained within the spectrum of the linear blur kernel. This regular structure is related to the length and the direction of the blur kernel. Normally, the stripes are corrupted by noise. If we choose a proper filter and reduce the noise level and enhance parallel stripes of the power spectrum, the parallel stripes will be clearer. The distance between neighboring stripes is proportional to parameter $L$ of blur kernel. The direction of the blur is perpendicular to the direction of such stripes. Therefore, if we are able to find this regular pattern, we would be able to infer the blur kernel. Sakano suggested an algorithm based on the Hough transform which can accurately and robustly estimate the PSF, even in low signal-to-noise ratio cases, but the method is computationally complex. Dobes et al. first computed the power spectrum of the image gradient and applied a band pass Butterworth filter to smooth the spectrum and remove the unwanted noise, then the blur parameters is found using Radon transform [4]. Motivated by this, this paper proposes a fast and robust algorithm, that is, the Ridgelet transform which is capable of linear expression, will be utilized to enhance the frequency spectral image with noises, then a robust algorithm based on Radon transform is used to estimate blur kernel.

3.1. The Radon Transform

The Radon transform of a signal $f(x,y)$ is the collection of integrals of $f(x,y)$ along straight lines, where the straight lines can be conveniently parameterized using the offset $\rho$ and the orientation $\beta$. As the location of the lines are determined from the line equation: $\rho = xcos\beta + ysin\beta$, so it can be expressed as [8]:

$$R_{\beta}(\rho, \beta) = \int \int g(x,y) \delta(xcos\beta + ysin\beta - \rho) dxdy,$$

where $\beta$ is the orientation of the radial line that we integrate over and $\rho$ is the offset of that line from the origin of the coordinate, $\delta()$ is the Dirac delta function. We define $R_{\beta}(\beta, \rho)$ is radon transform of $G(u,v)$, then

$$R_{\beta}(I, \rho) = \int \int G(u,v) \delta(u\cos\beta + v\sin\beta - \rho) dudv$$

215
3.2. The Ridgelet Transform

Now we start by briefly reviewing on the continuous Ridgelet transform defined by Candès and Donoho in [9]. We have introduced the Radon transform, from which the continuous Ridgelet transform stems. The Continuous Ridgelet transform (CRT) is simply the application of a mono-dimensional wavelet to the slices of the Radon transform. The CRT of the frequency spectrum \( G(u, v) \) is expressed as [10]:

\[
CRT_G(a, b, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \psi_{a, b, \theta}(u, v)G(u, v)dudv, \tag{7}
\]

where the Ridgelets \( \psi_{a, b, \theta}(u, v) \) in 2D are defined from a wavelet-type function in 1D \( \psi() \) as:

\[
\psi_{a, b, \theta}(u, v) = a^{ab} \psi(a \cos\theta + v \sin\theta - b) / a \tag{8}
\]

The Ridgelet function is constant along the lines of \( u \cos\theta + v \sin\theta = \text{const} \). Line singularities can be mapped to point singularities with Radon transform. Wavelets are expert in catching zero-dimensional or point singularities. The Ridgelet Transform is capable of linear expression. We use this capability to enhance the part of line in the frequency spectrum of the image [11].

\[
C_{\text{CRT}}((i, j)) = \chi * \text{sgn}(C_{\text{CRT}}(i, j))(C_{\text{CRT}}(i, j) - \chi)
\]

\[
= \begin{cases} 
C_{\text{CRT}}(i, j) + \chi, & C_{\text{CRT}}(i, j) < -\chi \\
0, & \text{otherwise} \\
C_{\text{CRT}}(i, j) - \chi, & C_{\text{CRT}}(i, j) > \chi 
\end{cases} \tag{9}
\]

where \( \chi \) is the threshold value, \( \gamma \) is the enhancement coefficients. Two results are shown in Fig. 1.

![Fig. 1. Frequency spectrogram of Ridgelet enhancement.](image)

3.3. Blur Direction Estimation

In recent years many researches in natural image statistics have shown that, although images of real-world scenes vary greatly in their absolute color distributions, their one-dimensional cross-sections of the logarithm of the power spectrum of the natural image is independent of angle of cross-sections [12-13], as shown in Fig. 2.

![Fig. 2. One-dimensional cross-sections of the power spectrum (only the positive part is shown).](image)

In fact, more thorough investigations have found that the log-power spectrum obey exponent distributions. Namely, \( \log |F(u, u \tan \phi)| \simeq \epsilon \|u\|^\zeta \), where \( \epsilon, \zeta \) are constants in the same image. We define \( G_j(u, v) = \log |G(u, v)| \), \( H_j(u, v) = \log |H(u, v)| \) and \( F_j(u, v) = \log |F(u, v)| \). The shape the logarithm of the power spectrum of the natural image along the line of \( v = u \tan \phi \) is independent of angle \( \phi \). According to the linear property of Radon transform and we can write the equation:

\[
R_{c_j} (r, \theta)(G_j(u, v)) = E(R_{c_j} (\theta)(G_j(u, v))) - E(R_{c_j} (\theta)(F_j(u, v)))
\]

\[
= R_{c_j} (\theta)(K_j(u, v)) - E(R_{c_j} (\theta)(K_j(u, v))) + E(R_j(\theta)(F_j(u, v)))
\]

\[
= R_{c_j} (\theta)(K_j(u, v)) - E(R_{c_j} (\theta)(K_j(u, v))) + E(R_j(\theta)(F_j(u, v)))
\]

\[
\leq R_{c_j} (\theta)(K_j(u, v)) - E(R_{c_j} (\theta)(K_j(u, v))) + R_{c_j} (\theta)(F_j(u, v))
\]

\[
\leq R_{c_j} (\theta)(K_j(u, v)) - E(R_{c_j} (\theta)(K_j(u, v))) + R_{c_j} (\theta)(F_j(u, v))
\]

\[
\leq R_{c_j} (\theta)(F_j(u, v)) - E(R_{c_j} (\theta)(K_j(u, v)))
\]

\[
\leq R_{c_j} (\theta)(F_j(u, v)) - E(R_{c_j} (\theta)(K_j(u, v)))
\]

\[
\leq R_{c_j} (\theta)(F_j(u, v)) - E(R_{c_j} (\theta)(K_j(u, v)))
\]

\[
\leq R_{c_j} (\theta)(F_j(u, v)) - E(R_{c_j} (\theta)(K_j(u, v)))
\]

\[
\leq R_{c_j} (\theta)(F_j(u, v)) - E(R_{c_j} (\theta)(K_j(u, v)))
\]

As the value of \( F_j(u, v) \) is irrelevant with angle \( \theta \), the second term of the Eqn. (10) is approximately constant. When the angle \( \theta \) equals the direction of motion, there exists an extreme value in equation. We can define a new expression:

\[
Q(\theta) = |R_{c_j} (\theta)(G_j(u, v)) - E(R_{c_j} (\theta)(G_j(u, v)))|
\]

\[
\tilde{\theta}_m = \arg \max_\theta \|Q(\theta)\| \tag{12}
\]
3.3. Blur Length Estimation

Once we have \( \theta \), we can estimate the motion blur length of the blur kernel. The method of inferring the blur length is by identifying the distance of parallel stripes. We rotate the power spectrum by \( \theta \) degree, and method [7] will be used to estimate the blur length.

4. Experiments of Kernel Estimation

To validate our algorithm, we tested our method on synthetic cases to estimate the blur angle and blur length. We considered several different motion blur lengths and motion blur angles to produce motion blur kernel, then we obtained an artificial blurred and noise images resulted from the convolution of a sharp image with a known shift-invariant motion blur kernel plus additive white Gaussian noise. The test sharp images 'Lena' 512x512 is chosen, the results are also briefly compared to the approach published in [14] as are shown in Fig. 3 and Fig. 4 (Algorithm I is our method).

![Fig. 3. L=4; BSNR=10 error of \( \theta \).](image)

![Fig. 4. Blur length estimation.](image)

The experiments revealed that determining the length and angle of blur is more accurate when the power of the additive noise is large, and the relative motion is small. From these results, it can be also seen the superiority of the present algorithm. Especially, in the case that power of noise is relatively large, that is, BSNR is relatively low, the estimation of performance of our algorithm is good.

5. Image Deblurring

Once we have the motion blur kernel, we can use a variety of non-blind deconvolution method to recover the clear image \( f(x,y) \) from the blurred image \( g(x,y) \), such as the simplest Richardson-Lucy [15], Total Variation regularization method [16] and so on. The disadvantage of RL is that this method is sensitive to an error estimation of kernel, which results in ringing artifacts in \( f(x,y) \). As our method used a linear motion blur kernel to replace a real blur kernel, so it will lead to error estimation of blur kernel. Therefore, we applied this modified version of non-blind Total Variation deconvolution algorithm.

5.1. Total Variation Algorithm

One of the most successful regularization approaches is the Total Variation (TV) regularization method which can effectively recover edges of an image. The mathematical formulation used in the image recovery problem is stated as follow:

\[
\min_f \| h \otimes f - g \|_2^2 + \lambda \| \nabla f \|_2 \text{dxdy} \quad (13)
\]

For simplicity, where \( f, g, h \) indicate \( f(x,y), g(x,y), h(x,y) \), \( \Omega \) is an open rectangular domain in \( \mathbb{R}^2 \), and \( |\nabla f| \) is the gradient of the image, \( \lambda \) is balancing weight. We rearrange the elements of the images \( f \) and \( g \) into column vectors. \( H_f \) is a linear blurring operator, \( D_h \) and \( D_v \) are horizontal and vertical discrete derivative operators, \( D \) is a gradient operator. We have:

\[
\min_f \| H_f f - g \|_2^2 + \lambda \left( (D_h)^2 + (D_v)^2 \right) \quad (14)
\]

where \( \| H_f f - g \|_2^2 \) is the fidelity term and \( \left( (D_h)^2 + (D_v)^2 \right) \) is the regularization term. We define:

\[
H_f f = g = \begin{pmatrix} e_x \\ \vdots \\ e_{xy} \end{pmatrix} ; \quad D f = \begin{pmatrix} D_h f \\ D_v f \end{pmatrix} ; \quad \frac{\partial f}{\partial y} = D_v f = \begin{pmatrix} \sigma_x \\ \sigma_y \\ \sigma_{xy} \end{pmatrix} \]

\[
\text{Algorithm I}
\]

\[
\text{Algorithm II}
\]
\[
\frac{\partial f}{\partial x} = D, f = \begin{pmatrix} \sigma_{x1} \\ \sigma_{x2} \\ \vdots \\ \sigma_{xw} \end{pmatrix} ; \text{ then } \nabla f = \begin{pmatrix} \sqrt{\sigma_{x1}^2 + \sigma_{y1}^2} \\ \sqrt{\sigma_{x2}^2 + \sigma_{y2}^2} \\ \vdots \\ \sqrt{\sigma_{xw}^2 + \sigma_{yw}^2} \end{pmatrix} \]

\[
\left( \begin{array}{c}
\frac{\partial f}{\partial x} \\
\frac{\partial f}{\partial y}
\end{array} \right) = W_x \left( \begin{array}{c}
\sigma_{x1} \\
\sigma_{x2} \\
\vdots \\
\sigma_{xw}
\end{array} \right);
\text{ then } \| f - \text{NLM}(f) \|_2^2 \text{, so we minimize the following energy function:}
\]

\[
\min_{f} J(f) = \min \| H_x f - g \|_2^2 + \kappa \| \text{NLM}(f) \|_2^2 + \lambda \| D_x f \|_2^2 + \| D_y f \|_2^2,
\]

where \( \text{NLM}(\cdot) \) is the non-local means denoising operation [17], \( \kappa \) is the weight. We use the Iteratively Reweighted algorithm [18] to optimize the Eq. (15). We set
\[
W_x = \text{diag} \left( \begin{array}{c}
\frac{1}{\sqrt{\sigma_{x1}^2 + \sigma_{y1}^2}}, \text{ if } \sqrt{\sigma_{x1}^2 + \sigma_{y1}^2} > \epsilon_x \\
0, \text{ otherwise}
\end{array} \right),
\]

where \( \epsilon_x \) is the infinitesimal threshold value, we have:

\[
\left\| \begin{pmatrix}
\frac{\partial f}{\partial x} \\
\frac{\partial f}{\partial y}
\end{pmatrix} \right\|_2^2 = W_x^{-1} \left( \begin{array}{c}
\frac{\partial f}{\partial x} \\
\frac{\partial f}{\partial y}
\end{array} \right) \left( \begin{array}{c}
\frac{\partial f}{\partial x} \\
\frac{\partial f}{\partial y}
\end{array} \right)^T \]

5.2. Improved Total Variation Algorithm

The standard ROF model Eqn. (13) is well known to have certain limitations. One important issue is the loss of contrast in solutions even for noise-free observed images. Moreover, as our method used an approximate linear motion blur kernel to estimate the sharp image, it will produce an estimation error. In the meantime, in order to reduce noise, we add a new regularization \( \| f - \text{NLM}(f) \|_2^2 \), so we minimize the following energy function:

\[
\min_{f} J(f) = \min \| H_x f - g \|_2^2 + \kappa \| f - \text{NLM}(f) \|_2^2,
\]

where \( \| \cdot \|_2 \) is the Euclidean norm. We use the Iteratively Reweighted algorithm [18] to optimize the Eq. (15). We set
\[
W_x = \text{diag} \left( \begin{array}{c}
\frac{1}{\sqrt{\sigma_{x1}^2 + \sigma_{y1}^2}}, \text{ if } \sqrt{\sigma_{x1}^2 + \sigma_{y1}^2} > \epsilon_x \\
0, \text{ otherwise}
\end{array} \right),
\]

where \( \epsilon_x \) is the infinitesimal threshold value, we have:

\[
\left\| \begin{pmatrix}
\frac{\partial f}{\partial x} \\
\frac{\partial f}{\partial y}
\end{pmatrix} \right\|_2^2 = W_x^{-1} \left( \begin{array}{c}
\frac{\partial f}{\partial x} \\
\frac{\partial f}{\partial y}
\end{array} \right) \left( \begin{array}{c}
\frac{\partial f}{\partial x} \\
\frac{\partial f}{\partial y}
\end{array} \right)^T \]

6. Experiments

In this section, we tested our method on synthetic data and real data. As shown in Fig. 5. Our method is compared with Fergus’ method. Results show our method achieves equal or better accuracy than Fergus’ method and can effectively reduce the computation time.

![Fig. 5. Results: (a) are the synthetic blurred image; (b), (c) real-world images; (d), (e), (f) are the deblurred images by the method in [1], (j), (k), (l) are the deblurred images by our method, (g), (h), (i) are the blur kernels estimated by the method in [1], (m), (n), (o) are the blur kernels estimated by our method.](image-url)

7. Conclusions

This paper proposes a new single motion-blurred image blind deconvolution method that is more robust than Fergus’ method when the SNR is low, and the relative motion is small. Then we use a fast non-blind deconvolution algorithm based on prior of image gradient magnitudes that generates high-quality final results. Experiments reveal that our method achieves equal or better results than Fergus et al.’s and reduce the computation time.
Acknowledgements

This work was supported by the National Science Foundation of China under Grant 61175064.

References


2014 Copyright ©, International Frequency Sensor Association (IFSA) Publishing, S. L. All rights reserved.
(http://www.sensorsportal.com)