ACKNOWLEDGMENT

The author would like to thank the anonymous reviewers for their many valuable comments and suggestions that helped to improve both the technical content and the presentation quality of this paper.

REFERENCES

[1] D. Coltuc and P. Bolon, "Color image watermaking in HSI space," in *Proc. IEEE ICIP*, Vancouver, BC, Canada, Sep. 2000, vol. 3, pp. 698–701.

[2] R. C. Gonzales and R. W. Woods, *Digital Image Processing*. Upper Saddle River, NJ: Prentice-Hall, 1992.

[3] V. Caselles, J. Lisani, J. Morel, and G. Sapiro, "Shape preserving local histogram modification," *IEEE Trans. Signal Process.*, vol. 8, no. 2, pp. 220–230, Feb. 1999.

[4] J. Stark, "Adaptive image contrast enhancement using generalization of histogram equalization," *IEEE Trans. Signal Process.*, vol. 9, no. 5, pp. 889–896, May 2000.

[5] D. Coltuc, P. Bolon, and J. M. Chassery, "Exact histogram specification," *IEEE Trans. Image Process.*, vol. 15, no. 5, pp. 1143–1152, May 2006.

[6] A. Rosenfeld and K. Kak, *Digital Picture Processing*. Upper Saddle River, NJ: Prentice-Hall, 1982.

[7] L. Shapiro and G. Stockman, *Computer Vision*. Englewood Cliffs, NJ: Prenctice-Hall, 2001.

[8] Y. Zhang, "Improving the accuracy of direct histogram specification," *Electron. Lett.*, vol. 28, no. 3, pp. 213–214, Jan. 1992.

[9] J. Rolland, V. Vo, B. Bloss, C. Abbey, and G. Young, "Fast algorithm for histogram matching applications to texture synthesis," *J. Electron. Imag.*, vol. 9, no. 1, pp. 39–45, Jan. 2000.

[10] A. Bevilacqua and P. Azzari, "A high performance exact histogram specification algorithm," in *Proc. 14th IEEE ICIAP*, Modena, Italy, Sep. 2007, pp. 623–628.

[11] E. L. Hall, "Almost uniform distribution for computer image enhancement," *IEEE Trans. Comput.*, vol. C-23, no. 2, pp. 207–208, Feb. 1974.

[12] Y. Wang and D. Shi, "Joint exact histogram specification and image enhancement through the wavelet transform," *IEEE Trans. Image Process.*, vol. 16, no. 9, pp. 2245–2250, Sep. 2007.

[13] B. B. Mandelbrot, *The Fractal Geometry of Nature*. San Francisco, CA: Freeman, 1982.

[14] E. Simoncelli, "Statistical models for images: Compression, restoration and synthesis," in *Conf. Rec. 31st Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, Feb. 1997, pp. 673–678.

[15] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *J. Chem. Phys.*, vol. 21, no. 6, pp. 1087–1092, Jun. 1953.

[16] P.-M. Jodoin and M. Mignotte, "Markovian segmentation and parameter estimation on graphics hardware," *J. Electron. Imag.*, vol. 15, no. 3, pp. 033 015–1–033 015–15, Sep. 2006.

PSF Estimation via Gradient Domain Correlation

Wei Hu, Jianru Xue, *Member, IEEE*, and Nanning Zheng, *Fellow, IEEE*

***Abstract*—This paper proposes an efficient method to estimate the point spread function (PSF) of a blurred image using image gradients spatial correlation. A patch-based image degradation model is proposed for estimating the sample covariance matrix of the gradient domain natural image. Based on the fact that the gradients of clean natural images are approximately uncorrelated to each other, we estimated the autocorrelation function of the PSF from the covariance matrix of gradient domain blurred image using the proposed patch-based image degradation model. The PSF is computed using a phase retrieval technique to remove the ambiguity introduced by the absence of the phase. Experimental results show that the proposed method significantly reduces the computational burden in PSF estimation, compared with existing methods, while giving comparable blurring kernel.**

***Index Terms*—Deblur, phase retrieval (PR), point spread function (PSF) estimation.**

I. INTRODUCTION

The problem of blur kernel estimation, and more generally blind deconvolution, is a long-standing problem in computer vision and image processing. Recovering the point spread function (PSF) from a single blurred image is an inherently ill-posed problem due to the loss of information during blurring. The observed blurred image provides only a partial constraint on the solution as there are many combinations of PSFs and "sharp" images that can be convolved to match the observed blurred image. There are numerous papers on this subject in the signal and image processing literature. We refer the reader to a survey paper [1] for the earlier works. Recently, state-of-the-art PSF estimation algorithms have been proposed based on natural image statistics [2], [3] using a variational Bayesian method [2] or an advance optimization technique [4]. In [5], Levin *et al.* provide a good summary for these methods. In this paper, we propose a new simple and efficient method to extract a blur kernel using a gradient domain correlation technique. It has been proven in various fields of image processing that the gradients of the natural images are approximately independent to each other [2]. Many existing techniques for blind deconvolution take advantage of this property to simplify the inference [2], [4]. In particular, when the problem is formulated into the Bayesian framework, this assumption usually substantially simplifies the prior term [2]. These sophisticated algorithms [2], [4] give good deblurred results. However, they are usually very slow since most of them involve a complex alternating optimization process, which is very time consuming. In this paper, we have observed that the correlation property of the image gradients is

Manuscript received May 10, 2010; revised January 30, 2011 and May 17, 2011; accepted May 27, 2011. Date of publication June 20, 2011; date of current version December 16, 2011. This work was supported in part by the NSFC under Project 60875008 and Project 60805044 and in part by the 973 Program under Project 2010CB327902. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Farhan A. Baqai.

The authors are with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: whu@aiar.xjtu.edu.cn; jrxue@aiar.xjtu.edu.cn; nnzheng@aiar.xjtu.edu.cn).

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/TIP.2011.2160073

changed after the clean image is convolved with a blurred kernel. Intuitively, pixels of the blurred image are more correlated in the direction of the blurring trajectory of the PSF than other directions. Thus, the information of a convolution kernel is contained in the covariance matrix of the blurred image patches, which represents the correlation property of the image neighborhood. The theoretical relationship between the covariance matrices of a clean image and a blurred image is derived using a patch-based image degradation model. For a given blurred image, we first estimate its covariance matrix using the sample covariance matrix of small image patches. Then, we convert the sample covariance matrix into an autocorrelation matrix, which is proved to be the approximation of an autocorrelation function of the true kernel. Finally, the PSF is estimated from the autocorrelation function using a phase retrieval (PR) technique [6], [7], which imposes compact support constraint and positivity constraint to the unknown kernel to resolve the phase ambiguity. We test the performance of the proposed PSF estimation method with real blurred images. Experimental results show that the proposed algorithm produces promising results without involving computationally intensive optimization. The overall estimation only takes a few seconds. The rest of this paper is organized as follows: Section II describes the details of the proposed gradient-correlation-based algorithm. The experimental results are given in Section III. We conclude this paper in Section IV.

II. Proposed Method

Here, a patch-based image degradation model is proposed for image convolving. The relationships between original image patches and blurred image patches are built based on the linear convolution operator. We further show that the covariance matrix of small patches in gradient domain clean natural image is approximately a diagonal matrix, which means that the gradients of a clean image are approximately uncorrelated to each other. Then, we derive a new blur-kernel-finding algorithm based on this prior using the patch-based model.

A. Patch Based Image Degradation Model

Let vectors $\mathbf{x}$ and $\mathbf{y}$ represent image patches in the original and blurred images in a column stacking order, respectively. In addition, let $\mathbf{n}$ be the additive independent and identically distributed (i.i.d.) Gaussian noise, i.e., $\mathbf{n} \sim N(0, \sigma_n^2 \mathbf{I})$, where σ_n^2 is the variance of the noise. Let $A(\mathbf{k})$ represent the convolution operator given convolution kernel $\mathbf{k}$. Then, we have

$$\mathbf{y} = \mathbf{A}(\mathbf{k})\mathbf{x} + \mathbf{n}. \tag{1}$$

Let the sizes of the original image patch, the blurred image patch, and the convolution kernel be $s \times s$, $m \times m$, and $n \times n$, respectively. Let $s = m + n - 1$, then we have $\mathbf{y}, \mathbf{n} \in R^{m^2 \times 1}$, $\mathbf{x} \in R^{(m+n-1)^2 \times 1}$, and $\mathbf{A}(\mathbf{k}) \in R^{m^2 \times (m+n-1)^2}$.

Convolution operator $\mathbf{A}(\mathbf{k})$ is derived as follows. Let $\mathbf{k}$ be the 1-D convolution kernel, i.e.,

$$\mathbf{k} = [\,k_1 \quad k_2 \quad \cdots \quad k_n\,]^T. \tag{2}$$

For 1-D convolution, it is easy to show that the linear convolution matrix for kernel $\mathbf{A}^1(\mathbf{k})$ is given by

$$\mathbf{A^1}(\mathbf{k}) = \begin{bmatrix} k_1 & k_2 & \cdots & k_n & 0 & \cdots & 0 \\ 0 & k_1 & k_2 & \cdots & k_n & 0 & 0 \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & k_1 & k_2 & \cdots & k_n \end{bmatrix} \tag{3}$$

where $\mathbf{A}^1(\mathbf{k}) \in R^{m \times (m+n-1)}$. For a 2-D kernel, $\mathbf{A}(\mathbf{k}) \in R^{m^2 \times (m+n-1)^2}$ becomes a block Toeplitz with Toeplitz blocks (BTTB) matrix [8] as follows:

$$\mathbf{A}(\mathbf{k}) = \begin{bmatrix} A_1 & A_2 & \cdots & A_n & 0 & \cdots & 0 \\ 0 & A_1 & A_2 & \cdots & A_n & 0 & 0 \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & A_1 & A_2 & \cdots & A_n \end{bmatrix}. \tag{4}$$

Each submatrix in $\mathbf{A}(\mathbf{k})$ is given by

$$\mathbf{A_i} = \mathbf{A^1}(\mathbf{k}_i) \tag{5}$$

where $\mathbf{k}_i$ is the ith column in the 2-D convolution kernel $\mathbf{k}$.

B. Bayesian Framework and Marginalization

The two unknown variables, i.e., $\mathbf{k}$ and $\mathbf{x}$, make the blind deconvolution problem very challenging [2], [9]. Here, a Bayesian framework is proposed for this problem based on the patch-based image degradation model previously defined. The direct estimation based on MAP usually fails because it favors the delta kernel [2], [5]. In this paper, we marginalize the unknown clear image $\mathbf{x}$ in the model and derive the likelihood function between $\mathbf{k}$ and $\mathbf{y}$. Then, convolution kernel $\mathbf{k}$ is directly estimated from multiple observations $\{\mathbf{y}_i\}$ (blurry image patches) according to the maximum likelihood principle.

The posterior distribution of $\mathbf{k}$ and $\mathbf{x}$ can be expressed as

$$\begin{aligned} p(\mathbf{k}, \mathbf{x}|\mathbf{y}) &\propto p(\mathbf{y}|\mathbf{k}, \mathbf{x})p(\mathbf{k}, \mathbf{x}) \\ &= p(\mathbf{y}|\mathbf{k}, \mathbf{x})p(\mathbf{k})p(\mathbf{x}) \\ &= p(\mathbf{y}|\mathbf{k}, \mathbf{x})p(\mathbf{x}) \end{aligned} \tag{6}$$

where we assume that $\mathbf{k}$ and $\mathbf{x}$ are independent, i.e., $p(\mathbf{k}, \mathbf{x}) = p(\mathbf{k})p(\mathbf{x})$, and $\mathbf{k}$ is uniformly distributed, i.e., $p(\mathbf{k}) = 1$.

Instead of estimating $\mathbf{k}$ and $\mathbf{x}$ simultaneously, which is shown very hard [2], [5], we choose to inference $\mathbf{k}$ only. This is equivalent to marginalize latent image $\mathbf{x}$ in the model as

$$\begin{aligned} p(\mathbf{k}|\mathbf{y}) &= \int p(\mathbf{k}, \mathbf{x}|\mathbf{y})d\mathbf{x} \\ &\propto \int p(\mathbf{y}|\mathbf{k}, \mathbf{x})p(\mathbf{x})d\mathbf{x}. \end{aligned} \tag{7}$$

We assume that the noise is Gaussian distributed; based on the patch-based image degradation model proposed in the previous section, the likelihood term is given by

$$p(\mathbf{y}|\mathbf{k}, \mathbf{x}) = N\left(\mathbf{y}|\mathbf{A}(\mathbf{k})\mathbf{x}, \mathbf{L^{-1}}\right) \tag{8}$$

where $\mathbf{L}$ is the precision matrix of the noise. As many previous works [2], [3], [5], we use natural image gradients as the image prior. For the sake of simplicity, we assume that the image gradients are Gaussian distributed as follows:

$$p(\mathbf{G}\mathbf{x}) = N(\mathbf{G}\mathbf{x}|\mathbf{0}, \mathbf{\Lambda^{-1}}). \tag{9}$$

In the image domain, this prior can be expressed as

$$p(\mathbf{x}) = N\left(\mathbf{x}|\mathbf{0}, (\mathbf{G^T \Lambda G})^{-\mathbf{1}}\right) \tag{10}$$

where $\mathbf{G}$ is the gradient operator and $\mathbf{\Lambda}$ is the precision matrix of image gradients.

Then, the Bayesian model can be written as

$$\begin{aligned} p(\mathbf{k}|\mathbf{y}) &= \int p(\mathbf{k},\mathbf{x}|\mathbf{y})d\mathbf{x} \\ &\propto \int p(\mathbf{y}|\mathbf{k},\mathbf{x})p(\mathbf{x})d\mathbf{x} \\ &\propto \int \exp\left(-\frac{1}{2}\left(\mathbf{A}(\mathbf{k})\mathbf{x}-\mathbf{y}\right)^{\mathbf{T}}\mathbf{L}\left(\mathbf{A}(\mathbf{k})\mathbf{x}-\mathbf{y}\right)\right) \\ &\quad \times \exp\left(-\frac{1}{2}\mathbf{x}^{\mathbf{T}}\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G}\mathbf{x}\right)d\mathbf{x}. \end{aligned} \tag{11}$$

For the sake of convenience, we set $p(\mathbf{x},\mathbf{y}) = p(\mathbf{x},\mathbf{y}|\mathbf{k})$ and $\mathbf{A} = \mathbf{A}(\mathbf{k})$. Then, the joint distribution of $\mathbf{x}$ and $\mathbf{y}$ is given by

$$\begin{aligned} &\ln p(\mathbf{x},\mathbf{y}) \\ &\quad= \ln p(\mathbf{y}|\mathbf{x}) + \ln p(\mathbf{x}) \\ &\quad= -\frac{1}{2}(\mathbf{A}\mathbf{x}-\mathbf{y})^{\mathbf{T}}\mathbf{L}(\mathbf{A}\mathbf{x}-\mathbf{y}) - \frac{1}{2}\mathbf{x}^{\mathbf{T}}\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G}\mathbf{x} + \mathrm{const} \\ &\quad= -\frac{1}{2}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix}^{\mathbf{T}}\begin{bmatrix}\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G}+\mathbf{A}^{\mathbf{T}}\mathbf{L}\mathbf{A} & -\mathbf{A}^{\mathbf{T}}\mathbf{L}\\ -\mathbf{L}\mathbf{A} & \mathbf{L}\end{bmatrix}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix} + \mathrm{const} \\ &\quad= -\frac{1}{2}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix}^{\mathbf{T}}\mathbf{\Lambda}\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix} + \mathrm{const} \end{aligned} \tag{12}$$

where the precision matrix is given by

$$\mathbf{\Lambda} = \begin{bmatrix}\mathbf{\Lambda}_{\mathbf{xx}} & \mathbf{\Lambda}_{\mathbf{xy}}\\ \mathbf{\Lambda}_{\mathbf{yx}} & \mathbf{\Lambda}_{\mathbf{yy}}\end{bmatrix} = \begin{bmatrix}\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G}+\mathbf{A}^{\mathbf{T}}\mathbf{L}\mathbf{A} & -\mathbf{A}^{\mathbf{T}}\mathbf{L}\\ -\mathbf{L}\mathbf{A} & \mathbf{L}\end{bmatrix} \tag{13}$$

and the covariance matrix is given by

$$\begin{aligned} \mathbf{\Sigma} &= \begin{bmatrix}\mathbf{\Sigma}_{\mathbf{xx}} & \mathbf{\Sigma}_{\mathbf{xy}}\\ \mathbf{\Sigma}_{\mathbf{yx}} & \mathbf{\Sigma}_{\mathbf{yy}}\end{bmatrix} \\ &= \begin{bmatrix}(\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G})^{-1} & (\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G})^{-1}\mathbf{A}^{\mathbf{T}}\\ \mathbf{A}(\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G})^{-1} & \mathbf{L}^{-1}+\mathbf{A}(\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G})^{-1}\mathbf{A}^{\mathbf{T}}\end{bmatrix}. \end{aligned} \tag{14}$$

Because of the relationship of the precision and covariance matrices $\mathbf{\Sigma} = \mathbf{\Lambda}^{-1}$, the result in (14) is derived from (13) according to the block matrix inversion rule [10], [11], i.e.,

$$\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{bmatrix}^{-1} = \begin{bmatrix}\mathbf{M} & -\mathbf{M}\mathbf{B}\mathbf{D}^{-1}\\ -\mathbf{D}^{-1}\mathbf{C}\mathbf{M} & \mathbf{D}^{-1}+\mathbf{D}^{-1}\mathbf{C}\mathbf{M}\mathbf{B}\mathbf{D}^{-1}\end{bmatrix} \tag{15}$$

where

$$\mathbf{M} = (\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}. \tag{16}$$

As shown, $p(\mathbf{x},\mathbf{y}|\mathbf{k})$ is a joint Gaussian distribution. The marginalization over $\mathbf{x}$ is defined as

$$p(\mathbf{y}|\mathbf{k}) = \int p(\mathbf{x},\mathbf{y}|\mathbf{k})d\mathbf{x}. \tag{17}$$

The marginalization of multidimensional Gaussian variable is a common problem in statistics and machine learning [11]. We give the conclusion directly, and the details are provided in the Appendix.

According to the conclusion derived in the Appendix, i.e., (51) and (52), conditional probability $p(\mathbf{y}|\mathbf{k})$ is given by

$$p(\mathbf{y}|\mathbf{k}) = N(\mathbf{y}|\boldsymbol{\mu}_{\mathbf{y}},\mathbf{\Sigma}_{\mathbf{y}}) \tag{18}$$

where

$$\mathbf{\Sigma}_{\mathbf{y}} = \mathbf{\Sigma}_{\mathbf{yy}}.$$

As can be identified in (14)

$$\mathbf{\Sigma}_{\mathbf{yy}} = \mathbf{L}^{-1} + \mathbf{A}(\mathbf{k})(\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G})^{-1}\mathbf{A}(\mathbf{k})^{\mathbf{T}}$$

and since $\mathbf{\Sigma}_{\mathbf{y}} = \mathbf{\Sigma}_{\mathbf{yy}}$, we have

$$\mathbf{\Sigma}_{\mathbf{y}} = \mathbf{L}^{-1} + \mathbf{A}(\mathbf{k})(\mathbf{G}^{\mathbf{T}}\mathbf{\Lambda}\mathbf{G})^{-1}\mathbf{A}(\mathbf{k})^{\mathbf{T}} \tag{19}$$

where we have replaced $\mathbf{A}$ with $\mathbf{A}(\mathbf{k})$.

C. Patch-Based Gradient Domain Correlation

Equation (19) shows that blur kernel $\mathbf{k}$ is contained in the covariance matrix of blurred image $\mathbf{\Sigma}_{\mathbf{y}}$. Given $\{\mathbf{y}_{\mathbf{i}}\}$ as samples of $\mathbf{y}$, which is Gaussian distributed, the maximum likelihood estimation of $\mathbf{\Sigma}_{\mathbf{y}}$ is defined as

$$\tilde{\mathbf{\Sigma}}_{\mathbf{y}} = \frac{1}{M}\sum_{i=1}^{M}(\mathbf{y}_i-\mu)(\mathbf{y}_i-\mu)^T \tag{20}$$

where $\{\mathbf{y}_i\}$ represent the $m \times m$ image patches in the degraded image, $\mu = E(\mathbf{y})$ represents the sample mean, and M is the number of image patches. Since the size of these image patches is much smaller than the size of the image, we can usually get enough image patches to compute a faithful covariance matrix. We then transform (19) to get the covariance matrix of gradient domain blurred image patches. According to the commutativity property of convolution operator [12], (19) can be rewritten as

$$\mathbf{\Sigma}_{\mathbf{y}} = \mathbf{L}^{-1} + \mathbf{G}^{-1}\mathbf{A}(\mathbf{k})\mathbf{\Lambda}^{-1}\mathbf{A}(\mathbf{k})^{\mathbf{T}}(\mathbf{G}^{\mathbf{T}})^{-1}. \tag{21}$$

Then, the gradient domain covariance matrix is given by

$$\begin{aligned} \mathbf{R}_{\mathbf{yy}} &= \mathbf{G}\tilde{\mathbf{\Sigma}}_{\mathbf{y}}\mathbf{G}^{\mathbf{T}} \\ &= \mathbf{G}\mathbf{L}^{-1}\mathbf{G}^{\mathbf{T}} + \mathbf{A}(\mathbf{k})\mathbf{\Lambda}^{-1}\mathbf{A}(\mathbf{k})^{\mathbf{T}}. \end{aligned} \tag{22}$$

This covariance matrix can be estimated from the gradient image directly, since

$$\mathbf{R}_{\mathbf{yy}} = \frac{1}{M}\sum_{i=1}^{M}(\mathbf{G}\mathbf{y}_i-\mu)(\mathbf{G}\mathbf{y}_i-\mu)^T \tag{23}$$

where M is the number of image patch samples. Because the noise is i.i.d. Gaussian (8), noise covariance matrix $\mathbf{L}^{-1}$ is diagonal. Morever, we have

$$\mathbf{G}\mathbf{L}^{-1}\mathbf{G}^{\mathbf{T}} = \sigma_n^2\mathbf{G}\mathbf{G}^{\mathbf{T}}. \tag{24}$$

The gradients of the clear natural image are assumed to be independent, which indicates that the covariance of image gradients $\mathbf{\Lambda}^{-1}$ (9) is also diagonal, and we have

$$\mathbf{A}(\mathbf{k})\mathbf{\Lambda}^{-1}\mathbf{A}(\mathbf{k})^{\mathbf{T}} = \sigma_x^2\mathbf{A}(\mathbf{k})\mathbf{A}(\mathbf{k})^{\mathbf{T}} \tag{25}$$

where σ_x^2 is the variance of the image gradients.

Finally, the maximum likelihood estimation of $\mathbf{k}$ satisfies

$$\mathbf{R}_{\mathbf{yy}} = \sigma_x^2\mathbf{A}(\mathbf{k})\mathbf{A}(\mathbf{k})^{\mathbf{T}} + \sigma_n^2\mathbf{G}\mathbf{G}^{\mathbf{T}} \tag{26}$$

based on previous discussion.

A simple way of estimating convolution kernel $\mathbf{k}$ from the given covariance matrix is the covariance matching technique [13]. However, it

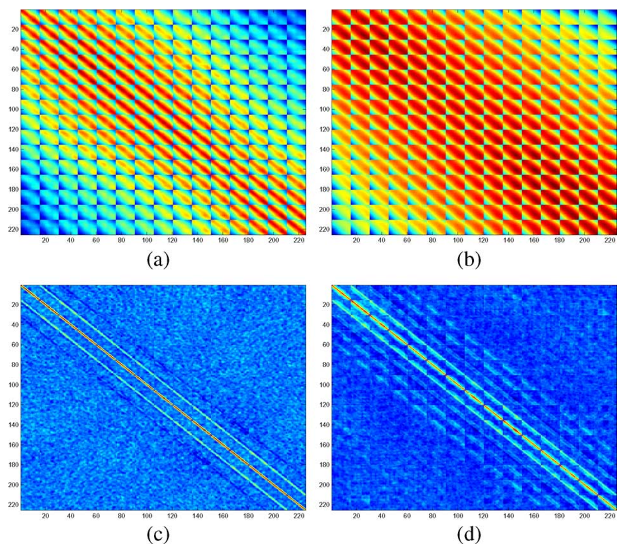

Fig. 1. Covariance matrices of the image patches. (a) Covariance matrix of the clean image. (b) Covariance matrix of the blurry image. (c) Covariance matrix of the gradient domain clean image. (d) Covariance matrix of the gradient domain blurry image. Note that all the covariance matrices here are approximately block Toeplitz matrices, and the pixels in the blurry image are more correlated than in the clean image.

cannot handle large kernels that commonly appear in a motion blur or a camera shake since the optimization is very difficult. As previously shown, unknown convolution kernel $\mathbf{k}$ is contained in sample covariance $\mathbf{R_{yy}}$; we now develop a simple and efficient method to extract the kernel.

D. Convert Covariance Matrix Into an Autocorrelation Function

For the sake of simplicity, we represent $\mathbf{A}(\mathbf{k})$ as $\mathbf{A}$ in the rest of this paper. Let

$$\mathbf{R_A} = \mathbf{A}\mathbf{A}^T \tag{27}$$

and

$$\mathbf{R_G} = \mathbf{G}\mathbf{G}^T. \tag{28}$$

We choose $\mathbf{G}$ to be a very compact gradient operator, whose convolution kernel is $\mathbf{g}$, i.e.,

$$\mathbf{g} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}. \tag{29}$$

Thus, $\mathbf{G}\mathbf{G}^T$ is approximately a diagonal matrix, which is verified by its autocorrelation function in Fig. 1; see the experimental section for details. According to (26), we have

$$\mathbf{R_{yy}} = \sigma_x^2 \mathbf{R_A} + \sigma_n^2 \mathbf{R_G}. \tag{30}$$

For the clean natural image, σ_x^2 is usually much larger than σ_n^2; therefore, $\mathbf{R_A}$ weighs much more than $\mathbf{R_G}$ in $\mathbf{R_{yy}}$. Since both $\mathbf{A}$ and $\mathbf{G}$ are convolution operators, it is easy to show that $\mathbf{R_A}$ and $\mathbf{R_G}$ are both symmetric Toeplitz matrices (4). We now focus on $\mathbf{R_A}$ first, and then, a similar conclusion can be derived for $\mathbf{R_G}$. Let $\mathbf{R_A}(|i-j|)$ be the $(i-j)$th diagonal elements of $\mathbf{R_A}$, according to the property of symmetric Toeplitz matrix, we have

$$\mathbf{R_A}(i,j) = \mathbf{R_A}(|i-j|). \tag{31}$$

This means that all the elements on each diagonal of $\mathbf{R_A}$ represent the same correlation property of the convolution kernel since we have

$$\mathbf{R_A}(i,j) = \mathbf{A}(i,:)\mathbf{A}^T(:,j) \tag{32}$$

where $\mathbf{A}$ is defined in (4) and $\mathbf{A}(i,:)$ and $\mathbf{A}^T(:,j)$ represent the elements in the ith row and jth column of $\mathbf{A}$, respectively.

Since $\mathbf{A}$ is a block Toeplitz matrix, the elements in the ith row of $\mathbf{A}$ can be expressed as

$$\mathbf{A}(i,:) = \mathbf{k}_{\text{ext}}^T(i \bmod (N)) \tag{33}$$

according to (3) and (4). Here, $\mathbf{k}_{\text{ext}}^T$ is a $1 \times N$ vector composed by vector form convolution kernel $\mathbf{k}$ with zero padding, i.e.,

$$\mathbf{k}_{\text{ext}}^T = [\mathbf{k}^T, 0, \ldots, 0] \tag{34}$$

where $N = (m+n-1)^2$ is the number of columns of $\mathbf{A}$.

Equation (32) shows that each element in $\mathbf{R_A}$ is actually the autocorrelation coefficient of convolution kernel $\mathbf{k}$. Specifically, the autocorrelation coefficient at the rth lag is

$$\text{ACF}_A(r) = \mathbf{R_A}(|r|). \tag{35}$$

Similarly, for $\mathbf{R_G}$, we also have

$$\text{ACF}_G(r) = \mathbf{R_G}(|r|) \tag{36}$$

based on the aforementioned analysis.

Let $m = 2n-1$, we estimate autocorrelation coefficient ACF(r) by averaging the rth diagonal elements of $\mathbf{R_{yy}}$ as follows:

$$\text{ACF}(r) = \begin{cases} \frac{1}{m^2-r} \sum_{i=1}^{m^2-r} \mathbf{R_{yy}}(i, i+r), & r \geq 0 \\ \frac{1}{m^2-|r|} \sum_{i=1}^{m^2-|r|} \mathbf{R_{yy}}(i+|r|, i), & r < 0 \end{cases} \tag{37}$$

where m is the size of the sample image patch and $-\lfloor m^2/2 \rfloor \leq r \leq \lfloor m^2/2 \rfloor$. Then, we have

$$\text{ACF}_A(r) = \text{ACF}(r)\text{ACF}_G(r). \tag{38}$$

Clearly, $\text{ACF}_G(r)$ should be approximated by a delta function since $\mathbf{R_G}$ is approximately a diagonal matrix, as shown in Fig. 1(c). Finally, 1-D autocorrelation function $\text{ACF}_A(r)$ is reshaped back to 2-D $m \times m$ matrix $\mathbf{S}$ as

$$\mathbf{S}(p,q) = \text{ACF}_A(r) \tag{39}$$

where $p = \bmod(r + \lfloor m^2/2 \rfloor + 1, m)$ and $q = [(r + \lfloor m^2/2 \rfloor + 1)/m]$.

Thus, $\mathbf{S}$ is an estimation of the autocorrelation matrix of $n \times n$ blur kernel k.

E. PR

It is well known that the Fourier transform of $\mathbf{S}$ is the power spectrum of convolution kernel $\mathbf{k}$. However, the phase information of the convolution kernel is lost since we only know its autocorrelation properties. The PR technique [6], [7] that recovers the phase given just the magnitude of a signal's Fourier transform is adopted here in order to derive the phase of the convolution kernel. We apply three constraints, i.e., module, compact support, and positivity constraints, to the convolution kernel in the PR algorithm. The algorithm iterates back and forth among three sets to get the kernel.

Letting F represent the Fourier transform, then the Fourier transforms of the kernel and its autocorrelation function can be expressed as $\hat{\mathbf{K}} = F\{\mathbf{k}\}$ and $\hat{\mathbf{S}} = F\{\mathbf{S}\}$. Since we only know the autocorrelation function of the kernel, according to the property of Fourier transform, we have

$$\left|\hat{\mathbf{K}}(u,v)\right| = \left(\left|\hat{\mathbf{S}}(u,v)\right|\right)^{\frac{1}{2}} \tag{40}$$

where

$$\hat{\mathbf{K}}(u,v) = \left|\hat{\mathbf{K}}(u,v)\right| e^{i\theta(u,v)}. \tag{41}$$

We can reconstruct $\mathbf{k}$, or equivalently retrieve phase θ, given $|\hat{\mathbf{K}}|$ and the three constraints on $\mathbf{k}$. Let $\mathbf{T}_1$, $\mathbf{T}_2$, and $\mathbf{T}_3$ be the temporary variables, and the algorithm can be expressed as follows.

1) Initial phase is set to be zero

$$\mathbf{T}_1(u,v) = \left(\left|\hat{\mathbf{S}}(u,v)\right|\right)^{\frac{1}{2}} e^{i\cdot 0}$$

2) **While iteration** <1000

a) Apply Fourier domain constraint

$$\mathbf{T}_2(u,v) = \left(\left|\hat{\mathbf{S}}(u,v)\right|\right)^{\frac{1}{2}} e^{\text{Phase}(\mathbf{T}_1(u,v))i}$$

b) Transform into spatial domain and apply spatial domain constraint

$$\mathbf{T}_3(p,q) = \left|F^{-1}\left\{\mathbf{T}_3(u,v)\right\}\right|$$

Set $\mathbf{T}_3(p,q) = 0$, if (p,q) is out of the support template.

c) Transform into Fourier domain

$$\mathbf{T}_1(u,v) = F\left\{\mathbf{T}_3(p,q)\right\}$$

End While

3) Finally

$$\begin{aligned}\hat{\mathbf{K}}_{\text{est}}(u,v) &= \left(\left|\hat{\mathbf{S}}(u,v)\right|\right)^{\frac{1}{2}} e^{\text{Phase}(\mathbf{T}_1(u,v))i} \\ \mathbf{k}_{\text{est}} &= \left|F^{-1}\{\hat{\mathbf{K}}_{\text{est}}\}\right|\end{aligned}.$$

The experimental results show that this simple PR algorithm gives promising results, as demonstrated as follows.

III. Experimental Results

Here, we use the database from [5], which is composed of four natural images blurred by eight different convolution kernels caused by a camera shake for evaluation. The image size is 255 × 255, and the kernels' support varies from 10 to 25 pixels. First, we compute the gradient of the input blurred image using the gradient operator in (29), and then, its sample covariance matrix is calculated according to (23). Second, the autocorrelation matrix of blur kernel $\mathbf{S}$ is estimated from (37) and (39). Finally, the true kernel is derived using the PR technique described in Section II.

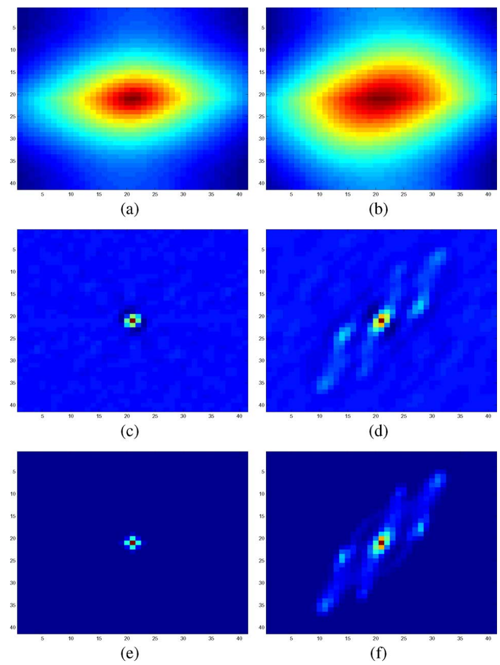

Fig. 2. Autocorrelation functions computed from the patch-based image model. (a) Image domain autocorrelation of the clean image. (b) Image domain autocorrelation of the blurry image. (c) Gradient domain autocorrelation of the clean image. (d) Gradient domain autocorrelation of the blurry image. (e) Autocorrelation of the gradient operator. (f) Autocorrelation of the true kernel.

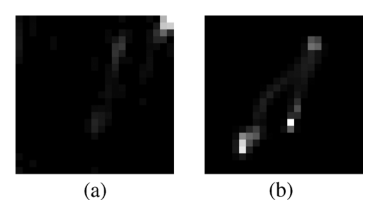

Fig. 3. Structure of the output kernel of the PR matches well with the ground truth, except for the flipping and shifting. (a) Output of phase retrieval algorithm. (b) Ground truth kernel.

In the first step, the covariance matrices of image patches are demonstrated in Fig. 1. We show both image and gradient domain covariance matrices of the clean and blurred images. As shown, all of these matrices are approximately block Teoplitz matrices, which verifies our theoretical analysis in Section II. Furthermore, the gradient domain covariance matrix is much sparser than the image domain covariance matrix. The covariance matrix of the gradient domain clean image is approximately diagonal, and it is more correlated in the blurry image. We will explore this correlation property to get the blurring kernel in the following steps.

In the second step, we convert covariance matrices derived from the first step into autocorrelation functions according to (37). For comparison, we demonstrate both image and gradient domain results. As shown in Fig. 2(a) and (b), it is hard to identify the convolution kernel

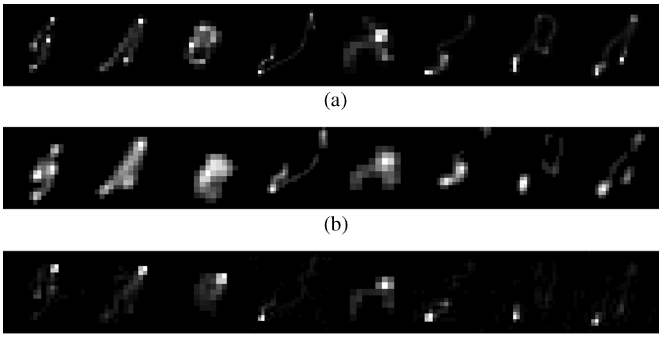

Fig. 4. Comparison of the estimated kernels from different methods. (a) Ground truth kernels. (b) Estimated results from [2]. (c) Our results.

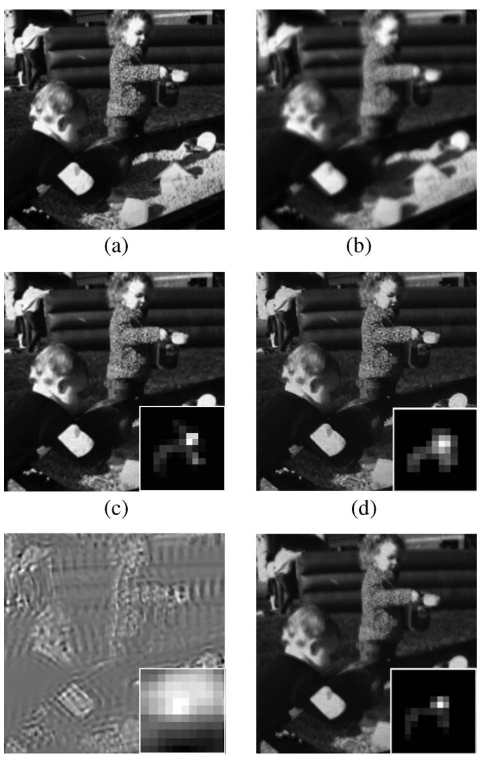

Fig. 5. Visual deconvolution results by various methods. (a) Ground truth. (b) Input, mse: 0.0030, ssim: 0.7897. (c) True kernel, mse: 0.0003, ssim: 0.9725. (d) [7]'s result, mse: 0.0007, ssim: 0.9465. (e) RL's result, mse: 0.1012, ssim: 0.1160. (f) Our result, mse: 0.0006, ssim: 0.9458.

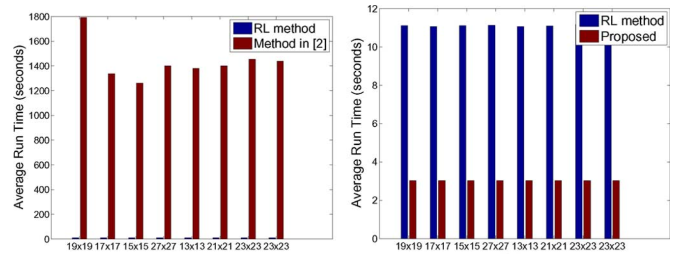

Fig. 6. Computational speed comparison. (a) Computational speed comparison of the RL method and the method in [2]. (b) Computational speed comparison of the RL method and the proposed method.

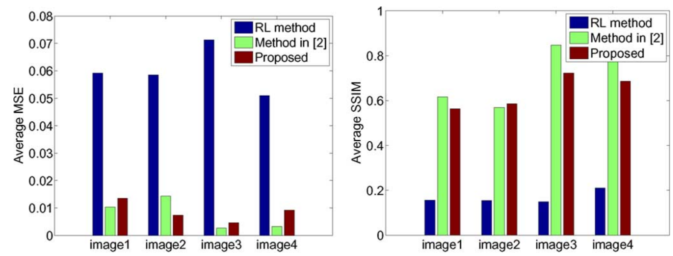

Fig. 7. Computational speed comparison. (a) MSE comparison of different blind deconvolution methods. (b) SSIM comparison of different blind deconvolution methods.

from image domain autocorrelation functions. However, the gradient domain autocorrelation functions [see Fig. 2(c) and (d)] show that their structures match well with the ground truth [see Fig. 2(e) and (f)]. Fig. 2(c) also verifies the assumption that the gradients of the clean image are uncorrelated to each other (9). Clearly, gradient domain statistics provides much more useful information than image domain statistics for blurring kernel identification.

In the third step, the blurring kernel is estimated from the autocorrelation function demonstrated in Fig. 2(d) through the PR algorithm proposed in Section II. It should be noted that the original result might be a mirrored or shifted version of the true kernel since all of them have the same autocorrelation function (see Fig. 3). However, it is easy to validate and correct since wrong kernels give very different deblurred result [5]. The final estimation results are demonstrated in Fig. 4 for comparison.

Finally, We compare the deblurred results of the proposed method with the classical Richardson–Lucy (RL) [14], [15] blind deconvolution method and the state-of-the-art method in [2].

As displayed in Fig. 5, RL gives poor results for natural images, whereas the proposed method provides better results in terms of MSE than the others. Since MSE is not an efficient metric for measuring perceptual fidelity, we also use SSIM index [16] for a higher perceptual relevance. As shown, the proposed method gives slightly worse SSIM than [2], but it is much better than RL. The method in [2] is based on variational Bayesian inference and multiscale estimation techniques, which are very time consuming. It takes more than 20 min to get a kernel on a 2.8-GHz central processing unit. On the other hand, our method only takes less than 10 s. The run times of different methods on 255×255 images blurred by varying kernels are demonstrated in Fig. 6.

The deblurred results of four images blurred by eight kernels are evaluated in Fig. 7(a) and (b) in both MSE and SSIM. We average the MSE and SSIM results of eight different kernels for each image. As shown, overall, the leading method in [2] gives the best results. Our method provides comparable results with much less computational burden, which is the major advantage of the proposed technique.

IV. Conclusion

In this paper, an efficient PSF estimation algorithm has been proposed based on natural image gradient domain correlation properties. The experimental results show that the proposed method produces promising PSF estimation results, compared with an existing method. The PSF estimated by the proposed method gives comparable deblurred results and also provides a good initialization for the more sophisticated PSF estimation methods such as in [2] and [4]. Moreover, the presented autocorrelation matrix provides an efficient way to extract and represent blurring information in the image, which is useful for image retrieval or image quality evaluation [17], [18]. These will be our future research topics.

Appendix

Suppose x and y are two Gaussian-distributed random variables, we define

$$\mathbf{z} = \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}. \tag{42}$$

The corresponding mean vector is given by

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_\mathbf{x} \\ \boldsymbol{\mu}_\mathbf{y} \end{bmatrix} \tag{43}$$

and the covariance matrix is given by

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{xx}} & \boldsymbol{\Sigma}_{\mathbf{xy}} \\ \boldsymbol{\Sigma}_{\mathbf{yx}} & \boldsymbol{\Sigma}_{\mathbf{yy}} \end{bmatrix}. \tag{44}$$

The precision matrix is defined as

$$\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1} = \begin{bmatrix} \boldsymbol{\Lambda}_{\mathbf{xx}} & \boldsymbol{\Lambda}_{\mathbf{xy}} \\ \boldsymbol{\Lambda}_{\mathbf{yx}} & \boldsymbol{\Lambda}_{\mathbf{yy}} \end{bmatrix}. \tag{45}$$

Joint distribution $p(\mathbf{z})$ can be expressed as

$$\begin{aligned} \ln p(\mathbf{z}) = & -\frac{1}{2}(\mathbf{z}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{z}-\boldsymbol{\mu}) + \mathrm{const} \\ = & -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_\mathbf{x})^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{xx}}(\mathbf{x}-\boldsymbol{\mu}_\mathbf{x}) \\ & -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_\mathbf{x})^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{xy}}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y}) \\ & -\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y})^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{yx}}(\mathbf{x}-\boldsymbol{\mu}_\mathbf{x}) \\ & -\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y})^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{yy}}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y}) + \mathrm{const}. \end{aligned} \tag{46}$$

Since we want to integrate out $\mathbf{x}$, we first identify the terms involving $\mathbf{x}$. We have

$$\begin{aligned} \ln p(\mathbf{z}) = & -\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y})^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{yy}}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y}) + \mathbf{y}^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{yx}}\boldsymbol{\mu}_\mathbf{x} \\ & -\frac{1}{2}\mathbf{x}^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{xx}}\mathbf{x} + \mathbf{x}^{\mathrm{Tm}} + \mathrm{const} \end{aligned} \tag{47}$$

where

$$\mathbf{m} = \boldsymbol{\Lambda}_{\mathbf{xx}}\boldsymbol{\mu}_\mathbf{x} - \boldsymbol{\Lambda}_{\mathbf{xy}}(\mathbf{y}-\boldsymbol{\mu}_y). \tag{48}$$

We can see that terms related to $\mathbf{x}$ show a standard quadratic form; we then complete the square for $\mathbf{x}$ and get

$$\begin{aligned} \ln p(\mathbf{x},\mathbf{y}) = & -\frac{1}{2}\left(\mathbf{x}-\boldsymbol{\Lambda}_{\mathbf{xx}}^{-1}\mathbf{m}\right)^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{xx}}\left(\mathbf{x}-\boldsymbol{\Lambda}_{\mathbf{xx}}^{-1}\mathbf{m}\right) \\ & -\frac{1}{2}\mathbf{m}^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{xx}}^{-1}\mathbf{m} - \frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y})^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{yy}}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y}) \\ & + \mathbf{y}^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{yx}}\boldsymbol{\mu}_\mathbf{x} + \mathrm{const}. \end{aligned} \tag{49}$$

As shown, $\mathbf{x}$ is Gaussian distributed, and integration over terms involving $\mathbf{x}$ gives constant; then, we have

$$\begin{aligned} \ln p(\mathbf{y}) = & -\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y})^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{yy}}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y}) \\ & + \mathbf{y}^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{yx}}\boldsymbol{\mu}_\mathbf{x} - \frac{1}{2}\mathbf{m}^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{xx}}^{-1}\mathbf{m} + \mathrm{const} \\ = & -\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_\mathbf{y})^{\mathrm{T}}(\boldsymbol{\Lambda}_{\mathbf{yy}} - \boldsymbol{\Lambda}_{\mathbf{xy}}^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{xx}}^{-1}\boldsymbol{\Lambda}_{\mathbf{xy}}) \\ & \times (\mathbf{y}-\boldsymbol{\mu}_\mathbf{y}) + \mathrm{const}. \end{aligned} \tag{50}$$

As shown, $p(\mathbf{y})$ is a Gaussian distribution function; then, we have

$$p(\mathbf{y}) = N(\mathbf{y}|\boldsymbol{\mu}_\mathbf{y}, \boldsymbol{\Sigma}_\mathbf{y}). \tag{51}$$

According to the relationship between covariance and precision matrices (45) and the block matrix inversion rule [10], [11], we have

$$\begin{aligned} \boldsymbol{\Sigma}_\mathbf{y} &= \left(\boldsymbol{\Lambda}_{\mathbf{yy}} - \boldsymbol{\Lambda}_{\mathbf{xy}}^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathbf{xx}}^{-1}\boldsymbol{\Lambda}_{\mathbf{xy}}\right)^{-1} \\ &= \boldsymbol{\Sigma}_{\mathbf{yy}}. \end{aligned} \tag{52}$$

Acknowledgment

The authors would like to thank the anonymous reviewers and the associate editor for their helpful suggestions and valuable comments. W. Hu would also like to thank Dr. S. Song for proofreading of the original manuscript. Part of this work was done when W. Hu was a visiting student at Massachusetts Institute of Technology.

References

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

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

[3] A. Levin, "Blind motion deblurring using image statistics," in *Proc. Adv. NIPS*, 2006, pp. 841–848.

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

[5] A. Levin, Y. Weiss, F. Durand, and W. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1964–1971.

[6] J. R. Fienup, "Phase retrieval algorithms: A comparison," *Appl. Opt.*, vol. 21, no. 15, pp. 2758–2769, Aug. 1982.

[7] M. Hayes, "The reconstruction of a multidimensional sequence from the phase or magnitude of its Fourier transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-30, no. 2, pp. 140–154, Apr. 1982.

[8] C. Hansen, J. G. Nagy, and D. O'Leary, *Deblurring Images: Matrices, Spectra, and Filtering*. Philadelphia, PA: SIAM, 2006.

[9] D. H. Brainard and W. T. Freeman, "Bayesian color constancy," *J. Opt. Soc. Amer. A*, vol. 14, no. 7, pp. 1393–1411, Jul. 1997.

[10] R. A. Horn and C. R. Johnson, *Matrix Analysis*. New York: Cambridge Univ. Press, 1986.

[11] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. New York: Springer-Verlag, 2006.

[12] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[13] B. Ottersten, P. Stoica, and R. Roy, "Covariance matching estimation techniques for array signal processing applications," *Digit. Signal Process.*, vol. 8, no. 3, pp. 185–210, Jul. 1998.

[14] W. H. Richardson, "Bayesian-based iterative method of image restoration," *J. Opt. Soc. Amer.*, vol. 62, no. 1, pp. 55–59, Jan. 1972.

[15] L. B. Lucy, "An iterative technique for the rectification of observed distributions," *Astronom. J.*, vol. 79, no. 6, pp. 745–754, Jun. 1974.

[16] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[17] R. Liu, Z. Li, and J. Jia, "Image partial blur detection and classification," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[18] M. J. Chen and A. Bovik, "No-reference image blur assessment using multiscale gradient," in *Proc. Int. Workshop QoMEx*, 2009, pp. 70–74.