Novel scheme for Lane-Bates' blind deconvolution: Determinant conditions for the zeros of blurs and a simple algorithm for eliminating blurs

Ikuo MORITANI[1)]
Souichirou AOGAKI[1)]
Takao SUGAI[1)]
Fujio TAKEUTCHI[2)]
Fumiharu Masafumi TOYAMA[1)]
1) *Department of Information and Communication Sciences, Kyoto Sangyo University, Japan*
2) *Department of Computer Sciences, Kyoto Sangyo University, Japan*

Abstract

The Lane-Bates method of blind deconvolution makes it possible to analytically recover the original image without prior knowledge of blurs convolved in a given image. The method utilizes the zeros of the z-transform of the given image. Its implementation, however, requires highly non-trivial analysis of the zeros. We have developed a novel scheme that considerably simplifies the analysis of the zeros. We have developed two versions of the scheme, i.e., determinant conditions (DCs) for the zeros of blurs and a search algorithm (SA) of blur images. The DCs consist of two forms, i.e., a derivative form and a multi-point form. The derivative form is given as a determinant form of conditions on derivatives of the zeros of assumed blurs that can be evaluated by using zeros of the z-transform of the given image. On the other hand, the multi-point form is given as a determinant form of conditions on the zeros of assumed blurs that are evaluated at multiple points in z space. The scheme is particularly powerful when the blurs have multiple structures as we illustrate. The SA is given as a form of simultaneous equations for blur elements of an assumed blur. The method is powerful when we try to find a single blur. This method is robust for compressed gray-scale images. These methods have been experimentally tested with model blurred images and shown to be powerful. In this report we illustrate how they are useful for the Lane-Bates blind deconvolution.

1. Introduction

Lane and Bates' (LB) blind deconvolution enables us to remove blurs convolved in a given image without prior knowledge of the point-spread mechanism that caused the blurring [1]. The method, however, requires highly nontrivial analysis of the zeros (zero-sheets) of the z-transform of the given image. We have recently presented a novel scheme that makes such an analysis almost unnecessary [2-9]. In this paper we give a comprehensive summary of the scheme. We have devised two versions of the scheme. One is given as determinant forms of conditions on the zeros and their derivatives of assumed blurs. The other is given as a form of search algorithm for finding assumed blurs. We refer to the former one as determinant conditions (DCs) and to the latter one as a search algorithm (SA) throughout this report.

The DCs enable us to find the zeros of blurs of assumed sizes without classifying the zero-sheets. The DCs are particularly powerful when the blurs have multiple structures as we illustrate. We have developed two forms of the DCs, i.e., a derivative form and a multi-point form. The derivative form is given as a determinant form of conditions on derivatives of the zeros of assumed blurs. The derivatives of the zeros are analytically given in terms of the whole zeros of the z-transform of a given image and are evaluated at a single point in z space. This is a great advantage of this method. On the other hand, the multi-point form is given as a determinant form of conditions on the zeros evaluated at multiple points in z space.

The SA is given as a form of simultaneous equations for blur elements of an assumed blur. This method is powerful when we try to find a single blur. This method is also robust even in a situation where the condition of the perfect convolution is broken.

We have tested these methods experimentally with model blurred images and shown to be powerful [2-9]. In this report we show that this novel scheme is useful in implementing the LB blind deconvolution. In Sec. 2 we present the details of the DCs and the search algorithm. In Sec. 3 we illustrate how they work for deconvolution of given images. Section 4 is for the summary.

2. Determinant conditions for zeros of blurs and a search algorithm

We consider a model of an observed image $g(x, y)$ that is given as the convolution of a true image $f(x, y)$ and a blurring function $h(x, y)$

$$g(x, y) = f(x, y) * h(x, y), \tag{1}$$

where no additional noise is involved and it is understood that the blurring is shift-invariant. Here x and y are nonnegative integers. The $f(x, y)$ and $h(x, y)$ are both unknown. The sizes of $f(x, y)$

and $h(x,y)$ are denoted with $M \times N$ and $m \times n$ respectively. The size of $g(x,y)$ is then given as $M' \times N' \equiv (M+m-1) \times (N+n-1)$. The z-transform $G(u,v)$ of the given image function $g(x,y)$ is written as

$$G(u,v) = \frac{1}{M'N'} \sum_{x=0}^{M'-1} \sum_{y=0}^{N'-1} g(x,y) u^x v^y, \tag{2}$$

where u and v are complex variables. Similarly, the z-transforms $F(u,v)$ and $H(u,v)$ of f and h are respectively written as

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) u^x v^y, \tag{3}$$

$$H(u,v) = \frac{1}{mn} \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} h(x,y) u^x v^y. \tag{4}$$

Then Eq. (1) implies

$$G(u,v) = F(u,v) H(u,v). \tag{5}$$

Consider the equation $H(u,v) = 0$ with a given u, which is an $(n-1)$-th order polynomial for unknown v. Let its $n-1$ roots be β_i with $i = 1, 2, \cdots, n-1$. Then $H(u,v)$ can be written as

$$H(u,v) = A(u) \prod_{i=1}^{n-1} (v - \beta_i), \tag{6}$$

where $A(u)$ is a function of u. The β_i's determined by $G(u,v)$ contain those of $F(u,v) = 0$ and $H(u,v) = 0$. Provided that we can identify the roots of $H(u,v) = 0$, we can determine $H(u,v)$ except for factors $A(u)$. We repeat the same procedure as above but with a given v. We denote the roots of this equation with γ_j with $j = 1, 2, \cdots, m-1$. As we discuss later, the consistency between the results of the above two procedures allows us to determine $H(u,v)$.

The problem is, how to separate the roots β_i's (γ_i's) that belong to $H(u,v) = 0$ from those of $F(u,v) = 0$. To perform this LB's fundamental procedure requires us to classify the zero-sheets of $G(u,v)$ when u (v) is varied. This is a highly nontrivial analysis. The novel scheme that we present in the following sections simplifies the analysis to a considerable extent.

2.1 Derivative form of DC

The root β_i of $H(u,v) = 0$ is a function with of u. Note that $H(u, \beta_i)$ is identically zero by definition. Therefore the following equations hold,

$$\frac{d^j}{du^j} H(u, \beta_i) = 0, \quad j = 0, \cdots, mn-1. \tag{7}$$

The above mn equations enable us to express the mn unknown elements of $h(x,y)$ in terms of u and the derivatives of β_i with respect to u, $\beta^{(j)} \equiv d^j \beta / du^j$ ($j = 0, 2, \cdots, mn-1$). Actually all higher derivatives of $H(u, \beta_i)$ are also zero. Hence we can use derivatives of any orders higher than those of Eq. (7). The results for $h(x, y)$, however, should be independent of the choice of the derivatives.

Let the matrix that consists of the coefficients of the mn elements of $h(x, y)$ be C and its determinant det$C$. Matrix C is complex in general. We require that Eq. (7) has a nontrivial solution, that is, the mn elements of $h(x, y)$ are not all zero. This requires det$C = 0$. We refer to this as version 1 of the DC. Hereafter we denote the DC for a $m \times n$ blur with $E^{\rho_u}_{m \times n}(\beta(u))$. As an illustration let us consider the case of $m = 2$ and $n = 3$. Then we obtain

$$C = \begin{pmatrix} 1 & \beta_i & \beta_i^2 & \rho_u e^{-i\phi_u} & \rho_u e^{-i\phi_u}\beta_i & \rho_u e^{-i\phi_u}\beta_i^2 \\ 0 & \beta_i^{(1)} & 2\beta_i\beta_i^{(1)} & e^{-i\phi_u} & e^{-i\phi_u}(\beta_i + \rho_u\beta_i^{(1)}) & e^{-i\phi_u}(\beta_i^2 + 2\rho_u\beta_i\beta_i^{(1)}) \\ 0 & \beta_i^{(2)} & 2({\beta_i^{(1)}}^2 + \beta_i\beta_i^{(2)}) & 0 & e^{-i\phi_u}(2\beta_i^{(1)} + \rho_u\beta_i^{(2)}) & 2e^{-i\phi_u}(2\beta_i\beta_i^{(1)} + \rho_u{\beta_i^{(1)}}^2 + \rho_u\beta_i\beta_i^{(2)}) \\ 0 & \beta_i^{(3)} & 2(3\beta_i^{(1)}\beta_i^{(2)} + \beta_i\beta_i^{(3)}) & 0 & e^{-i\phi_u}(3\beta_i^{(2)} + \rho_u\beta_i^{(3)}) & \begin{array}{c} 2e^{-i\phi_u}(3{\beta_i^{(1)}}^2 + 3\beta_i\beta_i^{(2)} + 3\rho_u\beta_i^{(1)}\beta_i^{(2)} \\ +\rho_u\beta_i\beta_i^{(3)}) \end{array} \\ 0 & \beta_i^{(4)} & \begin{array}{c} 2(3{\beta_i^{(2)}}^2 + 4\beta_i^{(1)}\beta_i^{(3)} \\ +\beta_i\beta_i^{(4)}) \end{array} & 0 & e^{-i\phi_u}(4\beta_i^{(3)} + \rho_u\beta_i^{(4)}) & \begin{array}{c} 2e^{-i\phi_u}(12\beta_i^{(1)}\beta_i^{(2)} + 4\beta_i\beta_i^{(3)} + 3\rho_u{\beta_i^{(2)}}^2 \\ +4\rho_u\beta_i^{(1)}\beta_i^{(3)} + \rho_u\beta_i\beta_i^{(4)}) \end{array} \\ 0 & \beta_i^{(5)} & \begin{array}{c} 2(10\beta_i^{(2)}\beta_i^{(3)} + 5\beta_i^{(1)}\beta_i^{(4)} \\ +\beta_i\beta_i^{(5)}) \end{array} & 0 & e^{-i\phi_u}(5\beta_i^{(4)} + \rho_u\beta_i^{(5)}) & \begin{array}{c} 2e^{-i\phi_u}(15{\beta_i^{(2)}}^2 + 20\beta_i^{(1)}\beta_i^{(3)} + 5\beta_i\beta_i^{(4)} \\ +10\rho_u\beta_i^{(2)}\beta_i^{(3)} \\ +5\rho_u\beta_i^{(1)}\beta_i^{(4)} + \rho_u\beta_i\beta_i^{(5)}) \end{array} \end{pmatrix}, \tag{8}$$

where we took as $u = \rho_u e^{-i\phi_u}$ and $\beta^{(j)} \equiv \partial^j \beta / \partial u^j \to \partial^j \beta_i / \partial \rho_u{}^j$ [4] and we dropped the factor 1/mn=1/6. The det$C$ leads to the DC [4, 7]

$$E^{\rho_u}_{2\times 3}(\beta_i) \equiv -135{\beta_i^{(2)}}^6 + 80{\beta_i^{(1)}}^3{\beta_i^{(3)}}^3 + 15{\beta_i^{(1)}}^4{\beta_i^{(4)}}^2 - 60{\beta_i^{(1)}}^3\beta_i^{(2)}\beta_i^{(3)}\beta_i^{(4)} - 12{\beta_i^{(1)}}^4\beta_i^{(3)}\beta_i^{(5)} + 18{\beta_i^{(1)}}^3{\beta_i^{(2)}}^2\beta_i^{(5)} + 270\beta_i^{(1)}{\beta_i^{(2)}}^4\beta_i^{(3)} - 180{\beta_i^{(1)}}^2{\beta_i^{(2)}}^2{\beta_i^{(3)}}^2 = 0. \tag{9}$$

For $\beta_i^{(j)}$, we can take it as derivatives with respect to other parameters. However, they are in fact identical to each other. For instance, when we take as $\beta_i^{(j)} \to \partial^j \beta / \partial \phi_u{}^j$ we have a relation $E^{\phi_u}_{2\times 3}(\beta_i) = \rho_u{}^{12} E^{\rho_u}_{2\times 3}(\beta_i)$ where $E^{\phi_u}_{2\times 3}(\beta_i)$ is the DC obtained by taking $\beta_i^{(j)} \to \partial^j \beta_i / \partial \phi_u{}^j$. This can be verified with the Cauchy-Riemann relation between derivatives of β_i with respect to ρ_u and ϕ_u, i.e., $\partial\beta_i / \partial\phi_u = i\rho_u (\partial\beta_i / \partial\rho_u)$ (the relations between higher degree derivatives are obtained from this relation).

This DC derived for 2×3 implicitly includes DCs for blurs of the sizes smaller than 2×3, i.e., 2×2, 1×3 and 1×2. This is because the det$C$ can be expressed as a linear combination of DCs for the blur elements of the smaller sizes. However, the det$C$ cannot vanish for $i \times 1$ where $i \geq 2$. For instance, we present an explicit decomposition of Eq. (9) into DCs for 2×2 blurs, i.e.,

$$E_{2\times 3}^{\rho_u}(\beta_i) = \sum_{k=1}^{6} A_k(\beta_i) E_{2\times 2}^{\rho_u}(\beta_i)_k, \tag{10}$$

where A_k and $E_{2\times 2}^{\rho_u}(\beta_i)_k$ are given as,

$$A_1(\beta_i) = -2\beta_i^2\beta_i^{(1)}\beta_i^{(3)} + 3\beta_i^{(1)^4} + 3\beta_i^2\beta_i^{(2)^2}, \tag{11}$$

$$A_2(\beta_i) = -6\beta_i\beta_i^{(1)}\beta_i^{(2)^2} + 4\beta_i\beta_i^{(1)^2}\beta_i^{(3)} + 2\beta_i^2\beta_i^{(1)}\beta_i^{(4)} - 12\beta_i^{(2)}\beta_i^{(1)^3} - 4\beta_i^2\beta_i^{(2)}\beta_i^{(3)}, \tag{12}$$

$$A_3(\beta_i) = -5\beta_i\beta_i^{(1)^2}\beta_i^{(4)} - 2\beta_i^2\beta_i^{(1)}\beta_i^{(5)} + 15\beta_i^{(1)^2}\beta_i^{(2)^2} + 15\beta_i\beta_i^{(2)^3} + 20\beta_i^{(3)}\beta_i^{(1)^3} + 5\beta_i^2\beta_i^{(2)}\beta_i^{(4)}, \tag{13}$$

$$A_4(\beta_i) = 27\beta_i^{(1)^2}\beta_i^{(2)^2} - 12\beta_i^{(3)}\beta_i^{(1)^3} - 3\beta_i\beta_i^{(1)^2}\beta_i^{(4)} - 9\beta_i\beta_i^{(2)^3} + 12\beta_i\beta_i^{(1)}\beta_i^{(2)}\beta_i^{(3)} - 3\beta_i^2\beta_i^{(2)}\beta_i^{(4)} + 4\beta_i^2\beta_i^{(3)^2}, \tag{14}$$

$$A_5(\beta_i) = -30\beta_i^{(2)}\beta_i^{(1)^2}\beta_i^{(3)} + 15\beta_i^{(1)^3}\beta_i^{(4)} + 3\beta_i\beta_i^{(1)^2}\beta_i^{(5)} + 15\beta_i\beta_i^{(2)^2}\beta_i^{(3)} + 3\beta_i^2\beta_i^{(2)}\beta_i^{(5)} - 45\beta_i^{(1)}\beta_i^{(2)^3} - 20\beta_i\beta_i^{(1)}\beta_i^{(3)^2} - 5\beta_i^2\beta_i^{(3)}\beta_i^{(4)}, \tag{15}$$

$$A_6(\beta_i) = 30\beta_i\beta_i^{(2)^2}\beta_i^{(4)} - 4\beta_i^2\beta_i^{(3)}\beta_i^{(5)} - 40\beta_i\beta_i^{(3)^2}\beta_i^{(2)} - 60\beta_i^{(1)^2}\beta_i^{(2)}\beta_i^{(4)} + 45\beta_i^{(2)^4} + 5\beta_i^2\beta_i^{(4)^2} + 80\beta_i^{(1)^2}\beta_i^{(3)^2} - 12\beta_i\beta_i^{(1)}\beta_i^{(2)}\beta_i^{(5)} + 20\beta_i\beta_i^{(1)}\beta_i^{(3)}\beta_i^{(4)}, \tag{16}$$

and

$$E_{2\times 2}^{\rho_u}(\beta_i)_1 = -5\beta_i^{(4)^2} + 4\beta_i^{(5)}\beta_i^{(3)}, \tag{17}$$

$$E_{2\times 2}^{\rho_u}(\beta_i)_2 = -5\beta_i^{(3)}\beta_i^{(4)} + 3\beta_i^{(5)}\beta_i^{(2)}, \tag{18}$$

$$E_{2\times 2}^{\rho_u}(\beta_i)_3 = -4\beta_i^{(3)^2} + 3\beta_i^{(4)}\beta_i^{(2)}, \tag{19}$$

$$E_{2\times 2}^{\rho_u}(\beta_i)_4 = -5\beta_i^{(4)}\beta_i^{(2)} + 2\beta_i^{(5)}\beta_i^{(1)}, \tag{20}$$

$$E_{2\times 2}^{\rho_u}(\beta_i)_5 = -4\beta_i^{(3)}\beta_i^{(2)} + 2\beta_i^{(4)}\beta_i^{(1)}, \tag{21}$$

$$E_{2\times 2}^{\rho_u}(\beta_i)_6 = -3\beta_i^{(2)^2} + 2\beta_i^{(3)}\beta_i^{(1)}. \tag{22}$$

Equations (17)-(22) are all DCs for 2×2 blurs. The simplest form $E_{2\times 2}^{\rho_u}(\beta_i)_6$ of Eq. (22) is obtained from Eq. (7) [2]. Other ones are obtained by taking the higher order derivatives than those in Eq. (7). Note that $E_{2\times 2}^{\rho_u}(\beta_i)_2$ and $E_{2\times 2}^{\rho_u}(\beta_i)_5$ are respectively obtained by differentiating $E_{2\times 2}^{\rho_u}(\beta_i)_3$ and $E_{2\times 2}^{\rho_u}(\beta_i)_6$ with respect to ρ_u. Thus, it is seen that $E_{2\times 3}^{\rho_u}(\beta_i)$ for 2×3 blurs implicitly includes the DCs for 2×2 blurs. Similarly, it can be explicitly shown that the DCs of Eqs. (17)-(22) for 2×2 blurs implicitly include the CEs for 1×2 blurs. This implies that $E_{2\times 3}^{\rho_u}(\beta_i)$ for 2×3 of Eq. (9) can detect the zeros of blurs of the sizes 2×3, 2×2, and 1×2 all at once. Equation (9) also includes DCs for $1\times j$ with $j > 3$ blurs (larger one-dimensional blurs). For such one-dimensional blurs, all zeros are independent of u and its derivatives with respective u are all zero. This is why Eq. (9) includes the DCs for $1\times j$ with $j > 3$ blurs implicitly. This feature of the DC, which holds for blurs of any sizes in general, enables us to detect all zeros of any blurs convolved in the given image if we choose sufficiently large values for m and n. For illustration we have shown the DC by assuming a 2×3 blur but it is straightforward, although tedious, to extend it to blurs of larger sizes.

Although we examined DCs for the zeros of $H(u,v)$, in the actual analysis we have to start with $G(u,v)$ that is defined in terms of the given image $g(x,y)$. Recall that we have no prior knowledge of the blurring function $h(x,y)$. With a given u, we numerically solve $G(u,v)=0$ for unknown v. Some of these roots can be the roots of $H(u,v)=0$ with the same given u. In that case, Eq. (7) with $H(u,v)$ replaced by $G(u,v)$ holds. Let one of such roots be β, i.e., assume that $H(u,\beta)=0$. Then Eq. (7) holds for this β. In order to work out the matrix elements of C, we have to evaluate derivatives of β with respect to u. Because $h(x,y)$ is still unknown, we cannot use Eq. (7) to determine these derivatives. The β, however, is a solution of Eq. (7) with $H(u,\beta)$ replaced by $G(u,\beta)$, which we can use to determine the derivatives of β. The derivatives can explicitly be written down in the form of rational functions of u. As examples we give analytical expressions for the derivatives of β_i, up to the fourth order. From $d^k G(\rho_u e^{-i\phi_u},\beta_i)/d\rho_u{}^k = 0$ ($k=1,\cdots,5$), the derivatives of the zeros β_i are given as,

$$\beta_i^{(1)} = -\frac{\left(\dfrac{\partial G}{\partial \rho_u}\right)}{\left(\dfrac{\partial G}{\partial \beta_i}\right)}, \tag{23}$$

$$\beta_i^{(2)} = -\frac{1}{\left(\dfrac{\partial G}{\partial \beta_i}\right)}\left(\frac{\partial^2 G}{\partial \rho_u{}^2} + 2\frac{\partial^2 G}{\partial \rho_u \partial \beta_i}\beta_i^{(1)} + \frac{\partial^2 G}{\partial \beta_i{}^2}\beta_i^{(1)^2}\right), \tag{24}$$

$$\beta_i^{(3)} = -\frac{1}{\left(\dfrac{\partial G}{\partial \beta_i}\right)}\left(\frac{\partial^3 G}{\partial \rho_u{}^3} + 3\frac{\partial^3 G}{\partial \rho_u{}^2 \partial \beta_i}\beta_i^{(1)} + 3\frac{\partial^3 G}{\partial \rho_u \partial \beta_i{}^2}\beta_i^{(1)^2} + 3\frac{\partial^2 G}{\partial \rho_u \partial \beta_i}\beta_i^{(2)} + 3\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(1)}\beta_i^{(2)} + \frac{\partial^3 G}{\partial \beta_i{}^3}\beta_i^{(1)^3}\right), \tag{25}$$

$$\begin{aligned}\beta_i^{(4)} = -\frac{1}{\left(\dfrac{\partial G}{\partial \beta_i}\right)}\Bigg(&\frac{\partial^4 G}{\partial \rho_u^4} + 4\frac{\partial^4 G}{\partial \rho_u^3 \partial \beta_i} + 6\frac{\partial^4 G}{\partial \rho_u^2 \partial \beta_i^2}\beta_i^{(1)^2} + 4\frac{\partial^4 G}{\partial \rho_u \partial \beta_i^3}\beta_i^{(1)^3} + \frac{\partial^4 G}{\partial \beta_i^4}\beta_i^{(1)4} + 6\frac{\partial^3 G}{\partial \rho_u^2 \partial \beta_i}\beta_i^{(2)} \\ &+ 4\frac{\partial^2 G}{\partial \rho_u \partial \beta_i}\beta_i^{(3)} + 12\frac{\partial^3 G}{\partial \rho_u \partial \beta_i^2}\beta_i^{(1)}\beta_i^{(2)} + 6\frac{\partial^3 G}{\partial \beta_i^3}\beta_i^{(1)2}\beta_i^{(2)} + 3\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(2)^2} + 4\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(1)}\beta_i^{(3)}\Bigg),\end{aligned} \tag{26}$$

$$\begin{aligned}\beta_i^{(5)} = -\frac{1}{\left(\dfrac{\partial G}{\partial \beta_i}\right)}\Bigg(&\frac{\partial^5 G}{\partial \rho_u^5} + 5\frac{\partial^5 G}{\partial \rho_u^4 \partial \beta_i}\beta_i^{(1)} + 10\frac{\partial^5 G}{\partial \rho_u^3 \partial \beta_i^2}\beta_i^{(1)^2} + 10\frac{\partial^4 G}{\partial \rho_u^3 \partial \beta_i}\beta_i^{(2)} + 10\frac{\partial^5 G}{\partial \rho_u^2 \partial \beta_i^3}\beta_i^{(1)^3} + 30\frac{\partial^4 G}{\partial \rho_u^2 \partial \beta_i^2}\beta_i^{(1)}\beta_i^{(2)} \\ &+ 5\frac{\partial^5 G}{\partial \rho_u \partial \beta_i^4}\beta_i^{(1)^4} + 30\frac{\partial^4 G}{\partial \rho_u \partial \beta_i^3}\beta_i^{(1)^2}\beta_i^{(2)} + \frac{\partial^5 G}{\partial \beta_i^5}\beta_i^{(1)^5} + 10\frac{\partial^4 G}{\partial \beta_i^4}\beta_i^{(1)^3}\beta_i^{(2)} + 10\frac{\partial^3 G}{\partial \rho_u^2 \partial \beta_i}\beta_i^{(3)} + 20\frac{\partial^3 G}{\partial \rho_u \partial \beta_i^2}\beta_i^{(1)}\beta_i^{(3)} \\ &+ 5\frac{\partial^2 G}{\partial \rho_u \partial \beta_i}\beta_i^{(4)} + 15\frac{\partial^3 G}{\partial \rho_u \partial \beta_i^2}\beta_i^{(2)^2} + 15\frac{\partial^3 G}{\partial \beta_i^3}\beta_i^{(1)}\beta_i^{(2)^2} + 10\frac{\partial^3 G}{\partial \beta_i^3}\beta_i^{(1)^2}\beta_i^{(3)} + 10\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(2)}\beta_i^{(3)} + 5\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(1)}\beta_i^{(4)}\Bigg),\end{aligned} \tag{27}$$

where

$$\frac{\partial^{(p+q)}G}{\partial\rho_u{}^p\partial\beta_i^q} = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} g(x,y)x(x-1)\cdots(x-p+1)\rho_u{}^{x-p}e^{-ix\phi_u}y(y-1)\cdots(y-q+1)\beta_i^{y-q}. \tag{28}$$

In Eq. (28) $p, q = 0, 1, 2, \cdots, 5$, with $p + q = 5$. In this way the derivatives of the zeros β_i are given in terms of known $g(x, y)$ and the zeros β_i that are evaluated at a point of u. If Eq. (9) is satisfied with the chosen β, it means that the given image contains the blurs of the sizes 2×3, 2×2 or $1\times j$ with $j \geq 2$. We repeat the same procedure for all of the roots of $G(u, \beta_i) = 0$ one by one. In this way we can identify the roots that belong to the blurs of the sizes 2×3, 2×2 or $1\times j$ with $j \geq 2$. We do not have to go through complicated analysis of the "zero-sheets". The size of the blurring function can be much larger. The det$C$ for $h(x, y)$ of larger sizes case can be worked out explicitly. If we test det$C = 0$ for all the roots of $G(u, \beta) = 0$, we can in principle identify all of the β's of $H(u, \beta) = 0$, including those of blurs of smaller sizes that are contained.

Even when we find all the β's associated with $H(u, v)$, we cannot determine $H(u, v)$ in the sense that we do not know factor $A(u)$ of Eq. (6). The $H(u, v)$, however, can be fully determined by repeating the same procedure for zeros with respect to variable u (with a fixed v). If $m = n$, the DC for γ_i's can be obtained from the DC for β_i by the following substitutions: $\beta_i \to \gamma_i$, $\rho_u \to \rho_v$, $\phi_u \to \phi_v$, $x \rightleftarrows y$ in the DC for β_i and relevant equations, but $g(x, y)$ must be left as it is. However, if $m \neq n$, the form of the DC for γ_i is different from that of β_i. As a preparation to the next section, for the 2×3 case we give the explicit form of the DC for γ_i, i.e.,

$$E_{2\times3}^{\rho_v}(\gamma_i) \equiv -40\gamma_i^{(3)^3} + 60\gamma_i^{(2)}\gamma_i^{(3)}\gamma_i^{(4)} - 15\gamma_i^{(1)}\gamma_i^{(4)^2} - 18\gamma_i^{(2)^2}\gamma_i^{(5)} + 12\gamma_i^{(1)}\gamma_i^{(3)}\gamma_i^{(5)} = 0, \tag{29}$$

where we took as $v = \rho_v e^{-i\phi_v}$ and $\gamma_i^{(j)} = d^j\gamma_i/dv^j$ [4, 7].

Before ending this section we should emphasize an interesting feature of the DC of the derivative form. As mentioned before, we can use derivatives of any orders higher than those of Eq. (7) when we construct DCs for an assumed blur. If we use such higher derivatives we obtain more complicated DCs with higher order derivatives of zeros β_i or γ_i. We can also obtain such DCs by differentiating DCs constructed with lower order derivatives of zeros β_i or γ_i. The DCs constructed with the higher order derivatives of the zeros β_i or γ_i are more powerful in detecting the zeros of blurs. This is because, in the higher order derivatives of the zeros, the $\rho_{v(u)}$ dependences of the zeros β_i or γ_i of blurs are much more different from those of non-blur elements. We will demonstrate this aspect of the derivative form of the DCs in the next section.

2.2 Multi-point form of DC

In the preceding section we presented the derivative form of DC. The derivative form of the

DC is very sophisticated in the sense that it can be evaluated at a single point in z space. In this section we present yet another version of the DC, which is a simpler form than the derivative form.

We consider the same situation as that we considered in the previous section, i.e., a situation where noise is absent and a given image $g(x,y)$ can be modeled as the convolution of a true image $f(x,y)$ and a blur image $h(x,y)$ given as Eq. (1). Equation $H(u,v)=0$ with β_i defines β_i as a function of u, i.e., $\beta_i=\beta_i(u)$. The $H(u,\beta_i)$ is identically zero by definition. Therefore the following equations hold,

$$H(u_j,\beta(u_j))=0, \quad j=0,\cdots, mn-1, \tag{30}$$

where β can be any of the β_i's and u_j are different points of u. The mn equations enable us to express mn unknown elements $h(x,y)$ in terms of u_j and $\beta(u_j)\,(j=0,\cdots,mn-1)$. Let the $mn\times mn$ matrix that consists of the coefficients of mn elements $h(x,y)$ be D. The matrix D is given as,

$$D=\begin{pmatrix} 1 & \beta(u_0) & \cdots & \beta(u_0)^{n-1} & u_0 & u_0\beta(u_0) & \cdots & u_0\beta(u_0)^{n-1} & \cdots\cdots\cdots & u_0^{\,m-1} & u_0^{\,m-1}\beta(u_0)^{n-1} & \cdots & u_0^{\,m-1}\beta(u_0)^{n-1} \\ 1 & \beta(u_1) & \cdots & \beta(u_1)^{n-1} & u_1 & u_1\beta(u_1) & \cdots & u_1\beta(u_1)^{n-1} & \cdots\cdots\cdots & u_1^{\,m-1} & u_1^{\,m-1}\beta(u_1)^{n-1} & \cdots & u_1^{\,m-1}\beta(u_1)^{n-1} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & & \vdots & \vdots & & \vdots \\ 1 & \beta(u_q) & \cdots & \beta(u_q)^{n-1} & u_q & u_q\beta(u_q) & \cdots & u_q\beta(u_q)^{n-1} & \cdots\cdots\cdots & u_q^{\,m-1} & u_q^{\,m-1}\beta(u_q)^{n-1} & \cdots & u_q^{\,m-1}\beta(u_q)^{n-1} \end{pmatrix}, \tag{31}$$

where $q=mn-1$ and we dropped the factor $1/mn$ [6, 9]. We require that Eq. (30) has nontrivial solutions, that is, mn elements of $h(x,y)$ are not all zero. This requires $\det D=0$. We refer to this as version 2 of the DC. Hereafter we denote the DC for $m\times n$ blur with $E^u_{m\times n}(\beta(u_0),\cdots,\beta(u_{mn-1}))$. The $E^u_{m\times n}$ implicitly includes the DCs for any blurs of sizes smaller than $m\times n$, except for the size $r\times 1\,(r=1,\cdots,m)$. This follows from the structure of the determinant $\det D$. Further, the DC can detect the zeros of blurs of the size $1\times k\,(k=n+1,\cdots,N)$. This can be seen as follows. In such a one dimensional blur the zeros are all constant, i.e., do not depend on u. Hence when we substitute the zeros of $1\times k$ blurs into $\det D$, at least two column vectors become proportional to each other. This is why the DC for the $m\times n$ blurs detects the zeros of $1\times k$ blurs.

Version 2 is simpler than version 1 in the sense that we do not have to evaluate derivatives of the zeros. However, there is a drawback in the sense that in this version 2 we have to solve β_i at mn different values of $u_\ell\,(\ell=0,\cdots,mn-1)$ numerically. That is, we have to find β_i of the blurs for each of step $u_j\,(j=0,\cdots,M-1)$ by solving mn equations on β_i. To do this we take mn different points $u_\ell\,(\ell=0,1,\cdots,mn-1)$ in the vicinity of each u_j, i.e., as $u_{j,\ell}=u_j+\ell\Delta u$. This requires an optimization of the parameter Δu. Namely, to hold the "same" β_i we have to take Δu sufficiently small. By the "same" β_i, we mean the β_i's such that, when $u_{j,\ell}$ are all reduced to the same value, then all the associated β_i's coalesce. On the other hand, if we take Δu too small, β_i solved for mn different points $u_{j,\ell}$ become closer to each other. This may cause more than two row vectors in det$D$ to be nearly proportional to each other. Thus, det$D \approx 0$ may be caused, which does not necessarily mean that the condition is satisfied. Therefore, we have to determine an optimal Δu to accomplish the detection process successfully.

In order to complete the image restoration we need the DC also for the variable u. The DC for u is given in a form similar to det$D$. The DC for u is denoted as $E^{v}_{m\times n}(\gamma_i(v_0), \cdots, \gamma_i(v_{mn-1}),)$ where γ_i are the zeros of the blurs.

2.3 A simple algorithm for eliminating blurs

In this section we present an algorithm for finding blurs, which is a new tool for the LB method but different from the DCs that we presented in the previous sections. In this new algorithm, for a larger size of the blur the amount of computation stays still relatively small compared with those in the DCs. Although the DCs we have presented in the last two sections work very well in the situation where the condition of the perfect convolution holds, they are not so robust against a break of the condition of the perfect convolution. On the other hand, this new search algorithm can be also extended to a situation where the condition of the perfect convolution is broken by some reason.

First we consider the same situation as that we considered in the previous sections, i.e., a situation where noise is absent and a given image $g(x, y)$ can be modeled as the convolution of a true image $f(x, y)$ and a blur image $h(x, y)$ given as Eq. (1). For a given u, the solutions v of the equation $G(u, v) = 0$ are denoted by $\beta^u{}_i (i = 1, 2, \cdots, N'; N' \leq N + n - 1)$. Then $G(u, v)$ of Eq. (2) can be expressed as

$$G(u_j, v) = k_j \prod_{i=0}^{N'-1} \left(v - \beta^j{}_i\right), \tag{32}$$

where suffix j means different value of u.

The blur function H can also be expanded in the same manner as above;

$$H(u_j, v) = p_j \prod_{i=0}^{n-1} \left(v - \alpha^j{}_i\right), \tag{33}$$

where $\alpha^j{}_i$ is the i-th solution of $H(u_j, v)$. Actually Eq. (33) is the same as Eq. (6). From Eq. (2), it follows that $\{\alpha^j\} \subset \{\beta^j\}$.

The RHS of Eq. (33) can be expanded as

$$H(u_j, v) = p_j \sum_{y=0}^{n-1} c_y v^y, \tag{34}$$

where c_y is the coefficient of the degree y term, all containing $\alpha^j{}_0$ through $\alpha^j{}_{n-1}$ except for $c_{n\text{-}1}$ which is just 1. Thus we obtain n equations

$$\begin{aligned}
&\sum_{x=0}^{m-1} h(x,0)u_j^{\,x} = p_j c_0 = (-1)^{n-1} p_j \prod_{i=1}^{n-1} \alpha^j{}_i \\
&\sum_{x=0}^{m-1} h(x,1)u_j^{\,x} = p_j c_1 = (-1)^{n-2} p_j \left(\prod_{i=2}^{n-1} \alpha^j{}_i + \prod_{i=1(i\neq 2)}^{n-1} \alpha^j{}_i + \cdots + \prod_{i=1}^{n-2} \alpha^j{}_i \right) \\
&\qquad \vdots \\
&\sum_{x=0}^{m-1} h(x,n-1)u_j^{\,x} = p_j c_{n-1} = p_j,
\end{aligned} \tag{35}$$

which can be considered as a set of simultaneous equations for unknowns $h(x,y)$ and p_j [5, 8].

Our aim is to determine the blur functions $h(x,y)$ from Eq. (35). However, Eq. (35) has $mn+1$ unknown variables. On the other hand, the number of independent equations in Eq. (35) is n. Therefore, in order to obtain $h(x,y)$ uniquely, one has to evaluate Eq. (35) at q (i.e., $j=1,\cdots,q$) different values of u_j, where q can be taken as the smallest integer that satisfies $q \geq mn/(n-1)$. Then, one can solve Eq. (35) with $j=1,\cdots,q$ for $h(x,y)$ and p_j. If Eq. (35) has nontrivial solutions for $h(x,y)$ and p_j's, the zero-values $\alpha^j{}_i(i=1,2,\cdots,n-1)$ are those of an $m\times n$ blur and the solutions $h(x,y)$ form the blur matrix. On the other hand, if Eq. (35) has no solution, it means that the zero-values $\alpha^j{}_i(i=1,2,\cdots,n-1)$ are not those of the $m\times n$ blur. For all combinations of $n-1$ β^j's, one repeats this procedure. The number of the possible combinations is ${}_{N'-1}C_{n-1}$ for a given $M'\times N'$ image. Once an $m\times n$ blur matrix $h(x,y)$ is found, one can construct the z-transform $H(u,v)$ of it. Hence, one can obtain the z-transform $F(u,v)$ of a real image $f(x,y)$ simply by $F(u,v)=G(u,v)/H(u,v)$, which allows us to restore the true image by the inverse Fourier transform.

The search algorithm for finding $m\times n$ blur matrix $h(x,y)$ is summarized as follows:

1. Determine the smallest integer q that satisfies $q \geq mn/(n-1)$.
2. Pick up combinations of $n-1$ zero-values β from $\beta_i(i=1,2,\cdots,N';N'\leq N+m-1)$ of the variable v of the z-transform of the given image.
3. Evaluate the $n-1$ zero-values β at q different points. Then, solve (35) for the blur $h(x,y)$ and unknown constants $p_j(j=1,\cdots,q)$.
4. If the set of equations give a non-trivial solution, then those β's are identified as the α's that are solutions of $H(u_j,v)$.
5. Find nontrivial solutions by repeating the procedures 1. ~ 3. at most ${}_{N'-1}C_{n-1}$ times until the condition 4. is met.
6. Restore the true image $f(x,y)$ by removing the blur using $F(u,v)=G(u,v)/H(u,v)$.

When one chooses q different values for u_j's, it is advisable to select values that are not very different from each other. In that case, if the i-th solution $\beta^j{}_i$ is identified as a α^j, the chance that $\beta^{j'}{}_i$ at another value $u_{j'}$ is also a solution $\alpha^{j'}$ is large.

We have constructed the algorithm in terms of the zeros of the variable v. Of course we may use the zeros $\gamma_i (i = 1, 2, \cdots, N' - 1)$ of u instead of v in constructing the same algorithm. This version can simply be obtained by the replacements $N \rightarrow M$, $n \rightarrow m$, $\beta_i \rightarrow \gamma_i$, and $u_j \rightarrow v_j$.

Next we consider a situation where the perfect convolution given in Eq. (1) is broken by some reasons. In this situation, the simultaneous equations (35) do not hold exactly because $\alpha^j{}_i$ in Eq. (35) are not exact solutions of Eq. (35). We then define χ^2 by Eq. (36) and determine unknown $h(x,y)$ and p_j to minimize the χ^2,

$$\begin{aligned}\chi^2 = &\left|\sum_{x=0}^{m-1} h(x,0)u_j^x - (-1)^{n-1} p(u_j)\prod_{i=1}^{n-1}\alpha_i(u_j)\right|^2 + \\ &\left|\sum_{x=0}^{m-1} h(x,1)u_j^x - (-1)^{n-2} p(u_j)\left\{\prod_{i=2}^{n-1}\alpha_i(u_j) + \prod_{i=1(i\neq 2)}^{n-1}\alpha_i(u_j) + \cdots + \prod_{i=1}^{n-2}\alpha_i(u_j)\right\}\right|^2 + \cdots + \quad (j = 1, \cdots, q)\ , \qquad (36) \\ &\left|\sum_{x=0}^{m-1} h(x,n-1)u_j^x - p(u_j)\right|^2 .\end{aligned}$$

In the next section it is shown that this algorithm is robust against a break of the condition of the perfect convolution.

3. Illustrations : Tests of the DCs and search algorithm

In the preceding section we have given two types of the DCs and a simple search algorithm for blurs. In this section, we illustrate the image restoration with the DCs and the simple search algorithm and show how this novel scheme is useful mathematical tools in implementing the LB's blind deconvolution.

3.1 Tests of the DCs

Figure 1 shows model images used for illustration in this section. Figure 1(a) shows a model image that we regard as a true image. The image has been downloaded from the web site of Ref. [10], which has been often used in many references to demonstrate image restorations. Figures 1(b), 1(c), 1(d), and 1(e) represent blur images of the sizes 1×2, 2×1, 2×2 and 2×3, respectively. We convolved these four blurs into the true image of Fig. 1(a). Here, it should be stressed that the gray tones of these blur images have been chosen unintentionally. Fig. 1(f) shows the convolved image, of which size is 43×44.

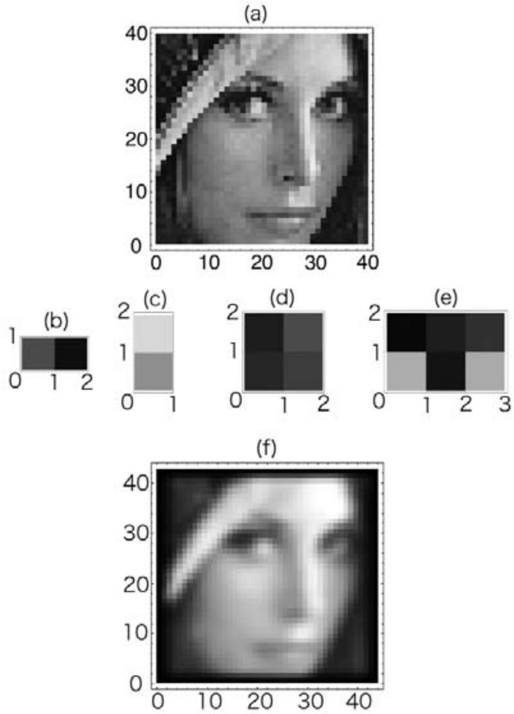

Fig. 1. (a): True image of 40×40 size that we took from [10]. (b): Blur image of 1×2 size. (c): Blur image of 2×1 size. (d): Blur image of size 2×2. (e): Blur image of size 2×3. (f): The image that was obtained by convolving the four blurs of (b)-(e) into the true image of (a). The size of the convolved image is 43×44.

We first illustrate how zeros of the blurs can be detected through $E^{\rho_u}_{2\times 3}(\beta)$ of Eq. (9) and $E^{\rho_v}_{2\times 3}(\gamma)$ of Eq. (29). For the parameter ρ_u and ρ_v we take them to be $\rho_u = \rho_v = 1$. We can take any values for ρ_u and ρ_v in principle. However, in order to accomplish the image-restoration effectively we have to choose optimal values for them. Figure 2 shows the results of numerical evaluations of the DCs for the convolved image shown in Fig. 1(f). We carried out the evaluation at $\phi_u = 2\pi j/43$ ($j = 0, \cdots, 42$) and $\phi_v = 2\pi k/44$ ($k = 0, \cdots, 43$). Note that when we restore the image by the inverse Fourier transform we need the representation of $G(u, v)$ at these points of ϕ_u and ϕ_v. In Fig. 2 we plot $\log[|E^{\rho_u}_{2\times 3}(\beta_i)| + 1]$ and $\log[|E^{\rho_v}_{2\times 3}(\gamma_i)| + 1]$ only for four points of each of ϕ_u and ϕ_v. As we have stressed in the preceding section, $E^{\rho_u}_{2\times 3}(\beta_i)$ and $E^{\rho_v}_{2\times 3}(\gamma_i)$ both include DCs for blurs of smaller sizes 1×2, 2×1, 2×2 and 2×3 implicitly. Therefore, $E^{\rho_u}_{2\times 3}(\beta_i)$ exhibits four ($= 1 + 1 + 2$) zeros. When there exists a degenerate zero in the 2×3 blur, the number of zeros is reduced to three. On the other hand, the number of zeros γ_i that can be detected through $E^{\rho_v}_{2\times 3}(\gamma_i)$ is just three ($= 1 + 1 + 1$).

As seen in Fig. 2(a), for $\phi_u = 0$, four zeros β_1, β_{13}, β_{20}, and β_{21} are exhibited by $E^{\rho_u}_{2\times3}(\beta_i)$. Also at other ϕ_u four zeros are well exhibited by $E^{\rho_u}_{2\times3}(\beta_i)$. Here, note that the solution-numbers of the detected zeros are different at each ϕ_u. In the evaluation of the DC we solved β_i numerically by *Mathematica*, separately at each ϕ_u. Accordingly, the ordering of the numerical solutions (zeros) varies depending on ϕ_u.

For $\phi_v = 0$, as seen in Fig. 2(b), three zeros γ_1, γ_3 and γ_{12} are detected through $E^{\rho_v}_{2\times3}(\gamma_i)$, as expected. Also at other points ϕ_v three zeros are clearly detected by $E^{\rho_v}_{2\times3}(\gamma_i)$. As we mentioned earlier, $E^{\rho_u}_{2\times3}(\beta_i)$ and $E^{\rho_v}_{2\times3}(\gamma_i)$ exhibit the zeros of blurs of not only the size 2×3 but

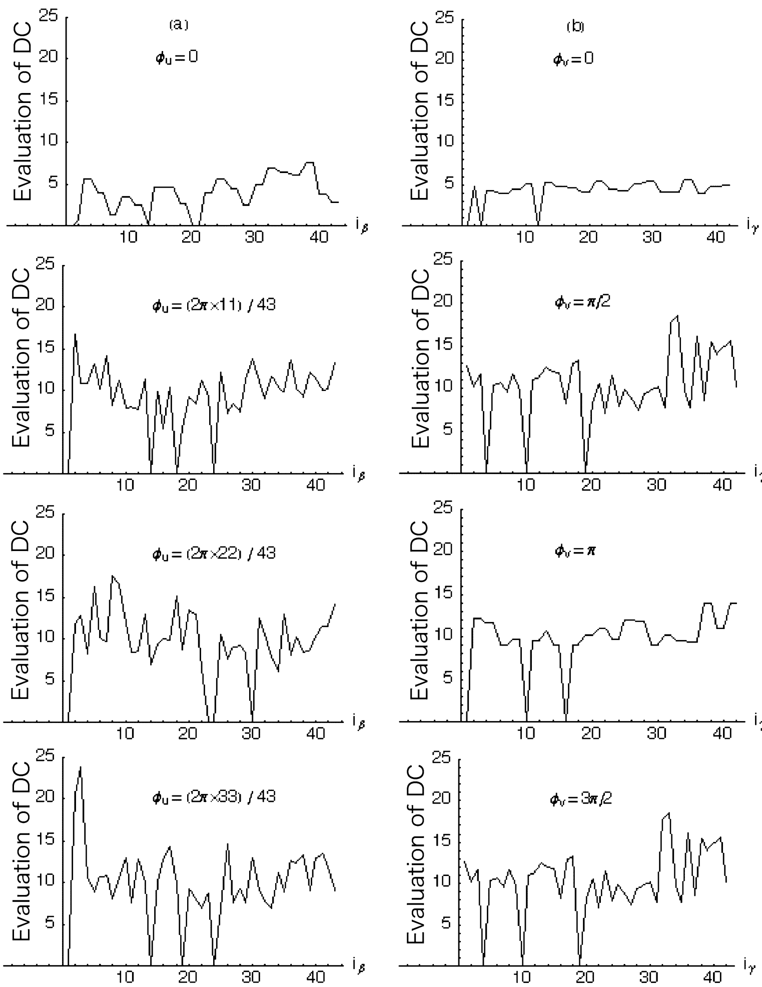

Fig. 2. Results of the evaluations of $E^{\rho_u}_{2\times3}(\beta_i)$ and $E^{\rho_v}_{2\times3}(\gamma_i)$ of Eq.(9) and Eq.(29). In (a) and (b), $\log[|E^{\rho_u}_{2\times3}(\beta_i)|+1]$ and $\log[|E^{\rho_v}_{2\times3}(\gamma_i)|+1]$ are plotted for the zero-value number i_β and i_γ. We took ρ_u and ρ_v as $\rho_u=\rho_v=1$.

also the sizes 1×2, 2×1 and 2×2, all at once. The results of Fig. 2 verify that the detection of the zeros was done successfully.

We have shown that version 1 of the DCs works very well in detecting zeros of multiple blurs that are convolved in a given image. Next, we illustrate version 2 by using the same image as that used for the illustration of version 1.

In Fig. 3 we show the results of the numerical evaluations of the DCs $E^{u}_{2\times 3}(\beta_i(u_0), \cdots, \beta_i(u_5))$ and $E^{v}_{2\times 3}(\gamma_i(v_0), \cdots, \gamma_i(v_5))$ where we plot $\log[|E^{u}_{2\times 3}(\beta_i(u_0), \cdots, \beta_i(u_5))| \times 10^{50} + 1]$ and

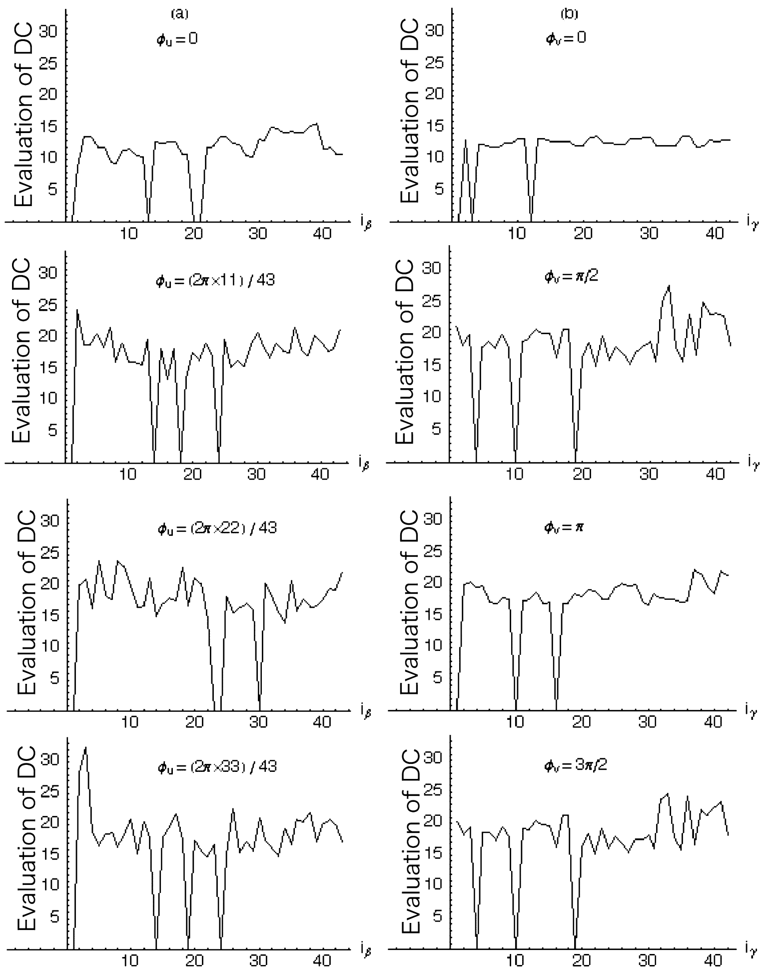

Fig. 3. Results of the evaluations of the DCs $E^{u}_{2\times 3}(\beta_i(u_0), \cdots, \beta_i(u_5))$ and $E^{v}_{2\times 3}(\gamma_i(v_0), \cdots, \gamma_i(v_5))$. (a) and (b) show the results of the evaluations for β_i and γ_i, respectively. We took ρ_u and ρ_v as $\rho_u = \rho_v = 1$.

$\log[|E^{v}_{2\times 3}(\gamma_i(v_0),\cdots,\gamma_i(v_5))|\times 10^{50}+1]$. We took Δu and Δv as $\Delta u=\Delta v=\rho e^{-i\Delta\phi}$ and used $\Delta\phi=\pi/2150$ and $\rho=1$, as optimized parameters. We took a very small value of $\Delta\phi$ so that the "same" β_i is hold in sampling mn points. We mentioned earlier that there can be misjudgment of the zeros of blurs. To avoid such a situation we need to evaluate the DC in high precision. This is why we multiplied $E^{u}_{2\times 3}$ and $E^{v}_{2\times 3}$ by the factor 10^{50} in evaluating the DC.

As we stressed in the preceding section, both of $E^{u}_{2\times 3}(\beta_i(u_0),\cdots,\beta_i(u_5))$ and $E^{v}_{2\times 3}(\gamma_i(v_0),\cdots,\gamma_i(v_5))$ include DCs for blurs of smaller sizes 1×2, 2×1 and 2×2 implicitly. Therefore, we can find altogether four $(=1+1+2)$ zeros β_i with $E^{u}_{2\times 3}(\beta_i(u_0),\cdots,\beta_i(u_5))$. When there exists a degenerate zeros in the 2×3 blur, the number of the zeros that we can find may be three $(=1+1+1)$. On the other hand, the number of zeros γ_i that must be found with $E^{v}_{2\times 3}(\gamma_i(v_0),\cdots,\gamma_i(v_5))$ is just three $(=1+1+1)$. As seen in Fig. 2(a), for $\phi_u=0$, four zeros β_1, β_{13}, β_{20}, and β_{21} are the blurs found through $E^{u}_{2\times 3}(\beta_i(u_0),\cdots,\beta_i(u_5))$. Also at other ϕ_u four zeros of the blurs are clearly found with $E^{u}_{2\times 3}(\beta_i(u_0),\cdots,\beta_i(u_5))$. As seen in Fig. 2(b), for $\phi_v=0$, three zeros γ_1, γ_3 and γ_{12} are found as those of the blurs with $E^{v}_{2\times 3}(\gamma_i(v_0),\cdots,\gamma_i(v_5))$. Also at other points ϕ_v three zeros are clearly detected through $E^{v}_{2\times 3}(\gamma_i(v_0),\cdots,\gamma_i(v_5))$. Note that the zeros detected through $E^{u}_{2\times 3}(\beta_i(u_0),\cdots,\beta_i(u_5))$ and $E^{v}_{2\times 3}(\gamma_i(v_0),\cdots,\gamma_i(v_5))$ are the same as those found with the DCs of version 1 (see Fig. 2). This is natural because we used the same test image for the two illustrations.

Version 2 of the DC obtained from Eq. (31) looks simpler than version 1 that we presented in section 2.1 in the sense that version 2 of the DC can be evaluated with only the zeros. However, when we use version 2 we have to spend extra time to optimize the sampling parameters Δu and Δv.

In Fig. 4 we show the image restored by removing four zeros β_i in v and three zeros γ_i in u. The restored image is the same as the true image of Fig. 1(a). Thus, we confirm that the zeros of

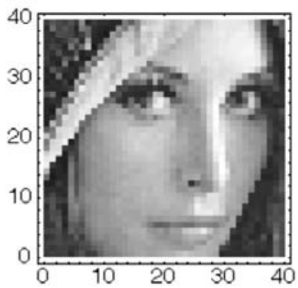

Fig. 4. Image restored by removing the four and three zeros that were respectively detected through $E^{u}_{2\times 3}(\beta_i)$ and $E^{v}_{2\times 3}(\gamma_i)$ of version 1 of the DC or $E^{u}_{2\times 3}(\beta_i(u_0),\cdots,\beta_i(u_5))$ and $E^{v}_{2\times 3}(\gamma_i(v_0),\cdots,\gamma_i(v_5))$ of version 2 of the DC.

blurs convolved in the given image are correctly detected through $E^{u}_{2\times3}(\beta_i(u_0), \cdots, \beta_i(u_5))$ and $E^{v}_{2\times3}(\gamma_i(u_0), \cdots, \gamma_i(u_5))$. The results show that the DC is very powerful tool for the LB blind deconvolution.

In the preceding section we mentioned that the higher order derivative forms of version 1 may be more powerful in detecting the zeros of blurs than the simplest form of the DC obtained by Eq. (7). Before ending this section we illustrate how they are powerful. Figure 5 shows the evaluations of three DCs constructed for 2×2 blurs, i.e, $E^{\rho_u}_{2\times2}(\beta_i)_5 = -4\beta_i^{(3)}\beta_i^{(2)} + 2\beta_i^{(4)}\beta_i^{(1)}$ of

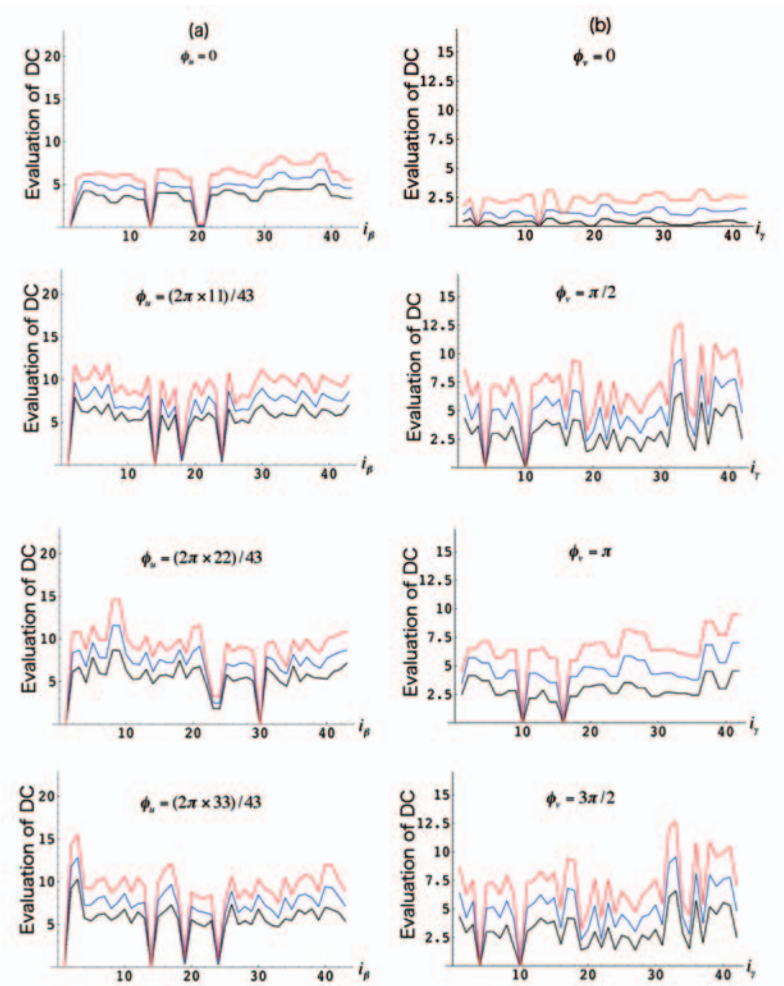

Fig. 5. Comparison of the evaluations of higher order derivative forms of the DCs for 2×2 blurs. (a) shows the evaluations done for the DCs $E^{\rho_u}_{2\times2}(\beta_i)_5' \equiv \partial E^{\rho_u}_{2\times2}(\beta_i)_5/\partial\rho_u = 2\beta_i^{(3)^2} + \beta_i^{(2)}\beta_i^{(4)} - \beta_i^{(1)}\beta_i^{(5)}$ (red lines), $E^{\rho_u}_{2\times2}(\beta_i)_6 = -4\beta_i^{(3)}\beta_i^{(2)} + 2\beta_i^{(4)}\beta_i^{(1)}$ (blue lines) and $E^{\rho_u}_{2\times2}(\beta_i)_6 = -3\beta_i^{(2)^2} + 2\beta_i^{(3)}\beta_i^{(1)}$ (black lines), where actually we plotted $\log[|E^{\rho_u}_{2\times2} \times 10^3 + 1]$. (b) shows the evaluations done for the DCs for the zeros γ_i. The forms of the DCs for γ_i of are the same as those for β_i.

Eq. (21), $E^{\rho_u}_{2\times 2}(\beta_i)_5{}' \equiv \partial E^{\rho_u}_{2\times 2}(\beta_i)_5/\partial\rho_u = 2\beta_i^{(3)^2} + \beta_i^{(2)}\beta_i^{(4)} - \beta_i^{(1)}\beta_i^{(5)}$ and $E^{\rho_u}_{2\times 2}(\beta_i)_6 = -3\beta_i^{(2)^2} + 2\beta_i^{(3)}\beta_i^{(1)}$ of Eq. (22), where we plotted $\log[|E^{\rho_u}_{2\times 2}(\beta)|\times 10^3 + 1]$. Here, note that the forms of the DCs for the zeros γ_i of the variable u are the same as those for the zeros β_i for variable v because the assumed blurs are symmetric 2×2 blurs. The $E^{\rho_u}_{2\times 2}(\beta_i)_6$ is the simplest form that is obtained by Eq. (7). The model image used in this illustration is of Fig. 1(f), in which 2×1, 1×2, 2×2 and 2×3 blurs of Fig. 1(b-e) are convolved. As we mentioned earlier, the DCs constructed for 2×2 blurs should detect the zeros of blurs of the sizes 2×1, 1×2 and 2×2. It should be noted that they never detect the zeros of 2×3 blur. The black, blue and red lines show the evaluations of $E^{\rho_u}_{2\times 2}(\beta_i)_6$, $E^{\rho_u}_{2\times 2}(\beta_i)_5$ and $E^{\rho_u}_{2\times 2}(\beta_i)_4$, respectively. It is seen that the differences between the evaluations of the DCs for the zeros of the 2×2 blur and those for the other zeros are larger in the higher order derivative forms of the DCs. Thus, higher order derivative forms are more powerful in detecting the zeros of the blurs.

3.2 Test of the simple search algorithm for finding blurs

In section 2 we discussed a search algorithm for finding a single blur convolved in a given image. In this section we present one of the results of the tests of image restoration done by means of the search algorithm.

Fig. 6(a) shows a 40×40 model image that we regard as the true image, which is the same as that we used for the tests of the DCs in the preceding section. Figs. 6(b), 6(c) and 6(d) are blurs of sizes 2×2, 2×3 and 3×3, respectively. We convolve the three blurs into the true image. The blurs of the sizes 2×2 and 2×3 are the same as those used in the tests for the DCs. We introduced the 3×3 blur in the present test. Figure 1(e) shows the convolved image. The size of the convolved image is 44×45.

We test how the search algorithm represented by Eq. (35) works in finding each single blur

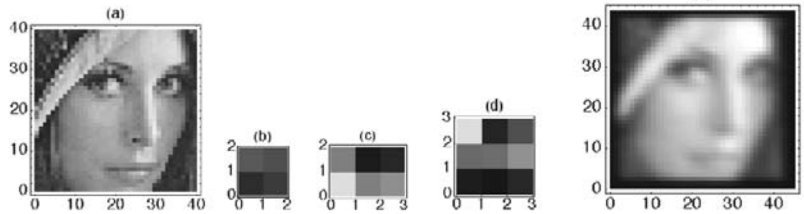

Fig. 6. (a): True image of size 40×40 that we took from [10]. (b): Blur image of size 2×2. (c): Blur image of size 2×3. (d): Blur image of size 3×3. (e): Image obtained by convolving the three blurs of (b), (c), and (d) into (a). The image size is 44×45.

convolved in the true image. To choose u_j's we imposed $|u_j| = 1$, and changed only its phase. Parameter q is 4, 3, and 5 for each blur of Fig. 1(b), 1(c) and 1(d). Figure 7 shows the results of the test. Figure 7(a) shows the image restored by removing the 2×2 blur that has been detected by searching the 2×2 blur in Fig. 7(e). Fig. 7 (b) shows the image restored by removing the 2×3 blur that has been detected by searching for the 2×3 blur in Fig. 7 (a). Finally, Fig. 7 (c) shows the image restored by removing the 3×3 blur that has been detected by searching for the 3×3 blur in Fig. 7 (b). In obtaining the final restored image of Fig. 7 (c) we applied the search algorithm three times. In each search, the algorithm worked very well for blurs of different sizes. In this way we verified that the search algorithm works well in finding a single blur convolved in a given image. This test is just one of the many tests that we have carried out.

Next we show how the search algorithm is robust in a situation where the perfect convolution is broken. Figure 8(a) is the true image that is the same as that used in the tests for the DCs and the search algorithm. Figure 8(c) is the image obtained by convolving the 2×2 blur image of Fig. 8(b). The gray levels are 3852. Figure 8(d) is the image obtained from the image of (c) by com-

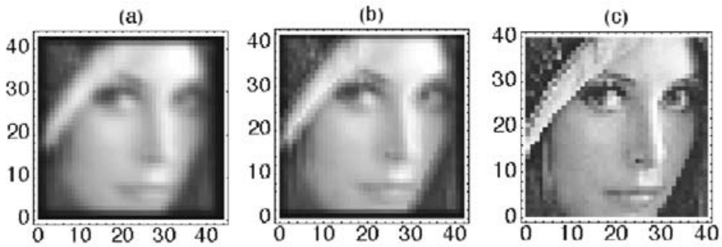

Fig. 7. Restored images by removing three blurs that were found by the search algorithm. (a): restored image by removing the 2×2 blur from the image of Fig. 6(e); (b): restored image by removing the 2×3 blur from the image of (a); (c): restored image by removing the 3×3 blur from the image of (b).

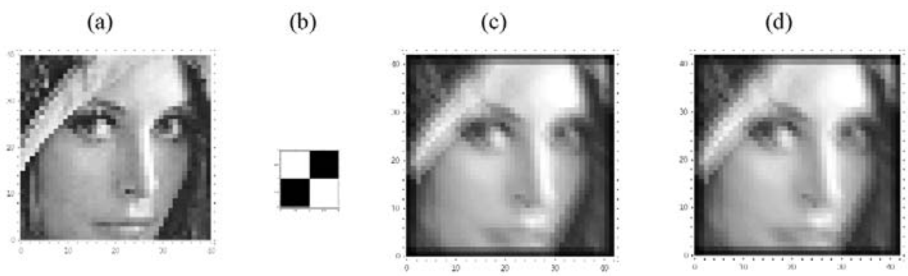

Fig. 8 (a): True image of the size 40×40 that we took from [10]; (b): Blur image of the size 2×2. (c): Image (3852 gray levels) obtained by convolving the blur of (b) in to (a); (d) Image obtained by compressing image (c) to 256 gray levels.

Fig. 9 Restored image from the image of Fig. 8(d).

pressing the gray levels to 256. In the compressed image (d) the condition of the perfect convolution, i.e., $G(u, v) = F(u, v)H(u, v)$ is broken. As we discussed in the preceding section, in this situation the simultaneous equations (35) does not hold exactly. Then, we tried to find blur elements $h(x, y)$ with Eq. (36). Figure 9 shows the restored image by removing the blur of Fig. 8(b). The restored image is surely enhanced. In this way, one can see that the modified search algorithm presented by Eq. (36) works for images with a broken convolution.

4. SUMMARY AND CONCLUSION

We have presented two versions of DCs for finding blurs convolved in a given image, in a situation where no noise is involved. Version 1 of the DCs is given in terms of the derivatives of the zeros of the given image. The derivatives of the zeros are given as rational functions of the zeros themselves of the given image. Hence, the DCs can be evaluated with only the zeros evaluated at a single point in z space. This is a great advantage of the DCs. By using the DCs we can avoid a complicated analysis of the zero-sheets of the given image and reduce the processing time. The DCs constructed for $m \times n$ blurs can actually detect any blurs of the sizes smaller than $m \times n$ all at once. Therefore, when we apply the DCs to the given image, it is advisable to start with the DCs for blurs of sufficiently large size, although we have to take account of the computational complexity of the image restoration process. The DCs are very useful to make the LB blind deconvolution a more practical one. In particular, higher order forms of the DCs are more powerful in detecting the zeros of blurs than the simplest form.

However, version 1 of the DCs are very complicated for blurs of large sizes. This causes a big computational load in the image restoration. Then, we gave the other version of the DCs to solve this problem. Version 2 of the DCs is given in terms of only the zeros of the given image evaluated at multiple points in z space. Therefore, we do not need any derivatives of the zeros of blurs to evaluate the DCs. Hence, in version 2 the computational load is much reduced compared with that of version 1. By using the DC constructed for $m \times n$ blurs we can actually find blurs of any sizes smaller than $m \times n$ all at once. This feature is the same as that of version 1 of the DCs.

The two versions of the DCs are mathematically elegant in the sense that they detect multiple blurs at once. On the other hand, the sizes of the detected blurs cannot be determined, except for a single blur. Then, we presented yet another form of a search method, i.e., a simple search algorithm for finding blurs of a specified size.

The search algorithm is given in the form of solving simultaneous equations for a blur matrix elements. The algorithm is for finding a single blur of a specified size. Once a blur of a given size is detected, and the matrix elements are obtained, one can easily reconstruct the unblurred image. The advantage of this method is that it can be extended easily to blurs of larger sizes. In particular, this search algorithm can be extended to a situation where the perfect convolution is broken. In a compressed image the condition of the perfect convolution is broken. Even in such a situation this search algorithm can be applied to find assumed blurs convolved in given images.

We experimentally tested the DCs and the search algorithm by using test images and examined how they work for finding blurs convolved in original images. We have verified that the algorithm works, in a practical processing time, very well for blurs of small sizes convolved in a middle size image. This novel scheme will be very useful in making use of the LB blind deconvolution.

ACKNOWLEDGMENT

We are grateful to Professor Y. Nogami for discussions and constructive comments. This work was supported in part by the Science Research Promotion from the Promotion and Mutual Aid Corporation for private Schools of Japan, and a grant from Institute for Comprehensive Research, Kyoto Sangyo University.

REFERENCES

[1] Automatic multidimensional deconvolution, R.G. Lane and R.H. Bates, J.Opt. Soc. Am, vol. A **4**, 1987, 180-188 (1987).

[2] Enhancing PET images by means of Bates' blind deconvolution, D. Okano, S. Aogaki, F. Takeutchi and F.M. Toyama, 先端科学技術研究所所報, 第4号 (2005) 69-106.

[3] Enhancing PET images by means of Bates' blind deconvolution, D. Okano, S. Aogaki, F. Takeutchi and F.M. Toyama, 情報処理学会第68回全国大会, 2006年3月9日, 工学院大学, 情報処理学会講演概要集, 第2分冊 2-91.

[4] Novel scheme for blind deconvolution: Derivative form, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, IEEE Fourth Intern. Conf. in Info. Tech. & Appl. 2007, 1/ 211-215;

[5] Simple method of eliminating blurs based on Lane and Bates algorithm, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, IEEE Fourth Intern. Conf. in Info. Tech. & Appl. 2007, 1/227-229.

[6] Novel scheme for blind deconvolution: Multi-point form, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, IEEE Fourth Intern. Conf. in Info. Tech. & Appl. 2007, 1/230-233;

[7] Conditional expressions for blind deconvolution: derivative form, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, 情報処理学会第69回全国大会, 早稲田大学, 2007年3月7日第2分冊, 2-21~22.

[8] Simple method to eliminate blur using Lane Bates alrorithm, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, 情報処理学会第69回全国大会,早稲田大学, 2007年3月7日,第2分冊, 2-25~26.

[9] Conditional expressions for blind deconvolution: multi-point form, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, 情報処理学会第69回全国大会, 早稲田大学, 2007年3月7日, 第2分冊, 2-23~24.

[10] http://www.imageprocessingplace.com/; Rafael C. Gonzalez, Richard E. Woods, Digital image processing, (R.C. Gonzalez, R.E. Woods), 329-341, Second Edition, 2002.