Efficient Methods of Blind Deconvolution Based on the Lane-Bates Algorithm: Comprehensive Summary

Ikuo MORITANI
Souichirou AOGAKI
Hideki KOTAKA
Fujio TAKEUTCHI[1)]
Fumiharu Masafumi TOYAMA

Department of Information and Communication Sciences, Kyoto Sangyo University
1) *Department of Computer Sciences, Kyoto Sangyo University*

Abstract

We developed efficient methods of blind deconvolution on the basis of the Lane-Bates algorithm. The methods consist of two kinds of mathematical tools and their modified versions. We give a comprehensive summary of them in this report.

1. Introduction

Lane and Bates' (LB's) 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 makes use of what the authors call "zero-sheets" of a given image. The zero-sheets are constructed with the zeros of the z-transform of the given image. We will show the detail of this method in Chaps. 2.1, 2.2 and 2.3, and will illustrate how this method works by using a sample image convolving blurs in Chap. 2.4. We also analyzed a medical image that was obtained with positron emission tomography (PET), by applying LB's zero-sheet method. We will present the details of the analysis in Chap. 2.5.

As one can see, the zero-sheets method requires highly nontrivial numerical analysis (classification) of the zeros. That has been a major impediment in implementing LB's method. We have recently developed a novel scheme that makes such an analysis almost unnecessary [2-9]. The rest of the report is devoted to a comprehensive summary of the new scheme.

We have devised a powerful mathematical tool, which is called Determinant Condition (DC) throughout this report. The DCs enable us to find zeros of the blurs of assumed sizes without classifying the zero-sheets of the blurs. The DCs are particularly powerful when the blurs have multiple structures as we illustrate later. We will present the detail of the DC in Chap. 3. We have developed two forms of DCs, i.e., a derivative form and a multi-point form. With the derivative form, which will be explained in Chap. 3.1, the derivatives of the zeros are evaluated using the given image at a single point in z-space, whereas the multi-point form, presented in Chap. 3.2 uses zeros evaluated at several points in the vicinity in z-space.

We tried two ways to reconstruct the image by using DCs. In the first, we follow the LB method i.e., we reconstruct the full true image in the frequency domain and obtain the true image by the inverse Fourier transformation. This method is demonstrated in Chap. 3.3. In the second, which is much simpler, and which we call "Highly efficiency method" in this report, we make use of the matrix evaluated for the DC to calculate the matrix elements of the blur directly, and remove it from the given image. The introduction of this method, together with its demonstration, is done in Chap. 4.

We also tried an approach, which is slightly different from what has been shown so far, namely without making use of the DC. We call this method a "Simple algorithm (SA) " as is presented in Chap. 5. The SA is given as a form of simultaneous equations that the assumed blur elements should satisfy. This method is powerful in finding a single blur. Up to this point, we assumed that no errors are involved in the given image. The main error gives origin to the com-

pression of the gray levels after the convolution of the blur. This problem is explained in the beginning of Chap. 5. It is important to make the algorithm of the deconvolution robust against the error caused by the reduction of the gray levels. The advantage of SA is that it can be modified so that the improved method, to a certain extent, is robust against this error. This procedure is demonstrated in Chap. 5.3.

From the point of view of the robustness against reduced gray levels, the DC, and thus the highly efficient method can also be modified such that one can handle images with reduced gray levels. This trial is explained in Chap. 6. Chap. 7 is for the discussion and summary.

2. Lane and Bates' blind deconvolution

Often blurs are convolved in images. They may originate from measuring equipments (quality of lens or circuit characteristics), environment (atmospheric fluctuation) or movement of the target. LB's method of blind deconvolution is an analytical method that makes it possible to remove the blurs convolved without prior knowledge of the blurs.

In this section, we give the basics of the LB blind deconvolution. The LB method of blind deconvolution makes use of so-called zero-sheets that are constructed with the zeros of the z-transform of an given image. The basic idea of the zero-sheets method was published in 1987 [1].

2.1 Basics of the Zero-sheets method

We consider a model image $g(x, y)$ that is given as the convolution of a true image $f(x, y)$ and a blur $h(x, y)$, i.e.,

$$g(x, y) = f(x, y) * h(x, y). \tag{2.1}$$

No additional noise is included. The pair (x, y) denotes a pixel, and thus x and y are both non-negative integers. The $f(x, y)$ and $h(x, y)$ are both assumed 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 $(M + m - 1) \times (N + n - 1)$. The z-transform $G(u, v)$ of the given image $g(x, y)$ of size $M' \times N'$ is written as

$$G(u, v) = \frac{1}{M' N'} \sum_{x=0}^{M'-1} \sum_{y=0}^{N'-1} g(x, y)\, u^x v^y, \tag{2.2}$$

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

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)\, u^x v^y, \tag{2.3}$$

$$H(u,v) = \frac{1}{mn}\sum_{x=0}^{m-1}\sum_{y=0}^{n-1} h(x,y)\, u^x v^y. \tag{2.4}$$

Then Eq. (2.1) implies

$$G(u,v) = F(u,v)H(u,v). \tag{2.5}$$

That means that, in z-space the convolved image is given as the product of $F(u,v)$ and $H(u,v)$. Consequently, if we can determine $H(u,v)$ by some means, we can reconstruct the true image $f(x,y)$ as the inverse Fourier transform of $G(u,v)$ divided by $H(u,v)$. Although the $G(u,v)$ is given, we have no prior knowledge either of $F(u,v)$ or $H(u,v)$. In order to obtain $F(u,v)$ in such a situation, a highly nontrivial analysis of the zero-sheets is required as we mentioned in the introduction, which we explain in the following.

Consider the equation $G(u_0,v) = 0$ with a fixed $u = u_0$. The left-hand side is an $(N'-1)$-th degree polynomial of the unknown v. Let its $N'-1$ roots be β_k with $k = 1, 2, \cdots, N'-1$. Then $G(u_0,v)$ can be written as

$$G(u_0,v) = A(u_0)\prod_{k=1}^{N'-1}(v-\beta_k), \tag{2.6}$$

where $A(u_0)$ is a constant dependent on u_0. Thus, $G(u,v)$ is determined by the roots (zeros) β_k, except for the constant $A(u_0)$. From Eq. (2.5), one can tell that a zero β_k is either a zero of the equation $F(u_0,v) = 0$ or that of $H(u_0,v) = 0$, (or both). Our objective, therefore, is to find the zeros of $F(u_0,v) = 0$ among zeros of $G(u_0,v) = 0$. We will treat the classification of the zeros in detail in the next section.

2.2 Classification of zeros

Let u_0 be equal to $\rho_u e^{-i\phi_u}$ where ρ_u and ϕ_u are both real $(\rho_u > 0)$. The complex solutions for v of the equation $G(u_0,v) = 0$, plotted on the Gaussian plane form trajectories when ϕ_u varies from 0 to 2π. Each trajectory, alone or some of them combined, form a closed curve. The set of these closed curves is called a "Zero-sheet". As the roots of $G(u_0,v) = 0$ are either those of $F(u_0,v) = 0$ or of $H(u_0,v) = 0$ (or both), the zero-sheet of $G(u_0,v)$ is the superposition of the zero-sheet of $F(u_0,v)$ and that of $H(u_0,v)$. The aim of the zero-sheet method is to facilitate the classification of the roots by making the splitting possible of the zero-sheet of $G(u,v)$ into that of $F(u,v)$ and that of $H(u,v)$.

One of the problems with this method is that it is sometimes difficult to judge if the removed closed curve really belongs to the zero-sheet of $H(u,v)$. However, apart from subtle cases, a removal of wrong trajectories leads to some disastrous results as will be demonstrated later, and the task is relatively easy.

To examine the trajectories one needs to solve the equation $G(u,v)=0$ by varying u with fine steps. As these trajectories cross each other, one needs to solve the equation with a high precision and the computation takes a huge amount of time. This is another problem with this method. We have to deal with these serious problems in order to develop a practical method on the basis of LB's method.

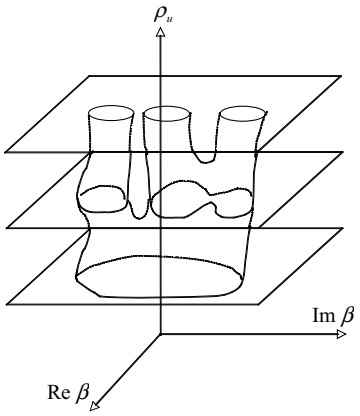


Fig. 2.1 Concept of the ρ_u-dependence of closed curves. For fixed values of ρ_u, ϕ_u is varied from 0 to 2π. A closed curve is formed with one root at some value of ρ_u, but with multiple roots at different values of ρ_u.

2.3 Reconstruction of true image

As we mentioned earlier, if we identify the closed curves of the true image, we can reconstruct the true image only with those closed curves. In the following we present a fundamental method for that.

The discrete Fourier transform of $f(x,y)$ is defined as,

$$\mathcal{F}(\xi,\eta)=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\,e^{-2\pi i\xi x/M}\,e^{-2\pi i\eta y/N}. \tag{2.7}$$

Consequently, if we can determine $\mathcal{F}(\xi,\eta)$ at discrete points $\xi=0,1,\cdots,M-1$ and $\eta=0,1,\cdots,N-1$, we can obtain $f(x,y)$ by the inverse Fourier transformation shown by Eq. (2.8).

$$f(x,y)=\sum_{\xi=0}^{M-1}\sum_{\eta=0}^{N-1}\mathcal{F}(\xi,\eta)\,e^{2\pi i\xi x/M}\,e^{2\pi i\eta y/N}. \tag{2.8}$$

By taking u and v as,

$$u=\exp\left(-2\pi i\,\frac{\xi}{M}\right), \tag{2.9}$$

$$v = \exp\left(-2\pi i \frac{\eta}{N}\right), \tag{2.10}$$

then Eq. (2.7) is written as,

$$\mathcal{F}(\xi,\eta) \to F(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\, u^x v^y. \tag{2.11}$$

Equation (2.11) is just the z-transform of $f(x,y)$. In the following we explain how to determine $\mathcal{F}(\xi,\eta)$ by using the roots of $F(u,v)$.

After classifying the zeros β_k by the zero-sheets method, we determine the zeros β_k of $F(u,v)$ at discrete points of Eq. (2.9). Let us fix ξ to ξ_0. For $u_0 = \exp\left(-2\pi i \xi_0 / M\right)$, Eq. (2.3) is rewritten as,

$$F(u_0,v) = A'(\xi_0)\prod_{k=1}^{N-1}(v-\beta_k). \tag{2.12}$$

Thus, for any v, $F(u_0,v)$ is determined at discrete points of u, except for the constant $A'(\xi_0)$. By putting v of Eq. (2.10) in Eq. (2.12) we have,

$$\mathcal{F}(\xi_0,\eta) = A'(\xi_0)\prod_{k=1}^{N-1}\left(e^{-2\pi i\eta/N} - \beta_k\right). \tag{2.13}$$

To determine the relative sizes of $A'(\xi_0)$, we repeat the same procedure as above but with a given v. Let us fix η to η_0. For $v_0 = \exp\left(-2\pi i \eta_0 / M\right)$, Eq. (2.3) is rewritten as,

$$F(u,v_0) = B'(\eta_0)\prod_{l=1}^{M-1}(u-\gamma_l), \tag{2.14}$$

where $B'(\eta_0)$ is not determined. We substitute u of Eq. (2.9) into Eq. (2.14), then we have,

$$\mathcal{F}(\xi,\eta_0) = B'(\eta_0)\prod_{l=1}^{M-1}\left(e^{-2\pi i\xi/M} - \gamma_l\right). \tag{2.15}$$

One can "normalize" $A'(\xi_0)$ by using Eq. (2.15). Let us fix η_0 to 0 (in fact, it can be any number). Equation (2.15) gives us the relative sizes of $\mathcal{F}(0,0), \mathcal{F}(1,0), \mathcal{F}(2,0), \cdots, \mathcal{F}(M-1,0)$. This suffices to determine the relative sizes of $A'(0), A'(1), A'(2), \cdots, A'(M-1)$ since we already know $\mathcal{F}(0,0), \mathcal{F}(1,0), \mathcal{F}(2,0), \cdots, \mathcal{F}(M-1,0)$ from Eq. (2.13). Thus all the (relative) sizes of $\mathcal{F}(\xi,\eta)$, $(\xi = 0,1,2,\cdots,M-1;\quad \eta = 0,1,2,\cdots,N-1)$ can be determined by the inverse Fourier transformation of $\mathcal{F}(\xi,\eta)$ we have $f(x,y)$.

As we mentioned in the preceding section, it is in fact difficult to classify closed curves with

$\rho_u = 1$. Next, we discuss the case of $\rho_u \neq 1$. By taking u and v such that

$$u = \rho_u \exp\left(-2\pi i \frac{\xi}{M}\right), \tag{2.16}$$

$$v = \rho_v \exp\left(-2\pi i \frac{\eta}{N}\right). \tag{2.17}$$

Eq. (2.3) is rewritten as

$$\mathcal{F}(\xi, \eta) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \rho_u^{\,x} e^{-2\pi i \xi x/M} \rho_v^{\,y} e^{-2\pi i \eta y/N}. \tag{2.18}$$

By the inverse Fourier transform of Eq. (2.18), $\rho_u^{\,x} \rho_v^{\,y} f(x, y)$ is obtained. Consequently, in the case of $\rho_u \neq 1$, the following transformation is needed to obtain the true image,

$$\frac{f_p(x, y)}{\rho_u^{\,x} \rho_v^{\,y}} \rightarrow f(x, y), \tag{2.19}$$

where $f_p(x, y)$ represents the true image obtained by the inverse Fourier transform with $\rho_u \neq 1$.

2.4 Illustration with a sample image

In this section, we illustrate the image restoration by means of the zero-sheet method. Figure 2.2 shows model images used for the illustration. Figure 2.2 (a) indicates an image that we regard as a true image. Figures 2.2 (b), (c), (d) and (e) represent blur images of sizes 4×4, 3×3, 1×4, and 4×1, respectively. We convolved these four blurs into the true image of Fig. 2.2 (a). Here, it should be stressed that these blurs have been chosen only accidentally. Figure 2.2 (f) shows the convolved image, of which the size is 108×108.

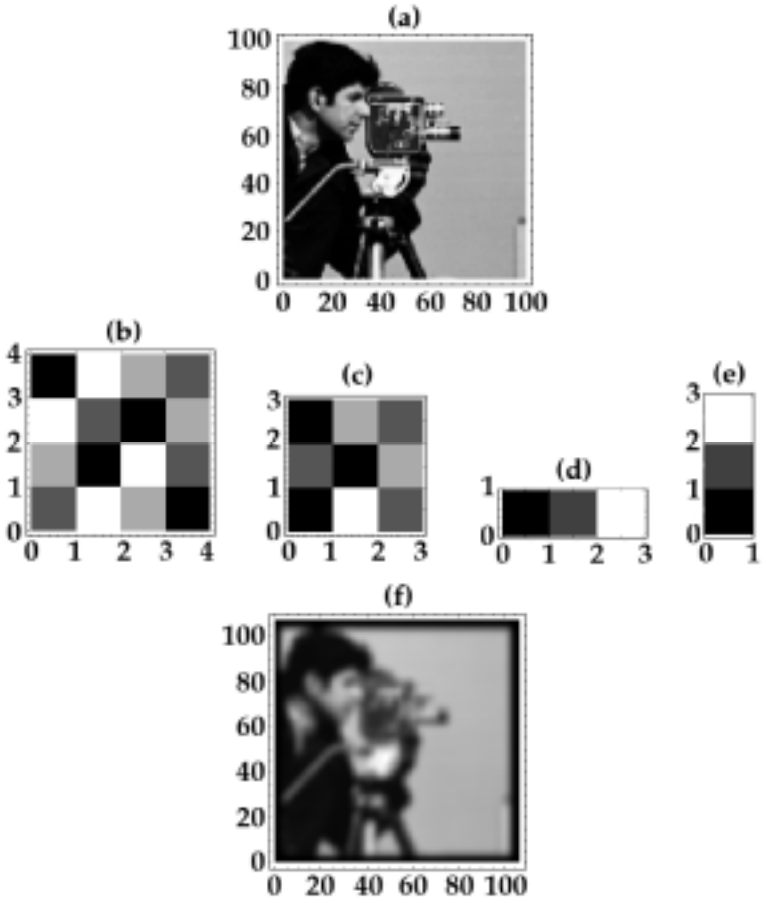


Fig. 2.2 (a) : True image of size 100×100 that we took from [10]. (b) : Blur image of size 4×4. (c) : Blur image of size 3×3. (d) : Blur image of size 1×3. (e) : Blur image of size 3×1. (f) : The image that was obtained by convolving the four blurs of (b) - (e) into the true image of (a). The size of the convolved image is 107×107.

We analyze the zero-sheet of Fig. 2.2 (f) to obtain the true image by removing the zero-sheets of the blurs. First we have to determine an optimal value for ρ_u to facilitate the distinction of the zero-sheets of blurs from those of the true image. For $\rho_u = 1$, the zeros tend to aggregate around

a circle with radius 1. In such a situation, it is extremely difficult to split the zero-sheets. Consequently, in order to classify the zero-sheets successfully we should choose a value of ρ_u far from $\rho_u = 1$. On the other hand, a value of ρ_u far from 1 requires a computation at high precision with many-digits because of the polynomial containing high power of ρ_u. Thus, it is seen that the classification of the zero-sheet by LB's method causes a dilemmatic problem.

Fig. A2.1 given in Appendix shows the zero-sheet obtained with various values of ρ_u. In the zero-sheet the zeros are plotted at only discrete points that correspond to points needed for the inverse Fourier transformation. In determining an optimal value of ρ_u, we give importance to the following two points:

(1) To resolve clustering of zeros as much as possible.
(2) To keep the value of ρ_u in the vicinity of unity as much as possible.

In the present illustration, we chose $\rho_u = 0.6$. For this value of ρ_u, the aggregation of the zeros is sufficiently resolved and ρ_u is not too far from unity. We carried out all the analyses in this section with 80 digits precision.

For the next step, namely, to identify all the closed loops in order to split the zero-sheets, we plot the curves with many more steps in ρ_u. It is because we can obtain so many points by evaluating the roots at different values of u, but we ignore how these points are actually connected and form closed loops.

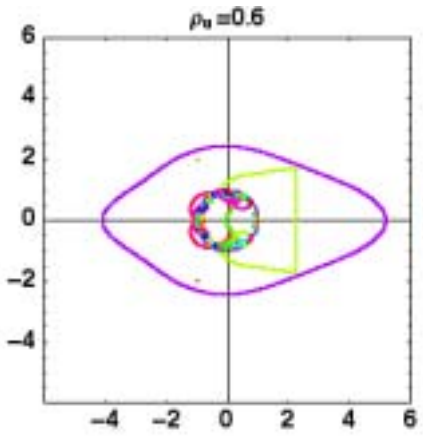


Fig 2.3 Zero-sheet for $\rho_u = 0.6$.

Fig. 2.3 shows the zero-sheet of the given model image at $\rho_u = 0.6$. More than 1000 times more points are plotted in Fig. 2.3 than in Fig. A2.1 (in Appendix). The zero-sheet is found to be consisting of 82 closed curves. We show each closed curve in Fig. A2.2 (in Appendix). With Fig. A2.2 (in Appendix), we split the zero-sheet. The aim of the game is how to identify the closed curves due to the blurs among these 82 closed curves. This is a crucial problem in LB's method. A cut and try may lead to the success of the classification of the closed loops.

In the present illustration, we know the blurs in advance because we use the model blurs shown in Fig. 2.2. Figure 2.4 shows the zero-sheet of blurs. By removing the zero-sheet of blurs, we can obtain the zero-sheet of the true image for variable v. Figure 2.5 shows the zero-sheet obtained by removing the zero-sheet of blurs.

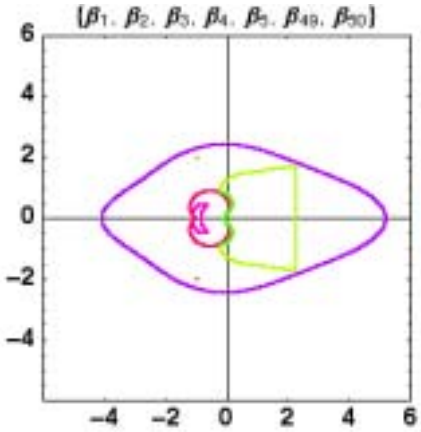


Fig. 2.4 Zero-sheet of blurs (variable v).

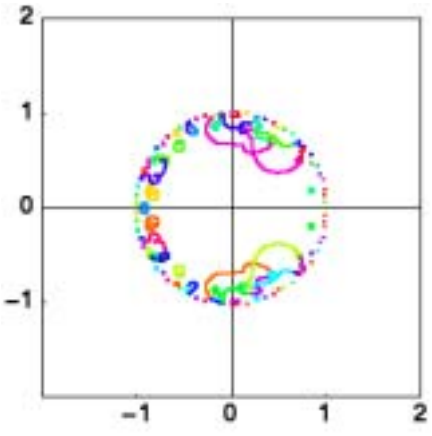


Fig. 2.5 Zero-sheet of true image (variable v).

We repeat the same procedure for the variable u. Figure A2.3 (in Appendix) shows zero-sheets of the given model image for various values of ρ_v. Figure 2.6 shows zero-sheet obtained with the chosen value of $\rho_v = 0.6$. The zero-sheet consists of 44 closed curves. Figure A2.4 (in Appendix) shows each closed curve. Figure 2.7 shows the zero-sheet of blurs for variable u.

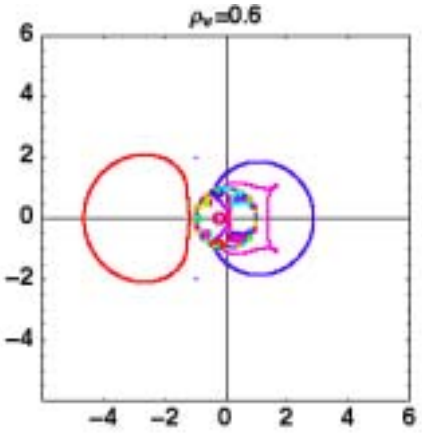


Fig. 2.6 Zero-sheet for $\rho_v = 0.6$.

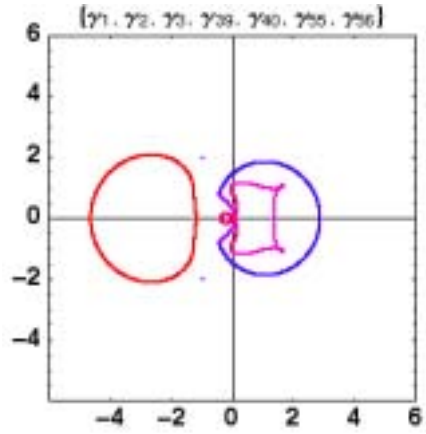


Fig. 2.7 Zero-sheet of blurs (variable u).

We remove the zero-sheet of blurs from Fig. A2.3 (in Appendix). Figure 2.8 shows the obtained zero-sheet by dropping the zero-sheets of blurs for variable u.

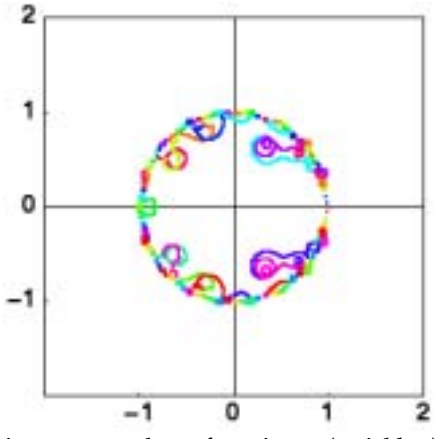


Fig. 2.8 Zero-sheet of true image (variable u).

Now we reconstruct the image with only the zeros of Fig. 2.5 and Fig. 2.8 following the prescription described in Sec. 2.3. Figure 2.9 shows the image obtained by the inverse Fourier transformation. The image of Fig. 2.9 is not the true image of Fig. 2.2 (a). The reconstructed image is in fact an image that is the product of the true image $f(x,y)$ and the factor $\rho_u^x \rho_v^y$. In order to obtain the true image we have to divide the reconstructed image by the factor $\rho_u^x \rho_v^y$. Figure 2.10 shows the image obtained after this procedure. The image is the same as the true image of Fig. 2.2 (a).

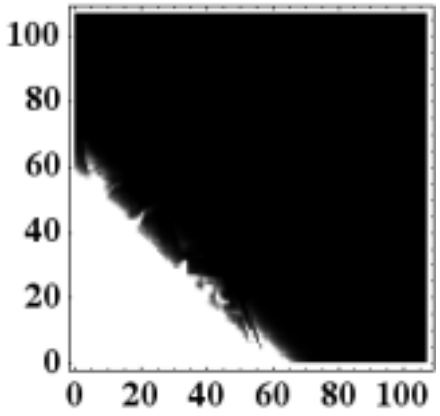


Fig. 2.9 Reconstructed image that is the product of the true image and the factor $\rho_u^x \rho_v^y$.

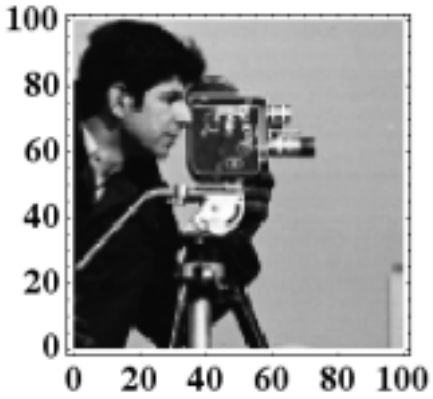


Fig. 2.10 True image reconstructed by removing blurs.

Before ending this section we illustrate how the dropping of the correct closed loops is a cumbersome procedure. Figure 2.11 shows an image reconstructed by removing the zeros represented by the closed curves of β_{14} of Fig. A2.2 and γ_{10} of Fig. A2.4 (in Appendix), which are not in reality due to the blurs. The restored image is completely dissimilar to the true image shown in Fig. 2.10. This implies that unless we remove the zeros of blurs correctly we usually expect to end up by obtaining an unexpected image.

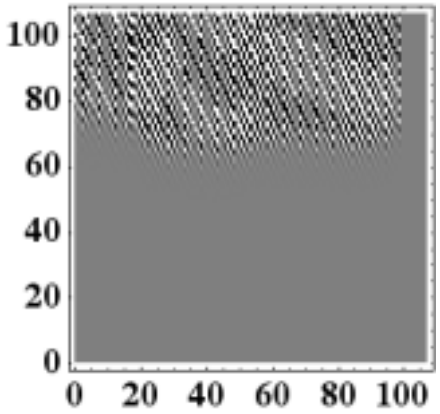


Fig. 2.11 Image reconstructed by wrongly removing the zeros β_{14} and γ_{10} that are not of blurs.

2.5 Enhancing a PET image

In this section, we illustrate the enhancement of a PET image by means of the zero-sheets method. Fig. 2.12 shows a PET image that was obtained from the DICOM database. We selected this PET image for the following two reasons. First, the spatial resolution of the image looks insufficient, and its improvement seems possible. Second, the background is dark. If the background is completely dark, namely 0, then the image is free from the error giving origin to the trimming. In such a situation, the convolution condition presented in Eq. (2.1) may be preserved well.

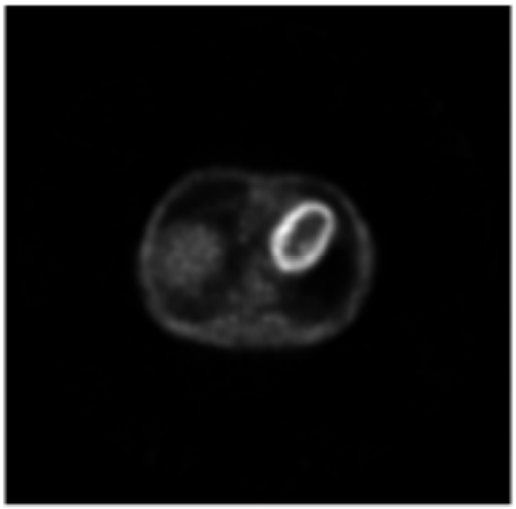


Fig. 2.12 PET image of size 168 × 168, the gray levels of which are 21345.

Fig. A2.5 (in Appendix) shows the zero-sheets of the variable v obtained with various values of ρ_u. First we have to determine an optimal value for ρ_u as we mentioned in Sec 2.4. From Fig. A2.5 (in Appendix), in this illustration, we chose $\rho_u = 1.3$ as an optimal value. We carried out all the analyses in 40 digits precision. Fig. 2.13 shows the zero-sheet calculated, with $\rho_u = 1.3$, for the image of Fig. 2.12 in smaller steps.

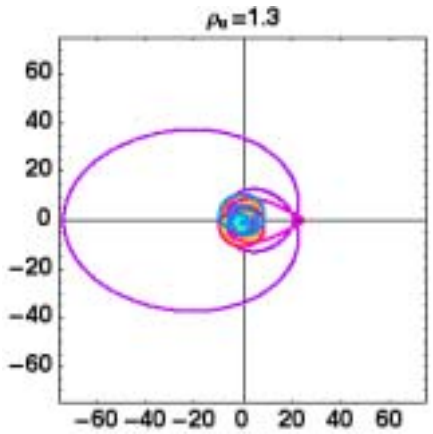


Fig. 2.13 Zero-sheet calculated for the image of Fig. 2.12, where $\rho_u = 1.3$ is used.

By analyzing the zero-sheet shown in Fig. 2.13, we found that the zero-sheet consists of 8 closed curves. We show each closed curve in Fig. A2.6 (in Appendix), where the numbers in the parentheses indicate the number of zeros that form each closed curve.

We repeat the same analyses for variable u. Fig. A2.7 (in Appendix) shows zero-sheets, obtained with various values of ρ_v, for the image given in Fig. 2.12. Fig. 2.14 shows the zero-sheet calculated with finer steps with $\rho_v = 1.3$ that was chosen as an optimal value. The zero-sheet consists of 12 closed curves. Fig. A2.8 (in Appendix) shows each closed curve.

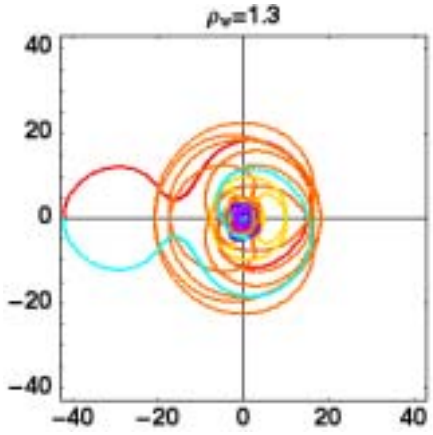


Fig. 2.14 Zero sheets obtained with $\rho_v = 1.3$.

Now we reconstruct the image by removing the zeros of a blur. First we have to spot the closed curves giving origin to the blur in the zero-sheet shown in Figs. A2.6 and A2.8 (in Appendix) without prior knowledge of the blur. In Fig. A2.6 (in Appendix), we regard the closed curves 1, 4 and 5 (β_1, β_5 and β_6, respectively) as ones for the candidates for the blur. Fig. 2.15 (a) shows

the image reconstructed by removing the zero β_1 that forms closed curve 1 in Fig. A2.6 (in Appendix). As for the zeros β_5 and β_6 we have to remove both of them as a pair because these two zeros are complex conjugates of each other. Fig. 2.15 (b) shows the image reconstructed by removing the zeros β_5 and β_6 that form the closed curves 4 and 5 in Fig. A2.6 (in Appendix).

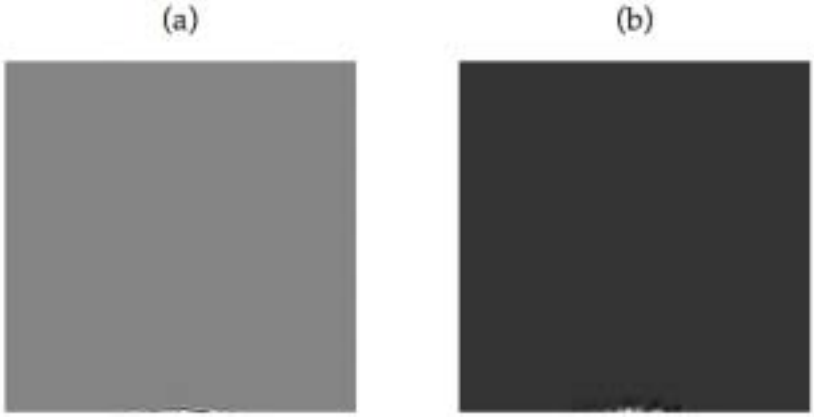


Fig. 2.15 Image reconstructed by removing the zeros β_1, β_5 and β_6 that form the closed curves 1, 4 and 5 (variable v). (a) The image reconstructed by removing the zero β_1 that forms the closed curve 1. (b) The image reconstructed by removing the zeros β_5 and β_6 that form the closed curves 4 and 5. All closed curves refer to those shown in Fig. A2.6 (in Appendix).

As seen in Fig. 2.15, the reconstructed images are completely dissimilar to the given image of Fig. 2.12. Consequently, all the zeros that we removed are not the zeros of the blur.

Next we consider Fig. A2.8 (in Appendix) that shows closed curves in the zero-sheet for variable u. We regard the closed curves 2, 10 and 11 (γ_2, γ_{75} and γ_{76}, respectively) as ones for the candidates for blurs. Fig. 2.16 (a) shows the reconstructed image by removing the zero γ_2 that forms closed curve 2 in Fig. A2.8 (in Appendix). Fig. 2.16 (b) shows reconstructed image by removing the zeros γ_{75} and γ_{76} that form the closed curves 10 and 11 in Fig. A2.8 (in Appendix).

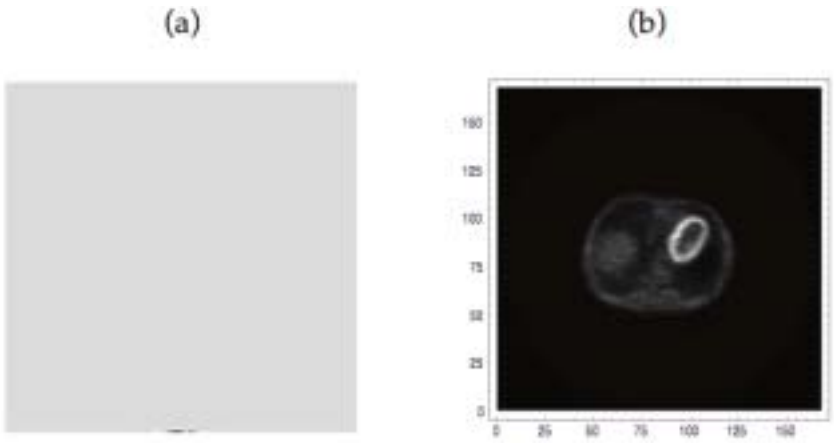


Fig. 2.16 Reconstructed image by removing the zeros that are deemed to be of blurs (variable u). (a) The reconstructed image by removing the zero γ_2 that forms the closed curve 2 of Fig. A2.8 (in Appendix).. (b) The reconstructed image by removing the zeros γ_{75} and γ_{76} that form the closed curves 10 and 11 of Fig. A2.8 (in Appendix)..

The reconstructed image of Fig. 2.16 (a) is by no means a restored image. On the other hand, the reconstructed image of Fig. 2.16 (b) looks like the given image. In order to evaluate how much the reconstructed image has been enhanced w.r.t. the original, we compare the brightness of the given image shown in Fig. 2.16 with that in Fig. 2.16 (b) after normalizing the maximum brightness of it to that of the original. Fig. 2.17 shows such a comparison. The comparison is done at 93rd column and 93rd row. The red and black curves show the brightness's of the restored and original images, respectively. Curves shown in Fig. 2.17 (a) are rather similar, while the variation in the brightness is clearly enhanced around 75-100 pixels in Fig. 2.17 (b).

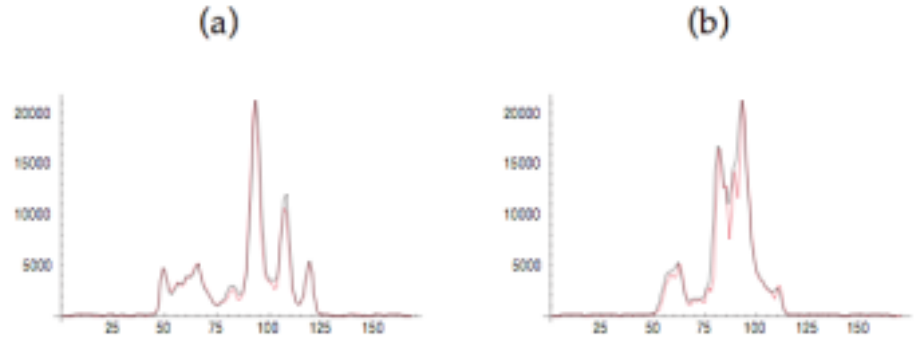


Fig. 2.17 Comparison of brightness between two images in Figs. 2.12 and 2.16 (b). The black curves indicate the brightness of the original image in Fig. 2.12. The red curves show the brightness of the restored image in Fig. 2.16 (b). (a) The comparison is done at 93rd row. (b) The comparison is done at 93rd column.

We could spot the zeros of the blur for the variable u, but we failed to do it for the variable v, although we thoroughly examined the zero-sheet for the variable v. If the blur we spotted is one-dimensional, the zeros of the blur do not exist for variable v. In such a case, the zeros of the blur for the variable u have no v-dependence. That means those zeros stay still in the complex plane when ϕ_v varies from 0 to 2π. However, in reality, the closed curves 10 and 11 in Fig. A2.8 (in Appendix) apparently have v-dependence. This seems puzzling. To solve this puzzle, we assumed that the v-dependence of the zeros (for the variable u) giving origin to the blur have roots in a broken convolution, although we regarded at the beginning of this analysis that the convolution condition is maintained quasi-perfectly. We examine this situation more in detail in the following.

Fig. 2.18 (a) shows the one-dimensional blur that is reconstructed with the zeros γ_{75} and γ_{76} of the blur that form the closed curves 10 and 11 in Fig. A2.8 (in Appendix). We experimentally convolved it with the image shown in Fig. 2.16 (b), and obtained the image shown in Fig. 2.18 (b). The precision of the restored image in Fig. 2.18 (b) is 40.

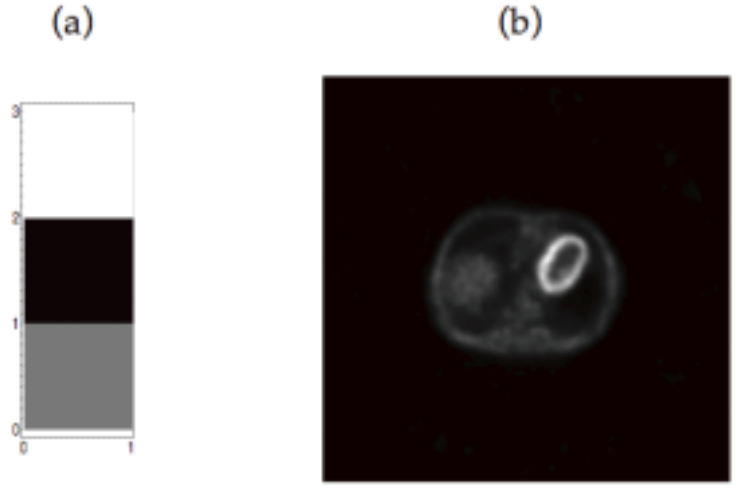


Fig. 2.18 (a) Blur reconstructed with the zeros that form the closed curves 10 and 11 in Fig. A2.8 (in Appendix). (b) Image obtained by convolving the blur (a) and the image in Fig. 2.16 (b).

Fig. A2.9 (in Appendix) shows the closed curves in the zero-sheet for variable u of the image given in Fig. 2.18 (b), where $\rho_v = 1.3$ is used. Since the image in Fig. 2.18 (b) convolves the one-dimensional blur in Fig. 2.18 (a), there must be "closed curves" composed of standing points in Fig. A2.9 (in Appendix). As seen in the last two figures in Fig. A2.9 (in Appendix), there exist seemingly two zeros that are constant and complex conjugate of each other, which are represented as dots in the figures.

Next, we reduced the gray levels of the image given in Fig. 2.18 (b) to 21345 that is the num-

ber of gray levels of the image given in Fig. 2.12. Fig. 2.19 (a) shows the image with the reduced gray levels. Fig. 2.19 (b) shows the difference between two images in Fig. 2.12 and Fig. 2.19 (a). The difference is recognizable near the bottom edge. Except for the slight difference, the two images are identical.

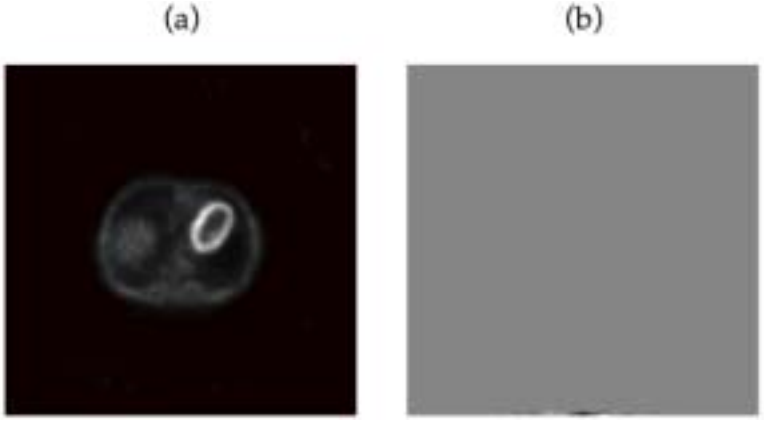


Fig. 2.19 (a) Image shown in Fig. 2.18 (b) whose maximum brightness was normalized to the image in Fig. 2.12. (b) Difference between the image in (a) and that in Fig. 2.12.

As a result of the reduction of the gray levels of the image given in Fig 2.18 (b), the convolution represented in Eq. (2.1) is broken. Fig. A2.10 (in Appendix) shows the closed curves obtained from the zero-sheet for the image of Fig. 2.19 (a), in which $\rho_v = 1.3$ is used. As seen in Fig. A2.10 (in Appendix), there are no zeros that are represented as dots. However, there must be closed curves giving origin to the one-dimensional blur that we convolved. We guess that the closed curve 9 is that due to the blur. We then reconstructed an image by removing the zeros that form the closed curve 9. Fig. 2.20 shows the reconstructed image, which is very close to Fig. 2.18 (b).

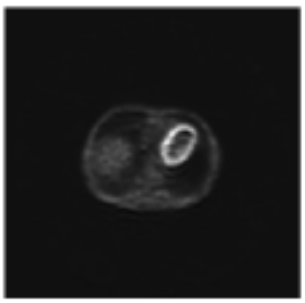


Fig. 2.20 Restored image by removing the zeros that form the closed curve 9 in Fig. A2.10 (in Appendix).

Now we must conclude that the zeros that form the closed curve 9 belong to the zero-sheet of the one-dimensional blur that we convolved. The results of the analyses that we have presented so far lead us to the conjecture that a broken convolution due to the reduced gray levels causes a v (u)-dependence in the zeros of one-dimensional blurs. In the following we show how the conjecture is plausible, more in detail. The matrix elements of the restored image shown in Fig. 2.18 (b) is in reality real with a precision of about 40 digits. We try to introduce gradually the effect of the reduction of gray levels, and for that, we consider the following three cases: (1) We used matrix elements directly as were obtained after the reconstruction, (2) We truncate the matrix elements to 10 digits after the decimal point after the convolution, while the maximum matrix element was about 3117150, (3) same but 2 digits after the decimal point. Figs. A2.11, A2.12 and A2.13 (in Appendix) show the closed curves obtained for the cases (1), (2) and (3), respectively. As seen in Fig. A2.11 (in Appendix), there exist the zero-sheets composed of dots, i.e., the closed curves 7 and 8. As for Fig. A2.12 (in Appendix), the closed curves 7 and 8 seem to be composed of the zeros represented as dots. However, a close scrutiny reveals that the zeros are not really dots but have v-dependence as shown in Fig. A2.14 (in Appendix). In Fig. A2.13 (in Appendix), it is evident that all the zeros have v-dependence. Consequently, it seems very plausible that the zeros of blurs have complicated v (u) -dependence, even if they are one-dimensional blurs.

In conclusion, we are rather sure that we could spot a one-dimensional blur in the PET image given in Fig. 2.12 and remove it.

3. Determinant Conditions

In the preceding chapter, we showed the image restoration by means of the zero-sheet method. This method requires as demonstrated, huge computational resources. For a practical implementation of the LB method, we have recently developed mathematical tools which are referred to as Determinant Conditions (DCs) throughout this report.

By definition, the size of a blur is smaller than that of a true image. And most of the time, it is even much smaller. The DC is constructed on the basis of this fact. By assuming the size of a blur, we can construct the DCs. Using the DCs one can detect the zeros of a blur without analyzing the zero-sheets of a given image

The DCs can be expressed in two different forms, derivative form and multi-point form. The derivative form is given as a condition to the determinant derived from the derivatives relative to the zeros of assumed blurs. The derivatives can be analytically derived using whole zeros of a given image. On the other hand, the multi-point form is given as a condition to the determinant evaluated at multiple points of u (or v).

3.1 Derivative form of DC

A root β_i of $H(u,v)=0$ varies as a function of u. That means that $H(u,\beta_i)$ is identically zero independently of u. Therefore one can derive equations such as,

$$\frac{d^j}{du^j}H(u,\beta_i)=0, \quad j=0,\cdots,mn-1. \tag{3.1}$$

The above mn equations, if they are independent, 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, $\beta^{(j)}\equiv d^j\beta/du^j$, $j=0,1,2,\cdots,mn-1$. Actually all even higher derivatives of $H(u,\beta_i)$ are also zero. Hence we can use derivatives of any order. 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_{D} and its determinant $\det C_{\mathrm{D}}$. Matrix C_{D} is complex in general. Eq. (3.1) can be rewritten in a vector notation as

$$C_{\mathrm{D}}\,\mathrm{h}=0, \tag{3.2}$$

where

$$\mathrm{h}^t=(h_{00}\quad h_{01}\quad h_{02}\quad\cdots\quad h_{10}\quad h_{11}\quad h_{12}\quad\cdots\quad h_{20}\quad h_{21}\quad h_{22}\quad\cdots\quad\cdots) \tag{3.3}$$

and

$$C_{\mathrm{D}}=\begin{pmatrix} 1 & \beta_i & \beta_i^{\,2} & \cdots & u & u\beta_i & u\beta_i^{\,2} & \\ & & & \cdots & u^2 & u^2\beta_i & u^2\beta_i^{\,2} & \cdots \\ 0 & \beta_i' & 2\beta_i\beta_i' & \cdots & 1 & \beta_i+u\beta_i' & \beta_i^{\,2}+2u\beta_i\beta_i' & \\ & & & \cdots & 2u & 2u\beta_i+u^2\beta_i' & 2\left(u\beta_i^{\,2}+u^2\beta_i\beta_i'\right) & \cdots \\ 0 & \beta_i'' & 2\left(\beta_i'^{\,2}+\beta_i\beta_i''\right) & \cdots & 0 & 2\beta_i'+u\beta_i'' & 2\left(2\beta_i\beta_i'+u\beta_i'^{\,2}+\beta_i\beta_i''\right) & \\ & & & \cdots & 2 & 2\beta_i+4u\beta_i'+u^2\beta_i'' & 2\left(\beta_i^{\,2}+4u\beta_i\beta_i'+u^2\beta_i'^{\,2}+u^2\beta_i\beta_i''\right) & \cdots \\ & & & & & \vdots & & \end{pmatrix} \tag{3.4}$$

We require that **h** have a nontrivial solution, that is, the mn elements of $h(x,y)$ are not all zero. This requires $\det C_{\mathrm{D}}=0$. We refer to this as the derivative form of the determinant condition (DC). The determinant used in DC for H of size $m\times n$ hereafter would be denoted by $E_{m\times n}(\beta_i)$, i.e.,

$$E_{m\times n}(\beta_i)\equiv \det C_{\mathrm{D}}(\beta_i). \tag{3.5}$$

Now, if ϕ_u is fixed, then the derivative du in Eq. (3.1) can be replaced by $d\rho_u$, and the DC would be denoted by $E^{\rho_u}_{m\times n}(\beta_i)$.

As an illustration, let us consider the case of $m=2$ and $n=3$. We obtain the explicit form of C_{D} in this case as

$$C_{\mathrm{D}}=e^{-2i\phi_u}\begin{pmatrix} 1 & \beta_i & \beta_i^{\,2} & \rho_u & \rho_u\beta_i & \rho_u\beta_i^{\,2} \\ 0 & \beta_i^{(1)} & 2\beta_i\beta_i^{(1)} & 1 & \beta_i+\rho_u\beta_i^{(1)} & \beta_i^{\,2}+2\rho_u\beta_i\beta_i^{(1)} \\ 0 & \beta_i^{(2)} & 2\left(\beta_i^{(1)^2}+\beta_i\beta_i^{(2)}\right) & 0 & 2\beta_i^{(1)}+\rho_u\beta_i^{(2)} & 2(2\beta_i\beta_i^{(1)}+\rho_u\beta_i^{(1)^2}+\rho_u\beta_i\beta_i^{(2)}) \\ 0 & \beta_i^{(3)} & 2\left(3\beta_i^{(1)}\beta_i^{(2)}+\beta_i\beta_i^{(3)}\right) & 0 & 3\beta_i^{(2)}+\rho_u\beta_i^{(3)} & \begin{matrix}2(3\beta_i^{(1)^2}+3\beta_i\beta_i^{(2)}+3\rho_u\beta_i^{(1)}\beta_i^{(2)}\\ +\rho_u\beta_i\beta_i^{(3)})\end{matrix} \\ 0 & \beta_i^{(4)} & 2\left(3\beta_i^{(2)^2}+4\beta_i^{(1)}\beta_i^{(3)}+\beta_i\beta_i^{(4)}\right) & 0 & 4\beta_i^{(3)}+\rho_u\beta_i^{(4)} & \begin{matrix}2(12\beta_i^{(1)}\beta_i^{(2)}+4\beta_i\beta_i^{(3)}+3\rho_u\beta_i^{(2)^2}\\ +4\rho_u\beta_i^{(1)}\beta_i^{(3)}+\rho_u\beta_i\beta_i^{(4)})\end{matrix} \\ 0 & \beta_i^{(5)} & 2\left(10\beta_i^{(2)}\beta_i^{(3)}+5\beta_i^{(1)}\beta_i^{(4)}+\beta_i\beta_i^{(5)}\right) & 0 & 5\beta_i^{(4)}+\rho_u\beta_i^{(5)} & \begin{matrix}2(15\beta_i^{(2)^2}+20\beta_i^{(1)}\beta_i^{(3)}+5\beta_i\beta_i^{(4)}\\ +10\rho_u\beta_i^{(2)}\beta_i^{(3)}\\ +5\rho_u\beta_i^{(1)}\beta_i^{(4)}+\rho_u\beta_i\beta_i^{(5)})\end{matrix} \end{pmatrix}, \tag{3.6}$$

where we took as $u=\rho_u e^{-i\phi_u}$ and $\beta_i^{(j)}\to \partial^{\,j}\beta_i/\partial\rho_u^{\,j}$ [4]. The $\det C_{\mathrm{D}}$ leads to the DC [4, 7] such as

$$E^{\rho_u}_{2\times3}(\beta_i) \equiv -135\beta_i^{(2)^6} + 80\beta_i^{(1)^3}\beta_i^{(3)^3} + 15\beta_i^{(1)^4}\beta_i^{(4)^2} - 60\beta_i^{(1)^3}\beta_i^{(2)}\beta_i^{(3)}\beta_i^{(4)} - 12\beta_i^{(1)^4}\beta_i^{(3)}\beta_i^{(5)} + 18\beta_i^{(1)^3}\beta_i^{(2)^2}\beta_i^{(5)} + 270\beta_i^{(1)}\beta_i^{(2)^4}\beta_i^{(3)} - 180\beta_i^{(1)^2}\beta_i^{(2)^2}\beta_i^{(3)^2} = 0. \tag{3.7}$$

For $\beta^{(j)}$, we can take it as derivatives with respect to ϕ_u. However, they are in fact identical to each other. For instance, if we take the derivative of β_i with respect to ϕ_u namely $\beta_i^{(j)} \to \partial^j \beta_i / \partial \phi_u^j$ we obtain the relation such as $E^{\phi_u}_{2\times3}(\beta_i) = \rho_u^{12} E^{\rho_u}_{2\times3}(\beta_i)$, where $E^{\phi_u}_{2\times3}(\beta_i)$ 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).

Next, we would like to show that the DC derived for 2×3 implicitly includes DCs for blurs of sizes smaller than 2×3, i.e., 2×2, 1×3 and 1×2. This is because the $\det C_{\mathrm{D}}$ can be expressed as a linear combination of DCs for the blur elements of each of the assumed smaller sizes.

For instance, we can show an explicit decomposition of Eq. (3.6) into DCs for 2×2 blurs, i.e.,

$$E^{\rho_u}_{2\times3}(\beta_i) = \sum_{k=1}^{6} A_k(\beta_i) E^{\rho_u}_{2\times2}(\beta_i)_k \ , \tag{3.8}$$

where A_k and $E^{\rho_u}_{2\times2}(\beta_i)_k$ are given as,

$$A_1 = -2\beta_i^2\beta_i^{(1)}\beta_i^{(3)} + 3\beta_i^{(1)^4} + 3\beta_i^2\beta_i^{(2)^2}, \tag{3.9}$$

$$A_2 = -6\beta_i\beta_i^{(1)}\beta_i^{(2)^2} + 4\beta_i\beta_i^{(1)^2}\beta_i^{(3)} + 2\beta_i^2\beta_i^{(1)}\beta_i^{(4)} - 12\beta_i^{(2)}\beta_i^{(1)^3} - 4\beta_i^2\beta_i^{(2)}\beta_i^{(3)}, \tag{3.10}$$

$$A_3 = -5\beta_i\beta_i^{(1)^2}\beta_i^{(4)} - 2\beta_i^2\beta_i^{(1)}\beta_i^{(5)} + 15\beta_i^{(1)^2}\beta_i^{(2)^2} + 15\beta_i\beta_i^{(2)^3} + 20\beta_i^{(3)}\beta_i^{(1)^3} + 5\beta_i^2\beta_i^{(2)}\beta_i^{(4)}, \tag{3.11}$$

$$A_4 = 27\beta_i^{(1)^2}\beta_i^{(2)^2} - 12\beta_i^{(3)}\beta_i^{(1)^3} - 3\beta_i\beta_i^{(1)^2}\beta_i^{(4)} - 9\beta_i\beta_i^{(2)^3} + 12\beta_i\beta_i^{(1)}\beta_i^{(2)}\beta_i^{(3)} - 3\beta_i^2\beta_i^{(2)}\beta_i^{(4)} + 4\beta_i^2\beta_i^{(3)^2}, \tag{3.12}$$

$$A_5 = -30\beta_i^{(2)}\beta_i^{(1)^2}\beta_i^{(3)} + 15\beta_i^{(1)^3}\beta_i^{(4)} + 3\beta_i\beta_i^{(1)^2}\beta_i^{(5)} + 15\beta_i\beta_i^{(2)^2}\beta_i^{(3)} + 3\beta_i^2\beta_i^{(2)}\beta_i^{(5)} - 45\beta_i^{(1)}\beta_i^{(2)^3} - 20\beta_i\beta_i^{(1)}\beta_i^{(3)^2} - 5\beta_i^2\beta_i^{(3)}\beta_i^{(4)}, \tag{3.13}$$

$$A_6 = 30\beta_i\beta_i^{(2)^2}\beta_i^{(4)} - 4\beta_i^2\beta_i^{(3)}\beta_i^{(5)} - 40\beta_i\beta_i^{(3)^2}\beta_i^{(2)} - 60\beta_i^{(1)^2}\beta_i^{(2)}\beta_i^{(4)} + 45\beta_i^{(2)^4} + 5\beta_i^2\beta_i^{(4)^2} + 80\beta_i^{(1)^2}\beta_i^{(3)^2} - 12\beta_i\beta_i^{(1)}\beta_i^{(2)}\beta_i^{(5)} + 20\beta_i\beta_i^{(1)}\beta_i^{(3)}\beta_i^{(4)} \tag{3.14}$$

and

$$E^{\rho_u}_{2\times2}(\beta_i)_1 = -5\beta_i^{(4)^2} + 4\beta_i^{(5)}\beta_i^{(3)}, \quad (0,1,4,5) \tag{3.15}$$

$$E^{\rho_u}_{2\times2}(\beta_i)_2 = -5\beta_i^{(3)}\beta_i^{(4)} + 3\beta_i^{(5)}\beta_i^{(2)}, \quad (0,1,3,5) \tag{3.16}$$

$$E^{\rho_u}_{2\times 2}\left(\beta_i\right)_3 = -4\beta_i^{(3)^2} + 3\beta_i^{(4)}\beta_i^{(2)}, \quad (0,1,3,4) \tag{3.17}$$

$$E^{\rho_u}_{2\times 2}\left(\beta_i\right)_4 = -5\beta_i^{(4)}\beta_i^{(2)} + 2\beta_i^{(5)}\beta_i^{(1)}, \quad (0,1,2,5) \tag{3.18}$$

$$E^{\rho_u}_{2\times 2}\left(\beta_i\right)_5 = -4\beta_i^{(3)}\beta_i^{(2)} + 2\beta_i^{(4)}\beta_i^{(1)}, \quad (0,1,2,4) \tag{3.19}$$

$$E^{\rho_u}_{2\times 2}\left(\beta_i\right)_6 = -3\beta_i^{(2)^2} + 2\beta_i^{(3)}\beta_i^{(1)}, \quad (0,1,2,3) \tag{3.20}$$

where the numbers in parenthesis in Eq. (3.15) - (3.20) indicate the order of derivatives used for the derivation of the DC (see Eq. (3.1) and the statement immediately after). Equations (3.15) - (3.20) are all DCs for 2×2 blurs. The simplest form $E^{\rho_u}_{2\times 2}\left(\beta_i\right)_6$ of Eq. (3.20) is obtained from Eq. (3.1) [2]. Other ones are obtained by taking the higher order derivatives than those in Eq. (3.1). Note that $E^{\rho_u}_{2\times 2}\left(\beta_i\right)_2$ and $E^{\rho_u}_{2\times 2}\left(\beta_i\right)_5$ are respectively obtained by differentiating $E^{\rho_u}_{2\times 2}\left(\beta_i\right)_3$ and $E^{\rho_u}_{2\times 2}\left(\beta_i\right)_6$ with respect to ρ_u. Thus, it is seen that $E^{\rho_u}_{2\times 3}\left(\beta_i\right)$ for 2×3 blurs can be rewritten as a linear combination solely of the DCs for 2×2 blurs. Similarly, it can be explicitly shown that the DCs of Eqs. (3.15) - (3.20) for 2×2 blurs can be expressed solely in terms of the DCs for 1×2 blurs. This indicates that $E^{\rho_u}_{2\times 3}\left(\beta_i\right)$ for 2×3 of Eq. (3.7) can detect the zeros of blurs of sizes 2×3, 2×2 and 1×2 all at the same time. It should be noted that Equation (3.7) also includes DCs for blurs $1 \times j$ with $j > 3$ (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 feature that we have demonstrated for the size of 2×3 can be generalized to any size. The DC calculated for the size of $m \times n$ can be expressed solely in terms of DC for a size $m' \times n'$ where m' and n' are smaller or equal to m and n respectively. *This means that if we see that the DC is satisfied for a sufficiently large size, then we can certify the convolution of a blur of that size or smaller.*

Although we examined DCs for the zeros of $H(u,v)$, in the actual analysis we have to deal only with $G(u,v)$ that is defined in terms of the given image $g(x,y)$ since 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 $H(u,v) = 0$. In that case, Eq. (3.1) with $H(u,v)$ replaced by $G(u,v)$ holds. Let one of such roots be β, i.e., assume that $H(u,\beta) = 0$. In order to work out the matrix elements of C, we have to evaluate derivatives of β with respect to u which we can do using Eq. (3.1) in which $H(u,\beta)$ replaced by $G(u,\beta)$. 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 fifth order. From $d^k G\left(\rho_u e^{-i\phi_u}, \beta_i\right)/d\rho_u^k = 0\,(k = 1,\cdots,5)$, the derivatives of the zeros β_i are given as,

$$\beta_i^{(1)} = -\frac{\left(\frac{\partial G}{\partial \rho_u}\right)}{\left(\frac{\partial G}{\partial \beta_i}\right)}, \tag{3.21}$$

$$\beta_i^{(2)} = -\frac{1}{\left(\frac{\partial G}{\partial \beta_i}\right)}\left(\frac{\partial^2 G}{\partial \rho_u^2} + 2\frac{\partial^2 G}{\partial \rho_u \partial \beta_i}\beta_i^{(1)} + \frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(1)^2}\right), \tag{3.22}$$

$$\begin{aligned}\beta_i^{(3)} = -\frac{1}{\left(\frac{\partial G}{\partial \beta_i}\right)}\Bigg(&\frac{\partial^3 G}{\partial \rho_u^3} + 3\frac{\partial^3 G}{\partial \rho_u^2 \partial \beta_i}\beta_i^{(1)} + 3\frac{\partial^3 G}{\partial \rho_u \partial \beta_i^2}\beta_i^{(1)^2} \\ &+ 3\frac{\partial^2 G}{\partial \rho_u \partial \beta_i}\beta_i^{(2)} + 3\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(1)}\beta_i^{(2)} + \frac{\partial^3 G}{\partial \beta_i^3}\beta_i^{(1)^3}\Bigg),\end{aligned} \tag{3.23}$$

$$\begin{aligned}\beta_i^{(4)} = -\frac{1}{\left(\frac{\partial G}{\partial \beta_i}\right)}\Bigg(&\frac{\partial^4 G}{\partial \rho_u^4} + 4\frac{\partial^4 G}{\partial \rho_u^3 \partial \beta_i} + 6\frac{\partial^4 G}{\partial \rho_u^2 \partial \beta_i^2}\beta_i^{(1)^2} + 4\frac{\partial^4 G}{\partial \rho_u \partial \beta_i^3}\beta_i^{(1)^3} + \frac{\partial^4 G}{\partial \beta_i^4}\beta_i^{(1)^4} + 6\frac{\partial^3 G}{\partial \rho_u^2 \partial \beta_i}\beta_i^{(2)} \\ &+ 4\frac{\partial^2 G}{\partial \rho_u \partial \beta_i}\beta_i^{(3)} + 12\frac{\partial^3 G}{\partial \rho_u \partial \beta_i^2}\beta_i^{(1)}\beta_i^{(2)} + 6\frac{\partial^3 G}{\partial \beta_i^3}\beta_i^{(1)^2}\beta_i^{(2)} + 3\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(2)^2} + 4\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(1)}\beta_i^{(3)}\Bigg),\end{aligned} \tag{3.24}$$

$$\begin{aligned}\beta_i^{(5)} = -\frac{1}{\left(\frac{\partial G}{\partial \beta_i}\right)}\Bigg(&\frac{\partial^5 G}{\partial \rho_u^5} + 5\frac{\partial^5 G}{\partial \rho_u^4 \partial \beta_i}\beta_i^{(1)} + 10\frac{\partial^5 G}{\partial \rho_u^3 \partial \beta_i^2}\beta_i^{(1)^2} + 10\frac{\partial^4 G}{\partial \rho_u^3 \partial \beta_i}\beta_i^{(2)} + 10\frac{\partial^5 G}{\partial \rho_u^2 \partial \beta_i^3}\beta_i^{(1)^3} + 30\frac{\partial^4 G}{\partial \rho_u^2 \partial \beta_i^2}\beta_i^{(1)}\beta_i^{(2)} \\ &+ 5\frac{\partial^5 G}{\partial \rho_u \partial \beta_i^4}\beta_i^{(1)^4} + 30\frac{\partial^4 G}{\partial \rho_u \partial \beta_i^3}\beta_i^{(1)^2}\beta_i^{(2)} + \frac{\partial^5 G}{\partial \beta_i^5}\beta_i^{(1)^5} + 10\frac{\partial^4 G}{\partial \beta_i^4}\beta_i^{(1)^3}\beta_i^{(2)} + 10\frac{\partial^3 G}{\partial \beta_i^2 \partial \beta_i}\beta_i^{(3)} + 20\frac{\partial^3 G}{\partial \rho_u \partial \beta_i^2}\beta_i^{(1)}\beta_i^{(3)} \\ &+ 5\frac{\partial^2 G}{\partial \rho_u \partial \beta_i}\beta_i^{(4)} + 15\frac{\partial^3 G}{\partial \rho_u \partial \beta_i^2}\beta_i^{(2)^2} + 15\frac{\partial^3 G}{\partial \beta_i^3}\beta_i^{(1)}\beta_i^{(2)^2} + 10\frac{\partial^3 G}{\partial \beta_i^3}\beta_i^{(1)^2}\beta_i^{(3)} + 10\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(2)}\beta_i^{(3)} + 5\frac{\partial^2 G}{\partial \beta_i^2}\beta_i^{(1)}\beta_i^{(4)}\Bigg),\end{aligned} \tag{3.25}$$

where

$$\frac{\partial^{(p+q)} G}{\partial \rho_u^{\,p} \partial \beta_i^{\,q}} = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} g(x,y)\, x(x-1)\cdots(x-p+1)\,\rho_u^{\,x-p} e^{-ix\phi_u}\, y(y-1)\cdots(y-q+1)\,\beta_i^{\,y-q}. \tag{3.26}$$

In Eq. (3.26) $p, q = 0, 1, 2, \cdots, 5,$ with $p + q = 5$. In this way the derivatives of the root β_i are

given in terms of known $g(x,y)$ and the root β_i itself, that are evaluated at a point of u, hence $\det C_{\mathrm{D}}$ can be evaluated for that root.

What we showed so far can be summarized as follows:

1. Choose a value u_0 for u.
2. Find a solution β_i of $G(u_0, v) = 0$ for v.
3. Calculate higher derivatives $\beta_i^{(j)}$ using the elements $g(x,y)$ and β_i itself.
4. If $\det C_{\mathrm{D}} = 0$ for a size $m \times n$, then a blur exists, and the root β_i satisfies $H(u_0, \beta_i) = 0$ where the size of h is $m \times n$ or smaller.

We can also obtain a DC just by replacing the role of u and v of the expressions above. Finding a solution γ_i for u of the equation $H(u,v) = 0$ for a fixed value of v, one can derive derivatives of the equation $H(\gamma_i, v) = 0$. One can obtain a similar equation as Eq. (3.2), but C_{D} now contains v, γ_i, and derivatives of γ_i. For example, for the case of size 2×3 case we can derive the explicit form of the DC for γ_i, i.e.,

$$E_{2\times 3}^{\rho_v}(\gamma_i) \equiv -40\gamma_i^{(3)^3} + 60\gamma_i^{(2)}\,\gamma_i^{(3)}\,\gamma_i^{(4)} - 15\gamma_i^{(1)}\,\gamma_i^{(4)^2} - 18\gamma_i^{(2)^2}\,\gamma_i^{(5)} + 12\gamma_i^{(1)}\,\gamma_i^{(3)}\,\gamma_i^{(5)} = 0, \tag{3.27}$$

where we took as $v = \rho_v e^{-i\phi_v}$ and $\gamma_i^{(j)} = d^j \gamma_i / d\rho_v^j$ [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. (3.1) when we construct DCs for an assumed blur. If we use such higher derivatives we obtain DCs generally more complicated 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 roots β_i or γ_i.

3.2 Multi-point form of DC

In the preceding section we presented the derivative form of the 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 little more primitive yet simpler to introduce and easier to use.

We consider the same situation as the preceding 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 $h(x,y)$ as given in Eq. (2.1). For a given u, equation $H(u,v) = 0$ has in general multiple roots for v. let $v = \beta_i$ be one of those roots. When u is varied, the root also varies as a func-

tion of u, i.e., $\beta_i = \beta_i(u)$. The expression $H(u, \beta_i)$ is identically zero as we already mentioned in the preceding chapter. Now we make mn equations by varying u, i.e.,

$$H(u_j, \beta_i(u_j)) = 0, \quad j = 0, \cdots, mn - 1. \tag{3.28}$$

Note that any of these equations contains all the matrix elements $h(x, y)$. Also note that if the equation has multiple roots $(m > 2)$, we can make Eq. (3.28) by picking any one of the roots (i.e., they can belong to different closed loops in the definition of Lane and Bates' "zero-sheet"). 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 C_{M}, i.e.,

$$C_{\mathrm{M}}\mathbf{h} = 0, \tag{3.29}$$

where $\mathbf{h}$ is defined in Eq. (3.2), and the matrix C_{M} is given as,

$$C_{\mathrm{M}} = \begin{pmatrix} 1 & \beta(u_0) & \cdots & \beta(u_0)^{n-1} & u_0 & u_0\beta(u_0) & \cdots & u_0\beta(u_0)^{n-1} & \cdots\cdots & u_0^{m-1} & u_0^{m-1}\beta(u_0)^{n-1} \\ 1 & \beta(u_1) & \cdots & \beta(u_1)^{n-1} & u_1 & u_1\beta(u_1) & \cdots & u_1\beta(u_1)^{n-1} & \cdots\cdots & u_1^{m-1} & u_1^{m-1}\beta(u_1)^{n-1} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & & \vdots & \vdots \\ 1 & \beta(u_q) & \cdots & \beta(u_q)^{n-1} & u_q & u_q\beta(u_q) & \cdots & u_q\beta(u_q)^{n-1} & \cdots\cdots & u_q^{m-1} & u_q^{m-1}\beta(u_q)^{n-1} \end{pmatrix}, \tag{3.30}$$

where $q = mn - 1$ and we dropped the factor $1/mn$ [6, 9]. We require that Eq. (3.28) have a non-trivial solution, that is, mn elements of $h(x, y)$ are not all zero. This requires $\det C_{\mathrm{M}} = 0$. We refer to this as the multi-point form of the determinant condition (DC). Hereafter we denote the DC for $m \times n$ blur with $E^{u}_{m\times n}\left(\beta(u_0), \cdots, \beta(u_{mn-1})\right)$. Like in the case of derivative form of DC, the DC for the size of $m \times n$ is satisfied if any blur of size smaller than $m \times n$ is convolved. This follows from the structure of the determinant $\det C_{\mathrm{M}}$. Further, the DC can detect the zeros of blurs of size $1 \times k\,(k = n + 1, \cdots, N)$ like in the case of derivative form of DC.

The multi-point form of the DC is simpler to use than the derivative form of the DC since we do not have to evaluate derivatives of the zeros. However, there is a drawback in the multi-point form. We mentioned above that the roots we use in Eq. (3.30) can be any one in case there are multiple roots. This is true when only one blur is convolved in the image. If multiple blurs are convolved as we see in Sec. 3.3, and the sizes of the blurs are different, then the multi-point DC is satisfied if the roots are selectively taken from the same loop in the sense of Lane and Bates' "zero-sheet". This is only a minor drawback since the convolution of multiple blurs is rare.

More important in the real application of DC to an image is, suppose we choose one value of u, and all the roots of $G(u,v)=0$ for v are calculated. We pick up one of the roots, and want to identify if this root is actually the root of $H(u,v)=0$, or that of $F(u,v)=0$. We can instantly tell if this root satisfies the derivative form of DC. But to use the multi-point form, we have to select a few points of u, calculate the roots, and choose roots all giving origin to $H(u,v)=0$. This is not an easy task if we do not know the structure of the "zero-sheet". As we know that for a small change of u, the change of the roots is also small. Thus we can take $mn-1$ different points $u_\ell\,(\ell=0,1,\cdots,mn-1)$ in the vicinity of chosen u_j, i.e., as $u_{j,\ell}=u_j+\ell\Delta u.$, and pick β_j's also in the vicinity of the target β_i. This requires an optimization of the parameter Δu. Namely, to be able to pick the "same" β_i we have to take Δu sufficiently small. (By "same" β_i, we mean the β_j's such that, when $u_{j,\ell}$ are all reduced to the same value, then all the associated β_j's coalesce.) If Δu we take is too small, β_i solved for mn different points $u_{j,\ell}$ become too close to each other. This may cause more than two row vectors in $\det C_{\mathrm{M}}$ to be nearly proportional to each other, and $\det C_{\mathrm{M}}\approx 0$ may be caused, which does not necessarily mean that the condition is satisfied.

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 $\det C_{\mathrm{M}}$ for v. The DC for u is denoted as $E^{v}_{m\times n}\left(\gamma_i(v_0),\cdots,\gamma_i(v_{mn-1})\right)=0$ where γ_i are the zeros of u for the blurs.

In the discussion hereafter, the matrix C_{D} and C_{M} can be interchangeably used. We use C, which can stand for both C_{D} and C_{M}. When the discussion is restricted to one form of C, then we explicitly write C_{D} or C_{M}. We did most of the computation for the illustration or demonstration using C_{D} for C. But of course, C_{M} can be used in most of the parts without any problem.

3.3 Reconstruction of the true image using DC

In the preceding section we have given two variations of the DCs. In this section, we illustrate the image restoration with the DCs and show how this novel scheme is useful in implementing the LB blind deconvolution.

We use the same model images as in section 2.4. We apply the derivative form DC for a blur of size 4×4 to a given image.

We first illustrate how zeros of the blurs can be detected through $E^{\rho_u}_{4\times4}(\beta_i)$ and $E^{\rho_v}_{4\times4}(\gamma_i)$. For the parameters 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. Figure 3.1 shows the results of numerical evaluations of the DCs for the convolved image shown in Fig. 2.2 (f). We carried out the evaluation at $\phi_u = 2\pi j/107, \quad j = 0, \cdots, 106$ and $\phi_v = 2\pi k/107, \quad k = 0, \cdots, 106$. Note that when we restore the image by inverse Fourier transformation, we need the representation of $G(u,v)$ at these arguments of ϕ_u and ϕ_v. In Fig. 3.1 we plot the size of $E^{\rho_u}_{4\times4}(\beta_i)$ namely, $\log\left[\left|E^{\rho_u}_{4\times4}(\beta_i)\right|+1\right]$ and $\log\left[\left|E^{\rho_v}_{4\times4}(\gamma_i)\right|+1\right]$ for four arguments of ϕ_u and ϕ_v, respectively. As we have stressed in the preceding section, $E^{\rho_u}_{4\times4}(\beta_i)$ and $E^{\rho_v}_{4\times4}(\gamma_i)$ both include DCs for blurs of smaller sizes 1×3, 3×3 and, 4×4 implicitly. Therefore, $E^{\rho_u}_{4\times4}(\beta_i)$ should detect seven (= 2 + 2 + 3) zeros. When there exists a degenerate zero in the 3×3 blur, the number of zeros is reduced to six. On the other hand, the number of zeros γ_i that should be also detected through $E^{\rho_v}_{4\times4}(\gamma_i)$ is seven (= 2 + 2 + 3).

As seen in Fig. 3.1 (a), for $\phi_u = 0$, seven zeros $\beta_1, \beta_2, \beta_3, \beta_4, \beta_{26}, \beta_{49}$ and β_{50} are detected by $E^{\rho_u}_{4\times4}(\beta_i)$. Also at other ϕ_u, seven zeros are detected by $E^{\rho_u}_{4\times4}(\beta_i)$. Here, note that the order of roots detected are different at each ϕ_u (They are sorted according to the rule of *Mathematica,* namely in the ascending order of the real part, and then ascending order of the imaginary part.). In the evaluation of the DC we obtained zeros β_i by solving the 106th degree polynomial of v numerically by *Mathematica*, separately at each ϕ_u with a precision of 150 digits.

For $\phi_v = 0$, as seen in Fig. 3.1 (b), seven zeros $\gamma_1, \gamma_2, \gamma_3, \gamma_{41}, \gamma_{42}, \gamma_{57}$ and γ_{58} are detected through $E^{\rho_v}_{4\times4}(\gamma_i)$, as expected. Also at other arguments ϕ_v, seven zeros are clearly detected by $E^{\rho_v}_{4\times4}(\gamma_i)$. As we mentioned earlier, $E^{\rho_u}_{4\times4}(\beta_i)$ and $E^{\rho_v}_{4\times4}(\gamma_i)$ detect the zeros of blurs of not only the size 4×4 but also the sizes 1×3 and 3×3 simultaneously. The results of Fig. 3.1 assure that the detection of the zeros was done successfully.

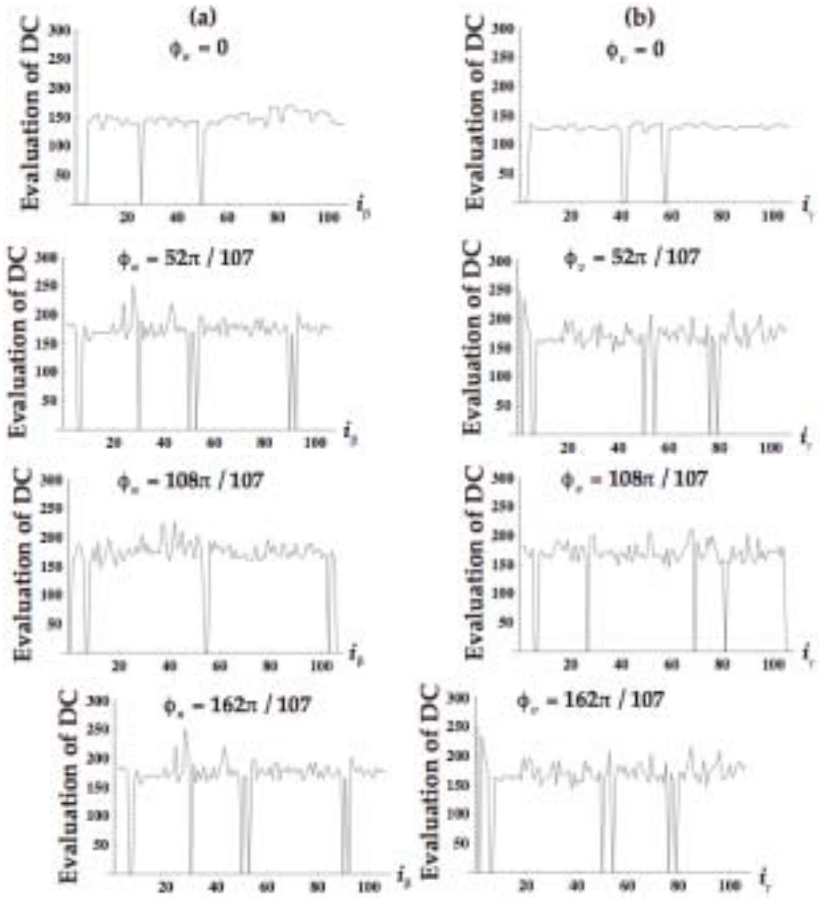


Fig. 3.1 Results of the evaluations of $E^{\rho_u}_{4\times4}(\beta_i)$ and $E^{\rho_v}_{4\times4}(\gamma_i)$. In (a) and (b), $\log\left[\left|E^{\rho_u}_{4\times4}(\beta_i)\right|+1\right]$ and $\log\left[\left|E^{\rho_v}_{4\times4}(\gamma_i)\right|+1\right]$ are plotted for the zeros numbers i_β and i_γ. We took ρ_u and ρ_v as $\rho_u = \rho_u = 1$.

We have shown that the derivative form of the DCs functions very well in detecting zeros of multiple blurs that are convolved in a given image. Next, we illustrate the multi-point form of the DC by using the same image as above.

In Fig. 3.2 we show the results of the numerical evaluations of the multi-point form of the DCs $E^{u}_{4\times4}\left(\beta_i(u_0),\cdots,\ \beta_i(u_{15})\right)$ and $E^{v}_{4\times4}\left(\gamma_i(u_0),\cdots,\ \gamma_i(u_{15})\right)$, where we plot the sizes of them namely, $\log\left[\left|E^{u}_{4\times4}\left(\beta_i(u_0),\cdots,\ \beta_i(u_{15})\right)\right|\times10^{400}+1\right]$ and $\log\left[\left|E^{v}_{4\times4}\left(\gamma_i(v_0),\cdots,\ \gamma_i(v_{15})\right)\right|\times10^{400}+1\right]$. We took Δ_u and Δ_v as $\Delta_u = \Delta_v = \rho e^{-i\Delta\phi}$ and used $\Delta\phi = \pi/20000$ and $\rho = 1$, as optimized parameters. We took $\Delta\phi$ to be very small so that β_i and γ_i picked up belong to the same family, respectively. However, as we mentioned earlier, if we take too small $\Delta\phi$ there is a risk of misjudgment due to the rounding errors. Thus we need to

evaluate the DC in a high precision. Note that the absolute values of $E^{u}_{4\times4}$ and $E^{v}_{4\times4}$ are of the order of 10^{-300}. This is why we multiplied $E^{u}_{4\times4}$ and $E^{v}_{4\times4}$ by 10^{400} for the plotting in Fig. 3.2.

As seen in Fig. 3.2 (a), for $\phi_u = 0$ seven zeros, $\beta_1, \beta_2, \beta_3, \beta_4, \beta_{26}, \beta_{49}$ and β_{50} are found through $E^{u}_{4\times4}\left(\beta_i(u_0), \cdots, \beta_i(u_{15})\right)$ as roots giving origin to the blurs. Also at other arguments ϕ_u, seven zeros due to the blurs are found with $E^{u}_{4\times4}\left(\beta_i(u_0), \cdots, \beta_i(u_{15})\right)$. As seen in Fig. 3.2 (b), for $\phi_v = 0$ seven zeros $\gamma_1, \gamma_2, \gamma_3, \gamma_{41}, \gamma_{42}, \gamma_{57}$ and γ_{58} are found as those due to the blurs with $E^{v}_{4\times4}\left(\gamma_i(v_0), \cdots, \gamma_i(v_{15})\right)$. Also at other arguments ϕ_v, seven zeros are detected through $E^{v}_{4\times4}\left(\gamma_i(v_0), \cdots, \gamma_i(v_{15})\right)$. Note that the zeros detected through $E^{u}_{4\times4}\left(\beta_i(u_0), \cdots, \beta_i(u_{15})\right)$ and $E^{v}_{4\times4}\left(\gamma_i(v_0), \cdots, \gamma_i(v_{15})\right)$ are the same as those found with the DCs of the derivative form (see Fig. 3.1).

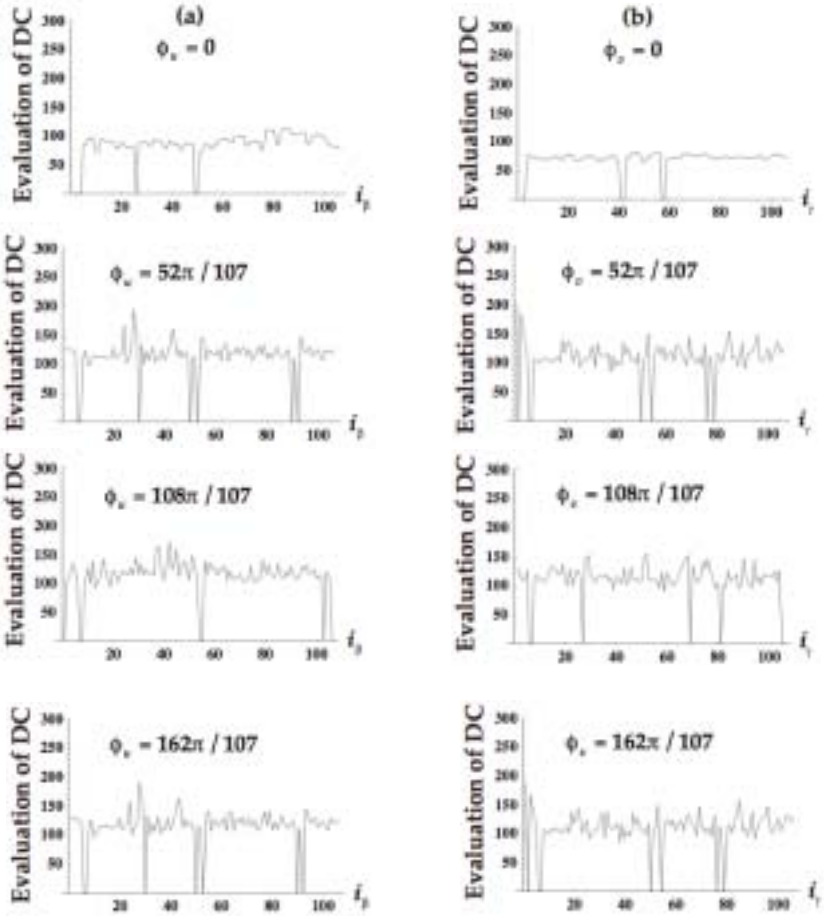


Fig. 3.2 Results of the evaluations of the DCs $E^{u}_{4\times4}\left(\beta_i(u_0), \cdots, \beta_i(u_{15})\right)$ and $E^{v}_{4\times4}\left(\gamma_i(v_0), \cdots, \gamma_i(v_{15})\right)$. (a) and (b) show the results of the evaluations for β_i and γ_i, respectively. We took ρ_u and ρ_v as $\rho_u = \rho_v = 1$. See text for the plotted value.

In Fig. 3.3 we show the image restored by removing seven zeros β_i in v and seven zeros γ_i in u. The restored image is the same as the true image of Fig. 2.2 (a). Thus, we confirm that the zeros of blurs convolved in the given image are correctly detected through $E_{4\times4}^{\rho_u}(\beta_i)$ and $E_{4\times4}^{\rho_v}(\gamma_i)$. These results demonstrate that the DC is very powerful tool for the LB blind deconvolution.

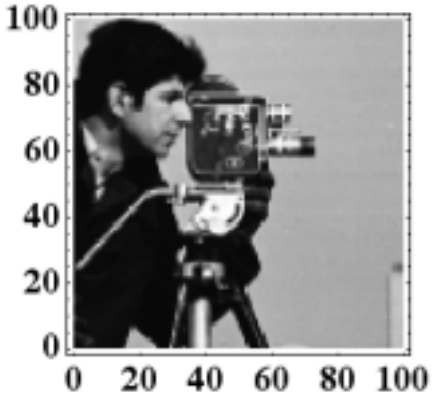


Fig. 3.3 Image restored by removing seven zeros detected through $E_{4\times4}^{\rho_u}(\beta_i)$ and $E_{4\times4}^{\rho_v}(\gamma_i)$, respectively of the derivative form of the DC. Same restored image was obtained by seven zeros spotted through $E_{4\times4}^{u}\left(\beta_i(u_0),\cdots,\ \beta_i(u_{15})\right)$ and $E_{4\times4}^{v}\left(\gamma_i(v_0),\cdots,\ \gamma_i(v_{15})\right)$, respectively of the multi-point form of the DC.

In this section, we illustrated the image restoration by removing the all the zeros of blurs detected at every discrete points of variable u and v. This is the method indicated by LB, and thus one can conclude that DC is a powerful tool for the real implementation of the LB blind deconvolution. However, in reality, once at one value of u or v, a root satisfying DC is found, one can reconstruct the true image using the matrix C. We describe the detail of this highly efficient method in Chap. 4.

4. Highly efficient method using the DC

In the preceding section, we presented Determinant Conditions (DCs). It is based on the idea that the size of the blur is smaller, and often much smaller than that of the true image. The DCs allow one to determine existence of a blur of an assumed size convolved in the image. By virtue of it, the classification of zeros becomes unnecessary, and the trial and error in the choice of zero-sheets is no more needed. Thus, if the assumption about the size of the blur is right, one can instantly tell whether or not a zero is due to the blur. We developed a method to restore the true image by dropping all the zeros coming from the blur at all discrete values of u needed for the inverse Fourier transformation in Sec. 3.3. In this chapter, we further show that it is even not necessary to evaluate the DCs at all the discrete values of u for the reconstruction. One can pick up a single zero satisfying the DCs at any one point of u. One can reconstruct the blur matrix by using the determinant employed in the DCs (det$C$'s). This method allows one to skip huge amount of computation.

4.1 Restoration of the original image

If Eq. (3.2) gives a unique solution for **h** namely h_{xy}, then F can be calculated by dividing G by H that is a z-transform of h_{xy}. The detail of this method will be shown in Sec. 4.2. This is clear that this method works when the size of the assumed blur in constructing the DCs is the same as that of a convolved blur. In such a case, we can uniquely determine the blur by taking h_{00} to be 1, for example. However, there are cases where Eq. (3.2) does not give a unique solution for **h.** We will consider those cases in the following sections.

4.1.1 Blurs smaller than assumed

As we mentioned in Sec 3.1, it must be noted that if a $\det C$ is evaluated at β_i whereas the real size of the blur $m' \times n'$ is smaller than $m \times n$, the DC is always satisfied. Therefore if a root β_i is found for $G(u,v) = 0$, and this is actually a root of $H(u,v) = 0$, $\det C$ calculated for $m' \times n'$ is equal to 0 if the real size of the blur n' is smaller than n. Therefore the DC can be used for the detection of the existence of a smaller blur. However, in that case, Eq. (3.2) fails to give the blur matrix elements. We will show that even in such a case, it possible to identify the real size of blur and also determine h_{xy}.

The real size of the blur can be found from the rank of the matrix C. If $\det C$ is calculated for 4×4 and the real size of blur is 2×2 that is, $m = 2$, $n = 2$, the rank of the matrix C is 7. This number represents 9 degrees of freedom since the size of the matrix C is 16×16. Taking the case of

$\det C$ for the size 4×4 as an example, we show in Table, how the rank of the matrix C is related to the real size of blur convolved.

Table When the real size of the blur convolved is actually smaller than the size of the DC, the rank of the matrix C changes according to the size of the convolved blur. The relation between the size of the blur and the rank is shown for the case of DC for size 4×4.

The size of blur $m \times n$	The rank of matrix C	Degree of freedom
2×2	7	9
2×3 (3×2)	10	6
2×4 (4×2)	13	3
3×3	12	4
3×4 (4×3)	14	2
4×4	15	1
1×4	4	12

One can construct such table for DC for any size $m \times n$. We do not show the proof here, but one can make this table as follows. First, calculate the degree of freedom. The degree of freedom represents how many possibilities there are to place a piece of $m' \times n'$ in the frame of size $m \times n$. The rank is the difference between mn and the degree of freedom. (The rank of matrix C is always m, and this is an exception of the reasoning above.)

Using such a table, one can find the real size of the blur spotted with the DC from the rank of matrix C. If one finds the real size of blur, one can obtain the elements of the blur imposing that all the elements included in the matrix $m \times n$, but not in the matrix $m' \times n'$ be zero (leaving the $m' \times n'$ elements as unknown). We will demonstrate this special case in Sec. 4.2.2 and Sec. 4.2.3.

4.1.2 One-dimensional blurs

Another case where this method fails is when the blur is a convolution of two one-dimensional blurs, one horizontal and one vertical. In other word, the blur of size $m \times n$ is separable into a blur of $1 \times n$ and a blur of $m \times 1$, namely

$$h(x, y) = h_1(x)\, h_2(y). \tag{4.1}$$

A typical case is when the point-spread function is a Gaussian. For a one-dimensional blur $h(y)$, the z-transform of it is

$$H(v) = \frac{1}{n} \sum_{y=0}^{n-1} h(y)\, v^y, \tag{4.2}$$

which has no u dependence. If one of the roots of $H(v) = 0$ is β_i, then

$$H(\beta) = \frac{1}{n}\sum_{y=0}^{n-1} h(y)\beta^y = 0. \tag{4.3}$$

The det$C$ for any $m \times n$ evaluated at β_i gives 0. In this case, the rank of the matrix C is m, and one can spot the existence of one-dimensional blur convolved. However, in such a case, the det C in general fails to give the blur matrix. In this case, one needs to pick up all the roots $\beta_1, \beta_2, \cdots, \beta_n$ always making use of DC, then the elements of **h** can be calculated as

$$H(v) = A\prod_{i=0}^{n-1}(v - \beta_i) = \frac{1}{n}\sum_{y=0}^{n-1} h(y)v^y. \tag{4.4}$$

To detect the one-dimensional blur in the other direction, one has to repeat the same procedure with respect to the zeros of the variable u. The full procedure of a reconstruction of an image with two one-dimensional blurs is demonstrated in Sec. 4.2.3.

4.2 Demonstration

4.2.1 Blur of assumed size

Fig. 4.1 shows model images used for the illustration purposes in this section. Fig. 4.1 (a) shows an image of size 245 × 245 considered as a true image. Fig. 4.1 (b) represents an arbitrary blur of size 4 × 4. The convolution of these two forms the sample image of size 248 × 248 that is shown in Fig. 4.1 (c).

In this section, a DC of size 4 × 4 $\left(E_{4\times4}^{\rho_u}(\beta_i) \right)$ is used for the detection of the blur.

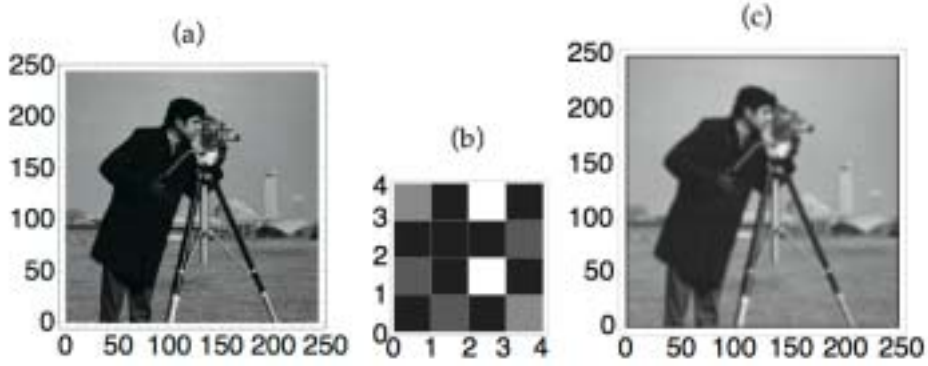


Fig. 4.1 (a) True image of size 245 × 245. (b) Blur image of size 4 × 4 used for the demonstration. (c) The image obtained by convolving the blur (b) and the true image (a). The size of the convolved image is 248 × 248.

How the zeros of the blurs are detected with the aid of $E^{\rho_u}_{4\times4}(\beta_i)$ is illustrated first. The DC is evaluated at $\rho_u = 1$. In principle one can evaluate it at any value of ρ_u.

At $u = 1$, $G(u,v) = 0$ gives 247 solutions, among those, 244 must be the solutions of $F(u,v) = 0$, and 3, those of $H(u,v) = 0$. The latter can be spotted with the aid of the DC. Fig. 4.2 shows the results of numerical evaluation of the det C_D's for each of the 247 solutions.

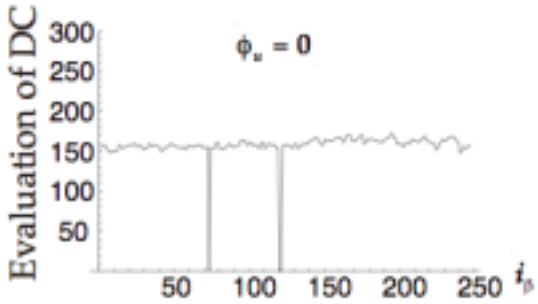


Fig. 4.2 Evaluated result of $E^{\rho_u}_{4\times4}(\beta_i)$ of the image shown in Fig. 4.1 (c). On the vertical axis, the value $\log\left[\left|E^{\rho_u}_{4\times4}(\beta_i)\right| + 1\right]$ is plotted for each β_i. The evaluation is done at $u = 1$, namely with $\rho_u = 1$ and $\phi_u = 1$. One can see that the value is 0 with β_{73}, β_{120} and β_{121}.

As seen in Fig. 4.2, $E^{\rho_u}_{4\times4}(\beta_i)$ give very small value for three solutions β_{73}, β_{120} and β_{121}, and therefore, those are identified as the solution of $H(u,v) = 0$. Equation (3.2) allows one to obtain the matrix elements of **h** with any one of these 3 solutions. As the rank of C_D is 15 and its determinant is 0, Eq. (3.2) gives only the relative ratios of the matrix elements. Then by dividing the Fourier transform of the sample image by the Fourier transform of **h** the true image can be restored.

4.2.2 Multiple blurs smaller than assumed

Next, a sample image which is a convolution of the true image and two blurs, of size 2×2, and 3×3 each, thus both smaller than 4×4, is shown in Fig. 4.3 (c). The size of the sample image is 248×248.

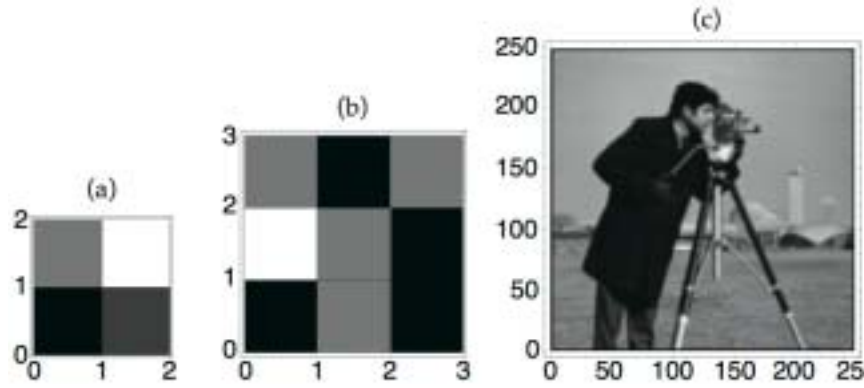


Fig. 4.3 (a) Blur image of size 2 × 2. (b) Blur image of size 3 × 3. (c) The image that was obtained by convolving two blurs (b) and (c) into the true image shown in Fig. 4.1 (a). The size of the convolved image is 248 × 248.

Again, $G(u,v)=0$ is numerically solved at $u=1$, which gives 247 solutions. For each of them, DC is examined, and the results are shown in Fig. 4.4. As was mentioned earlier, $E^{\rho_u}_{4\times4}(\beta_i)$ detects solutions giving origin to blurs not only of size 4 × 4 but also smaller. In the current case, $H(u,v)=0$. for the blur of size 2 × 2 should give 1 solution, and $H(u,v)=0$. for the blur of size 3 × 3 should give 2 solutions. Thus in total 3 solutions should be spotted.

The results shown in Fig. 4.4 (a) demonstrate that the detection of the zeros is done successfully. To find the size of the blur corresponding to those 3 solutions, one calculates the rank of C_{D} for each of the solutions.

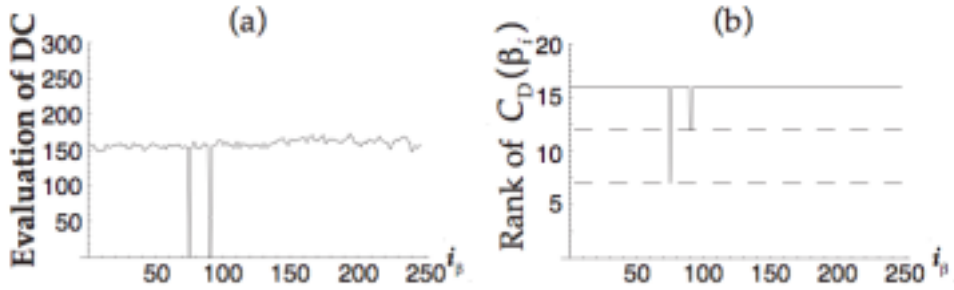


Fig. 4.4 (a) Evaluated result of $E^{\rho_u}_{4\times4}(\beta_i)$ of the image shown in Fig. 4.3 (c). On the vertical axis, the value $\log\left[\left|E^{\rho_u}_{4\times4}(\beta_i)\right|+1\right]$ is plotted for each $C_{\mathrm{D}}(\beta_i)$. The evaluation is done at $u=1$, namely with $\rho_u=1$ and $\phi_u=0$. One can see that the value is 0 with β_{75}, β_{84} and β_{85}. (b) The rank of the matrix $C_{\mathrm{D}}(\beta_i)$ is calculated for each β_i. One can see that the rank is 7 for β_{75}, and 12 for β_{84} and β_{85}.

Fig. 4.4 (b) shows that the calculated rank of C_D for β_{75} is 7, thus it is giving origin to a blur of size 2×2, and the rank of $C_D(\beta_{84})$ and $C_D(\beta_{85})$ are both 12, specifying the size of the corresponding blur to be 3×3.

To calculate the matrix elements of the blur of size 2×2, one solves the Eq. (3.2) by imposing all the h_{ij} elements to be 0 except for h_{00}, h_{01}, h_{10} and h_{11}. Similarly to reconstruct the blur of size 3×3, one solves Eq. (3.2) either with β_{84} or β_{85}, imposing all the elements of **h** to be 0 except for h_{00}, $\cdots$, h_{33}. One can restore the true image by dividing the Fourier transform of the sample image by those of the two blurs.

Fig. 4.5 shows the results. Fig. 4.5 (a) shows the image restored by removing the blur 2×2 alone. 4.5 (b) shows the image restored by removing the blur of size 3×3 alone. Finally, Fig. 4.5 (c) shows the image restored by removing both blurs.

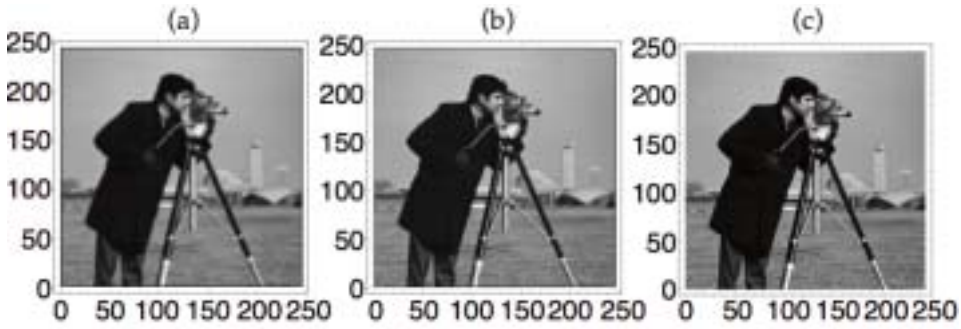


Fig. 4.5 (a) Restored image by removing the 2×2 blur from the image shown in Fig. 4.3 (c). (b) Restored image by removing the 3×3 blur from the image shown in Fig. 4.3 (c). (c) Restored image by removing the 2×2 and 3×3 blurs from the image shown in Fig. 4.3 (c).

4.2.3 One-dimensional blurs

Finally, a sample image which is a convolution of the true image and a Gaussian blur that consists of two one-dimensional blurs, is processed. The size of the sample image shown in Fig. 4.6 (d) is 248×248.

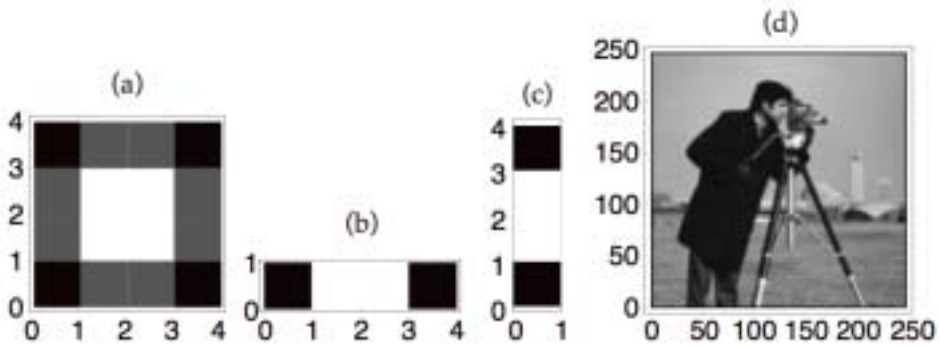


Fig. 4.6 (a) : Blur image of size 4 × 4. It can be decomposed to two one-dimensional blurs shown in (b) and (c). (b) One-dimensional blur of size 1 × 4. (c) One-dimensional blur of size 4 × 1. (d) The image obtained by convolving blur (a) into the true image shown in Fig. 4.1 (a). The size of the convolved image is 248 × 248.

Again, $G(u,v) = 0$ is numerically solved at $u = 1$, which gives 247 solutions. For each of them, $\det C_D$ was calculated, and the results are shown in Fig. 4.7 (a). It shows the values of $E^{\rho_u}_{4\times4}(\beta)$ for all 247 solutions. The solutions β_3, β_{84} and β_{85} that satisfy DC are the candidates for the blur. Fig. 4.7 (b) shows the rank of C_D for each of the solutions. According to Table, β_3, β_{84} and β_{85}, are identified as one-dimensional blur since the rank of $C_D(\beta_3), C_D(\beta_{84})$ and $C_D(\beta_{85})$ are all 4. By using all of β_3, β_{84} and β_{85} in Eq. (4.4) one can reconstruct the elements of one-dimensional blur apart from the common factor A. The reconstructed blur is found to be identical to the one shown in Fig. 4.6 (b). One repeats the same procedure for the other dimension namely for v = 1. The results are shown in Fig. 4.8. Fig. 4.8 (a) shows that the solutions β_3, β_{84} and β_{85} among 247 are the candidates for a blur. Fig. 4.8 (b) shows that the rank of C_D for these solutions are all 4, and thus one concludes tha γ_1, γ_{84} and γ_{85} are all from a one-dimensional blur. Therefore using Eq. (4.4) in the same manner as above, one reconstructs the elements of one-dimensional blur shown in Fig. 4.6 (c). One can restore the true image by dividing the Fourier transform of the given image by that of the convolution of two one-dimensional blurs.

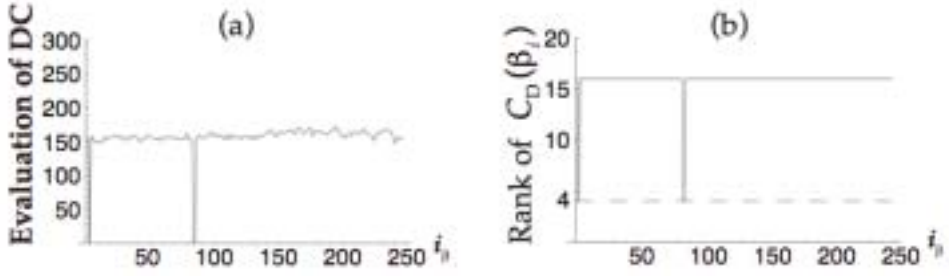


Fig. 4.7 (a) Evaluated result of $E^{\rho_u}_{4\times4}(\beta_i)$ of the image shown in Fig. 4.6 (d). On the vertical axis, the value $\log\left[\left|E^{\rho_u}_{4\times4}(\beta_i)\right|+1\right]$ is plotted for each β_i. The evaluation is done at u = 1, namely with ρ_u = 1 and ϕ_u = 0. One can see that the value is 0 with β_3, β_{84} and β_{85}. (b) The rank of the matrix $C_D(\beta_i)$ is calculated for each beta. One can see that the rank is 4 fo β_3, β_{84} and β_{85}.

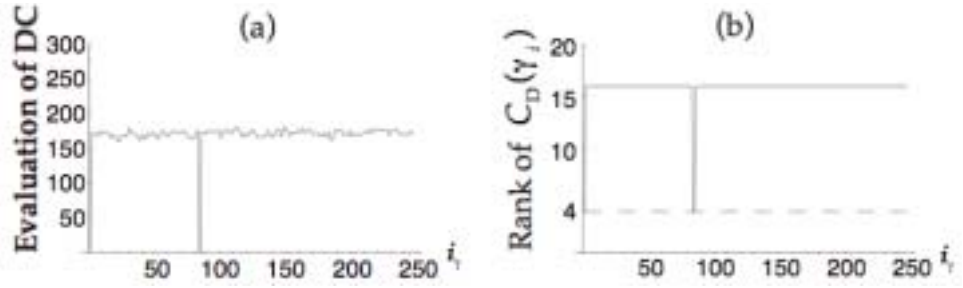


Fig. 4.8 (a) Evaluated result of $E^{\rho_v}_{4\times4}(\gamma_i)$ of the image shown in Fig. 4.6 (d). On the vertical axis, the value $\log\left[\left|E^{\rho_v}_{4\times4}(\gamma_i)\right|+1\right]$ is plotted for each γ_i. The evaluation is done at $v = 1$, namely with $\rho_v = 1$ and $\phi_v = 0$. One can see that the value is 0 with γ_1, γ_{84} and γ_{85}. (b) The rank of the matrix $C_D(\gamma_i)$ is calculated for each beta. One can see that the rank is 4 for γ_1, γ_{84} and γ_{85}.

5. A simple algorithm for eliminating blurs

As we illustrated in Secs. 3.3 and 4.2, the DCs function for finding and eliminating blurs convolved in given images very well, even if the given images are convolution of multiple blurs. In this chapter we present yet another type of algorithm for finding and eliminating blurs, which we refer to as "Simple Search Algorithm" throughout this report. The algorithm makes use of all the zeros found for a given set of u (or v). The detail of this method will be presented in Sec. 5.1, and a demonstration of the reconstruction using this method will be shown in Sec. 5.3

This algorithm is useful in pinpointing only a single blur of a specified size at a time, even multiple blurs are convolved in the given image. But the great advantage of this algorithm is elsewhere. That is, this method is easily extendable to the case of "reduced gray scale". By this term "reduced gray scale", we mean the following. Suppose the full gray scale of the original true image is p bits. When the convolution of a blur occurs, the full gray scale of the blurred image is more than p bits. Now, if the image is "compressed" to p bits, then the convolution is no more mathematically exact due to the rounding error, but rather it is approximative. Let us call this situation a "reduced gray scale". If an algorithm is applicable also to such an image, we call the method "robust" against the reduction of the gray scale.

In this chapter we will further present a variation of the Simple Search Algorithm, which functions to a certain extent when the gray level of the given image is reduced. In such a situation, we can assume an approximate solution for the blurring function, which we will find by introducing an evaluation function χ^2 and by minimizing it as a function of the blur elements and additional parameters. This modification will be presented in Sec. 5.2. A demonstration of this modified version applied to an image with reduced gray levels will be shown in the second half of Sec. 5.3.

5.1 Simple algorithm

First we consider the same situation as that we considered in the previous Chapters, 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. (2.1). For a given u_j, the solutions v of the equation $G(u, v) = 0$ are denoted by ${\beta^j}_i (i = 1, 2, \cdots, N' - 1)$ where suffix j stands for different values of u. Then $G(u, v)$ of Eq. (2.2) can be expressed as

$$G(u_j, v) = k_j \prod_{i=0}^{N-1} \left(v - {\beta^j}_i\right). \tag{5.1}$$

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

$$H(u_j, v) = p_j \prod_{i=0}^{n-1} \left(v - \alpha^j{}_i \right), \tag{5.2}$$

where $\alpha^j{}_i$ is the i-th solution of $H(u, v) = 0$. From Eq. (2.2), it follows that $\left\{ \alpha^j \right\} \subset \left\{ \beta^j \right\}$. The RHS of Eq. (5.2) can be expanded as

$$H(u_j, v) = p_j \sum_{y=0}^{n-1} c_y v^y, \tag{5.3}$$

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-1} which is just 1. Thus we obtain n equations

$$\begin{aligned}
&\sum_{x=0}^{m-1} h(x, 0)\, u_j^x = p_j c_0 = (-1)^{n-1} \; p_j \prod_{i=1}^{n-1} \alpha^j{}_i \\
&\sum_{x=0}^{m-1} h(x, 1)\, u_j^x = p_j c_1 = (-1)^{n-2} \; p_j \left(\prod_{i=2}^{n-1} \alpha^j{}_i + \prod_{i=1(i\neq 2)}^{n-1} \alpha^j{}_i + \cdots + \prod_{i=1}^{n-2} \alpha^j{}_i \right) \\
&\qquad\qquad \vdots \\
&\sum_{x=0}^{m-1} h(x, n-1)\, u_j^x = p_j c_{n-1} = p_j \;,
\end{aligned} \tag{5.4}$$

which can be considered as a set of simultaneous equations for $mn + 1$ unknowns $h(x, y)$ and p_j [5, 8].

Our aim is to determine the blur functions $h(x, y)$ from these equations. The number of independent equations in Eq. (5.4) is n. Therefore, in order to determine $h(x, y)$ uniquely, one needs to obtain Eq. (5.4) for several different values of u_j. Let the number of repetition needed be q. Thus, we have to solve q times the equation $G(u_j, v) = 0$ at different values of u_j. We will then have qn equations, but also will have $mn+q$ unknowns (q being the number of parameters p_j). We, however, need to find only the relative size of h, thus one of the p_j's, say p_0, can be put to 1. Thus the total number of unknowns is $mn + q - 1$. Solving the inequality $mn + q - 1 \leq qn$, one finds that the minimum repetition needed is $q \geq (mn - 1)/(n - 1)$. Then, one can solve $mn + q - 1$ simultaneous equations picked up from these qn equations.

If these equations have a unique set of solutions for $h(x, y)$ and p_j's, then the selected $\beta^j{}_i$ are in reality $\alpha^j{}_i$'s. 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 transformation. However, if the simultaneous equations have no solutions, then it means that the

choice of $\beta^j{}_i$'s as candidates for $\alpha^j{}_i$'s was wrong. If we assume that the q choices of u_j are just right so that in each q steps, one can easily choose a $\beta^j{}_i$ of the same family as in the first step, then the number of possible choices of $\beta^j{}_i$'s for $\alpha^j{}_i$'s is ${}_{N-1}C_{n-1}$ for a given $M \times N$ image. Thus even if we are unlucky, we can reach the right blur after this many trials.

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

1. Assume the size of the blur, and determine accordingly the smallest integer q that satisfies $q \geq (mn - 1)/(n - 1)$.
2. Choose any value u_0 as a starting value of u. Solve the equation $G(u_0, v) = 0$.
3. Solve q-1 more times the equation $G(u_j, v) = 0$ at different values of u_j. Note that we change the values of u_j only slightly every time.
4. Pick up n-1 roots as candidates for α^0's among N roots (β^0's) at value u_0.
5. Select each time, among the N roots found (β^j's), n-1 roots that are close to the ones selected at step 4 as candidates for α^j's at that u_j.
6. Set up Eq. (5.4) for all u_j in order to obtain qn equations. Then, pick up $mn + q - 1$ equations out of these qn equations
7. Solve the set of equations as simultaneous equations. If these equations have a set of solutions, we can reconstruct $H(u, v)$. Furthermore, those $\beta^j{}_i$'s are actually identified as $\alpha^j{}_i$'s that satisfy $H\left(u_j, \alpha^j{}_i\right) = 0$.
8. If the simultaneous equations have no solutions, repeat the step 4 to 7 until the set of equations have solutions, but at most ${}_{N-1}C_{n-1}$ times.
9. Restore the true image $f(x, y)$ by removing the blur using $F(u, v) = G(u, v)/H(u, v)$.

As stated above, the reason why we make small steps in the choice of u_j's in step 4 is, we hope in that case the $\beta^j{}_i$ in a same family are also close. Then the right selection of $\beta^j{}_i$'s in step 5 is easy. However, if the steps are too small, then the whole procedure fails due to the rounding error. Thus, this is the most delicate point in this method.

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

5.2 A modified Simple algorithm

Next we consider a situation where the gray level of the given image is reduced. In such a situation, the convolution given in Eq. (2.1) is broken by the errors caused by the compression of the gray levels. This modified version is, however, based on the assumption that the errors are small. In such a situation, we can assume an approximate solution for the blurring function, which we will find by defining an evaluation function χ^2 and by minimizing it.

In this situation, the simultaneous equations (5.4) do not hold exactly because $\alpha^j{}_i$ in Eq. (5.4) are not exact solutions of Eq. (5.4). We then define χ^2 by

$$\chi^2 = \sum_{i=1}^{q} \chi_j^{\,2}, \tag{5.5}$$

where,

$$\begin{aligned}\chi_j^{\,2} = &\left| \sum_{x=0}^{m-1} h(x,0)\, u_j^{\,x} - (-1)^{n-1} p_j \prod_{i=1}^{n-1} \gamma_i(u_j) \right|^2 + \\ &\left| \sum_{x=0}^{m-1} h(x,1)\, u_j^{\,x} - (-1)^{n-2} p_j \left\{ \prod_{i=2}^{n-1} \gamma_i(u_j) + \prod_{i=1(i\neq 2)}^{n-1} \gamma_i(u_j) + \cdots + \prod_{i=1}^{n-2} \gamma_i(u_j) \right\} \right|^2 + \cdots + \\ &\left| \sum_{x=0}^{m-1} h(x,n-1)\, u_j^{\,x} - p_j \right|^2 \end{aligned} \tag{5.6}$$

and determine unknown $h(x,y)$ and p_j's by minimizing χ^2 by

$$\begin{aligned} &\frac{\partial \chi^2}{\partial h_{00}} = 0, \quad \frac{\partial \chi^2}{\partial h_{01}} = 0, \quad \frac{\partial \chi^2}{\partial h_{10}} = 0, \quad \cdots \\ &\frac{\partial \chi^2}{\partial h_{10}} = 0, \quad \frac{\partial \chi^2}{\partial h_{11}} = 0, \quad \frac{\partial \chi^2}{\partial h_{12}} = 0, \quad \cdots \\ &\quad \vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\quad \cdots \\ &\frac{\partial \chi^2}{\partial p_1} = 0, \quad \frac{\partial \chi^2}{\partial p_2} = 0, \quad \frac{\partial \chi^2}{\partial p_3} = 0, \quad \cdots. \end{aligned} \tag{5.7}$$

When χ^2 is minimized, then we regard $h(x,y)$ and p_j as a set optimal solutions.

In the next section we show how this algorithm functions in restoring blurred images and is robust against reduced gray levels.

5.3 Test of a simple algorithm for finding blurs

In the preceding section, 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. 5.1 (a) shows a 245 × 245 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. 5.1 (b), 5.1 (c) and 5.1 (d) are blurs of sizes 2 × 2, 2 × 3 and 3 × 3, respectively. We convolve these three blurs separately into the true image in each test. The 3 × 3 blur is newly introduced for the present test. Figure 1 (e) shows the convolved image with this blur. The size of the convolved image is 249 × 250.

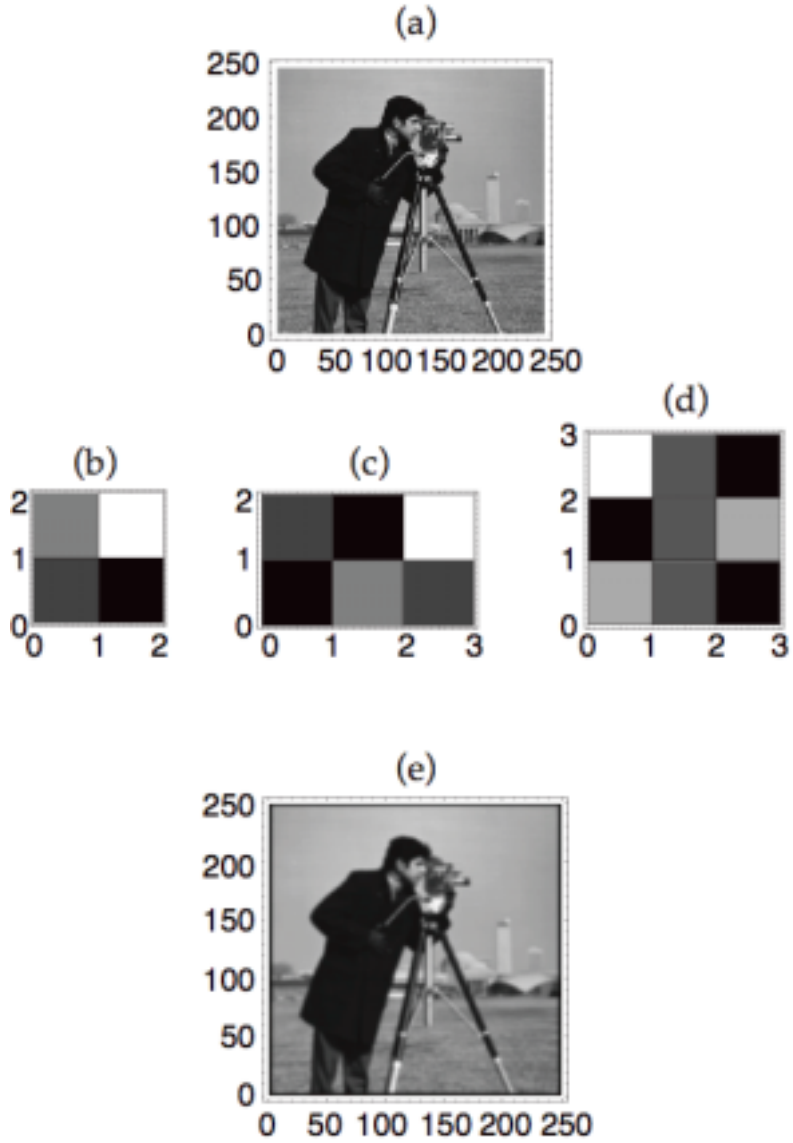


Fig. 5.1 (a) : True image of size 245 × 245. (b) : Blur image of size 2 × 2. (c) : Blur image of size 2 × 3. (d) : Blur image of size 3 × 3. (e) : Image obtained by convolving the three blurs of (b), (c), and (d) into (a). The image size is 249 × 250.

We test how the search algorithm represented by Eq. (5.4) functions in finding each single blur convolved in the true image. To choose u_j's we imposed $|u_j| = 1$, and changed only their arguments. Parameter q is 4, 3, and 5 for each blur of Fig. 5.1 (b), 5.1 (c) and 5.1 (d). Figure 5.2 shows the results of the test. Figure 5.2 (a) shows the image restored by removing the 2×2 blur that has been detected by searching for a 2×2 blur in Fig. 5.1 (e). Fig. 5.2 (b) shows the image restored by removing the 2×3 blur that has been detected by searching for a 2×3 blur in Fig. 5.2 (a). Finally, Fig. 5.2 (c) shows the image restored by removing the 3×3 blur that has been detected by searching for a 3×3 blur in Fig. 5.2 (b). In obtaining the final restored image of Fig. 5.2 (c) we applied the search algorithm three times. In each search, the algorithm functioned very well for blurs of different sizes. In this way we verified that the search algorithm works well in finding at a time a single blur convolved in a given image. This test is just one of the many tests that we have carried out.

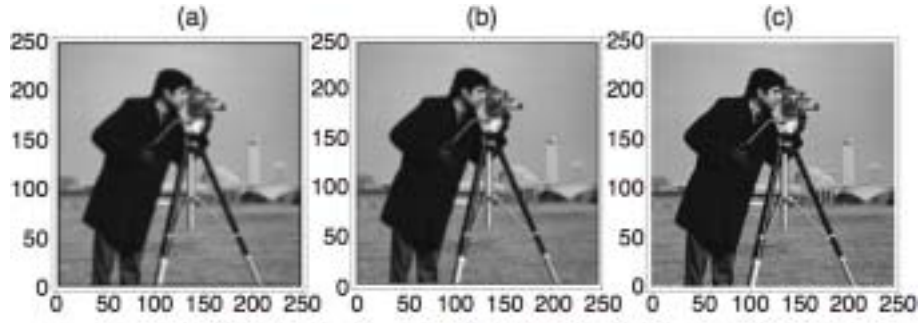


Fig. 5.2. Restored images by removing three blurs that were searched for by the search algorithm. (a) : restored image by removing the 2×2 blur from the image of Fig. 5.2 (e) ; (b) : restored image by removing the 2×3 blur from the image of (a) ; (c) : restored image by removing the 3×3 blur from the image of (b).

Next we show how the search algorithm is robust against noises that disturb the convolution. Figure 5.3 (a) is the true image of size 100×100. Figure 5.3 (c) is the image obtained by convolving the 2×2 blur image of Fig. 5.3 (b). The gray levels are 2676. Figure 5.3 (d) is the image obtained from the image of (c) by reducing the gray levels to 256. In the image with gray level reduced 5.3 (d), the 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 Eq. (5.3) does not hold exactly. Then, we tried to find blur elements $h(x, y)$ with Eq. (5.4). Figure 5.4 shows the restored image by removing the blur that is close to the blur of Fig. 5.3 (b). The restored image is surely enhanced compared to the given image shown in Fig. 5.3 (d). In this way, one can see that the modified search algorithm presented by Eq. (5.4) functions for images with a broken convolution due to the gray-scale reduction.

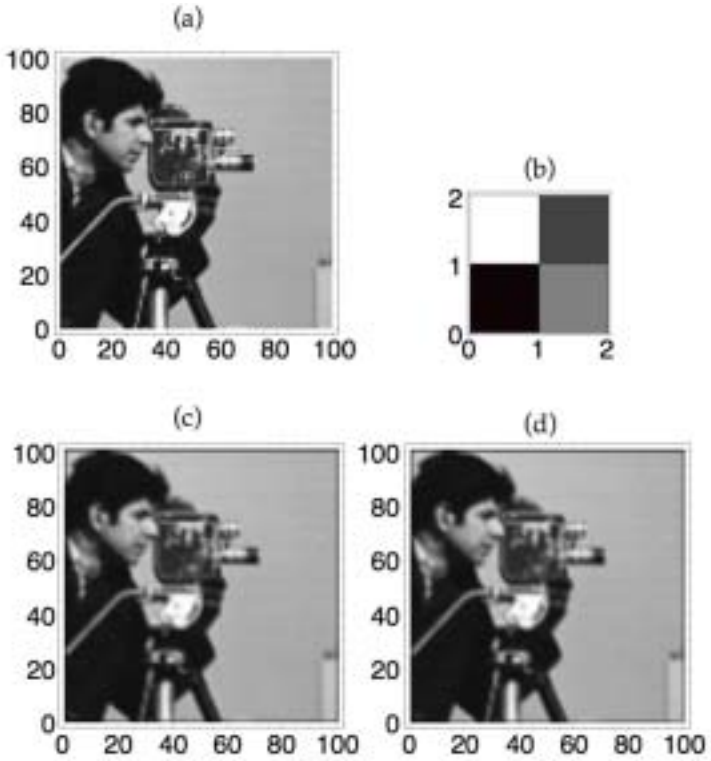


Fig. 5.3 (a) : True image of size 100×100. (b) : Blur image of size 2×2. (c) : Image (2676 gray levels) obtained by convolving the blur of (b) in to (a) ; (d) Image obtained by reducing the gray level of image (c) to 256 gray levels.

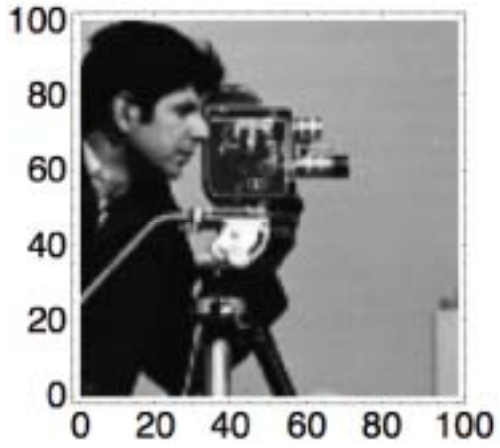


Fig. 5.4 Restored image from the image of Fig. 5.2 (d).

6. A modified highly efficient method

In this section we present a modified highly efficient method, on the basis of the algorithm presented in Chap. 4. Although the highly efficient method given in Chap. 4 is extremely useful in eliminating multiple blurs, the method is not robust against the broken convolution due to the reduction of the gray scale. To improve this point, we developed a modified highly efficient method which is presented in this chapter.

The method functions to a certain extent for an image with gray level reduced. This modified version is based on the fact that errors caused by the gray level reduction of the given image are small. In such a situation, we can assume an approximate solution for the blurring function, which we will find by minimizing an evaluation function χ^2 in the similar way to that used for the modified Simple Search Algorithm in Sec. 5.2.

6.1 Robustness of the derivative form of the DC

We use the same model image as that used in the preceding section, i.e., the one shown in Fig. 5.3. First we show the evaluation of the DC for the convolved image of Fig. 5.3 (c). Next, we test the same DC for the image shown in Fig. 5.3 (d) that was obtained by reducing the gray level of image shown in Fig. 5.3 (c). Figure 6.1 (a) and (b) show the evaluation of $E_{2\times 2}^{\rho_v}(\gamma_i)$ for the convolved image with gray level non-reduced and the image with gray level reduced, respectively. We took the parameter ρ_v and ϕ_v to be 1.3 and 0, respectively.

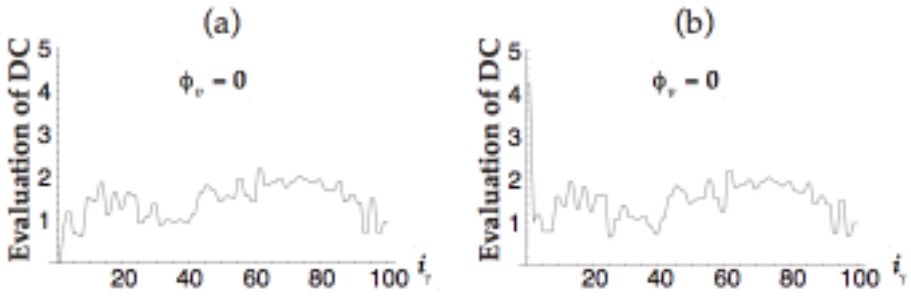


Fig. 6.1 (a) : The result of the evaluation of $E_{2\times 2}^{\rho_v}(\gamma_i)$ for the image with gray level non-reduced of Fig. 5.3 (c). We plotted $\log\left[\left|E_{2\times 2}^{\rho_v}(\gamma_i) \times 10^5\right| + 1\right]$ on the vertical axis. We took ρ_v to be 1.3 and ϕ_v to be 0. $E_{2\times 2}^{\rho_v}(\gamma_i)$ spotted the root γ_1. (b) : The result of the evaluation of $E_{2\times 2}^{\rho_v}(\gamma_i)$ for the image with gray level reduced of Fig. 5.3 (d). No roots were spotted.

Figure 6.1 (a) clearly indicates the zero of blurs. We verified that the true image is certainly reconstructed by removing γ_1. On the other hand, as shown in Fig. 6.1 (b), for the image with the gray levels reduced, it seems very difficult to judge the zero of the blur. Further, we evaluated $E_{2\times2}^{\rho_u}(\beta_i)$, $E_{2\times2}^{\rho_v}(\gamma_i)$ at other various values of ρ and ϕ, but it turned out to be always difficult to identify the zero of the blur for the image with gray level reduced. Then we devised a little modified method that we explain in the next section.

6.2 A variation of the highly efficient method

Next we consider the situation where the gray levels of the given image are reduced. We define χ^2 on the basis of the simultaneous Eq. (3.1),

$$\begin{aligned}\chi^2(\beta_i) = &\left| h_{00} + \beta_i h_{01} + \beta_i^{\,2} h_{02} + \cdots + uh_{10} + u\beta_i h_{11} + u\beta_i^{\,2} h_{12} + \cdots + u^2 h_{20} + u^2 \beta_i h_{21} + \cdots \right|^2 + \\ &\left| \beta_i' h_{01} + 2\beta_i \beta_i' \ h_{02} + \cdots + u' h_{10} + (u\beta_i)' h_{11} + \left(u\beta_i^{\,2}\right)' h_{12} + \cdots + 2uh_{20} + \left(u^2 \beta_i\right)' h_{21} + \cdots \right|^2 + \\ &\left| \beta_i'' h_{01} + \left(2\beta_i \beta_i'\right)' h_{02} + \cdots + u'' h_{10} + (u\beta_i)'' h_{11} + \left(u\beta_i^{\,2}\right)'' h_{12} + \cdots + 2h_{20} + \left(u^2 \beta_i\right)'' h_{21} + \cdots \right|^2 + \cdots .\end{aligned} \tag{6.1}$$

We obtain an approximate solution h_{ij} by minimizing $\chi^2(\beta_i)$ as

$$\begin{aligned}&\frac{\partial \chi^2(\beta_i)}{\partial h_{00}} = 0, \quad \frac{\partial \chi^2(\beta_i)}{\partial h_{01}} = 0, \quad \frac{\partial \chi^2(\beta_i)}{\partial h_{10}} = 0, \ \cdots \\ &\frac{\partial \chi^2(\beta_i)}{\partial h_{10}} = 0, \quad \frac{\partial \chi^2(\beta_i)}{\partial h_{11}} = 0, \quad \frac{\partial \chi^2(\beta_i)}{\partial h_{12}} = 0, \ \cdots \\ &\ \vdots \qquad\qquad \vdots \qquad\qquad \vdots \qquad\quad \cdots .\end{aligned} \tag{6.2}$$

Figure 6.2 (a) shows the evaluation of Eq. (6.1) for the image of Fig 5.3 (c) whose gray levels are not reduced, in which we took the parameters to be $\rho_v = 1.3$, $\phi_v = 0$. As seen is the Fig. 6.2 (a), happens that two zeros were spotted. One of them agrees with the zero γ_1 in Fig. 6.1 (a). The other root, however, is not a zero giving origin to a blur. It was wrongly spotted by accident.

Fig. 6.2 (b) shows the result of the evaluation of $\chi^2(\gamma_i)$ for the image with gray levels reduced of Fig. 5.3 (d). Two zeros were also spotted. One of them must be fictitious (fictive) because there can be only one root with a blur of size 2×2. We show the image reconstructed by using γ_i in Fig. 6.3 (a). The restored image is very close to the true image. On the other hand,

the image reconstructed shown in Fig. 6.3 (b) by using the other γ_i is completely dissimilar to the true image. In this way, although this method may lead to misjudgments by accident, we see that it functions for a image with gray level reduced to a certain extent.

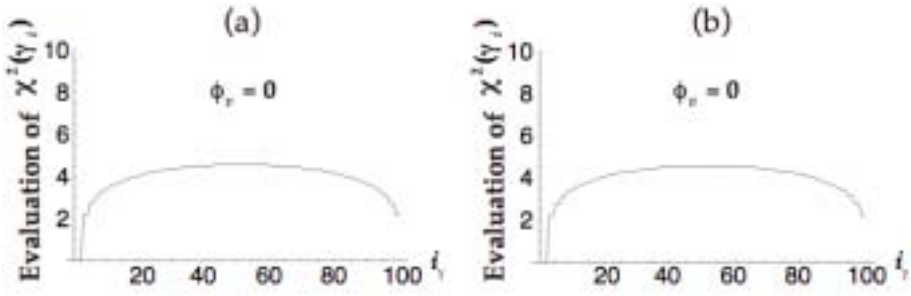


Fig. 6.2 We took the parameters ρ_v and ϕ_v to be 1.3 and 0, respectively. We plotted $\log\left[\left|\chi^2(\gamma_i)\times 10^5\right|+1\right]$. (a) : The result of the evaluation of $\chi^2(\gamma_i)$ for the image with the gray level non-reduced of Fig. 5.3 (c). $\chi^2(\gamma_i)$ spotted the zeros, γ_1 and γ_2. (b) : The result of the evaluation of $\chi^2(\gamma_i)$ for image with the gray level reduced of Fig. 5.3 (d). The zeros, γ_1 and γ_2, are spotted by the $\chi^2(\gamma_i)$. It looks very similar to Fig. 6.2 (a).

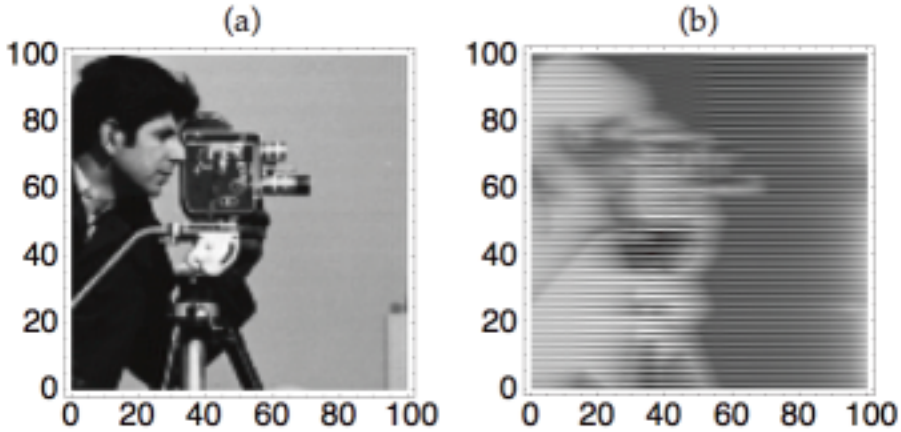


Fig. 6.3 (a) : Reconstructed image by using the root γ_1. It is close to the true image of Fig. 5.3 (a). (b) : Image reconstructed by using the root γ_2. It is clearly dissimilar to the true image of Fig. 5.3 (a).

Figures 6.4 shows $\chi^2(\gamma_i)$ evaluated at other values of v, i.e., $\rho_v = 1.3$, $\phi_v = 26\pi/101$. In this case, only one zero was correctly detected as a zero of the blur, that is γ_1. We can restore the same image as that shown in the Fig. 5.3 (a) by using the root γ_1, although we do not show it.

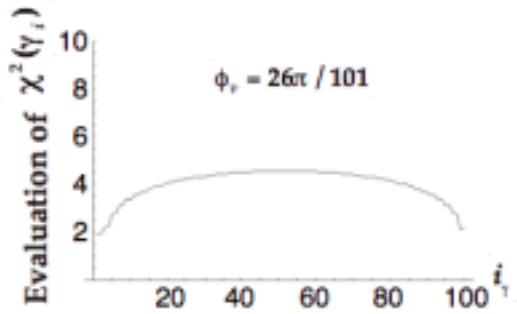


Fig. 6.4 The result of the evaluation of $\chi^2(\gamma_i)$ for the image with the gray level reduced of Fig. 5.3 (d). γ_1 gives the minimum of the $\chi^2(\gamma_i)$, $\chi^2(\gamma_1) = 0.000788$ L. We can restore the image of Fig. 6.3 (a). We took the parameters ρ_v to be 1.3 and ϕ_v to be $26\pi/101$. We plotted $\log\left[\left|\chi^2(\gamma_i) \times 10^5\right| + 1\right]$ on the vertical axis.

7. Summary and Discussion

We have presented two variations of DCs for finding blurs convolved in a given image, namely the derivative form and the multi-point form. The derivative form of the DCs is given in terms of the derivatives of the zeros of the z-transform of the given image. $\det C_{\mathrm{D}}$, the determinant to be used in DC has a form a little bit complicated especially for large sizes, but one can instantly calculate it only by using the values of $g(x, y)$ and β.

Then, we devised the other variation of the DCs, i.e., the multi-point form. This variation is given in terms of only the zeros of the given image evaluated at multiple points of u or v. Its form is quite simple but for the real computation, it might need an optimization procedure of a sampling parameter.

The DCs constructed for $m \times n$ blurs can actually spot blurs of any size 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 total image restoration process. The DCs are very powerful in making the LB blind deconvolution a more practical method.

We also developed a highly efficient method of reconstructing the true image on the basis of the DCs. The DCs are mathematically elegant in the sense that they spot multiple blurs simultaneously. Further the sizes of the blurs can be determined by examining the rank of the determinant of the DC.

We presented yet another form of a search method, i.e., a simple algorithm for finding blurs with a specified size. The search algorithm is given in the form of simultaneous equations for blur matrix elements. The algorithm is meant to spot a single blur of a specified size. Once a blur of a specified size is detected, and the blur elements are obtained, one can easily reconstruct the true image. The advantage of this method is first, it can be easily extended to blurs of larger sizes. And second, this search algorithm can be made robust against a broken convolution due to a reduced gray scale.

We experimentally tested the DCs and the search algorithm by using test images and examined how they function for finding blurs convolved in original images. We have verified that the algorithms work, in a practical processing time, very well for blurs of small sizes convolved in a middle size image.

Now, we examine the methods proposed in this report from the viewpoint of practicality. With the original method proposed by LB, it takes days to process even a relatively simple image.

When the object image is large, one needs to repeat the root finding with very fine steps, and also with a high precision to resolve and classify the solutions forming a Gordian knot. As a rule of thumb, for small and simple images, a precision of 40 digits in the computation suffices, but for larger images, 100 or more digits are needed.

The use of the DC largely simplifies the above procedure. Only at N points in u or in v, one needs to solve the equation. The DC allows one to drop the solutions giving origin to the blur among all the solutions, and then reconstruct the true image. Thus the processing of an image is just a matter of hours. Even if the size of image is large, as large as 1000×1000 pixels for example, it takes about two days to obtain the true image.

The method proposed as highly efficient method in the present report now allows to process an image with a blur of a size 4×4 within a few minutes to one hour. Thus, it can be considered as a first practical method based on the original idea of Lane and Bates. Even if one cannot obtain the blur at one point of u or in v as was mentioned as special cases in Secs. 4.3.2 and 4.3.3, the whole procedure won't take much longer time. These special cases are treated separately in Sec. 4.2.1, and for each case, a prescription was given.

However, the problem with the LB method arises from the noise that is added to the convolution. The random noise as treated by LB becomes less important with the development of the imager devices and data transmission system. The more important source of noise, or rather error, making the numerical treatment difficult is stemming from the image quantization, namely digitization with limited number of gray levels. This is referred to as broken convolution due to the reduced gray scale throughout this thesis. With a development of the storage system and the data transfer facilities, a new standard about the quantization is becoming established. A pixel of an image of DICOM that is for the application of medical sciences is from 12 bits to 16 bits. However, the most popular standard is still 8 bits with 256 levels, in which case the error coming from the quantization is a serious challenge to the numerical treatment of the images. We presented, in this report, a few trials to make the proposed algorithms robust against the reduced gray levels.

All the analyses were executed using *Mathematica*. As an interpreter language, *Mathematica* is not very fast, but it is flexible so that one can for example try different algorithms with desired precisions quite easily. The logarithm used in the evaluation of the $\det C$ shown in figures in this report is with a base of 10.

Acknowledgments

This work has been supported by the Grant-in-Aid for Scientific Research 15650107 from Japan Society for the Promotion of Science, the Science Research Promotion Fund 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] R.G. Lane and R.H. Bates, Automatic multidimensional deconvolution, Opt. Soc. Am, A **4** 180-188 (1987).

[2] Enhancing PET images by means of Bates' blind deconvolution, D. Okano, S. Aogaki, F. Takeutchi and F.M. Toyama, the bulletin of the research institute of advance technology, **4** (2005) 69-106.

[3] Enhancing PET images by means of Bates' blind deconvolution, D. Okano, S. Aogaki, F. Takeutchi and F.M. Toyama, Proc. of the 68th National Convention of Information Processing Society of Japan, March 2006, Tokyo, vol. 2, 2 – 91.

[4] Novel scheme for blind deconvolution: Derivative form, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, IEEE 4th Intern. Conf. in Info. Tech. & Appl., Jan. 2007, Halbin **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 4th Intern. Conf. in Info. Tech. & Appl., Jan. 2007, Halbin **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 4th Intern. Conf. in Info. Tech. & Appl., Jan. 2007, Halbin **1**/230-233;

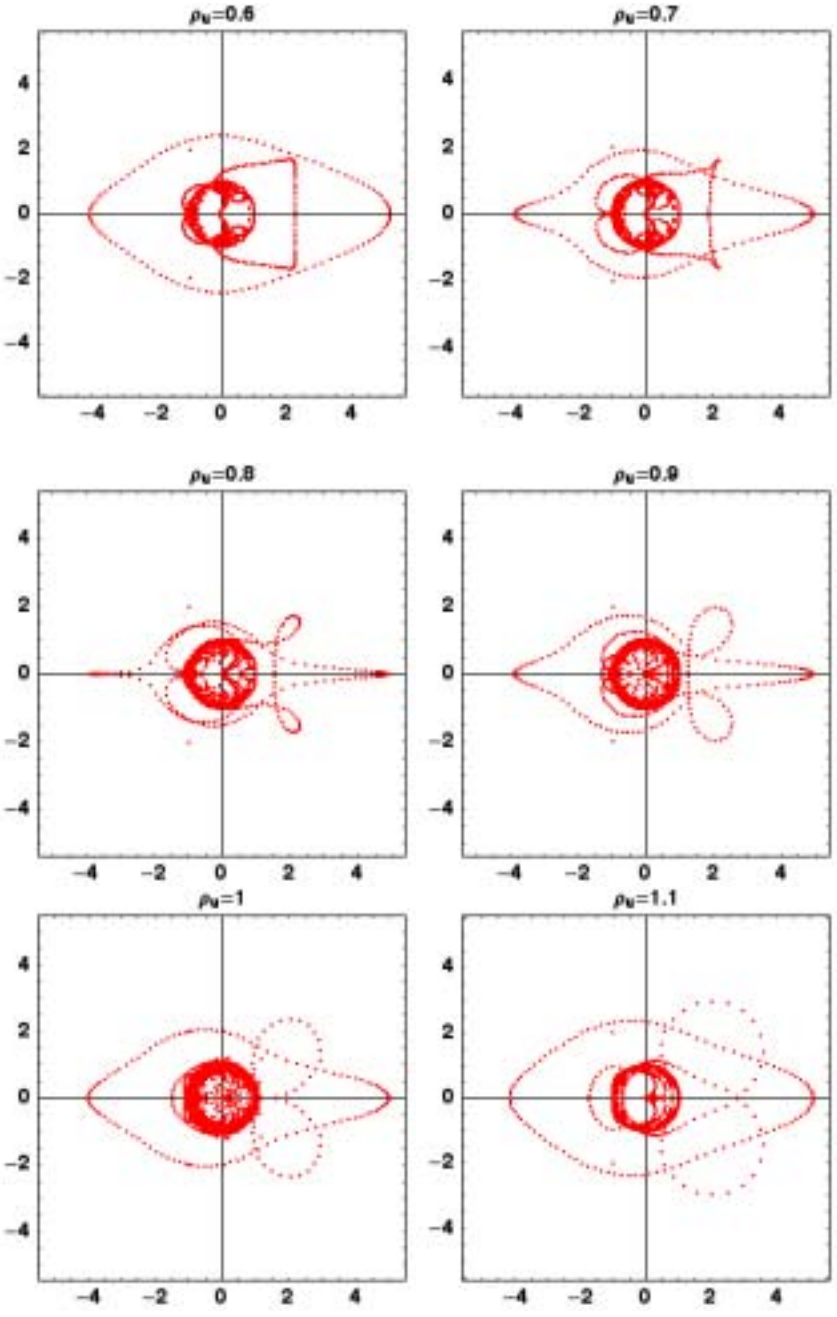
[7] Conditional expressions for blind deconvolution: derivative form, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, Proc. of the 69th National Convention of Information Processing Society of Japan, March 2007, Tokyo, Vol. 2, 2-23~24

[8] Simple method to eliminate blur using Lane Bates algorithm, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, Proc. of the 69th National Convention of Information Processing Society of Japan, March 2007, Tokyo, Vol. 2, 2-25~26.

[9] Conditional expressions for blind deconvolution: multi-point form, S. Aogaki, I. Moritani, T. Sugai, F. Takeutchi and F.M. Toyama, Proc. of the 69th National Convention of Information Processing Society of Japan, March 2007, Tokyo, Vol. 2, 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.

Appendix



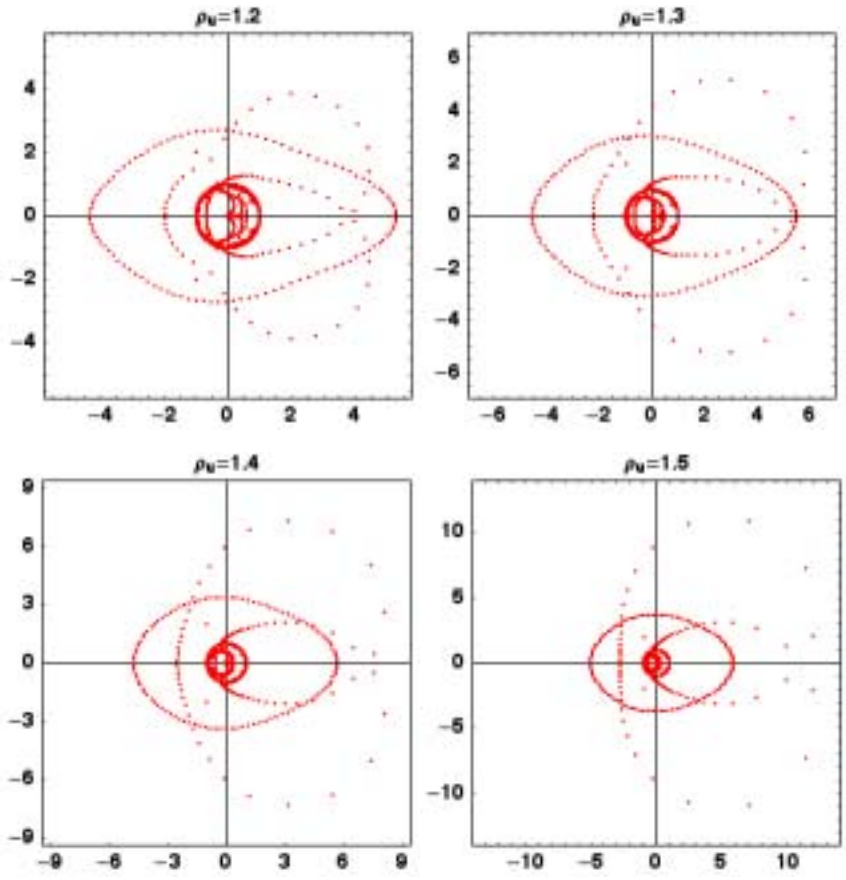
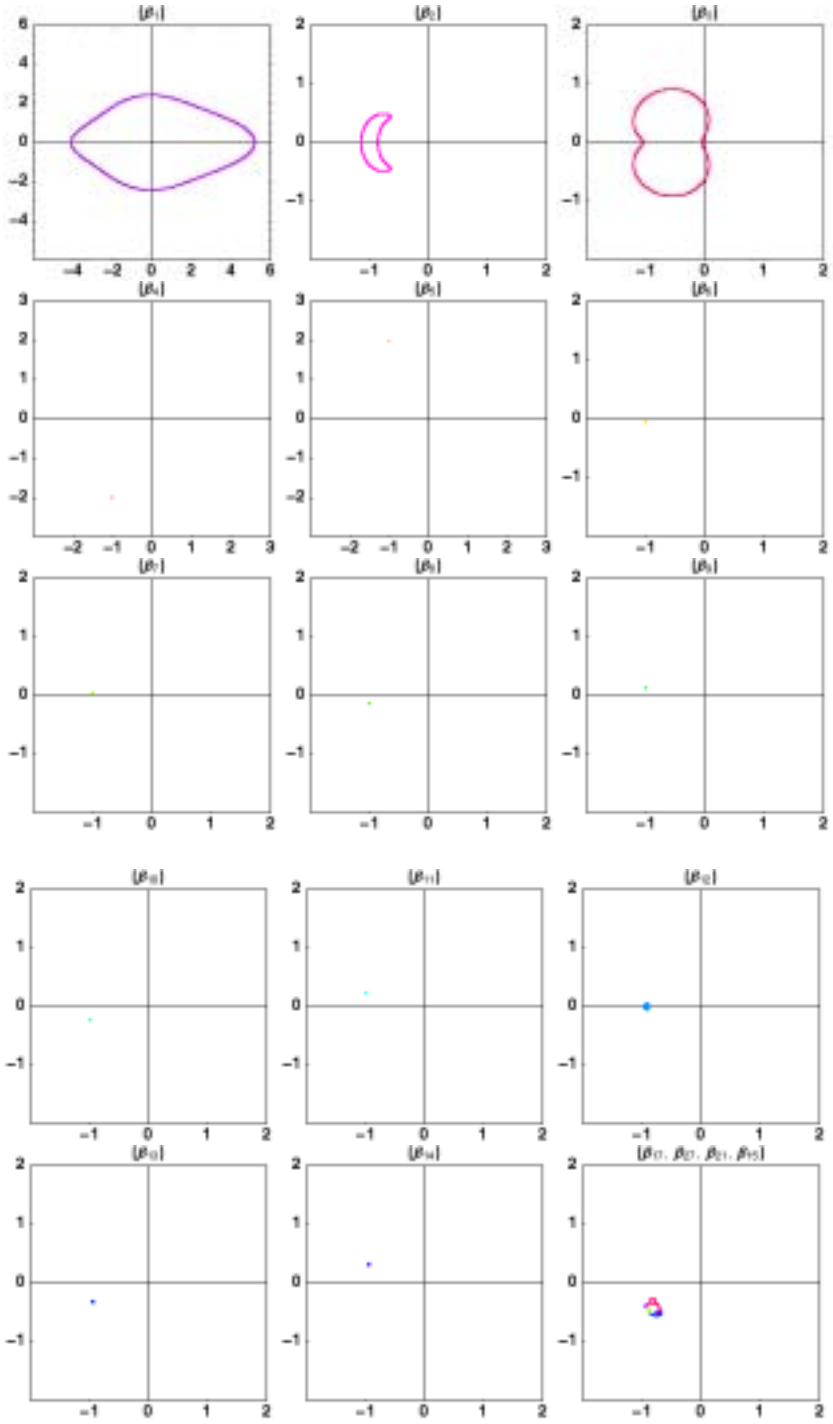
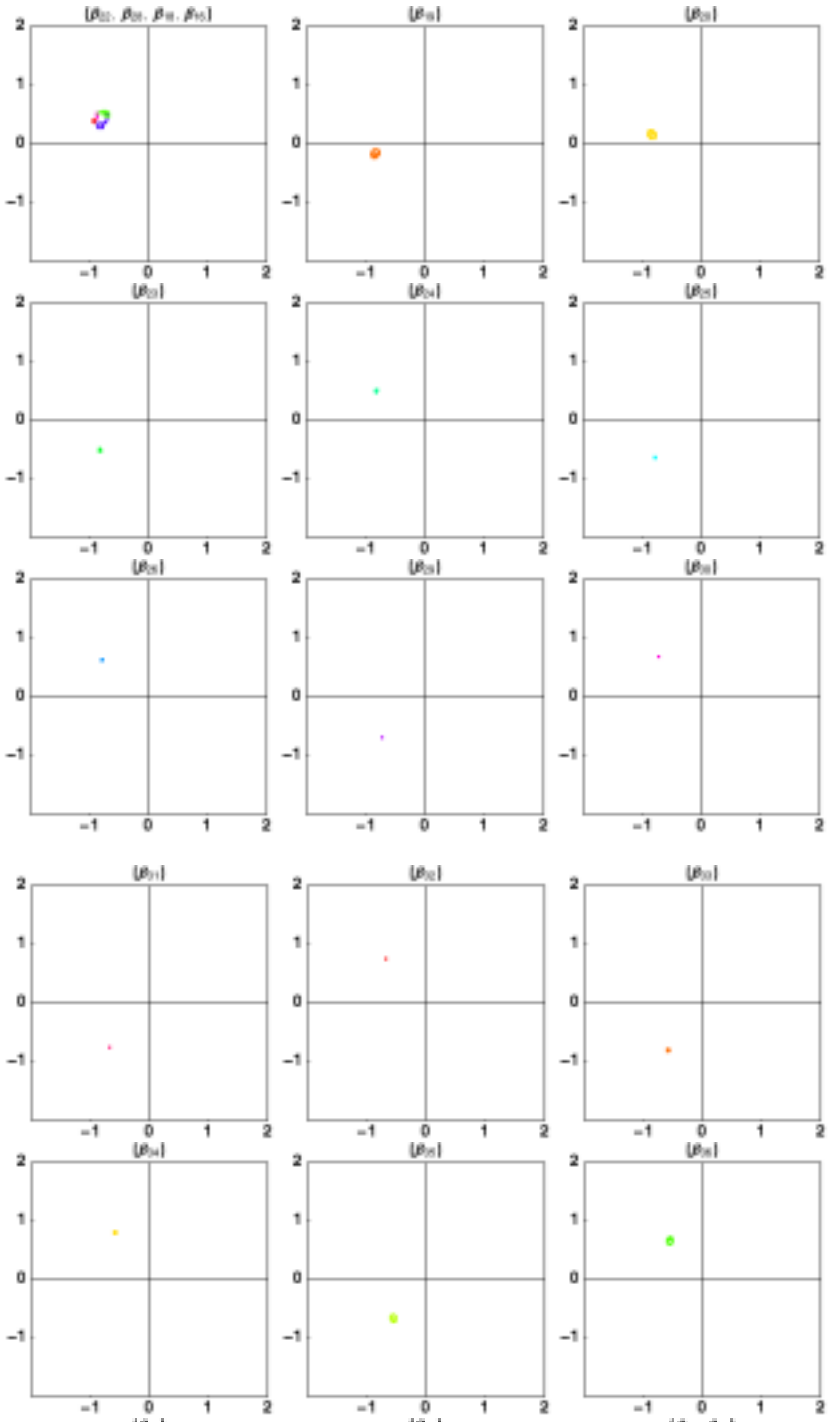
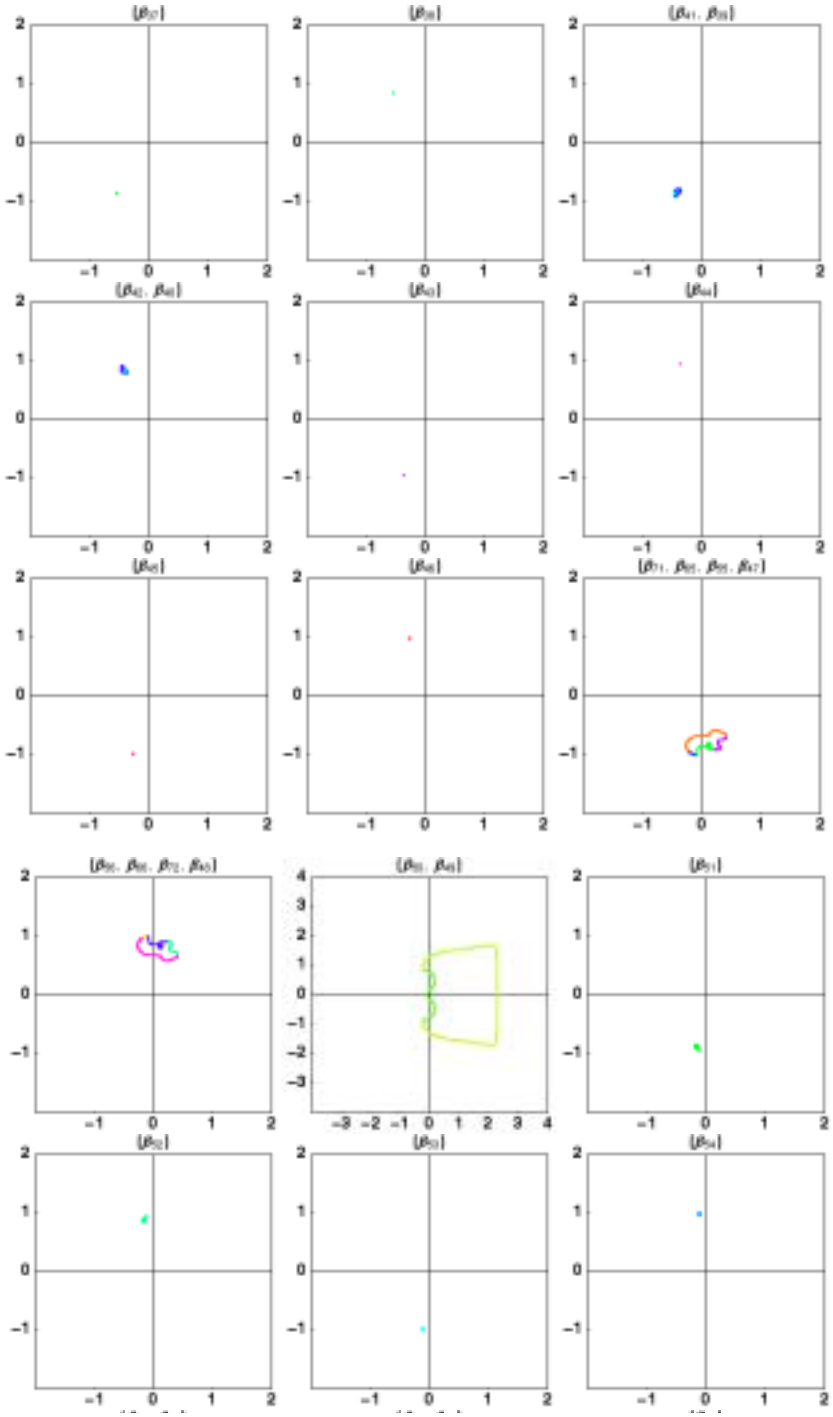
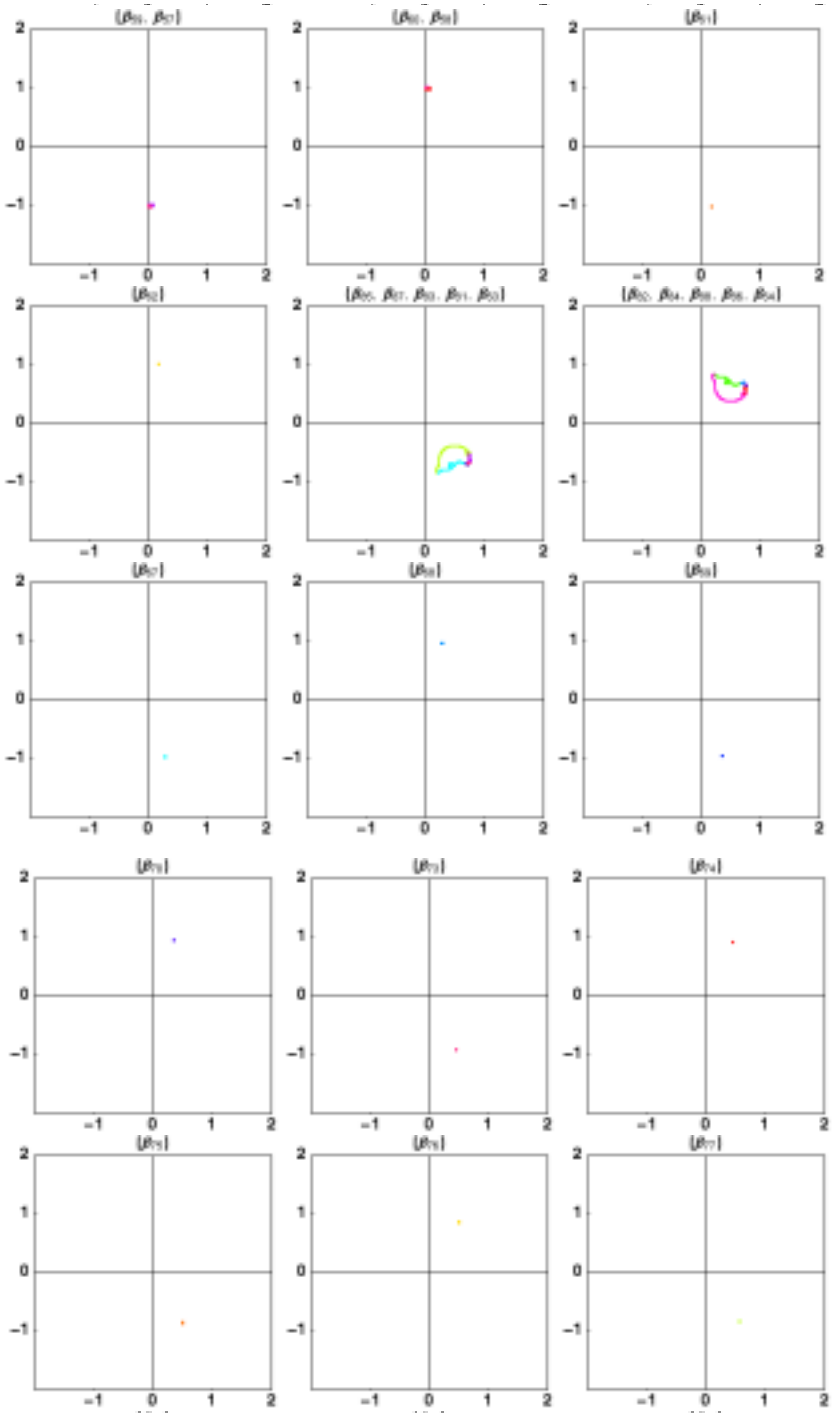


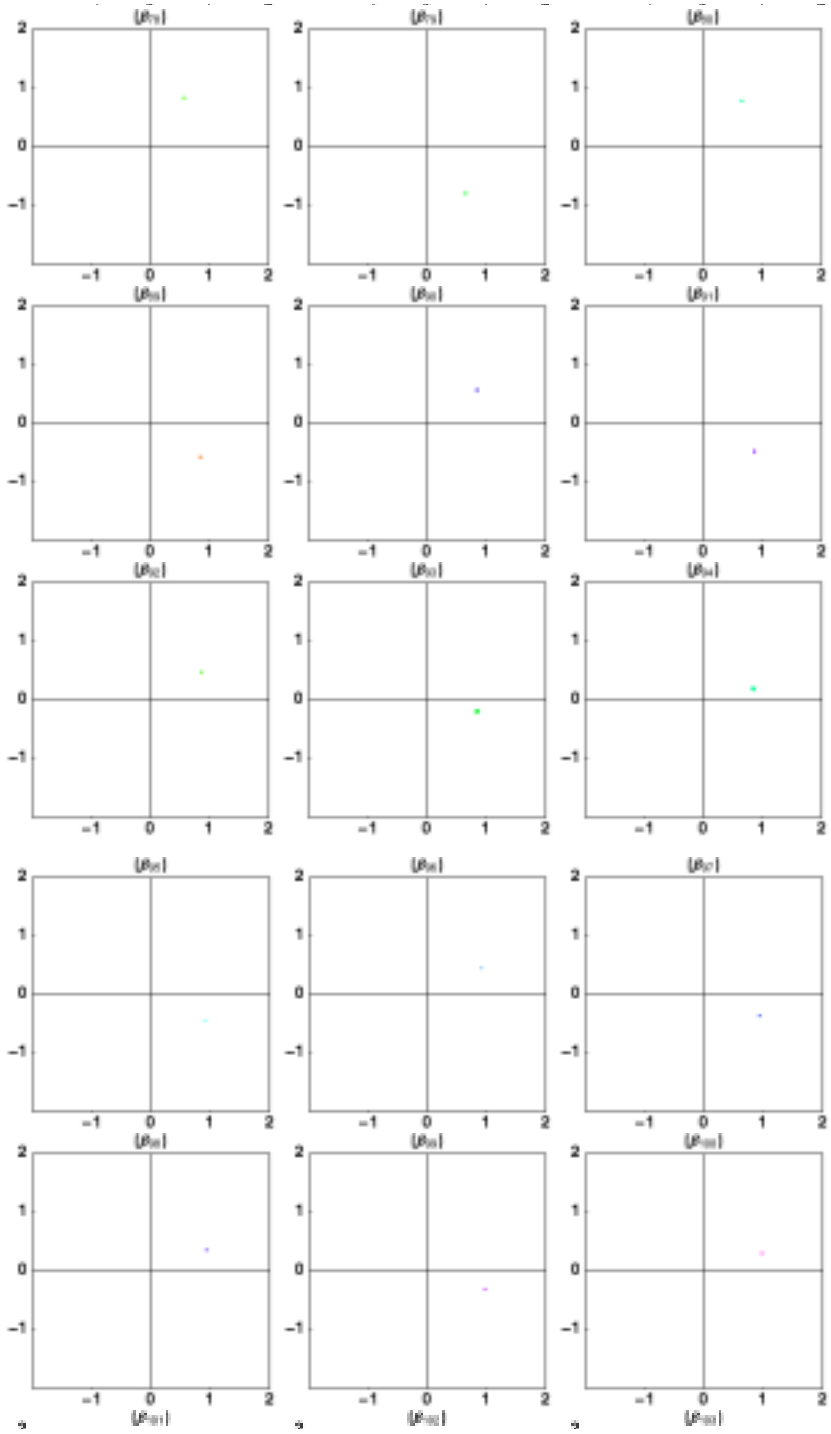
Fig. A2.1 Zero-sheets obtained with various ρ_u.











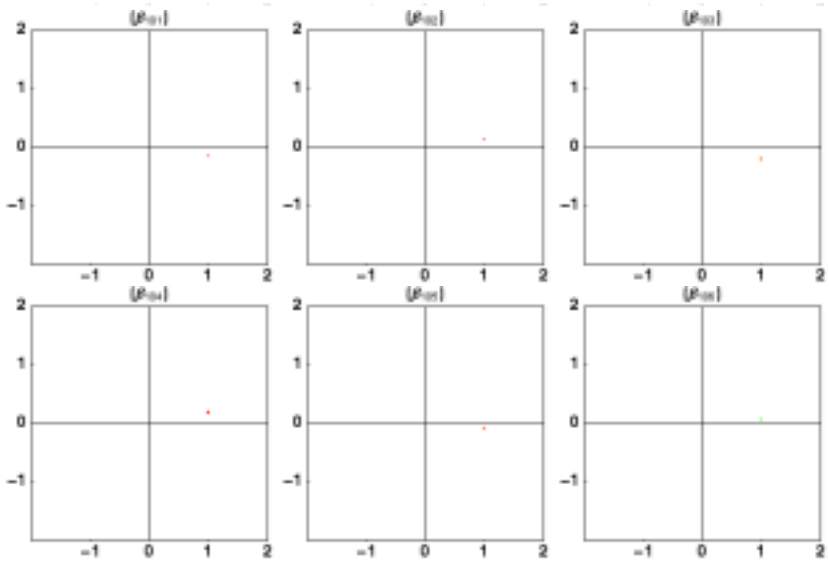
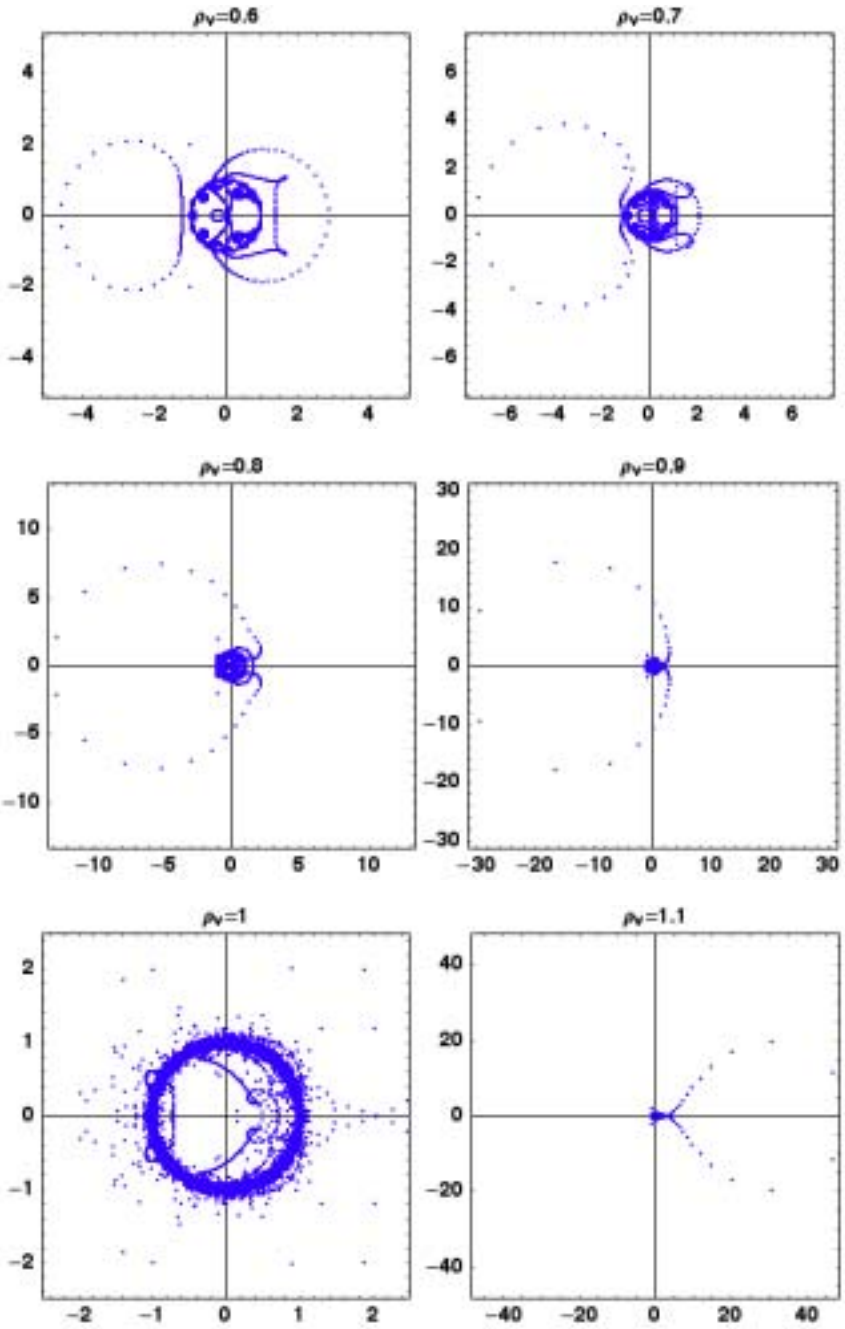


Fig. A2.2 Closed curves of zeros (variable v).


ρV=0.6
ρV=0.7
ρV=0.8
ρV=0.9
ρV=1
ρV=1.1

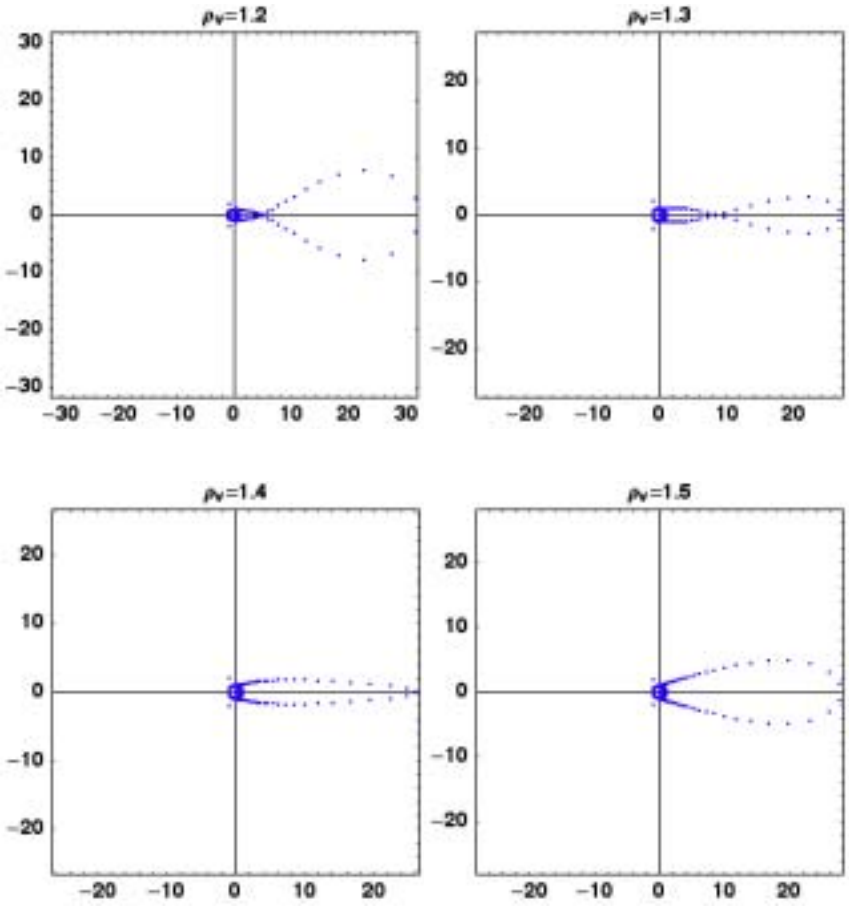
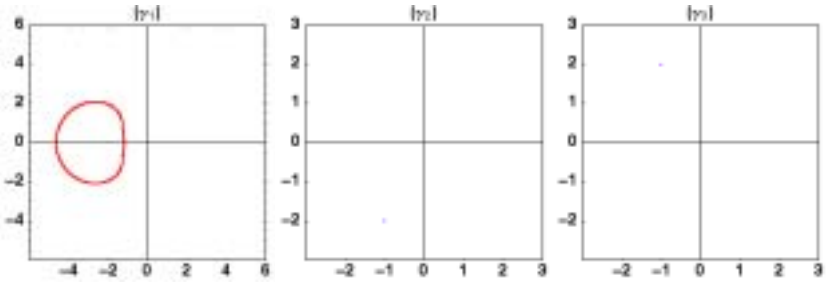
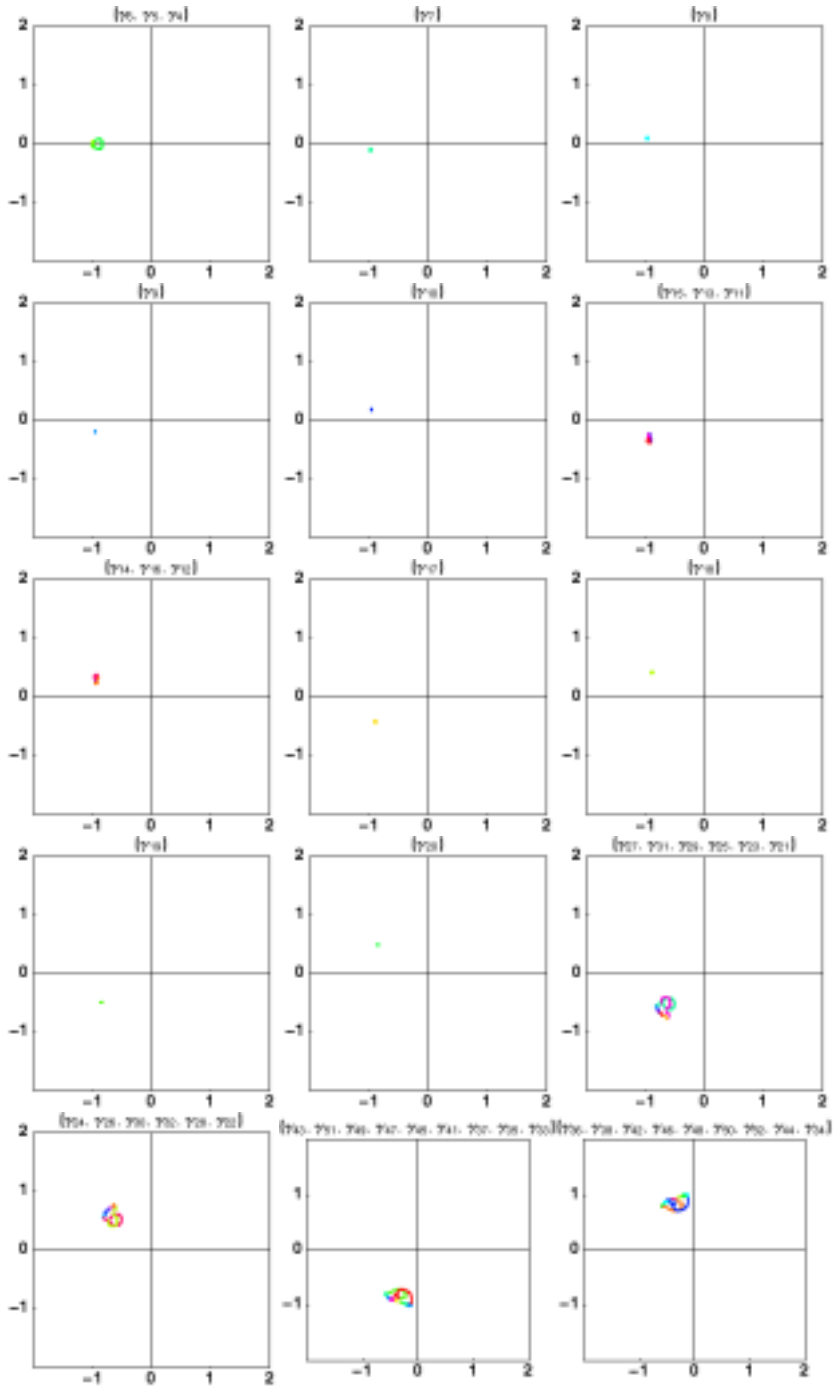
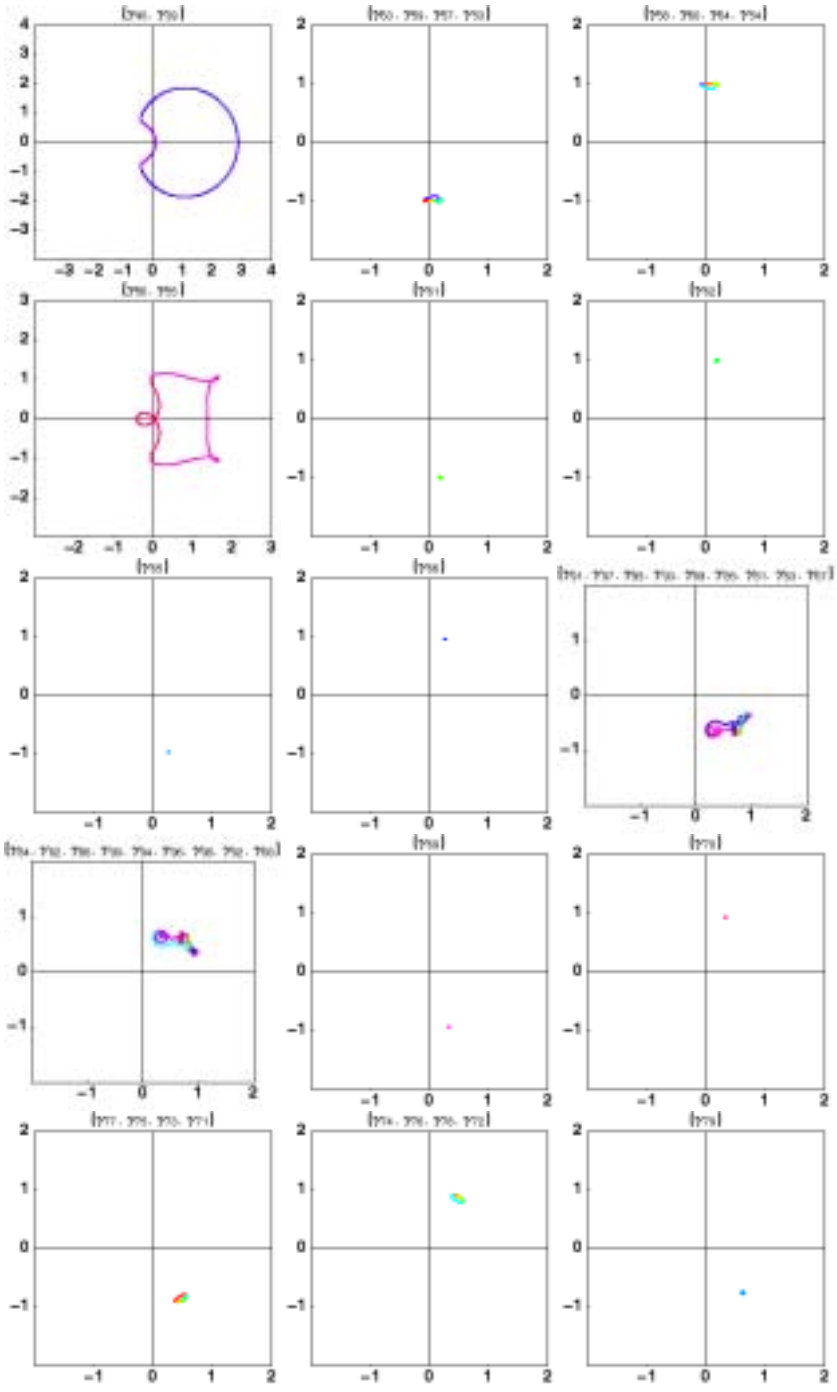


Fig. A2.3 Zero-sheets for various ρ_v.







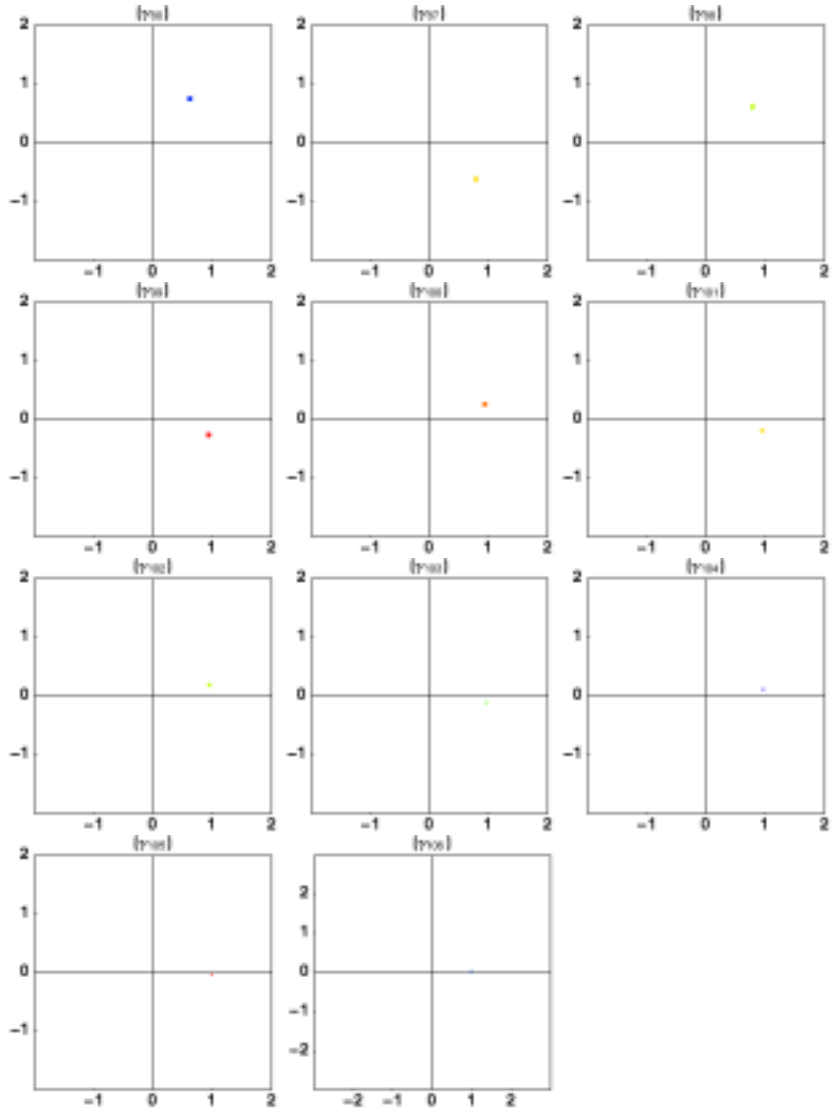


Fig. A2.4 Closed curves of formed with zeros (variable u).

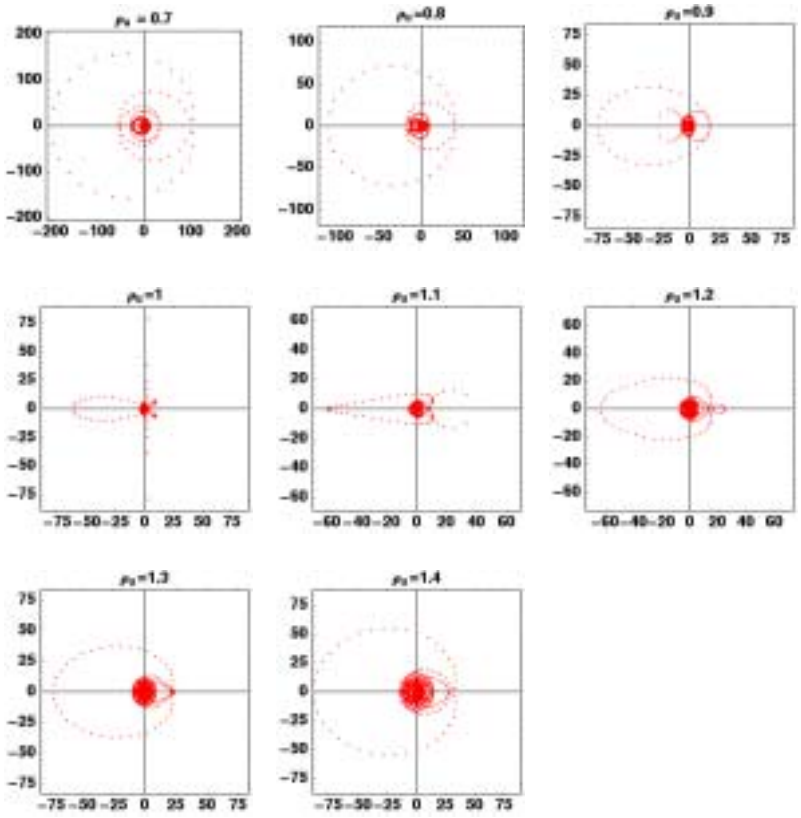
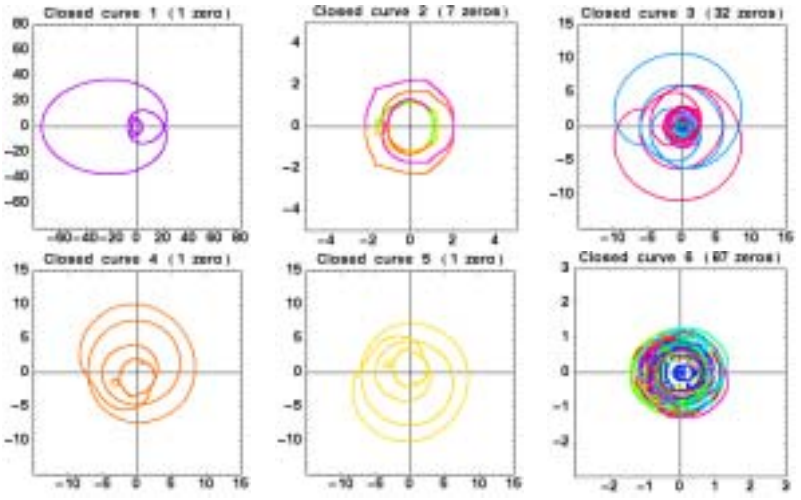


Fig. A2.5 Zero-sheets obtained with various values of ρ_u.



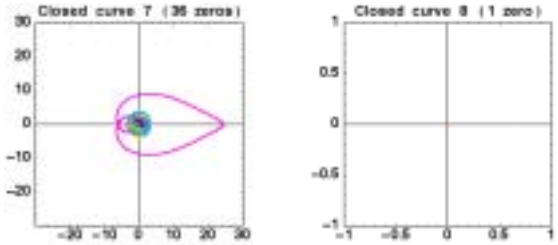


Fig. A2.6 Closed curves of zeros (variable v).

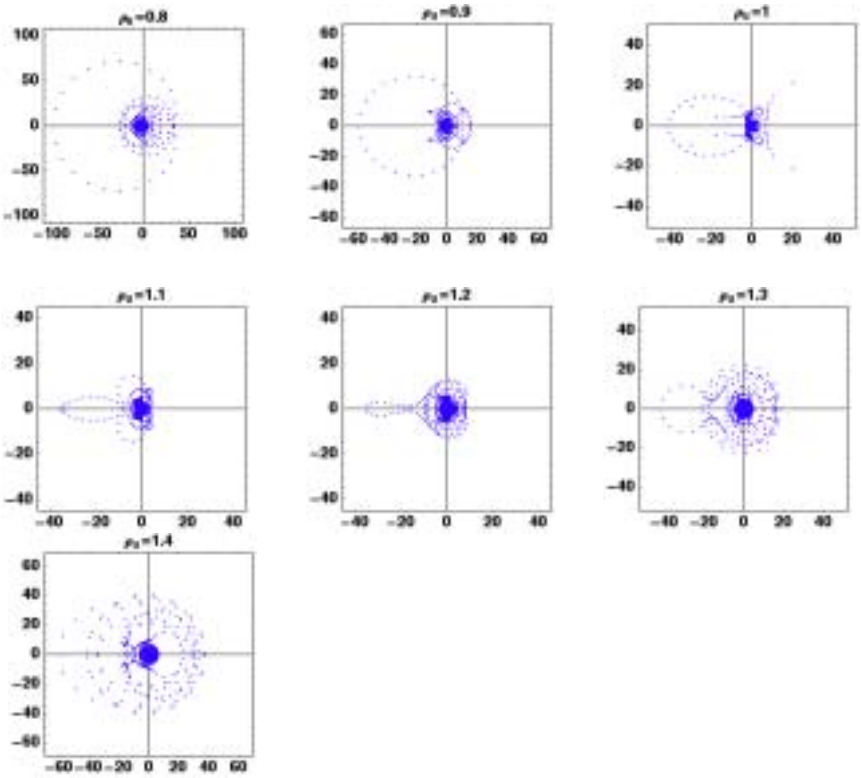


Fig. A2.7 Zero-sheets for various values of ρ_v.

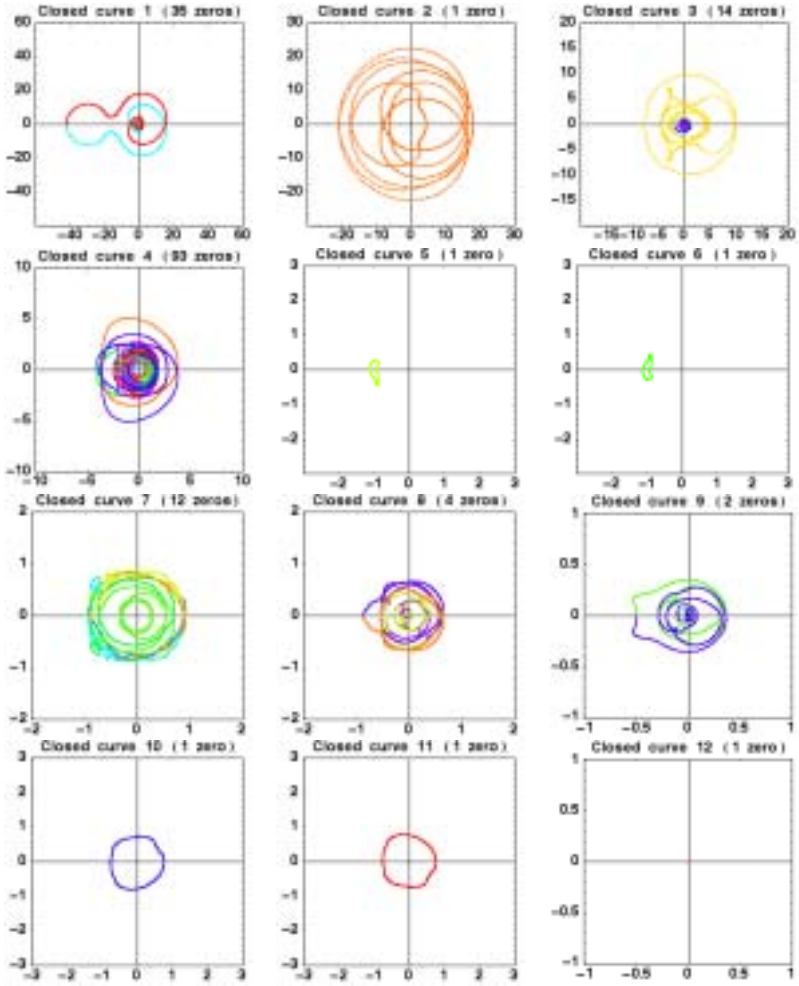
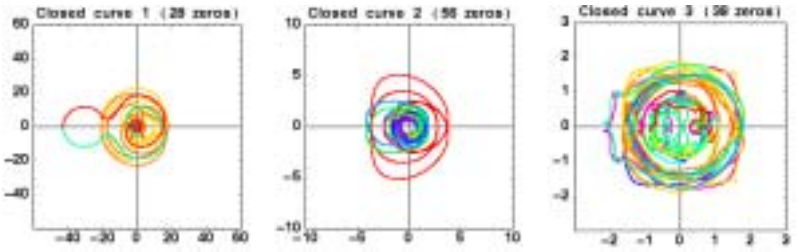


Fig. A2.8 Closed curves of zeros (variable u).



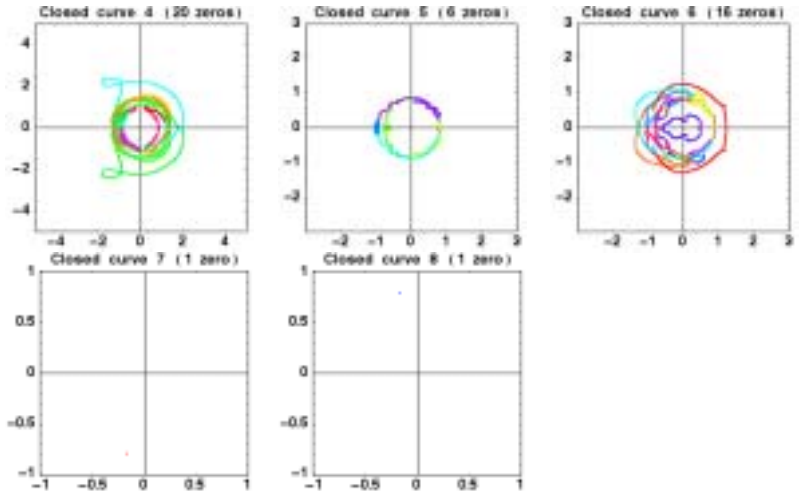


Fig. A2.9 Zero-sheet of the image given in Fig. 2.18 (b), where $\rho_v = 1.3$ is used (variable u).

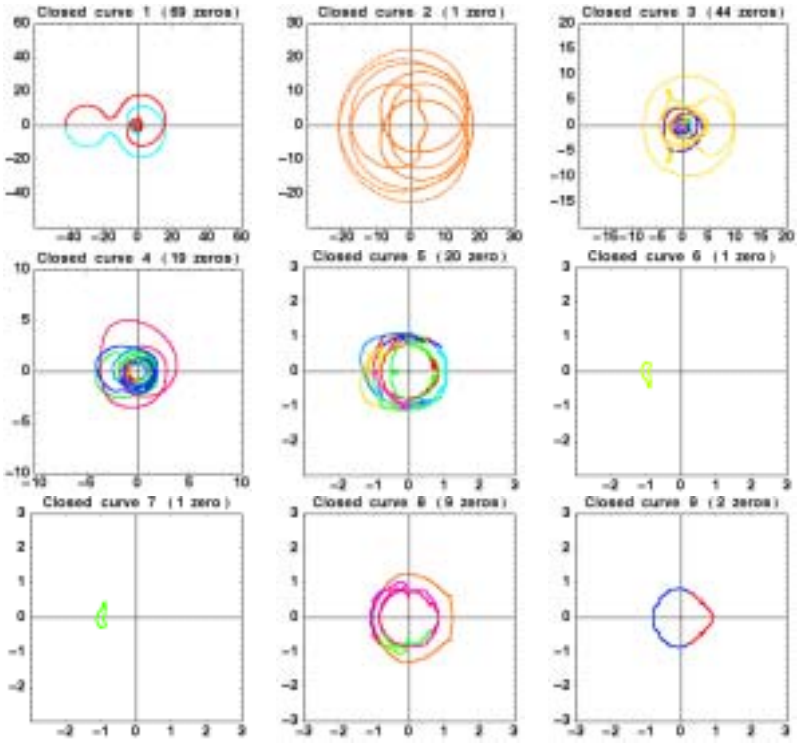


Fig. A2.10 Zero sheets of the image given in Fig. 2.19 (a). $\rho_v = 1.3$ is used. (variable u)

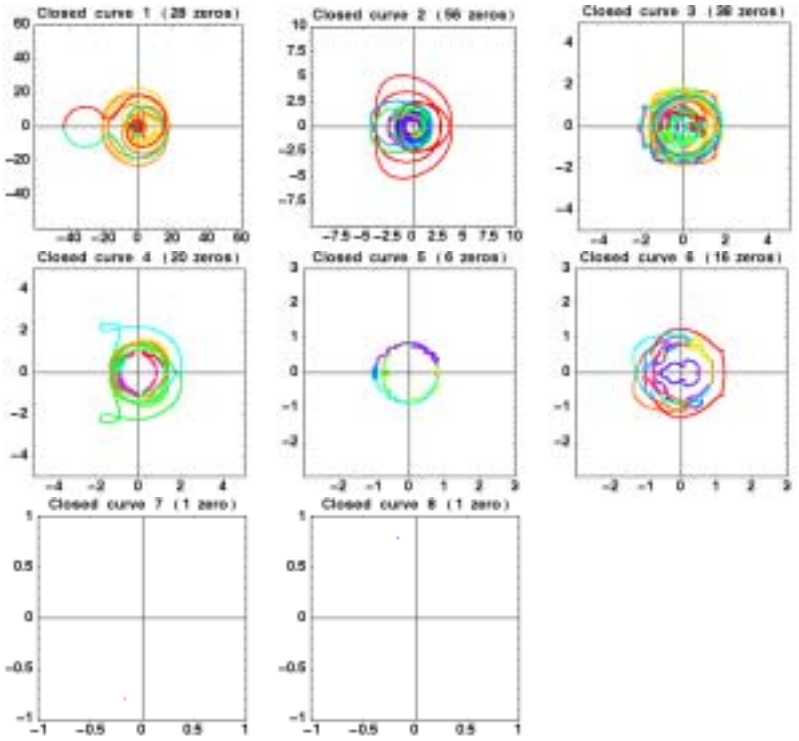
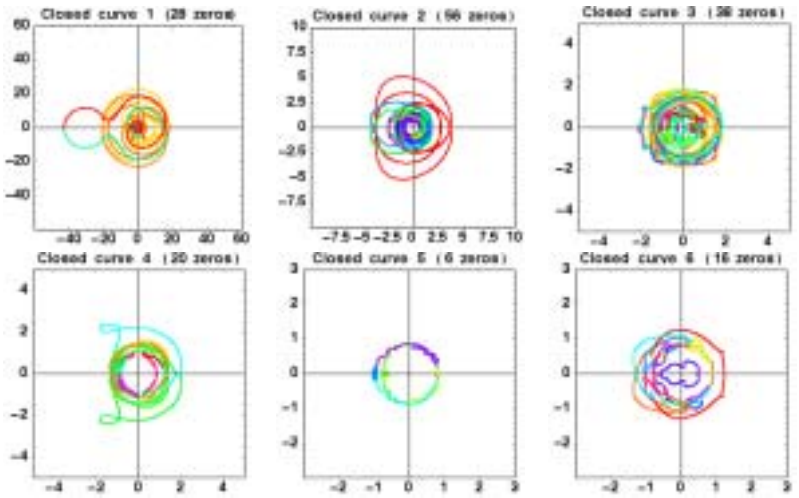


Fig. A2.11 Zero-sheets in the case of (1) (variable u).



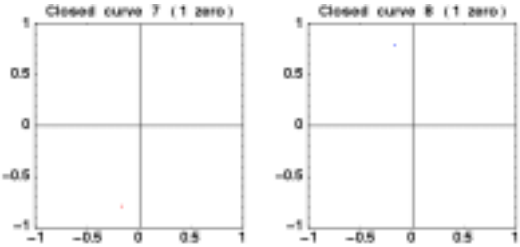


Fig. A2.12 Zero-sheets in the case of (2) (variable u).

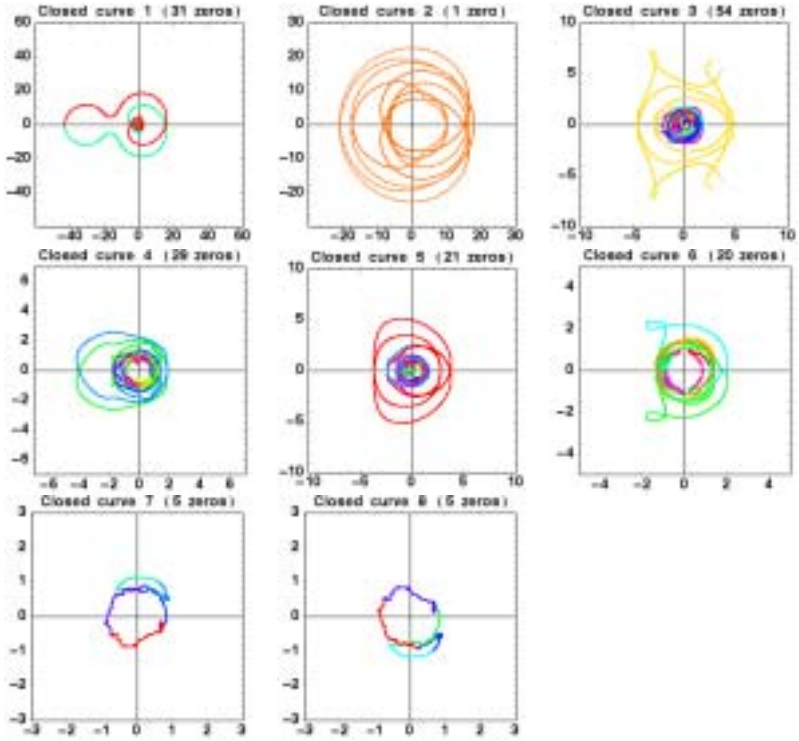


Fig. A2.13 Zero-sheets in the case of (3) (variable u).

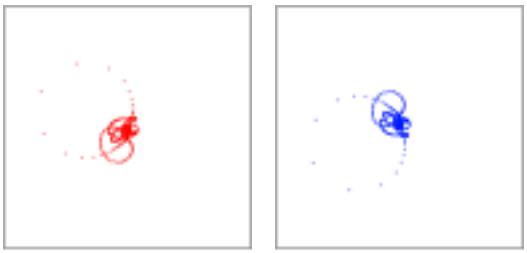


Fig. A2.14 Close-up of the closed curves 7 and 8.