PAPER *Special Section on Signal Processing*

Restoration of Images Degraded by Linear Motion Blurred Objects in Still Background

Karn PATANUKHOM[†a)], ***Nonmember and*** **Akinori NISHIHARA**[††], ***Fellow***

SUMMARY A blur restoration scheme for images with linear motion blurred objects in still background is proposed. The proposed scheme starts from a rough detection of locations of blurred objects. This rough segmentation of the blurred regions is based on an analysis of local orientation map. Then, parameters of blur are identified based on a linear constant-velocity motion blur model for every detected blurred region. After the blur parameters are estimated, the locations of blurred objects can be refined before going to a restoration process by using information from the identified blur parameters. Blur locations are refined by observing local power of the blurred image which is filtered by a high-pass filter. The high-pass filter has approximately a frequency characteristic that is complementary to the identified blur point spread function. As a final step, the image is restored by using the estimated blur parameters and locations based on an iterative deconvolution scheme applied with a regularization concept. Experimental examples of simulated and real world blurred images are demonstrated to confirm the performance of the proposed scheme.

key words: *blur identification, motion blur, image restoration, moving object, still background*

1. Introduction

A motion blur is one of frequently seen types of image degradation which is caused by relative motions between a camera and objects during exposure time. Restoration for the blurred images has been studied [1]–[14]. Most of the existing methods [1]–[9] are focusing on the restoration of space-invariant blur function for the entire images. However, in situations that the degradation is caused by moving objects, the space-invariant assumption is invalid. In order to restore the space-varying blurred image, most of existing methods need multiple images [10], [11] or input from a human [12], [13]. Jiaya Jia [13] developed a single-image-based deblurring scheme by using a concept of image matting. The digital matting is a process to extract foreground objects from the image or video sequence via analysis of a transparency map. The transparency map around a boundary of blurred object is used for estimation of the blur function. However, the scheme needs user to draw strokes to collect samples of blurred object and background elements before extraction of the transparency map. While Anat Levin [14] proposed a fully-automatic deblurring scheme for single blurred image caused by a few moving objects with linear and constant-velocity motion. The method, however, sometimes fails to identify a blur direction especially when the blurred object is small because the blur direction is still estimated by using statistics obtained from the entire image.

In this paper, a fully-automatic and more robust scheme for restoration of motion blur caused by a few moving objects are proposed. The proposed scheme is based on single image and an assumption of linear and constant-velocity motion. Our scheme includes identification of the blurred objects' locations, estimation of the blur parameters and reconstruction of the deblurred image. Section 2 gives models of blurred and desired images for the degradation caused by moving objects. Details of the proposed schemes are presented in Sect. 3. Two experimental examples are demonstrated in Sect. 4. Finally, Sect. 5 gives conclusions of this paper.

2. Image Blurring Model

For the space-invariant blur such as blurs caused by a planar motion of the camera, the blurred image, $g(\mathbf{x})$, has a relation to the desired original scene, $f(\mathbf{x})$, as

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

where $h(\mathbf{x})$ is an impulse response corresponding to the blur function called point spread function (PSF), and $\mathbf{x} = [x\ y]^T$ represents coordinate (x, y) in a vector form. On the other hand, when the blur caused by multiple non-overlapping moving objects in still background is considered, the blurred image can be obtained from parts of blurred objects, $O_i(\mathbf{x})$, and unblurred background, $\mathcal{B}(\mathbf{x})$, as

$$g = \sum_{i=1}^{N_O} [h_i * (v_i \cdot O_i)] + \left(1 - \sum_{i=1}^{N_O} h_i * v_i\right) \cdot \mathcal{B}, \tag{2}$$

while the desired unblurred image is

$$f = \sum_{i=1}^{N_O} (v_i \cdot O_i) + \left(1 - \sum_{i=1}^{N_O} v_i\right) \cdot \mathcal{B}, \tag{3}$$

where $O_i(\mathbf{x})$ is the i-th blurred object, $v_i(\mathbf{x})$ is a binary mask corresponding to the region occupied by the i-th object, and N_O denotes the number of blurred objects. Different from the cases of space-invariant blurs that the desired image can

Manuscript received December 8, 2008.

Manuscript revised February 17, 2009.

[†]The author is with the Department of Communications and Integrated Systems, Tokyo Institute of Technology, Tokyo, 152-8552 Japan.

[††]The author is with the Center for Research and Development of Educational Technology, Tokyo Institute of Technology, Tokyo, 152-8552 Japan.

a) E-mail: karn@nh.cradle.titech.ac.jp

DOI: 10.1587/transfun.E92.A.1920

be obtained by deconvolution of (1) after we know the PSF, in this case, the desired image is reconstructed as (3) by using $v(\mathbf{x})$, $\mathcal{O}(\mathbf{x})$ and $\mathcal{B}(\mathbf{x})$. $\mathcal{O}(\mathbf{x})$ and $\mathcal{B}(\mathbf{x})$ can be extracted based on the relation in (2) after the location of blur, $v(\mathbf{x})$, and the PSF, $h(\mathbf{x})$, have been estimated.

Since the single image provides a limited information, in our proposed scheme, the blur model is simplified by the linear motion blur with constant velocity. The corresponding PSF can be defined by

$$h(x, y; L, \theta) = h(x'_\theta, y'_\theta; L, 0°), \tag{4}$$

$$h(x, y; L, 0°) = \begin{cases} 1/L & ; 0 \leq x \leq L-1, y = 0, \\ 0 & ; \text{otherwise}, \end{cases} \tag{5}$$

where θ represents a motion direction, L represents a motion length, $x'_\theta = x \cdot \cos\theta + y \cdot \sin\theta$ and $y'_\theta = y \cdot \cos\theta - x \cdot \sin\theta$.

3. Four-Stage Deblurring Scheme

An overview of the proposed scheme is shown in Fig. 1. The proposed scheme starts from a rough detection of the blurred objects' locations. An output of the first step is represented by $\{\tilde{v}_i(\mathbf{x})\}_{i=1}^{\hat{N}_O}$ where $\tilde{v}_i(\mathbf{x})$ is a binary mask corresponding to a raw detected location of the i-th blurred object; and $\hat{N}_O$ is the number of detected objects. The rough detection stage is described in Sect. 3.1. Then, the blur parameters are identified based on the linear motion blur with constant velocity model for every detected blurred object. The details are presented in Sect. 3.2. An identified blur PSF for the i-th blurred object is denoted by $\hat{h}_i(\mathbf{x})$. After the PSFs are estimated, the locations of blurred objects can be refined before going to a restoration process. $\hat{v}_i(\mathbf{x})$ represents a refined binary mask for the i-th blurred object. Section 3.3 shows details of the refinement stage. In the last step, the image can be restored by using the estimated blur parameters and locations. The restoration process is described in Sect. 3.4. A restored image is denoted by $\hat{f}(\mathbf{x})$.

3.1 Rough Blurred Region Segmentation

The first step of the proposed scheme is to find the rough locations of the blurred objects, automatically. A flow diagram of this stage is shown in Fig. 2.

STEP 1: Compute variation functions along four directions of 0°, 90°, 45°, and −45°. The variation, $\Delta(\mathbf{x})$, of the blurred image along direction of α is defined as

$$\Delta_\alpha(\mathbf{x}) = |d_\alpha(\mathbf{x}) * g(\mathbf{x})|, \tag{6}$$

where $d_\alpha(\mathbf{x})$ is a filter in the direction of α whose coefficients are given in Table 1. The comparative performances for three types of filters in Table 1 are demonstrated in Sect. 4.

STEP 2: Extract a binary mask for an interesting region called $w_{\mathrm{IR}}(\mathbf{x})$. The interesting region is the region excluding regions with very low frequency or very high frequency components. The low frequency region is defined as a region where every computed directional variation is significantly low (lower than the predetermined lower bound, LB). We expect that there is almost no difference between blurred

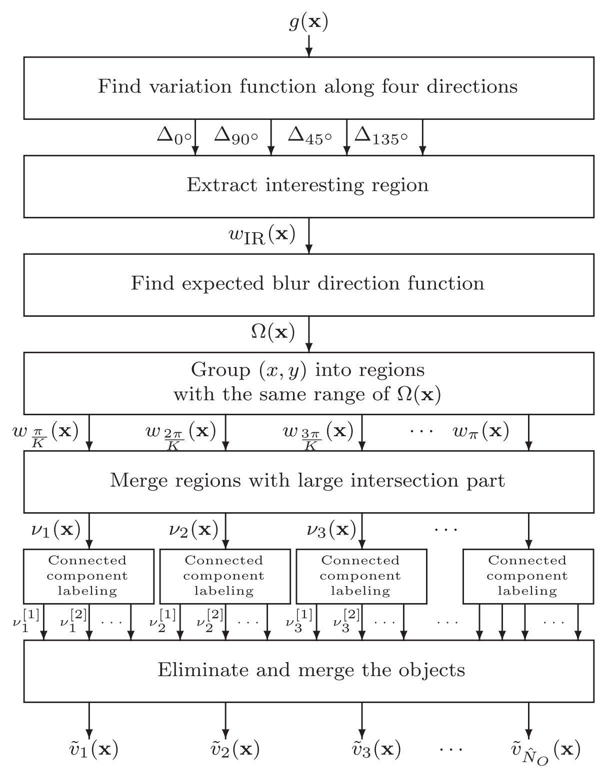


Fig. 2 Proposed rough blurred region segmentation scheme.

Table 1 Coefficients of the directional filters $d_\alpha(\mathbf{x})$ where $\alpha \in \{0°, 90°, 45°, -45°\}$ for the second-derivative operator, the Sobel operator, and the Prewitt operator [16].

	$d_{0°}(\mathbf{x})$	$d_{90°}(\mathbf{x})$	$d_{45°}(\mathbf{x})$	$d_{-45°}(\mathbf{x})$
The 2nd derivative operator	$\begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & -1 \\ 0 & 2 & 0 \\ -1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -1 \end{bmatrix}$
Sobel operator	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}$	$\begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}$
Prewitt operator	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$

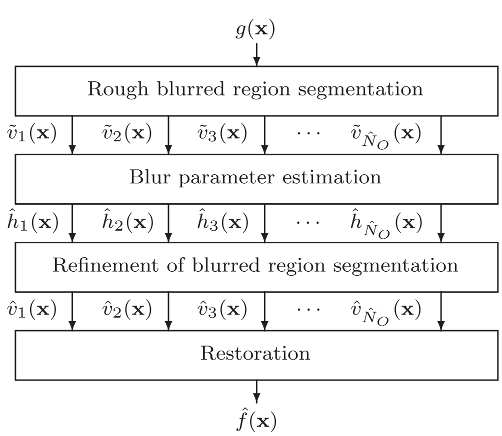


Fig. 1 Proposed deblurring scheme.

and unblurred pixels in this region and the restoration process can be avoided [5]. While the high frequency region is a region where every computed directional variation is significantly high (higher than the predetermined upper bound, UB). Since there is no attenuation of the variations in this region, we can expect that this region is not degraded by the linear motion blur. As a results, w_{IR} can be computed from

$$w_{\mathrm{IR}}(\mathbf{x}) = \mathcal{DT}\left\{\tilde{w}_{\mathrm{IR}}(\mathbf{x})\right\}, \tag{7}$$

where $\tilde{w}_{\mathrm{IR}}(\mathbf{x})$ is a raw function obtained from

$$\tilde{w}_{\mathrm{IR}}(\mathbf{x}) = \begin{cases} 1; & \Delta_\alpha(\mathbf{x}) \in [LB, UB] \text{ for every } \alpha, \\ 0; & \text{otherwise,} \end{cases} \tag{8}$$

and $\mathcal{DT}\{\cdot\}$ is a density-based hard thresholding function used for refining the raw results. The density-based hard thresholding function is defined by

$$\mathcal{DT}\{w(\mathbf{x})\} = \begin{cases} 1; & \frac{\sum_{\boldsymbol{\tau}} w(\mathbf{x})\cdot C(\mathbf{x}-\boldsymbol{\tau})}{\sum_{\boldsymbol{\tau}} C(\boldsymbol{\tau})} \geq \eta, \\ 0; & \text{otherwise,} \end{cases} \tag{9}$$

$$\text{where } C(\mathbf{x}) = \begin{cases} 1; & |\mathbf{x}| \leq R, \\ 0; & |\mathbf{x}| > R, \end{cases} \tag{10}$$

$\eta \in (0, 1]$ is a pre-determined cut-off ratio, and R is a radius of the circular filter $C(\mathbf{x})$. $\mathcal{DT}$ operator can be used for forming solid shapes from groups of high-density scattered pixels, filtering out low-density scattered pixels, and smoothing objects' contour. The behavior of $\mathcal{DT}$ operator varies by the cut-off ratio, η. $\mathcal{DT}$ can work as an erosion operator when $\eta = 1$, a median filtering operator of the binary data when $\eta = 0.5$, and a dilation operator when $\eta \to 0$. While the radius, R, affects details of the output's shape. In this work, we choose $R = 5$ which can remove the scattered pixels while keeping the satisfying details for the output's shape.

For the post-processing in (7), in consideration of the spatial characteristics of the image, it is better to remove the pixels of interesting region whose near neighbors have already been ignored in the raw version, $\tilde{w}_{\mathrm{IR}}(\mathbf{x})$. For this purpose, the cut-off ratio of $\mathcal{DT}$ in (7) is recommended as $\eta \in [0.8\ 1.0]$.

STEP 3: Compute an expected blur direction function, $\Omega(\mathbf{x})$. In consideration of a ratio of the horizontal and vertical variations of blurred image, the expected blur direction can be obtained from

$$\Omega_{\mathrm{A}}(\mathbf{x}) = \arctan\left(\frac{\Delta_{0^\circ}(\mathbf{x})}{\Delta_{90^\circ}(\mathbf{x})}\right). \tag{11}$$

The definition in (11) shows that if the horizontal variation is much greater than the vertical variation, we expect that the blur direction is near the vertical direction because the blur reduces the variation in vertical direction. On the other hand, if the horizontal variation is much less than the vertical variation, we expect that the blur direction is near the horizontal direction. As a result, the blur direction can be estimated by observing the ratio of the horizontal and vertical variations. Since both Δ_{0° and Δ_{90° are non-negative numbers according to (6), the result in (11) is always located in the first quadrant. A proper quadrant of Ω is estimated by comparing the variation in the diagonal directions as

$$\Omega_{\mathrm{S}}(\mathbf{x}) = \begin{cases} 1; & \Delta_{45^\circ}(\mathbf{x}) \leq \Delta_{-45^\circ}(\mathbf{x}), \\ -1; & \Delta_{45^\circ}(\mathbf{x}) > \Delta_{-45^\circ}(\mathbf{x}). \end{cases} \tag{12}$$

If the variation along the direction of 45° is greater than −45°, we expect that the blur direction locates in the first quadrant; otherwise, it locates in the forth quadrant. Finally, the expected blur direction, $\Omega(\mathbf{x})$, can be obtained from

$$\Omega(\mathbf{x}) = \begin{cases} \mathcal{MF}\{\Omega_{\mathrm{A}}(\mathbf{x})\}\cdot\mathcal{MF}\{\Omega_{\mathrm{S}}(\mathbf{x})\}; & w_{\mathrm{IR}}(\mathbf{x})=1, \\ \text{not defined;} & \text{otherwise,} \end{cases} \tag{13}$$

where $\mathcal{MF}\{\cdot\}$ denotes a 2-D median filtering operator. The median filter of any 2-D data $s(\mathbf{x})$ is defined as

$$\mathcal{MF}\{s(\mathbf{x})\} = \operatorname*{median}_{\boldsymbol{\tau}\in\mathcal{K}(\mathbf{x})}(\{s(\boldsymbol{\tau})\}), \tag{14}$$

where $\mathcal{K}(\mathbf{x})$ is a set of local neighbors around the pixel $\mathbf{x}$. The reason why the median filters are used is that the raw outputs in (11) and (12) are degraded by impulsive and speckle noises and the objects' edges should be preserved to obtain a good segmentation result. However, the coupling of $\Omega_{\mathrm{S}}(\mathbf{x})$ and $\Omega_{\mathrm{A}}(\mathbf{x})$ yields cyclic values of direction with period of π that can not be sorted. Therefore, the median filter can not be applied to $\Omega_{\mathrm{S}} \cdot \Omega_{\mathrm{A}}$, directly. We have to apply the median filter separately to each element of $\Omega_{\mathrm{S}}(\mathbf{x})$ and $\Omega_{\mathrm{A}}(\mathbf{x})$ as shown in (13).

STEP 4: In consideration of $\Omega(\mathbf{x})$, the image is decomposed into several layers as

$$\tilde{w}_{\frac{k\pi}{K}}(\mathbf{x}) = \begin{cases} 1; & \begin{array}{l}\Omega(\mathbf{x}) + n\pi \in \left[\frac{k\pi}{K} - \delta, \frac{k\pi}{K} + \delta\right] \\ \text{for any integer } n,\end{array} \\ 0; & \text{otherwise,} \end{cases} \tag{15}$$

$$w_{\frac{k\pi}{K}}(\mathbf{x}) = \mathcal{DT}\left\{\tilde{w}_{\frac{k\pi}{K}}(\mathbf{x})\right\}, \tag{16}$$

where K is the number of decomposed layers; $\tilde{w}_{\frac{k\pi}{K}}(\mathbf{x})$ is a binary mask of raw window when the center of the range is $\frac{k\pi}{K}$ and the radius is $\delta \geq \frac{\pi}{2K}$; and $w_{\frac{k\pi}{K}}(\mathbf{x})$ is a binary mask after refinement by using the density based thresholding function in (9). The high density of pixels where $w_{\frac{k\pi}{K}}(\mathbf{x}) = 1$ can be expected in the blurred area. Most of low-density scattered pixels trend to belong to unblurred area and should be removed. For this purpose, the cut-off ratio of $\mathcal{DT}$ in (16) is recommended as $\eta \in [0.5\ 0.8]$. While an approach for choosing δ and K in (15) is given in Sect. 4.

STEP 5: Merge groups of layers $w_{\frac{k\pi}{K}}(\mathbf{x})$ that have large intersection parts. In consideration of every pair of layers, if the intersection part of each pair occupies more than 80% of area of the smaller one, the layers are merged together. The result is a new set of binary mask denoted by $\{\nu_j(\mathbf{x})\}$.

STEP 6: To separate the object parts in each window, a connected component labeling algorithm [15], [16] is applied to every $\nu_j(\mathbf{x})$ (please see Appendix A for more information about the connected component labeling algorithm). The p-th connected region in $\nu_j(\mathbf{x})$ is denoted by $\nu_j^{[p]}(\mathbf{x})$.

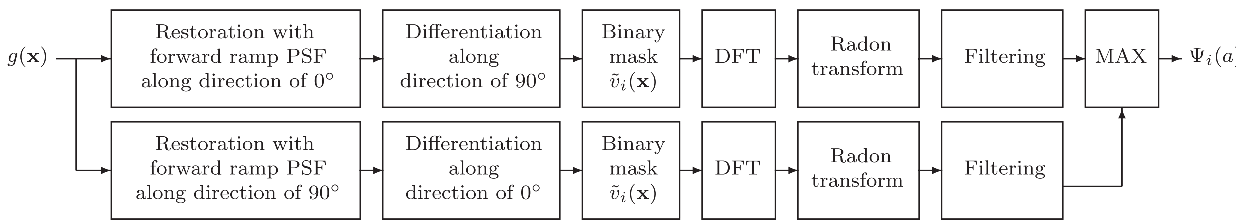


Fig. 3 Scheme for motion direction estimation.

STEP 7: Eliminate the objects $v_j^{[p]}(\mathbf{x})$, if their sizes are too small. For example, in this work, we consider only the blurred objects which occupy more than 2% of the entire image's area. As a result, the objects whose sizes are smaller than 2% of the image size can be ignored. There is a trade-off between fineness and computational cost. Reducing the threshold can improve the fineness and resolution of the detection scheme; however, the number of rough detected objects increase and higher computational cost will be required in the next steps. In addition, the performances of PSF identification and restoration schemes typically degrade due to an insufficient information when they are applied to the small objects. After small objects are eliminated, merge groups of objects $\{v_j^{[p]}(\mathbf{x})\}_{j,p}$ that have large intersection area (larger than 80% of the smaller object). The final result is denoted by $\{\tilde{v}_i(\mathbf{x})\}_{i=1}^{\hat{N}_O}$ which is the rough detected regions of blurred objects.

3.2 Identification of Motion Blur Parameter

Since the proposed scheme is based on assumption of linear motion blur with constant velocity, there are two parameters to be identified that are the motion blur direction and the length of motion.

3.2.1 Identification of Motion Direction

The estimation of motion direction is based on the algorithm used in [3]–[5] which is demonstrated in Fig. 3. In order to stimulate stripes along the blur direction, the algorithm starts from trial restoration of the blurred image by using a forward ramp PSF along the horizontal and vertical directions. The coefficients of forward ramp PSF are $[1\ 2\ 3\ \dots\ L_{\mathrm{tr}}]$ where L_{tr} is the trial length of motion which can be chosen according to [3]. Then, the stripes are emphasized by applying the vertical and horizontal differentiation to the horizontal and vertical trial restored images, respectively. For identification of the motion direction of the i-th blurred object, the image derivatives are multiplied with the mask $\tilde{v}_i$. Finally, two outputs are transformed by Fourier and Radon transforms to prepare for extraction of the directional information. For the linear motion blurs, the outputs after Fourier transform are characterized by single-stripe-like functions [3] corresponding to frequency characteristics of the PSFs. The blur direction can be obtained from a direction of the stripe which is extracted via the Radon transform (please see Appendix B for more information about the Radon transform). The outputs of the Radon transform are denoted by $\mathcal{D}_i(a, \rho; \phi)$ for the direction of trial restoration of $\phi \in \{0°, 90°\}$ where a and ρ represent the dimensions of an angle and a distance from the origin, respectively. The motion directions are corresponding to the dominant angles in $\mathcal{D}_i$. To locate the dominant angles, local maxima are emphasized by using filter as

$$\tilde{\mathcal{D}}_i(a;\phi) = \bar{\mathcal{D}}_i(a;\phi) - \frac{1}{2A}\int_{a-A}^{a+A} \bar{\mathcal{D}}_i(s;\phi)ds, \tag{17}$$

$$\bar{\mathcal{D}}_i(a;\phi) = \frac{1}{2P}\int_{-P}^{P} \mathcal{D}_i(a,\rho;\phi)d\rho, \tag{18}$$

where A and P are adjusting parameters for the filter, and $\tilde{\mathcal{D}}_i(a)$ is a peak emphasizing function. Since now we are trying to identify the direction of the narrow stripe located in a center part of frequency domain, the integration in (18) is used to focus on only the a-dimension and the center area of ρ-dimension. Then, the peaks of $\bar{\mathcal{D}}_i(a)$ are emphasized by the high-pass filter in (17). Finally, combining the two outputs together, we obtain the directional analysis function, $\Psi_i(a)$, as

$$\Psi_i(a) = \max(\tilde{\mathcal{D}}_i(a;0°), \tilde{\mathcal{D}}_i(a;90°)). \tag{19}$$

The processes in (17)–(19) are originally proposed for estimation of space-invariant non-linear motion blurs approximated in piecewise linear model. Since the blur directions can be estimated from the locations of local maxima in $\Psi_i(a)$ [4], $\Psi_i(a)$ extracted from the linear motion blurred object should be characterized by only one dominant peak. In this work, the scheme in (17)–(19) is applied to check whether the detected objects are degraded by the linear motion blur or not. If $\Psi_i(a)$ is characterized by a number of distinct peaks which can be caused by textures or nonlinear motion blurs, the corresponding detected object is considered as a detection fault. In consideration of the shape of the directional analysis function, the motion direction can be estimated from

$$\hat{\theta}_i = \begin{cases} \arg\max\limits_a \Psi_i(a); & \sum_k \Psi_i(\Lambda_k) \le \max\limits_a \Psi_i(a), \\ \text{not defined}; & \text{otherwise}, \end{cases} \tag{20}$$

where $\hat{\theta}_i$ represents the identified motion direction, and $\{\Lambda_k\}$ is a set of local maxima of $\Psi_i(a)$ whose values are greater than zero, excluding the global maximum. For the feasibility condition, we observe the number and the heights of the local peaks. If there are many peaks and a total height

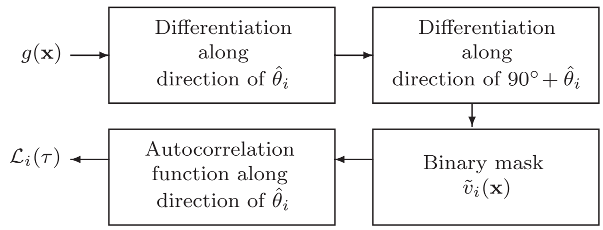


Fig. 4 Scheme for motion length estimation.

is over the global maximum value, the motion direction is assigned as 'not defined' which means that the corresponding detected region might not be valid for the assumption of linear motion blur with constant velocity or the analysis function is not clear enough for identification of the blur parameter.

3.2.2 Identification of Motion Length

The scheme for motion length estimation is modified from the method of Yitzhaky [1]. The motion length can be estimated from a location of global minimum of an autocorrelation function along the motion direction extracted from an image derivative as

$$\hat{L}_i = \begin{cases} \arg\min\limits_{\tau} \mathcal{L}_i(\tau); & \dfrac{\left|\mathcal{L}_{\text{2nd}} - \min\limits_{\tau} \mathcal{L}_i(\tau)\right|}{\left|\min\limits_{\tau} \mathcal{L}_i(\tau)\right|} \geq 0.5 \\ & \text{and } \arg\min\limits_{\tau} \mathcal{L}_i(\tau) \geq 5, \\ \text{not defined}; & \text{otherwise}, \end{cases} \tag{21}$$

where $\hat{L}_i$ is an identified motion length for the i-th detected object, $\mathcal{L}_i(\tau)$ is a motion length analysis function shown in Fig. 4, and $\mathcal{L}_{\text{2nd}}$ is a depth of the second lowest local minimum of $\mathcal{L}_i(\tau)$.

In this case, there are two feasibility conditions. For the first condition, we compare a difference between the global minimum and the second lowest local minimum. Since the characteristic of $\mathcal{L}_i(\tau)$ extracted from the linear motion blurred objects should have the small second lowest local minimum comparing to the global minimum, if the difference between those two values is small, the motion length is assigned as 'not defined.' For the second condition, we consider the characteristic of the identification scheme when it is tested on the normal images (unblurred images). In this case, the minimum location of the directional autocorrelation function is not affected by the PSF but affected by the characteristic of the image itself. In this work, we observed the locations of global minimum of the autocorrelation functions obtained from ten standard test images (e.g. *Lenna, Barbara, Boat, Peppers*, and etc.) along 180 directions. From this experiment, we found that the locations of minimum are mainly distributed in a range of [2 4] (20% are two, 65% are three, 14% are four, and 1% is more than four). Therefore, to avoid the detection fault, the condition of minimum identified motion length is applied as $\hat{L} \geq 5$.

Finally, after the motion direction and length are identified, the PSF for the i-th detected object can be estimated as $\hat{h}_i(\mathbf{x}) = h(\mathbf{x}; \hat{L}, \hat{\theta})$ according to (4).

3.3 Refinement of Blurred Region Segmentation

After the PSF of blur is identified for every detected region, the blurred region can be updated by using the information of the blur PSF as the following steps.

STEP 1: Create the ideal high-pass filter, $h'_i(\mathbf{x})$, which is roughly complementary to the identified blur PSF, $\hat{h}_i(\mathbf{x})$, as

$$H'_i(\boldsymbol{\omega}) = \begin{cases} 0; & \boldsymbol{\omega} \in \mathcal{S}, \\ 1; & \text{otherwise}, \end{cases} \tag{22}$$

where $H'_i(\boldsymbol{\omega})$ is a Fourier transform of $h'_i(\mathbf{x})$; and $\mathcal{S}$ represents an area of stop band. Then, the blurred image is filtered by $h'_i(\mathbf{x})$ as

$$g'_i(\mathbf{x}) = h'_i(\mathbf{x}) * g(\mathbf{x}). \tag{23}$$

$\mathcal{S}$ in (22) can be chosen to obtain the condition that $|h'_i(\mathbf{x}) * \hat{h}_i(\mathbf{x})| \approx 0$. For the general blur functions, $\mathcal{S}$ can be the area that $|\hat{H}_i(\omega)|$ is over some pre-determined threshold where $\hat{H}_i(\boldsymbol{\omega})$ is a Fourier transform of $\hat{h}_i(\mathbf{x})$. For the cases of linear motion blurs whose frequency responses are characterized by sinc functions, an area occupied by a main lobe of $\hat{H}_i(\omega)$ is appropriate to be used as $\mathcal{S}$ since the most power of $\hat{H}_i(\omega)$ dominate in the main lobe and an abrupt change in the frequency response can be avoided. We expect that the power of $g'_i(\mathbf{x})$ in the area of the blurred object v_i might be dramatically degraded since $|h'_i(\mathbf{x}) * \hat{h}_i(\mathbf{x})| \approx 0$ while the power in the other areas is kept by $|h'_i(\mathbf{x}) * f(\mathbf{x})| > 0$.

STEP 2: Based on the assumption that the power of the $g'_i(\mathbf{x})$ in the part of blur $v_i(\mathbf{x})$ is near zero, thresholding is applied as

$$\tilde{u}_i(\mathbf{x}) = \begin{cases} 1; & \left|g'_i(\mathbf{x})\right| < \varepsilon, \\ 0; & \text{otherwise}, \end{cases} \tag{24}$$

where ε is a threshold. An approach for choosing the threshold is given in Sect. 4. Then, the results from this stage and the rough estimation stage are combined as

$$u_i(\mathbf{x}) = \mathcal{DT}\left\{[\tilde{v}_i(\mathbf{x}) | \tilde{u}_i(\mathbf{x})] \cdot w_{\text{IR}}(\mathbf{x}) \prod_{j, j \neq i} [1 - \tilde{v}_j(\mathbf{x})]\right\}, \tag{25}$$

where | is a logical OR operator. Similar to (16), high density of pixels where $\tilde{u}_i(\mathbf{x}) = 1$ in the blurred area can be expected. In addition, the solid shape is already obtained in the rough estimation process. Most of the low-density scattered pixels trend to be errors and should be removed. The same cut-off ratio of $\mathcal{DT}$ in (16) can be applied here.

STEP 3: Apply connected component labeling algorithm [16] to decompose $u_i(\mathbf{x})$ to $u_i^{[q]}(\mathbf{x})$ and select the component with the largest intersection part to the rough estimated results, $\tilde{v}_i(\mathbf{x})$, as

$$Q = \arg\max_q \sum_{\mathbf{x}} u_i^{[q]}(\mathbf{x}) \cdot \tilde{v}_i(\mathbf{x}). \tag{26}$$

STEP 4: Apply morphology techniques [16] to smooth the contour of $u_i^{[Q]}(\mathbf{x})$, resulting in the final identified location of blur, $\hat{v}_i(\mathbf{x})$.

3.4 Restoration

To describe the restoration scheme, (2) and (3) are re-written in the matrix-vector form as

$$\mathbf{g} = \sum_{i=1}^{N_O} (\mathbf{H}_i \mathbf{V}_i \mathbf{o}_i) + \left(\mathbf{I} - \mathrm{diag}\left(\sum_{i=1}^{N_O} \mathbf{H}_i \mathbf{v}_i\right)\right)\mathbf{b}, \tag{27}$$

$$\mathbf{f} = \sum_{i=1}^{N_O} (\mathbf{V}_i \mathbf{o}_i) + \left(\mathbf{I} - \sum_{i=1}^{N_O} \mathbf{V}_i\right)\mathbf{b}, \tag{28}$$

where $\mathbf{f}$, $\mathbf{g}$, $\mathbf{b}$ and $\mathbf{o}_i$ are the vector representation of the intensities of the original image $f(\mathbf{x})$, the blurred image $g(\mathbf{x})$, the part of still background $\mathcal{B}(\mathbf{x})$, and the moving objects $O_i(\mathbf{x})$, respectively; $\mathbf{H}_i$ denotes a matrix which is identical to convolution with the PSF $h_i(\mathbf{x})$; $\mathbf{v}_i$ is a vector representation for the binary mask $v_i(\mathbf{x})$; $\mathbf{V}_i = \mathrm{diag}(\mathbf{v}_i)$ denotes a diagonal matrix which is identical to windowing operator with the binary mask $v_i(\mathbf{x})$; $\mathrm{diag}(\cdot)$ for any vector represents a diagonal matrix whose diagonal values starting in the upper left corner are respectively obtained from the elements of the vector; and $\mathbf{I}$ is an identity matrix. (27) and (28) can be simplified as

$$\mathbf{g} = \mathbf{\Phi}\varrho, \tag{29}$$

$$\mathbf{f} = \Xi\varrho, \tag{30}$$

where

$$\varrho = \begin{bmatrix} \mathbf{b}^T & \mathbf{o}_1^T & \mathbf{o}_2^T & \cdots & \mathbf{o}_{N_O}^T \end{bmatrix}^T, \tag{31}$$

$$\mathbf{\Phi} = \left[\mathbf{I} - \mathrm{diag}\left(\sum_{i=1}^{N_O} \mathbf{H}_i \mathbf{v}_i\right) \ \mathbf{H}_1\mathbf{V}_1 \cdots \mathbf{H}_{N_O}\mathbf{V}_{N_O}\right], \tag{32}$$

$$\Xi = \begin{bmatrix} \mathbf{I} - \sum_{i=1}^{N_O} \mathbf{V}_i & \mathbf{V}_1 & \mathbf{V}_2 & \cdots & \mathbf{V}_{N_O} \end{bmatrix}, \tag{33}$$

In order to solve linear equations in (29) and (30) to reconstruct $\mathbf{f}$, we use an iterative scheme based on Landweber method [6], [7] and regularization techniques [7]–[9] as the following steps.

Initialization: The proposed iterative restoration scheme starts form initialization of the objects and background. The blurred objects are initialized by

$$\mathbf{o}_i^{(0)} = \mathbf{V}_i \mathbf{g}, \tag{34}$$

where $\mathbf{o}_i^{(0)}$ denotes an initial guess for the i-th object. For the background, we can consider into three regions as

$$\mathcal{B}^{(0)}(\mathbf{x}) = \begin{cases} g(\mathbf{x}); & \sum_{i=1}^{N_O}(h_i * v_i)(\mathbf{x}) = 0, \\ 0; & \sum_{i=1}^{N_O}(h_i * v_i)(\mathbf{x}) = 1, \\ \mathcal{P}(\mathbf{x}); & \text{otherwise}, \end{cases} \tag{35}$$

where $\mathcal{B}^{(0)}(\mathbf{x})$ denotes a spatial domain representation of an initial guess for the background. In the region where $\sum_{i=1}^{N_O}(h_i * v_i)(\mathbf{x}) = 0$ which results in $v_i(\mathbf{x}) = 0$ for every i, the recorded image, $g(\mathbf{x})$, is obtained from pure background according to (2). For the region where $\sum_{i=1}^{N_O}(h_i * v_i)(\mathbf{x}) = 1$, based on an assumption that the blurred objects do not overlap, $g(\mathbf{x})$ is obtained from pure objects. In this area, the information of background is not used to reconstruct the desired image. The last area is corresponding to a transition area where the recorded image, $g(\mathbf{x})$, is obtained from an addition of background and objects with different ratio depending upon the location. The background value in the transition area can be predicted based on the information from the estimated objects, $\mathcal{P}_{\mathrm{obj}}(\mathbf{x})$, and based on the correlation to the known background, $\mathcal{P}_{\mathrm{bg}}(\mathbf{x})$. According to (2), if the objects are known, the background can be obtained from

$$\mathcal{P}_{\mathrm{obj}}(\mathbf{x}) = \frac{g(\mathbf{x}) - \sum_{i=1}^{N_O} h_i(\mathbf{x}) * O_i^{(0)}(\mathbf{x})}{1 - \sum_{i=1}^{N_O} h_i(\mathbf{x}) * v_i(\mathbf{x})}. \tag{36}$$

In addition, in consideration of the correlation between the unknown background and their known neighborhood values, the background can be estimated by using interpolation/extrapolation or inpainting techniques [17]–[19]. The predicted result is denoted by $\mathcal{P}_{\mathrm{bg}}(\mathbf{x})$. Finally, the predicted background value in the transition area can be obtained by combining the results from $\mathcal{P}_{\mathrm{obj}}(\mathbf{x})$ and $\mathcal{P}_{\mathrm{bg}}(\mathbf{x})$ as

$$\begin{aligned} \mathcal{P}(\mathbf{x}) = & \left[1 - \textstyle\sum_{i=1}^{N_O} h_i(\mathbf{x}) * v_i(\mathbf{x})\right] \cdot \mathcal{P}_{\mathrm{bg}}(\mathbf{x}) \\ & + \left[\textstyle\sum_{i=1}^{N_O} h_i(\mathbf{x}) * v_i(\mathbf{x})\right] \cdot \mathcal{P}_{\mathrm{obj}}(\mathbf{x}). \end{aligned} \tag{37}$$

From (37), the unknown background can be estimated from parts of $\mathcal{P}_{\mathrm{obj}}(\mathbf{x})$ with the weight of $\sum_{i=1}^{N_O} h_i(\mathbf{x}) * v_i(\mathbf{x})$ and $\mathcal{P}_{\mathrm{bg}}(\mathbf{x})$ with the weight of $1 - \sum_{i=1}^{N_O} h_i(\mathbf{x}) * v_i(\mathbf{x})$. The higher weight values are given to $\mathcal{P}_{\mathrm{bg}}(\mathbf{x})$ for prediction of the unknown pixels which locate near the known background. On the other hands, the higher weight values are given to $\mathcal{P}_{\mathrm{obj}}(\mathbf{x})$ for prediction of the unknown pixels which locate near the objects.

After the background and objects are initialized, the initial values of $\hat{\varrho}$ and $\hat{\mathbf{f}}$ can be obtained from

$$\hat{\varrho}^{(0)} = \begin{bmatrix} \mathbf{b}^{(0)T} & \mathbf{o}_1^{(0)T} & \mathbf{o}_2^{(0)T} & \cdots & \mathbf{o}_{N_O}^{(0)T} \end{bmatrix}^T, \tag{38}$$

$$\hat{\mathbf{f}}^{(0)} = \Xi\hat{\varrho}^{(0)}. \tag{39}$$

Updating: Based on the regularized iterative scheme, the restored image can be updated by

$$\begin{aligned} \hat{\varrho}^{(k)} = & \hat{\varrho}^{(k-1)} + \beta\mathbf{\Phi}^T\left(\mathbf{g} - \mathbf{\Phi}\hat{\varrho}^{(k-1)}\right) \\ & - \beta\gamma\mathbf{V}^{\sharp}\mathbf{C}^T\mathbf{A}\mathbf{C}\mathbf{f}^{(k-1)}, \end{aligned} \tag{40}$$

$$\hat{\mathbf{f}}^{(k)} = \Xi\hat{\varrho}^{(k)}, \tag{41}$$

where the superscript (k) denotes an iteration number; β is a step size; $\mathbf{C}$ is a matrix corresponding to a regularization operator which is typically a high-pass filtering operator; γ is a regularization parameter; $\mathbf{A}$ is a diagonal matrix corresponding to a spatial adaptive component for the regularization term;

$$\mathbf{V}^{\sharp} = \left\lceil \mathrm{diag}\left(\sum_{i=1}^{N_O} \mathbf{H}_i \mathbf{v}_i \right) \right\rceil; \tag{42}$$

and $\lceil \cdot \rceil$ is a ceiling operator for every element of matrix. The iterative scheme in (40) consists of two updating terms. The first part is corresponding to a discrepancy error, $\mathbf{g} - \mathbf{\Phi}\hat{\varrho}^{(k-1)}$, feeding back to the results with the updating step size of β. The second part is called the regularization term, $\gamma \mathbf{V}^{\sharp} \mathbf{C}^T \mathbf{A} \mathbf{C} \mathbf{f}^{(k-1)}$, which is applied to prevent an over-amplification of the high frequency components due to the ill-posed condition. The effect of the regularization term is reduction of noise amplification and ringing artifacts. The regularization operator, $\mathbf{C}$, regularization parameter, γ, and adaptive component, $\mathbf{A}$, can be chosen according to [7]–[9].

4. Experiments

In this section, the performance of our proposed method is demonstrated by using simulation and real world examples in Example I and II, respectively. We applied the schemes in Sects. 3.1–3.3 for identifying the locations and PSFs of blurred objects. The restored images, $\hat{f}(\mathbf{x})$, were obtained from the scheme presented in Sect. 3.4 by using the identified blur locations, $\hat{v}_i(\mathbf{x})$, and estimated PSFs, $\hat{h}_i(\mathbf{x})$. The parameters in each stage were assigned as shown in Table 2. Most of parameters were tuned based on the characteristics observed from pre-experiments on the set of test images (used in Sect. 3.2.2).

Firstly, the three derivative operators in Table 1 were tested on the images degraded by 200 random PSFs under five noise conditions. The expected blur direction function, $\Omega(\mathbf{x})$, and its estimation error, $|\Omega(\mathbf{x}) - \theta|$, corresponding to each derivative operator were observed. Figure 5 shows root-mean-square errors in different noise condition. The results show that the second-derivative operator which is the high-pass operator can provide the minimum error in low-noise conditions. However, in noisy conditions, the performance of the second-derivative operator is degraded and the Sobel and Prewitt operators which are the band-pass operators provide the similar performances with smaller estimation errors. Since further experiments were based on the assumption of low-noise condition, the second derivative operator was selected for computing the variation functions.

Table 2 Set of parameters using in the experiments.

Rough Estimation :	
$d_\alpha(\mathbf{x})$:	The second-derivative filter
$[LB\ UB]$:	[1 55]
$\mathcal{MF}$:	5×5
K :	18 layers
δ :	13°
Merging Threshold :	80%
Elimination Threshold :	2%
Motion Direction Estimation :	
L_{tr} :	10
A :	140°
P :	0.04π
Motion Length Estimation :	
Minimum Length :	5
Refinement Stage :	
ε :	Fig. 7
Density-Based Hard Thresholding :	
R :	5
η :	1.0 for (7)
	0.6 for (16) and (25)
Restoration Stage:	
β :	0.5
γ :	0.01
$\mathbf{C}$:	Laplacian
$\mathbf{A}$:	None

Next, to set up the bounds for interesting region in (8), the very low frequency region was considered as a homogeneous region where there is completely no variation or $\Delta_\alpha(\mathbf{x}) = 0$ because $\Omega(\mathbf{x})$ in this region can not be obtained due to the division in (13). In addition, the restoration process can be avoided because there is almost no difference between the blurred and unblurred pixels [5]. As a result, the lower bound for the interesting region in (8) can be set as $LB = 1$. On the other hand, the upper bound, UB, can be selected based on histograms of the variation functions. The test images were blurred by motion blur along 180 directions with the length of five (the minimum length for the identification scheme). The histograms of the variation functions calculated from both blurred and unblurred inputs were observed, resulting in Fig. 6. The results show that the variation functions of the blurred inputs have less distribution comparing to the unblurred inputs. Based on the results in this example, the variation functions of the blurred inputs were bounded by 360; therefore, an ideal upper bound for the interesting region can be set as $UB = 360$, then we can expect 100% of the blurred pixels including in the interesting region. However, in this case, only 0.35% of the unblurred pixels can be ignored. In practice, we should increase the number of ignored pixels by some allowance of small error. In this work, we allowed 1% of the blurred pixels that may be missing by assigning the upper bound as $UB = 55$ so that we can expect 7% of the unblurred pixels which be ignored.

In order to find the expected blur direction, $\Omega(\mathbf{x})$, in (13), the median filters are used to clean the function that can improve the segmentation results. However, the median filters smooth $\Omega(\mathbf{x})$ not only in the blurred region but also the unblurred region. A large size of neighborhood region, $\mathcal{K}(\mathbf{x})$, may induce more erroneously detected objects. Therefore, in this experiment, we chose $\mathcal{K}(\mathbf{x})$ as a square of size 5×5 which can reduce the variation of $\Omega(\mathbf{x})$ in the blurred region while keeping the variation of $\Omega(\mathbf{x})$ in the unblurred region.

After $\Omega(\mathbf{x})$ have been obtained, the image is separated into layers based on the range of $\Omega(\mathbf{x})$. For this step, we started by setting the cut-off ratio of $\mathcal{DT}$ in (16) as 0.6. It means that the blurred objects can be recovered if 60% of their pixels are provided. Based on this condition, to select the segmentation range, δ, in (15), a histogram of the estimation errors of the expected blur direction, $|\Omega(\mathbf{x}) - \theta|$, was observed from the test images. From the histogram, we found that 60% of $\Omega(\mathbf{x})$ have the estimation errors less than

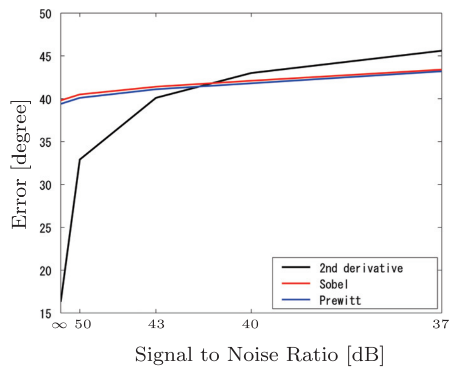


Fig. 5 Error characteristics of the expected blur direction obtained from three types of directional filters.

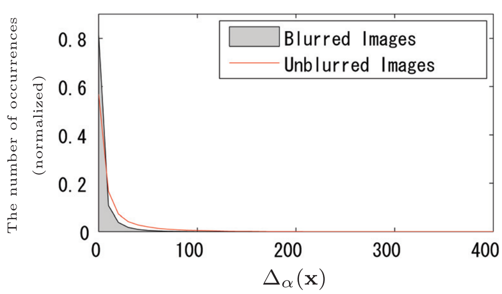


Fig. 6 Histograms of the variation functions.

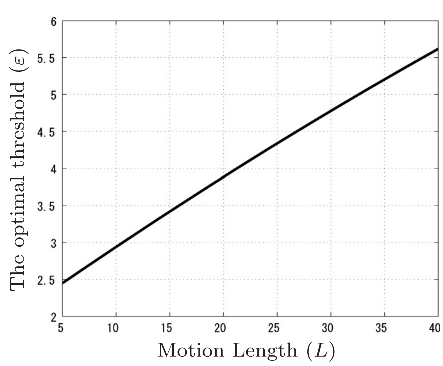


Fig. 7 The optimal threshold for $|g_i'(\mathbf{x})|$ in (24).

13°. As a results, the appropriate values for δ is 13°. While the number of decomposed layers, K, that should be greater than $180°/2\delta$ can be chosen arbitrary. However, the large number of decomposed layers will causes more redundancy because the similar layers will merge together in the next step. In this experiment, we choose $K = 18$.

For the PSF identification stage, the parameter P in (18) can be selected via observation of identification errors of the motion directions, $|\hat{\theta} - \theta|$, obtained from the test images. Based on our experiment, the optimal P which provided the minimum error was $P = 0.04\pi$. While the parameter A in (17) which directly affects the feasibility condition, should minimize $\sum_k \Psi_i(\Lambda_k)$ in (20) for the linear motion blurred objects while keeping the condition $\sum_k \Psi_i(\Lambda_k) > \max_a \Psi_i(a)$ for the unblurred or non-linear motion blurred objects. Based on the experiment, the optimal A was obtained as 140°.

For the refinement stage, the optimal threshold ε in (24) was analyzed by using the test images and the characteristic of the optimal threshold is illustrated in Fig. 7. The optimal threshold increases when the length of motion increases since the power allocated in the main lobe of $H_i(\boldsymbol{\omega})$ is reduced and the stop band of $H_i'(\boldsymbol{\omega})$ is narrowed.

Example I: The first example is the case of an image with one simulated motion blurred object in still outdoor background. Figures 8(a) and 8(b) show the desired image without blur and its degraded version, respectively. The object in Fig. 8(b) was blurred by linear motion blur along the direction of $\theta = 80°$ with the motion length of $L = 29$ and constant velocity. The proposed scheme started from the rough detection of blurred objects by using the expected blur direction function, $\Omega(\mathbf{x})$, which was obtained as in Fig. 10. The expected blur direction function in Fig. 10 has approximately homogeneous values around 80° in the area of the blurred object. By observing $\Omega(\mathbf{x})$, the proposed scheme detected roughly two blurred objects as illustrated in Fig. 9(a). The rough detection result still includes one erroneously detected object caused by an edge in the floor. However, this detection fault can be eliminated in the next parameter identification step.

The blur parameter identification process of two detected objects is demonstrated in Fig. 11. For the first object (floor), the characteristics in that region were not uniform enough to identify the length of motion blur according to

(a) Desired Image $f(\mathbf{x})$ (b) Blurred Image $g(\mathbf{x})$ (c) Restored Image $\hat{f}(\mathbf{x})$

Fig. 8 Example I: Simulation.

the first feasibility condition in (21). For the second object (can), the clear characteristics can be observed and the direction of blur can be estimated as $\hat{\theta}_2 = 80°$. Then, the motion length can be estimated as $\hat{L}_2 = 29$ and the identified PSF was obtained as shown in Fig. 11. While the conventional scheme in [14] failed in estimation of the motion direction. The motion direction was estimated as 22° which is very different from the actual value since the scheme used the entire image to estimate the motion direction of blurred object that makes the characteristics of the background interfere the identified result.

After the PSF had been identified, the location of blur was completed in the refinement stage. The better segmentation result was obtained as in Fig. 9(b). The detection fault was ignored by the PSF estimation process and the contour of the real blurred object was improved in the refinement stage. By using the identified location and PSF of blur, the image can be restored, resulting in Fig. 8(c). The restored result improves $PSNR$ from the blurred image by 2.5 dB and provides readable characters and more details in the objects.

Example II: In this example, the proposed scheme was tested by using a real world blurred image demonstrated in Fig. 12(a). The rough and refined detections for the blurred

(a) Rough Detection $\tilde{v}_i(\mathbf{x})$ (b) Refined Detection $\hat{v}_i(\mathbf{x})$

Fig. 9 Blurred region segmentation for Fig. 8(b).

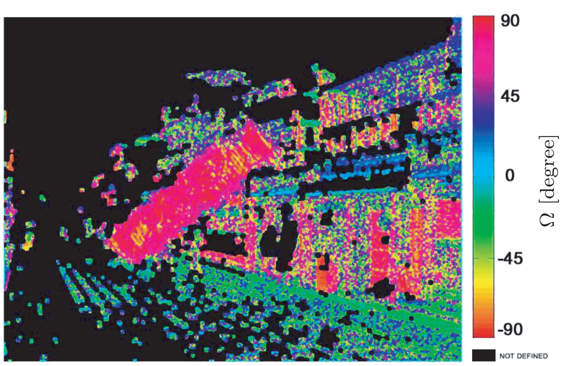
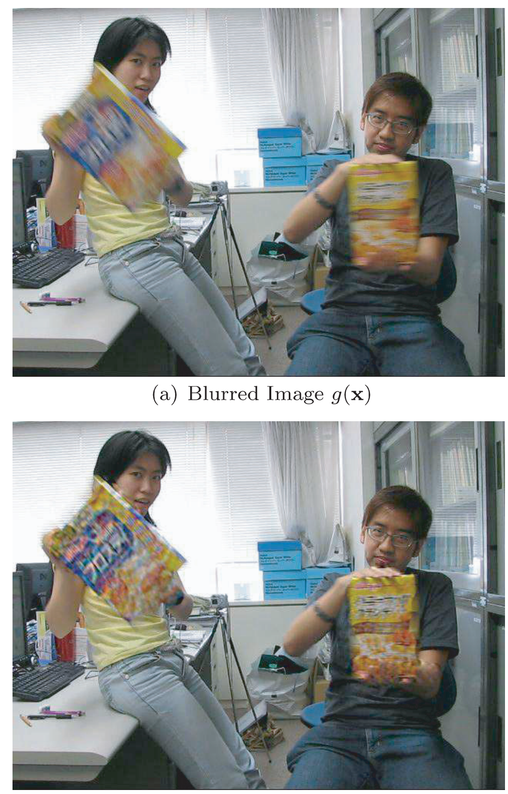


Fig. 10 Expected blur direction $\Omega(\mathbf{x})$ for Fig. 9(a).



(a) Blurred Image $g(\mathbf{x})$

(b) Restored Image $\hat{f}(\mathbf{x})$

Fig. 12 Example II: Real world image.

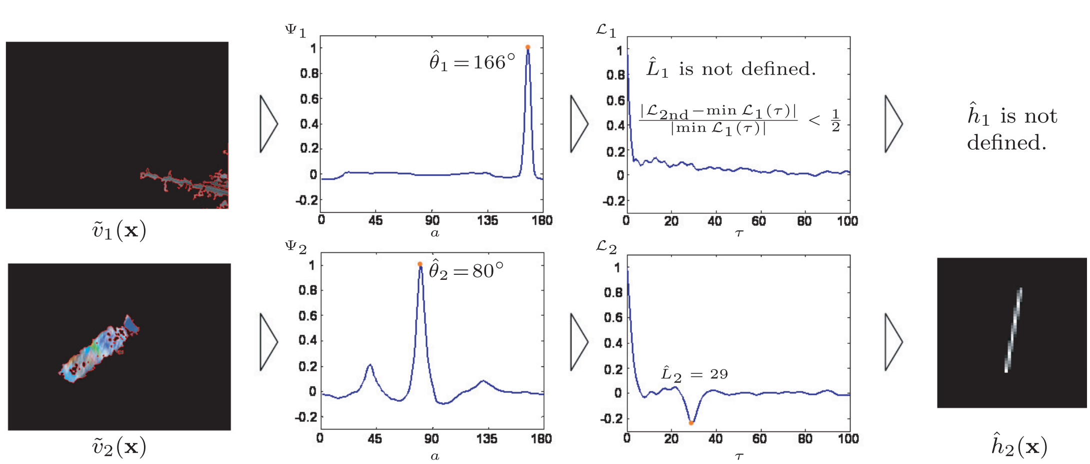


Fig. 11 Blur parameter identification of Fig. 8(b). From left to right, rough detected blurred objects, motion direction estimation, motion length estimation, and estimated PSFs.

objects are shown in Figs. 13(a) and 13(b), respectively. The rough estimates of blur locations were obtained from the expected blur direction, $\Omega(\mathbf{x})$, in Fig. 14. In the rough detection stage, we detected four blurred objects, including two actual blurred objects and two detection faults caused by a texture with horizontal parallel stripes of a curtain. The identification of PSFs is illustrated in Fig. 15. The PSFs failed to be identified in the motion length estimation steps for the two detection faults. As in Example I, the characteristics of the image in the erroneously detected regions were clear to identify the blur directions due to the parallel stripes appearing in the textures. However, the characteristics of the image derivatives were not uniform enough to identify the motion length based on the first condition in (21). While the characteristics of the actual blurred objects were homogeneous to identify the motion directions and lengths. As a result, the two detection faults were ignored and the locations of two real objects were improved in the refinement step. Finally, the restored result can be obtained as illustrated in Fig. 12(b) which provides more readable characters in the blurred objects. While the conventional scheme in [14] still failed for estimation of motion directions in this example. Since the motion directions were estimated by using entire image, only one dominant direction in the image can be de-

(a) Rough Detection $\tilde{v}_i(\mathbf{x})$ (b) Refined Detection $\hat{v}_i(\mathbf{x})$

Fig. 13 Blurred region segmentation of Fig. 12.

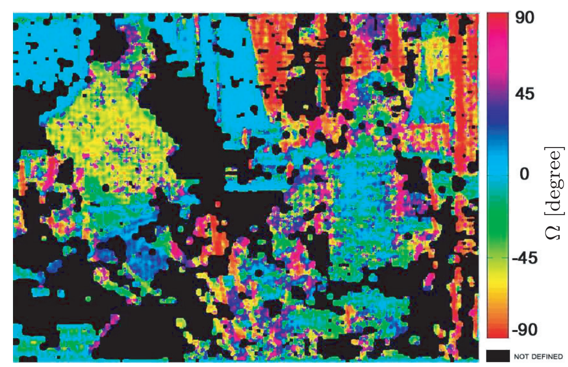


Fig. 14 Expected blur direction $\Omega(\mathbf{x})$ for Fig. 13(a).

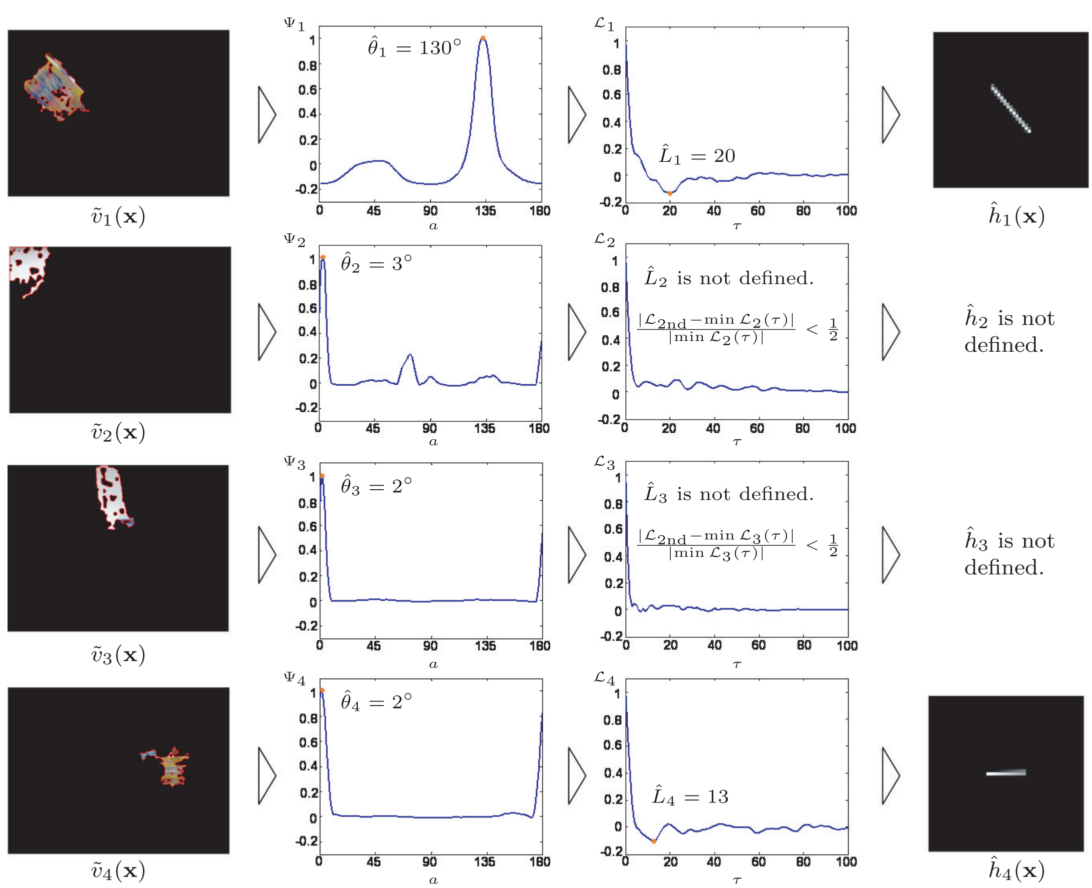


Fig. 15 Blur parameter identification of Fig. 12(a). From left to right, rough detected blurred objects, motion direction estimation, motion length estimation, and estimated PSFs.

tected while there are two blurred objects moving in different directions in this example. The estimated direction was obtained as 2° which is dominated by the horizontal parallel stripes in curtain and clothes.

5. Conclusion

A single-image-based scheme for identification of the locations and PSFs of blur caused by multiple moving objects along with iterative restoration scheme are proposed. To simplify the problem, the proposed scheme is based on the linear and constant-velocity motion blur model. Our proposed method consists of four stages, including rough detection of blur locations, identification of blur parameters, refinement of estimated blur locations and reconstruction of the desired image, respectively. The experimental examples were given in the both cases of simulated and real world blurred images to demonstrate that the proposed scheme provides the ability to automatically identify the PSFs and locations of blurred objects and recover the desired scene. Although the rough detected blurred regions may include some errors, they can be corrected by the blur parameter estimation or blur location refinement stages. However, the proposed scheme is still limited for the linear motion blur, and for the future work, the scheme for the non-linear motion blur should be considered.

References

[1] Y. Yitzhaky and N.S. Kopeika, "Identification of blur parameters from motion blurred images," CVGIP: Graphical Models and Image Processing, vol.59, no.5, pp.321–332, Sept. 1997.
[2] Y. Chen and I. Choa, "An approach to estimating the motion parameters for a linear motion blurred image," IEICE Trans. Inf. & Syst., vol.E83-D, no.7, pp.1601–1603, July 2000.
[3] W. Tan, J. Zhang, G. Rong, and H. Chen, "Identification of motion blur direction based on analysis of intentional restoration errors," 2004 IEEE Inter. Conf. on Networking, Sensing and Control, vol.2, pp.1253–1258, 2004.
[4] K. Patanukhom and A. Nishihara, "Identification of piecewise linear uniform motion blur," IEICE Trans. Fundamentals, vol.E91-A, no.6, pp.1416–1425, June 2008.
[5] K. Patanukhom and A. Nishihara, "Multiple-image-based restoration for motion blur with non-uniform point spread function," IEICE Trans. Fundamentals, vol.E91-A, no.8, pp.1924–1934, Aug. 2008.
[6] L. Liang and Y. Xu, "Adaptive landweber method to deblur images," IEEE Trans. Signal Process., vol.10, no.5, pp.129–132, May 2003.
[7] J. Biemond, R.L. Lagendjik, and R.M. Mersereau, "Iterative methods for image deblurring," Proc. IEEE, vol.78, pp.856–883, May 1990.
[8] A.K. Katsaggelos, J. Biemond, R.W. Schafer, and R.M. Mersereau, "A regularization iterative image restoration algorithm," IEEE Trans. Signal Process., vol.39, no.4, pp.914–929, April 1991.
[9] J. Lu, E. Poon, and K.N. Plataniotis, "Restoration of motion blurred images," IEEE International Conference on Multimedia and Expo, 2006, pp.1193–1196, July 2006.
[10] S. Cho, Y. Matsushita, and S. Lee, "Removing non-uniform motion blur from images," IEEE 11th Inter. Conf. on Computer Vision, ICCV 2007, Oct. 2007.
[11] M. Šorel and J. Flusser, "Space-variant restoration of images degraded by camera motion blur," IEEE Trans. Image Process., vol.17, no.2, pp.105–116, Feb. 2008.
[12] Q. Shan, W. Xiong, and J. Jia, "Rotational motion deblurring of a rigid object from a single image," IEEE 11th Inter. Conf. on Computer Vision, ICCV 2007, Oct. 2007.
[13] J. Jia, "Single image motion deblurring using transparency," IEEE Conf. on Computer Vision and Pattern Recognition, CVPR'07, June 2007.
[14] A. Levin, "Blind motion deblurring using image statistics," Advances in Neural Information Processing Systems (NIPS), Dec. 2006.
[15] L.D. Stefano and A. Bulgarelli, "A simple and efficient connected components labeling algorithm," Inter. Conf. on Image Analysis and Processing, pp.322–327, Sept. 1999.
[16] R.C. Gonzalez and R.E. Woods, Digital Image Processing, Third ed., Peason Prentice Hall, New Jersey, 2008.
[17] A. Rares, M.J.T. Reinders, and J. Biemond, "Edge-based image restoration," IEEE Trans. Image Process., vol.14, no.10, pp.1454–1468, Oct. 2005.
[18] B. Li, Y. Qi, and X. Shen "An image inpainting method," Ninth Inter. Conf. on Computer Aided Design and Computer Graphics, 2005, Dec. 2005.
[19] Z. Tauber, Z. Li, and M.S. Drew, "Review and preview: Disocclusion by inpainting for image-based rendering," IEEE Trans. Syst. Man Cybern. C, Appl. Rev., vol.37, no.4, pp.527–540, July 2007.

Appendix A: Connected Component Labeling Algorithm

For any image or 2-D data, connected component labeling algorithm is a process to devide the image into components based on a pixel connectivity. Any pair of pixels are connected, if they are neighbors and their values are from the same set of values. The neighbors of pixel can be considered as 4-neighbors (horizontal and vertical neighbors) or 8-neighbors (horizontal, vertical and diagonal neighbors).

Based on a classical approach, the algorithm starts by scanning the data (e.g. from left to right and up to down). Indicating $\mathbf{x}$ as a pixel to be labeled and p as a current label. The algorithm observes the neighbors of $\mathbf{x}$ which have already been encountered in the scan; if any neighbors are connected to $\mathbf{x}$, the smallest label of the connected neighbors is assigned to $\mathbf{x}$; otherwise, a new label ($p \Leftarrow p + 1$) is assigned to $\mathbf{x}$. In the cases that there is more than one neighbor connected to $\mathbf{x}$, all labels of the connected neighbors are registered as equivalences. There are a number of approaches for recording and resolving label equivalences. For example, after completing the first scan, the second scan is made to replace each group of equivalent labels with its corresponding new unique label.

In our application, since the binary masks $w(\mathbf{x})$ are the targets to be labeled, the labels are assigned only in the pixels where $w(\mathbf{x}) = 1$ while the pixels where $w(\mathbf{x}) = 0$ are ignored. The connectivity condition is based on 8-neighbors that have the same value.

Appendix B: Radon Transform

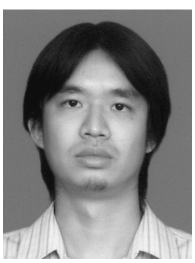
Radon transform is a famous method for detecting parameterized lines, edges, curves and shapes. The Radon transform is integral projections obtained from integration of a function over straight lines. A formulation of the Radon transform can be written as

$$\mathcal{R}_s(a,\rho)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}s(x,y)\cdot\delta(\rho-x\cos a-y\sin a)dxdy, \quad (\mathrm{A}\cdot 1)$$

or

$$\mathcal{R}_s(a,\rho)=\int_{-\infty}^{\infty}s(\rho\cos a-l\sin a,\rho\sin a+l\sin a)dl, \quad (\mathrm{A}\cdot 2)$$

where $\mathcal{R}_s(a,\rho)$ is the Radon transform of any function $s(x,y)$; a and ρ represent direction and distance from the origin of each line integration, respectively.



Karn Patanukhom was born in 1981. He received the B.S. and M.S. degrees in Electrical Engineering from Chiang Mai University in 2003 and Tokyo Institute of Technology in 2006, respectively. Since 2006 he has been working toward the Ph.D. degree at the Tokyo Institute of Technology. His research interest is in image processing, especially in image restoration problem.

Akinori Nishihara received the B.E., M.E. and Dr. Eng. degrees in electronics from Tokyo Institute of Technology in 1973, 1975 and 1978, respectively. Since 1978 he has been with Tokyo Institute of Technology, where he is now Professor of the Center for Research and Development of Educational Technology. His main research interests are in one- and multi-dimensional signal processing, and its application to educational technology. He has published more than 200 technical papers in international journals and conferences. He served as an Associate Editor of the IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences from 1990 to 1994, and then an Associate Editor of the Transactions of IEICE Part A (in Japanese) from 1994 to 1998. He was an Associate Editor of the IEEE Transactions on Circuits and Systems II from 1996 to 1997 and Editor-in-Chief of Transactions of IEICE Part A (in Japanese) from 1998 to 2000. He served in IEEE Region 10 Executive Committee in various positions, and the Executive Committee of IEEE Tokyo Section and IEEE Japan Council for the last 10 years. He was a member of the Board of Governors, IEEE Circuits and Systems Society (2004–2005), and was Chair of IEEE Circuits and Systems Society Japan Chapter (2006–2007). He was Chair of the IEICE Technical Group on Circuits and Systems from 1997 to 1998, and since 1998 he has been serving as an Adviser of that Technical Group. He is now serving as IEICE Director, Conferences, Student Activities, and Education. He received a Best Paper Award of the IEICE in 1999, and IEEE Third Millennium Medal in 2000. He also received a Distinguished Service Award for IEEE Student Activities in 2006. Prof. Nishihara is a Fellow of IEEE, and a member of EURASIP, European Circuits Society, Association for the Advancement of Computing in Education, and Japan Society for Educational Technology.