

# Removing Non-Uniform Motion Blur from Images\*

Sunghyun Cho  
POSTECH  
Pohang, Korea

sodomau@postech.ac.kr

Yasuyuki Matsushita  
Microsoft Research Asia  
Beijing, China

yasumat@microsoft.com

Seungyong Lee  
POSTECH  
Pohang, Korea

leesy@postech.ac.kr

## Abstract

We propose a method for removing non-uniform motion blur from multiple blurry images. Traditional methods focus on estimating a single motion blur kernel for the entire image. In contrast, we aim to restore images blurred by unknown, spatially varying motion blur kernels caused by different relative motions between the camera and the scene. Our algorithm simultaneously estimates multiple motions, motion blur kernels, and the associated image segments. We formulate the problem as a regularized energy function and solve it using an alternating optimization technique. Real-world experiments demonstrate the effectiveness of the proposed method.

## 1. Introduction

Motion blur caused by a relative motion between a camera and a scene is inevitable due to the nature of a camera sensor that accumulates incoming light over a certain period of time. Many computer vision algorithms rely on the assumption that a scene is captured without such motion blur. However, this assumption generally does not hold unless the scene and the camera are both static. It is therefore important to correctly remove motion blur from images so that the subsequent algorithms can neglect the effect of motion blur.

*Motion deblurring* has been studied by many researchers. Most methods solve the problem under an assumption that there is only a single motion blur kernel for the entire image. However, in real-world cases, photographed images often have spatially-varying motion blurs due to multiple relative motions caused by moving objects or depth variations from the camera (Fig. 1).

This paper proposes an algorithm for removing spatially-varying motion blurs from images. The removal of spatially-varying motion blurs involves three different problems: estimation of motions, segmentation into regions of homogeneous motions, and estimation of motion blur ker-

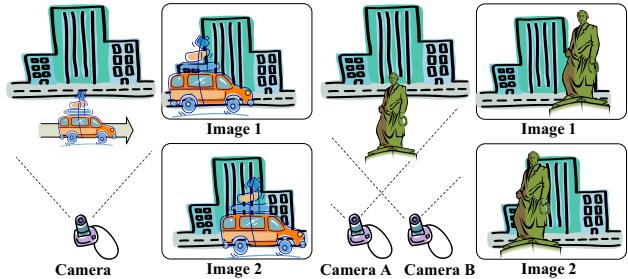


Figure 1. Different motions are introduced to the foreground and background objects due to a moving object (left) and a depth difference (right).

nels. Solutions of these problems affect to each other reciprocally. Therefore, we jointly solve these problems by an energy minimization approach. Our algorithm uses a regularized form of the energy function and minimizes it by an alternating optimization technique. From two or more input images, images are restored by removing spatially-varying motion blurs.

The primary contributions of our work are twofold. First, it proposes a new approach to restoring images that are contaminated by spatially-varying motion blurs. Second, in addition to the restoration, associated motion blur kernels, segmentation, and motions are simultaneously obtained. The proposed method has wider applicability compared with the prior approaches that only assume a single motion blur kernel.

### 1.1. Related work

The major difficulty of motion deblurring is accurate estimation of motion blur kernels, in other words, motion point spread functions (PSFs). Most prior approaches extensively focus on a single motion PSF caused by a camera, but not spatially-varying motion PSFs caused by multiple objects' motions. Yitzhaki et al. propose a method for motion deblurring by identifying the motion PSF using an isotropic image assumption [2]. Assuming an 1D directional motion blur in their model, they extract the motion

\*Part of this work was done while the first author was visiting Microsoft Research Asia as an intern.

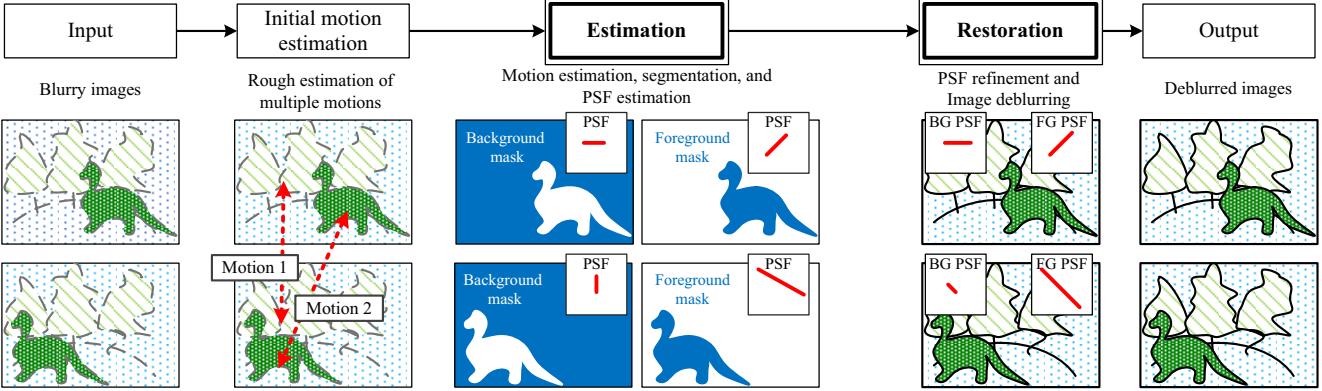


Figure 2. Overview of the proposed method. Given blurry images as input, the algorithm finds the initial guess for the multiple motion estimation using Bhat *et al.*'s method [1]. In the estimation step, accurate motion estimation, segmentation, and motion PSF estimation are performed. With these estimates, the restoration step deblurs input images with refinement of the motion PSF estimates.

component from the image in the frequency domain using autocorrelation. Afterwards, they extend their method to account for a high frequency motion blur caused by vibration that is observed along with the major 1D directional motion blur [3]. Recently, Jia proposes a deblurring method that uses the translucency information of blurry regions between two opaque regions [4]. The information is acquired by image matting and used to estimate a single motion PSF.

In the multiple image framework, Chen *et al.* propose a method for recovering the original image from two consecutively blurred images [5]. The method adopts an energy minimization technique that computes the best translation of the camera from the two consecutive images. Similarly, Rav-Acha and Peleg [6] use two blurry images (one is horizontally and the other is vertically blurred) for deblurring by simultaneously estimating parameters of image displacements using iterative energy minimization. Basclé *et al.* propose a method to remove motion blur and generate high-resolution images from an image sequence by finding pixelwise motion [7]. Bar *et al.* developed a unified framework of segmentation and blind restoration by treating the image segmentation and restoration as a coupled problem [8]. While they use a classical color-based image segmentation without using the motion blur kernel for the segmentation, convincing results are obtained with the proposed method. Yuan *et al.* use a blurred/noisy image pair to handle uniform motion blur [9]. They also propose a novel deconvolution method that reduces ringing artifacts.

Recently, a few works for removing spatially-variant motion blur have been proposed. Levin proposes a method to restore a blurry image that contains one motion blurred object and a sharp background [10]. The algorithm uses the statistical distributions of image gradient to estimate motion blurs and to locate a blurry foreground object. Bardsley *et al.* use a phase diversity method to find motion PSFs

in which segmentation is performed by splitting an image into regions that have homogeneous motion PSFs [11].

Apart from these post-processing approaches, a few hardware approaches have been proposed as well. Ben-Ezra and Nayar develop a hybrid imaging system [12] that consists of two cameras: one is high-resolution with a long exposure time, the other is low-resolution but can capture multiple frames with a shorter exposure time. Using the low-resolution camera for estimating the camera motion, their method estimates PSF for the high resolution camera. It is known that hardware approaches can handle more general camera motions than software approaches. However, such hardware systems are still expensive, and unfortunately they are not commonly available.

## 1.2. Overview

Our goal is to restore motion blurred images which contain spatially-varying motion blurs. Figure 2 illustrates the overview of our method.

Our algorithm takes multiple blurry images of roughly the same scene as input, e.g., consecutive frames from a video. To simplify the explanation, we describe our algorithm in the case of two input images although the algorithm can naturally work with more than two input images. Given the input images, our method first finds an initial guess of multiple different motions that appear in the images. For this step, we use Bhat *et al.*'s motion fitting method [1]. Since the accurate motion estimation is performed in the following step, the initial estimation result need not necessarily be precise.

Once the initial estimation of motion parameters is obtained, the first stage of the proposed method simultaneously refines the motion estimates, performs image segmentation into regions of homogeneous motions, and estimates the corresponding motion PSFs. The estimated PSFs are

further refined in the following restoration stage, where we perform the refinement and image restoration at the same time, using a method similar to Jin *et al.*'s work [13].

A few assumptions are used in the proposed method. First, a piecewise affine motion is assumed between image segments. Second, we assume that a motion PSF is dominated by a linear component. These assumptions are commonly used in many prior deblurring methods (e.g., [2], [3], [6], [7]) and reasonably hold with image sequences captured by a video camera.

## 2. Formulation

Section 2.1 describes the motion blur model used in this paper. Section 2.2 describes the algorithm of simultaneous estimation for obtaining multiple relative motions, segmentation, and motion PSFs. Section 2.3 explains the algorithm for image restoration and motion PSFs refinement.

### 2.1. Motion blur model

Let  $I$  be an image that contains a motion blur and  $I_s$  be the corresponding sharp image in which the motion blur does not exist. These two images  $I$  and  $I_s$  are related by the following equation

$$I(\mathbf{x}) = k_{\mathbf{v}} * I_s(\mathbf{x}) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-t^2/2} I_s(\mathbf{x} + \mathbf{v}t) dt, \quad (1)$$

where  $\mathbf{v}$  is a 2D motion vector corresponding to the relative motion between a camera and the scene, and  $k_{\mathbf{v}}$  is a motion blur kernel parameterized by the motion vector  $\mathbf{v}$ .

Since an image is composed of a collection of discrete pixels, Equation (1) can be expressed in a discrete form;

$$\mathbf{u} = \mathbf{K}_{\mathbf{v}} \mathbf{u}_s, \quad (2)$$

in which the column vectors  $\mathbf{u}$  and  $\mathbf{u}_s$  embed pixels of the image  $I$  and  $I_s$ , respectively, and  $\mathbf{K}_{\mathbf{v}}$  is a matrix representation of convolution with the motion blur kernel  $k_{\mathbf{v}}$ .

In a captured image, multiple different motions and spatially-varying motion blurs may exist. To represent  $n$  different relative motions and motion blurs, we use the subscript  $i$ , such as  $\mathbf{v}_i$  and  $\mathbf{K}_{\mathbf{v}_i}$ , where  $0 \leq i < n$ , to indicate the  $i$ -th motion. We also introduce a segmentation mask vector  $\mathbf{m}_i$  and its matrix representation  $\mathbf{M}_i$ , which is a diagonal matrix whose diagonal elements are equivalent to the elements of  $\mathbf{m}_i$ , to indicate the  $i$ -th image region that corresponds to the  $i$ -th relative motion. In the vector  $\mathbf{m}_i$ , we use floating-point numbers instead of binary numbers, and all elements are set in the range  $[0, 1]$ . For example, if  $\mathbf{m}_i[l] = 1$ , the  $l$ -th pixel belongs to the  $i$ -th motion. As  $\mathbf{m}_i[l]$  decreases, the  $l$ -th pixel is less likely to belong to the  $i$ -th motion. Using this notation, the motion blur caused by the  $i$ -th motion can be described as

$$\mathbf{M}_i \mathbf{u} = \mathbf{M}_i \mathbf{K}_{\mathbf{v}_i} \mathbf{u}_s. \quad (3)$$

Assume that there are two sharp images  $\mathbf{u}_{s0}$  and  $\mathbf{u}_{s1}$  of roughly the same scene. Image segments of these two images are associated by piecewise affine motions as

$$\mathbf{M}_i \mathbf{u}_{s0} = \mathbf{M}_i \mathbf{u}_{s1}^{\mathbf{A}_i}, \quad (4)$$

where  $\mathbf{u}_{s0}^{\mathbf{A}_i}$  is the vectorized image  $\mathbf{u}_{s0}$  transformed by the affine transform  $\mathbf{A}_i$ , and  $\mathbf{M}_i$  is a segmentation mask defined in the coordinates of  $\mathbf{u}_{s0}$ . When these image segments are blurred by spatially-varying motion PSFs, Equation (4) changes to

$$\begin{aligned} \mathbf{M}_i \mathbf{u}_0 &= \mathbf{M}_i \mathbf{K}_{\mathbf{v}_{0i}} \mathbf{u}_{s0}, \quad \text{and} \\ \mathbf{M}_i \mathbf{u}_1^{\mathbf{A}_i} &= \mathbf{M}_i \mathbf{K}_{\mathbf{v}_{1i}}^{\mathbf{A}_i} \mathbf{u}_{s1}^{\mathbf{A}_i}, \end{aligned} \quad (5)$$

where  $\mathbf{u}_0$  and  $\mathbf{u}_1$  are blurry images, and  $\mathbf{K}_{\mathbf{v}_{1i}}^{\mathbf{A}_i}$  is a matrix representation of convolution with the  $i$ -th blur kernel parameterized by a vector  $\mathbf{v}_{1i}^{\mathbf{A}_i}$ .  $\mathbf{v}_{0i}$  represents the  $i$ -th motion blur parameter in image  $\mathbf{u}_0$ . Likewise,  $\mathbf{v}_{1i}$  denotes the  $i$ -th motion blur parameter of image  $\mathbf{u}_1$ . In the following, we use  $\mathbf{K}_{0i}$  and  $\mathbf{K}_{1i}^{\mathbf{A}_i}$  to denote  $\mathbf{K}_{\mathbf{v}_{0i}}$  and  $\mathbf{K}_{\mathbf{v}_{1i}}^{\mathbf{A}_i}$ , respectively. For a 2D motion vector  $\mathbf{v}$ ,  $\mathbf{v}^{\mathbf{A}}$  indicates a vector warped by a transform  $\mathbf{A}$  without the translation component.

From Equations (4) and (5), we obtain

$$\mathbf{M}_i \mathbf{u}_1^{\mathbf{A}_i} = \mathbf{M}_i \mathbf{K}_{1i}^{\mathbf{A}_i} \mathbf{u}_{s0}. \quad (6)$$

Because of occlusions or image boundaries, we may have image areas that do not have corresponding regions in the other image. For these image areas, segments cannot be defined. We call such image areas occluded regions.

### 2.2. Estimation step

Since motion blurs are commutative, applying motion PSFs  $\mathbf{K}_{0i}$  and  $\mathbf{K}_{1i}^{\mathbf{A}_i}$  to the  $i$ -th segments of  $\mathbf{u}_1^{\mathbf{A}_i}$  and  $\mathbf{u}_0$  respectively will produce the same result, which can be written as

$$\begin{aligned} \mathbf{M}_i \mathbf{K}_{1i}^{\mathbf{A}_i} \mathbf{u}_0 &= \mathbf{M}_i \mathbf{K}_{1i}^{\mathbf{A}_i} \mathbf{K}_{0i} \mathbf{u}_s \\ &= \mathbf{M}_i \mathbf{K}_{0i} \mathbf{K}_{1i}^{\mathbf{A}_i} \mathbf{u}_s = \mathbf{M}_i \mathbf{K}_{0i} \mathbf{u}_1^{\mathbf{A}_i}, \end{aligned} \quad (7)$$

where  $\mathbf{u}_s = \mathbf{u}_{s0}$ . In the following, to simplify the notation, we will use  $\mathbf{u}_s$  to represent  $\mathbf{u}_{s0}$ . From Equation (7), the following equation is obtained;

$$\sum_i \mathbf{d}_i^T \mathbf{M}_i \mathbf{d}_i = 0, \quad (8)$$

where  $\mathbf{d}_i = \mathbf{K}_{0i} \mathbf{u}_1^{\mathbf{A}_i} - \mathbf{K}_{1i}^{\mathbf{A}_i} \mathbf{u}_0$ . Therefore, we expect that minimizing the left term of Equation (8) yields the solutions of  $\mathbf{K}$  and  $\mathbf{A}$ , if mask vectors  $\mathbf{m}_i$  are properly set. However, due to the non-linearity of the equation, the solution that minimizes the term cannot be determined uniquely. To assure the uniqueness of the optimal solution, we use additional constraints similar to Jin *et al.*'s method [13], which

penalize long motion blurs. For mask vectors  $\mathbf{m}_i$ , setting all the elements to zero can produce a numerically optimal result when minimizing the left term of Equation (8), however, this approach should be avoided. In addition, it is better to preserve edges in mask images while smooth transition of mask values should be observed in blurry image regions. Combining all these constraints, we design a regularized form of the energy function as follows.

$$E(\mathbf{v}, \mathbf{A}, \mathbf{m}) = \sum_i \underbrace{\mathbf{d}_i^T \mathbf{M}_i \mathbf{d}_i}_{\text{error term}} + \underbrace{\alpha \sum_i [\mathbf{v}_{0i}^T \mathbf{v}_{0i} + (\mathbf{v}_{1i}^{\mathbf{A}_i})^T (\mathbf{v}_{1i}^{\mathbf{A}_i})]}_{\text{motion blur term}} + \underbrace{\beta (\mathbf{1} - \sum_i \mathbf{m}_i)^T (\mathbf{1} - \sum_i \mathbf{m}_i)}_{\text{occlusion term}} + \underbrace{\gamma \sum_i [(\mathbf{D}_x \mathbf{m}_i)^T \mathbf{G} (\mathbf{D}_x \mathbf{m}_i) + (\mathbf{D}_y \mathbf{m}_i)^T \mathbf{G} (\mathbf{D}_y \mathbf{m}_i)]}_{\text{edge term}}, \quad (9)$$

where  $\alpha$ ,  $\beta$ , and  $\gamma$  are weighting factors.

In Equation (9), the error term computes the weighted differences between two segments. This is the core term in the energy function for estimating the multiple motions, motion PSFs, and segmentation. The motion blur term assures the uniqueness of the motion PSFs by penalizing excessively long motion blur vectors. The occlusion term accounts for the occluded pixels that are measured by  $(\mathbf{1} - \sum_i \mathbf{m}_i)$ , in which  $\mathbf{1}$  is a vector whose elements are all one. The edge term makes mask estimates rapidly change around edges, but slowly change in the smooth regions. To evaluate transitions of mask values, spatial gradients of mask values are computed.  $\mathbf{D}_x \mathbf{m}_i$  and  $\mathbf{D}_y \mathbf{m}_i$  are the spatial gradients, where  $\mathbf{D}_x$  and  $\mathbf{D}_y$  are matrix representations of convolution with vertical and horizontal gradient kernels, i.e.,  $[0 \ -1 \ 1]$  and  $[0 \ -1 \ 1]^T$ .  $\mathbf{G}$  is a diagonal matrix whose elements are  $1/(g_j^2 + \delta)$ , where  $g_j$  is the gradient magnitude of the  $j$ -th pixel of  $\mathbf{u}_0$ .  $\delta$  is a small positive constant used to prevent division by zero. The mask vectors are constrained by  $\sum_i m_{ij} \leq 1 \wedge m_{ij} \geq 0 \forall j$ , where  $m_{ij}$  is the  $j$ -th element of  $\mathbf{m}_i$ .

We optimize the energy function in Equation (9) using an alternating optimization approach and gradient-based minimization techniques. The implementation details about the optimization are described in Section 3.

### 2.3. Restoration step

The estimated PSFs in the previous stage are *relative* PSFs among images which are not necessarily *optimal* for generating sharp images via deconvolution. We call the

PSFs that associate a sharp image with the blurry image the *optimal PSFs* to distinguish from the relative PSFs.

As described by Jin *et al.* [13], the optimal PSFs and the relative PSFs are related as

$$\hat{\mathbf{v}}_{0i} = \sqrt{b^2 + 1} \mathbf{v}_{0i} - b \mathbf{v}_{1i}^{\mathbf{A}_i}, \quad \text{and} \quad (10)$$

$$\hat{\mathbf{v}}_{1i}^{\mathbf{A}_i} = -b \mathbf{v}_{0i} + \sqrt{b^2 + 1} \mathbf{v}_{1i}^{\mathbf{A}_i}, \quad (11)$$

where  $b$  is an unknown scalar. From these relations, the image  $\mathbf{u}_0$  can be deblurred to yield a sharp image  $\mathbf{u}_s$ , except for the occluded regions, by minimizing the following function:

$$E_{r1}(\mathbf{u}_s, b) = \sum_i (\mathbf{d}_{0i}^T \mathbf{M}_i \mathbf{d}_{0i} + \mathbf{d}_{1i}^T \mathbf{M}_i \mathbf{d}_{1i}) + \rho(\mathbf{u}_s), \quad (12)$$

where  $\mathbf{d}_{0i} = \mathbf{u}_0 - \mathbf{K}_{0i(b)} \mathbf{u}_s$  and  $\mathbf{d}_{1i} = \mathbf{u}_1^{\mathbf{A}_i} - \mathbf{K}_{1i(b)}^{\mathbf{A}_i} \mathbf{u}_s$ .  $\mathbf{K}_{0i(b)}$  and  $\mathbf{K}_{1i(b)}^{\mathbf{A}_i}$  are matrices parameterized by motion blur vectors  $\hat{\mathbf{v}}_{0i}$  and  $\hat{\mathbf{v}}_{1i}^{\mathbf{A}_i}$  computed by Equations (10) and (11).  $\rho(\mathbf{u}_s)$  is a regularization function. We used a regularization function introduced by Geman *et al.* [14],  $\rho(\mathbf{u}_s) = \zeta \sum_j g_j^2 / (1 + g_j^2)$ , where  $\zeta$  is a weighting factor for the regularization term.  $g_j$  is the spatial gradient magnitude of the  $j$ -th pixel of  $\mathbf{u}_s$ .  $\zeta$  should be relatively small to avoid strong bias to  $\rho(\mathbf{u}_s)$ .

In the proposed method, occluded regions need special handling because no relative motions can be defined for them. To handle occluded regions, we assume that such regions have the most dominant motion in the image. This assumption works for many images, e.g., where the most dominant motion is observed in the background of the scene by a camera motion. Using this assumption, we modify the mask matrix  $\mathbf{M}$  to assign new mask values in the occluded regions. With the modified mask matrix  $\mathbf{M}^o$ , the energy function  $E_{r1}$  is slightly modified so that the occluded regions are correctly handled;

$$E_{r2}(\mathbf{u}_s, b) = \sum_i (\mathbf{d}_{0i}^T \mathbf{M}_i^o \mathbf{d}_{0i} + \mathbf{d}_{1i}^T \mathbf{M}_i \mathbf{d}_{1i}) + \rho(\mathbf{u}_s). \quad (13)$$

By minimizing Equation (13), the deblurred image  $\mathbf{u}_s$  corresponding to  $\mathbf{u}_0$  and optimal PSFs  $\mathbf{K}$  are obtained.

## 3. Implementation

This section describes minimization methods for the energy functions in Equations (9) and (13).

### 3.1. Computation of estimation step

In the estimation step, motions  $\mathbf{A}$ , image segments  $\mathbf{m}$ , and non-uniform motion PSFs  $\mathbf{K}$  (which are parameterized by  $\mathbf{v}$ ) are estimated using Equation (9). To jointly optimize these parameters, we use an alternating optimization approach for three sets of parameters  $[\mathbf{m}_i]$ ,  $[\mathbf{K}_i]$  and  $[\mathbf{A}_i]$ .

One of these parameter sets is updated at one step, and the others are updated in the following steps in sequence. This iteration continues until the energy function converges. We use a gradient-based unconstrained non-linear optimization method, `fminunc` of Matlab optimization toolbox, to optimize motion PSFs  $[\mathbf{K}_i]$ , and a gradient projection method [15] to optimize segmentation masks  $[\mathbf{m}_i]$ . For estimating motions  $[\mathbf{A}_i]$ , we use a registration method described by Baker *et al.* [16].

**Segmentation masks** Segmentation masks  $\mathbf{m}_i$  are estimated by optimizing the terms of Equation (9) except the motion blur term. The gradient of  $\mathbf{m}_i$  is computed from Equation (9) as

$$\begin{aligned}\frac{\partial E'}{\partial \mathbf{m}_i} &= \mathbf{d}_i^T \mathbf{d}_i - 2\beta(1 - \sum_j \mathbf{m}_j) \\ &+ 2\gamma[\mathbf{D}_x^T \mathbf{G} \mathbf{D}_x + \mathbf{D}_y^T \mathbf{G} \mathbf{D}_y]\mathbf{m}_i.\end{aligned}\quad (14)$$

**Motion PSFs** Motion PSFs can be estimated by optimizing the error term and the motion blur term of Equation (9). By computing gradients of motion blur parameters  $\mathbf{v}_{0i}$  and  $\mathbf{v}_{1i}^{\mathbf{A}_i}$  in a similar way to Jin *et al.* [13], we can obtain

$$\frac{\partial E}{\partial \mathbf{v}_{0i}} = 2 \left( \mathbf{M}_i \dot{\mathbf{K}}_{0i} \nabla \mathbf{u}_1^{\mathbf{A}_i} \right)^T \mathbf{d}_i + \alpha \mathbf{v}_{0i}, \text{ and } (15)$$

$$\frac{\partial E}{\partial \mathbf{v}_{1i}^{\mathbf{A}_i}} = -2 \left( \mathbf{M}_i \dot{\mathbf{K}}_{1i}^{\mathbf{A}_i} \nabla \mathbf{u}_0 \right)^T \mathbf{d}_i + \alpha \mathbf{v}_{1i}^{\mathbf{A}_i}, \quad (16)$$

where  $\nabla \mathbf{u}_0 = [\mathbf{D}_x \mathbf{u}_0 \ \mathbf{D}_y \mathbf{u}_0]$  is a matrix consisting of two column vectors of image gradients, and  $\dot{\mathbf{K}}_v$  is a matrix representation of convolution with a kernel  $k_v$  defined as

$$k_v * I = - \int_{-\infty}^{\infty} \frac{t}{\sqrt{2\pi}} e^{-t^2/2} I(\mathbf{x} + \mathbf{v}t) dt. \quad (17)$$

**Motion parameters** Motion parameters  $\mathbf{A}_i$  can be computed by optimizing the error term of Equation (9). To optimize the error term,  $\mathbf{K}_{0i} \mathbf{u}_1^{\mathbf{A}_i}$  and  $\mathbf{K}_{1i}^{\mathbf{A}_i} \mathbf{u}_0$  should be computed at each iteration. However, if we assume that current motion parameters  $\mathbf{A}_i$  are close enough to the solution, we can approximate the minimization by image registration between  $\mathbf{w}_1^{\mathbf{A}_i} = \mathbf{K}_{0i} \mathbf{u}_1^{\mathbf{A}_i}$  and  $\mathbf{w}_0 = \mathbf{K}_{1i}^{\mathbf{A}_i} \mathbf{u}_0$ . Consequently, motion parameters can be computed by Lucas-Kanade based registration method between  $\mathbf{w}_0$  and  $\mathbf{w}_1^{\mathbf{A}_i}$  with weighting factors  $\mathbf{M}_i$ . For this computation, we used a weighted inverse-compositional method described by Baker *et al.* [16].

**Hierarchical estimation** For large images, estimation of these parameters is computationally intensive, and obtaining an optimal solution becomes difficult due to the number of unknown parameters. To avoid this problem, we used a

pyramidal approach for the computation in a coarse-to-fine manner. To assign initial values for the estimation in a finer level, segmentation masks are enlarged by bilinear interpolation. The estimated motion blur parameter at a pixel is magnified twice and propagated to the corresponding four pixels at the next level of the pyramid. Likewise, the estimated motion parameters are propagated to the next level of the pyramid with the translation parameters multiplied by two.

### 3.2. Computation of restoration step

The image restoration and PSF refinement are also performed by a gradient-based optimization. We use a gradient projection method to force pixel values of  $\mathbf{u}_s$  to be within the valid range, and an unconstrained gradient descent method to optimize the parameter  $b$  of Equation (13). We describe the details of optimizing Equation (13) for obtaining the deblurred image  $\mathbf{u}_s$  and optimal PSFs.

**Image deblurring** From Equation (13), the gradient of the deblurred image  $\mathbf{u}_s$  can be written as

$$\begin{aligned}\frac{\partial E'_{r2}}{\partial \mathbf{u}_s} &= -2\mathbf{K}_{0i(b)}^T \mathbf{M}_i^o \mathbf{d}_{0i} \\ &- 2(\mathbf{K}_{1i(b)}^{\mathbf{A}_i})^T \mathbf{M}_i \mathbf{d}_{1i} + \frac{d\rho(\mathbf{u}_s)}{d\mathbf{u}_s}.\end{aligned}\quad (18)$$

**Motion PSFs** To obtain the optimal PSFs, the gradient of parameter  $b$  in Equation (13) can be computed as

$$\begin{aligned}\frac{\partial E'_{r2}}{\partial b} &= -2\mathbf{d}_{0i}^T \mathbf{M}_i^o (\dot{\mathbf{K}}_{0i(b)} \nabla \mathbf{u}_s)(q\mathbf{v}_{0i} - \mathbf{v}_{1i}^{\mathbf{A}_i}) \\ &- 2\mathbf{d}_{1i}^T \mathbf{M}_i (\dot{\mathbf{K}}_{1i(b)}^{\mathbf{A}_i} \nabla \mathbf{u}_s)(q\mathbf{v}_{1i}^{\mathbf{A}_i} - \mathbf{v}_{0i}),\end{aligned}\quad (19)$$

where  $q = b/\sqrt{b^2 + 1}$ .

## 4. Experiments

To evaluate the proposed algorithm, we used both synthetic and real-world images. The input images are first converted into grayscale images to perform the estimation step. Once the parameters are obtained, the restoration step is performed for each color band to produce the final result. Throughout our experiments, we set parameters in Equation (9) as  $(\alpha, \beta, \gamma, \delta) = (10^{-2}, 10^{-3}, 2.0 \times 10^{-5}, 10^{-4})$ , and the weighting factor of  $\rho$  in Equation (13) as  $\zeta = 10^{-5}$ . The image intensity is normalized in the range  $[0, 1]$ .

We ran our algorithm over five synthetically-blurred images. Root mean squared errors (RMSE) from the ground truth (original sharp image) were measured in normalized intensity (Fig. 3). From left to right for each dataset, the first column shows RMSE of the blurry image, and the second and third columns show RMSEs computed from the restored images using the foreground PSF and the background

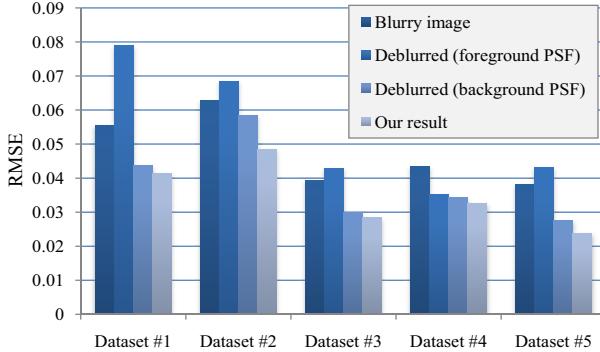


Figure 3. Comparison of RMSEs from the ground truth using synthetic datasets that contain background and foreground motion PSFs. From left to right bars in each dataset, RMSE of the input blurry image, the deblurred image using the background PSF, the deblurred image using the foreground PSF, and our result are shown.

PSF respectively. The right-most column shows the RMSE of our result. Images restored by only foreground PSFs show errors even larger than that of blurred images. Also, the images restored only by background PSFs show slightly larger errors than our results. This is because the area of the background region is usually large enough to make the RMSE numerically small. However, perceptual differences are large due to the large error in the foreground region. In these experiments, all the image sizes are  $640 \times 480$  pixels. The average computation time is about six minutes for the estimation step and 40 seconds for the restoration step.

Four different real-world scenes that contain spatially-varying motion blurs were deblurred by our method (Fig. 4). The first and second rows show the blurry input images. The third and fourth rows show the segmentation masks that correspond to the background and foreground, respectively. Brighter pixels indicate the higher confidence in being the part of background/foreground. Pixels that do not belong to either the foreground or the background are estimated to be in the occluded regions. In the fifth row, estimated PSFs are shown in a tabular form, e.g., 1-A corresponds to the PSF of Input 1, segment A (background). The following two rows show the deblurred results restored from input 1 and 2. For the deblurring, the estimated non-uniform PSFs are used for the corresponding image regions. For better visualization, the bottom row shows magnified image portions.

Input images containing three different motions were tested to show that our method is able to handle more than two different motions (Fig. 5). These three motions correspond to the background, a pencil vase, and a cigarette pack, respectively, resulting from depth differences among them. The cigarette pack is the nearest object to the camera, and the pencil vase lies between the cigarette pack and the background. Our method successfully segments the scene, estimates the motion PSFs, and deblurs the input images.

The bottom row of the figure shows the magnified views of the image portions for a better comparison.

Figure 6 depicts the power of handling spatially-varying motion PSFs. The foreground object is blurred by its own motion while background is also blurred by a camera motion. The image deblurred only with the background PSF shows severe artifacts on the foreground object due to the inconsistency with the foreground PSF (Fig. 6b). On the other hand, the image deblurred only with the foreground PSF yields a blurry background (Fig. 6c). Our method is able to avoid this problem because it uses non-uniform motion PSFs for image restoration (Fig. 6d).

## 5. Discussion

In this paper, we proposed a new method for removing non-uniform motion blurs from images. To achieve this goal, the problem is formulated as simultaneous estimation of multiple motions, segmentation, and spatially-varying motion PSFs. The problem is solved by optimizing a regularized form of the energy function. Furthermore, the estimated PSFs are refined to achieve the restoration of images. We have evaluated the proposed method using a variety of synthetic and real-world images and validated the effectiveness of the algorithm.

Although we used only two images as input, the algorithm can naturally take more than two images as well. In this case, the estimation step is performed for each pair of images, and the restoration step is performed by simultaneously optimizing the energy functions of all pairs of images.

**Limitations** Our algorithm has a few limitations. One is that it shows some artifacts around the boundaries of different motions in restored images. Blurry regions on boundaries of the foreground object still remain (Fig. 6). This artifact is inevitable due to missing information of hidden pixels behind the foreground objects. Second, like existing segmentation algorithms, our segmentation is not performed well on textureless regions because it is difficult to determine the motion in such regions. Therefore, the proposed method works better if the input images are more textured.

## Acknowledgements

We would like to thank Kyuman Jeong for the help in taking pictures used for our experiments, and Derek J. Lactin for his help on English editing. This research was supported in part by the ITRC support program.

## References

- [1] P. Bhat, K. Zheng, N. Snavely, A. Agarwala, M. Agrawala, M. Cohen, and B. Curless, “Piecewise image registration in the presence

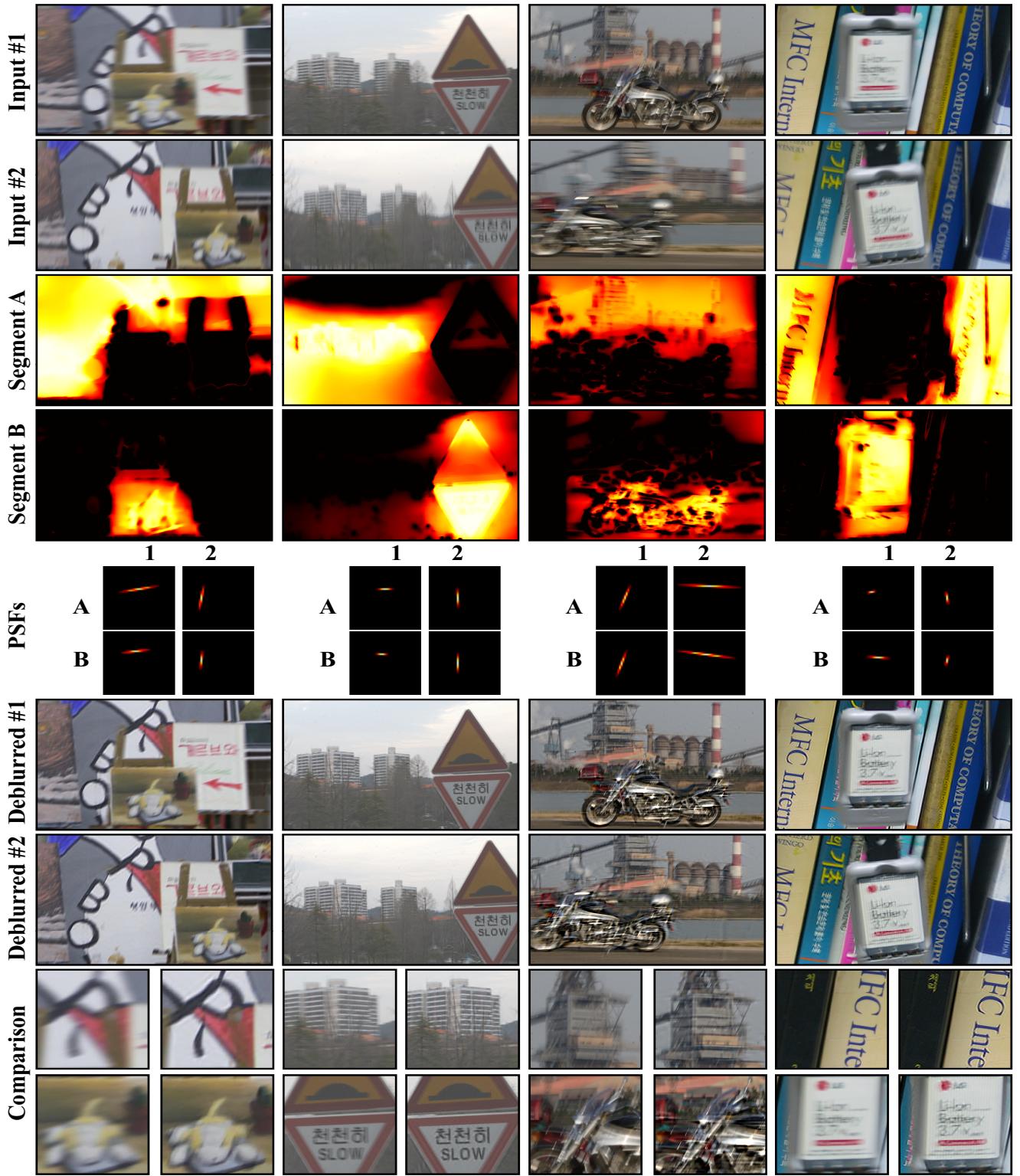


Figure 4. Results of our method. Among the four scenes, the input images for the left-most column were blurred synthetically, while the other input images were photographed from real-world scenes. The top two rows show blurry input images. Next two rows show segmentation masks followed by estimated motion PSFs. Below the motion PSFs, deblurred results are shown. The bottom row shows magnified views of original blurry images and the deblurred images for better visualization.

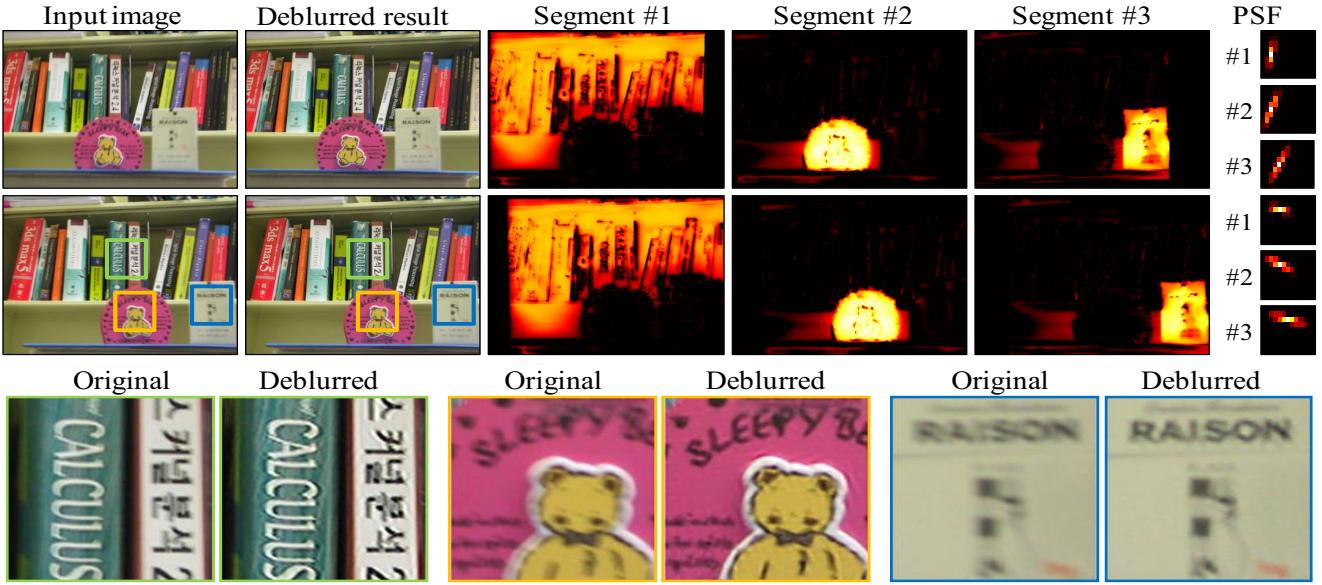


Figure 5. Result of deblurring images that contain three different motions and motion PSFs.



(a) One of two input images (b) Deblurred by background PSF (c) Deblurred by foreground PSF (d) Our result (non-uniform PSF)  
Figure 6. Comparison of deblurred results. (a) shows one of the two input images. (b) shows the deblurred result using the background motion PSF, and (c) shows the result using the foreground motion PSF. In (b) and (c), the foreground and background are not properly restored, respectively. (d) shows the result of our method. Since our method uses non-uniform motion PSFs for restoring images, the result shows less artifacts compared with (b) and (c).

- of multiple large motions,” in *Proc. of Computer Vision and Pattern Recognition*, vol. 2, pp. 2491–2497, 2006. 2
- [2] Y. Yitzhaky, I. Mor, A. Lantzman, and N. S. Kopeika, “Direct method for restoration of motion-blurred images,” *Journal of Opt. Soc. Am. A*, vol. 15, no. 6, pp. 1512–1519, 1998. 1, 3
- [3] Y. Yitzhaky, G. Boshusha, Y. Levi, and N. S. Kopeika, “Restoration of an image degraded by vibrations using only a single frame,” *Optical Engineering*, vol. 38, no. 8, 2000. 2, 3
- [4] J. Jia, “Single image motion deblurring using transparency,” in *Proc. of Computer Vision and Pattern Recognition*, pp. 1–8, 2007. 2
- [5] W.-G. Chen, N. Nandhakumar, and W. N. Martin, “Image motion estimation from motion smear - a new computational model,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 412–425, 1996. 2
- [6] A. Rav-Acha and S. Peleg, “Two motion-blurred images are better than one,” *Pattern Recognition Letters*, vol. 26, pp. 311–317, 2005. 2, 3
- [7] B. Basile, A. Blake, and A. Zisserman, “Motion deblurring and superresolution from an image sequence,” in *Proc. of European Conference on Computer Vision*, vol. 2, pp. 573–582, 1996. 2, 3
- [8] L. Bar, N. Sochen, and N. Kiryati, “Variational pairing of image segmentation and blind restoration,” in *Proc. of European Conference on Computer Vision*, vol. 2, pp. 166–177, 2004. 2

- [9] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum, “Image deblurring with blurred/noisy image pairs,” *ACM Trans. on Graphics (SIGGRAPH)*, vol. 26, no. 3, p. 1, 2007. 2
- [10] A. Levin, “Blind motion deblurring using image statistics,” in *Advances in Neural Information Processing Systems*, 2006. 2
- [11] J. Bardsley, S. Jefferies, J. Nagy, and R. Plemmons, “Blind iterative restoration of images with spatially varying blur,” *Optics Express*, vol. 14, pp. 1767–1782, 2006. 2
- [12] M. Ben-Ezra and S. K. Nayar, “Motion-based motion deblurring,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 689–698, 2004. 2
- [13] H. Jin, P. Favaro, and R. Cipolla, “Visual tracking in the presence of motion blur,” in *Proc. of Computer Vision and Pattern Recognition*, vol. 2, pp. 18–25, 2005. 3, 4, 5
- [14] S. Geman and D. McClure, “Bayesian image analysis: An application to single photon emission tomography,” in *Proc. of the Statistical Computing Section, American Statistical Association*, vol. 54, pp. 281–289, 1985. 4
- [15] J. B. Rosen, “The gradient projection method for nonlinear programming: Part I. Linear constraints,” *Journal of the Society for Industrial and Applied Mathematics*, vol. 8, no. 1, pp. 181–217, 1960. 5
- [16] S. Baker, R. Gross, I. Matthews, and T. Ishikawa, “Lucas-kanade 20 years on: A unifying framework: Part 2,” Tech. Rep. CMU-RI-TR-03-01, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, February 2003. 5