This ECCV 2018 paper, provided here by the Computer Vision Foundation, is the author-created version.
The content of this paper is identical to the content of the officially published ECCV 2018
LNCS version of the paper as available on SpringerLink: https://link.springer.com/conference/eccv

Joint Blind Motion Deblurring and Depth Estimation of Light Field

Dongwoo Lee[1], Haesol Park[1], In Kyu Park[2], and Kyoung Mu Lee[1]

[1] Department of ECE, ASRI, Seoul National University
dongwoo.lee@snu.ac.kr, haesol.park@gmail.com, kyoungmu@snu.ac.kr
[2] Department of Information and Communication Engineering, Inha University
pik@inha.ac.kr

Abstract. Removing camera motion blur from a single light field is a challenging task since it is highly ill-posed inverse problem. The problem becomes even worse when blur kernel varies spatially due to scene depth variation and high-order camera motion. In this paper, we propose a novel algorithm to estimate all blur model variables jointly, including latent sub-aperture image, camera motion, and scene depth from the blurred 4D light field. Exploiting multi-view nature of a light field relieves the inverse property of the optimization by utilizing strong depth cues and multi-view blur observation. The proposed joint estimation achieves high quality light field deblurring and depth estimation simultaneously under arbitrary 6-DOF camera motion and unconstrained scene depth. Intensive experiment on real and synthetic blurred light field confirms that the proposed algorithm outperforms the state-of-the-art light field deblurring and depth estimation methods.

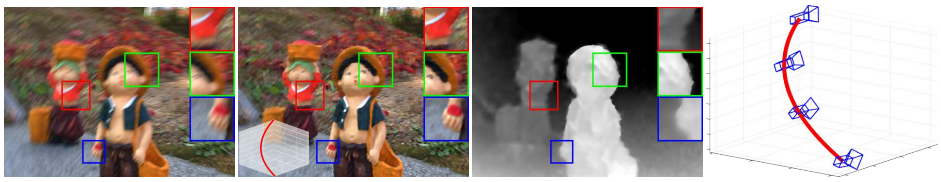
Keywords: light field, 6-DOF camera motion, motion blur, blind motion deblurring, depth estimation

1 Introduction

For the last decade, motion deblurring has been an active research topic in computer vision. Motion blur is produced by relative motion between camera and scene during the exposure where blur kernel, *i.e.* point spread function (PSF), is spatially non-uniform. In blind non-uniform deblurring problem, pixel-wise blur kernels and corresponding sharp image are estimated simultaneously.

Early works on motion deblurring [5, 8, 12, 27, 36] focus on removing spatially uniform blur in the image. However, the assumption of uniform motion blur is often broken in real world due to nonhomogeneous scene depth and rolling motion of camera. Recently, a number of methods [9, 13–15, 17, 19, 30, 33, 38] have been proposed for non-uniform deblurring. However, they still can not completely handle non-uniform blur caused by scene depth variation. The main challenge lies in the difficulty of estimating the scene depth with only single observation, which is highly ill-posed.

A light field camera ameliorates the ill-posedness of single-shot deblurring problem of the conventional camera. 4D light field is equivalent to multi-view



(a) (b) (c) (d)

Fig. 1: The proposed algorithm jointly estimates latent image, depth map, and camera motion from a single light field. (a) Center-view of blurred light field sub-aperture image. (b) Deblurred image of (a). (c) Estimated depth map. (d) Camera motion path and orientation (6-DOF).

images with narrow baseline, *i.e.* sub-aperture images, taken with an identical exposure [23]. Consequently, motion deblurring using light field can be leveraged by its multi-dimensional nature of captured information. First, strong depth cue is obtained by employing multi-view stereo matching between sub-aperture images. In addition, different blurs in the sub-aperture images can help the optimization converge more fast and precise.

In this paper, we propose an efficient algorithm to jointly estimate latent image, sharp depth map, and 6-DOF camera motion from a blurred single 4D light field as shown in Figure 1. In the proposed light field blur model, latent sub-aperture images are formulated by 3D warping of the center-view sharp image using the depth map and the 6-DOF camera motion. Then, motion blur is modeled as the integral of latent sub-aperture images during the shutter open. Note that the proposed center-view parameterization reduces light field deblurring problem in lower dimension comparable to a single image deblurring. The joint optimization is performed in an alternating manner, in which the deblurred image, depth map, and camera motion are refined during iteration. The overview of the proposed algorithm is shown in Figure 2. In overall, the contribution of this paper is summarized as follows.

- We propose a joint method which simultaneously solves deblurring, depth estimation, and camera motion estimation problems from a single light field.
- Unlike the previous state-of-the-art algorithm, the proposed method handles blind light field motion deblurring under 6-DOF camera motion.
- Practical and extensible blur formulation that can be extended to any multi-view camera system.

2 Related Works

Conventional Single Image Deblurring. One way to effectively remove the spatially-variant motion in a conventional single image is to first find the motion density function (MDF) and then generate the pixel-wise kernel from this function [13–15]. Gupta et al. [13] modeled the camera motion in discrete 3D motion

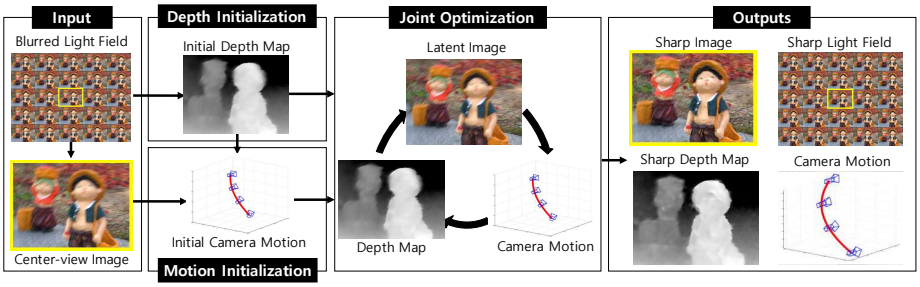


Fig. 2: Overview of the proposed algorithm. The proposed algorithm jointly estimates the latent image, depth map, and camera motion from a single light field.

space comprising x, y translation and in-plane rotation. They performed deblurring by iteratively optimizing the MDF and the latent image that best describe the blurred image. Similar model was used by Hu and Yang [15] in which MDF was modeled with 3D rotations. These methods of using MDF well parameterize the spatially-variant blur kernel into low dimensions. However, modeling the motion blur using MDF only in depth varying images is difficult, because the motion blur is determined by both camera motion and scene depth. In [14], the image was segmented by the matting algorithm, and the MDF and representative depth values of each region were found through the expectation-maximization algorithm.

A few methods [19, 30] estimated linear blur kernels locally, and they showed acceptable results for the arbitrary scene depth. Kim and Lee [19] jointly estimated the spatially varying motion flow and the latent image. Sun et al. [30] adopted a learning method based on convolutional neural network (CNN) and assumed that the motion was locally linear. However, the locally linear blur assumption does not hold in large motion.

Video and Multi-View Deblurring. Xu and Jia [37] decomposed the image region according to the depth map obtained from a stereo camera and recombined them after independent deblurring. Recently, several methods [10, 20, 24, 26, 35] have addressed the motion blur problem in video sequences. Video deblurring shows good performance, because it exploits optical flow as a strong guide for motion estimation.

Light Field Deblurring. Light field with two plane parameterization is equivalent to multi-view images with narrow baseline. It contains rich geometric information of rays in a single-shot image. These multi-view images are called sub-aperture images and individual sub-aperture images show slightly different blur pattern due to the viewpoint variation. In last a few years, several approaches [6, 11, 18, 28, 29] have been proposed to perform motion deblurring on the light field. Chandramouli et al. [6] addressed the motion blur problem in the light field for the first time. They assumed constant depth and uniform motion to alleviate the complexity of the imaging model. Constant depth means that

the light field has little information about 3D scene structure, which depletes the advantages of light field. Jin et al. [18] quantized the depth map into two layers and removed the motion blur in each layer. Their method assumed that the camera motion is in-plane translation and utilized depth value as a scale factor of translational motion. Although their model handles non-uniform blur kernel related to the depth map, a more general depth variation and camera motion should be considered for application to real-world scenes. Dansereau et al. [11] applied the Richardson-Lucy deblurring algorithm to the light field with non-blind 6-DOF motion blur. Although their method dealt with 6-DOF motion blur, it was assumed that the ground truth camera motion was known. Unlike [11], in this paper, we address the problem of blind deblurring which is a more highly ill-posed problem. Srinivasan et al. [29] solved the light field deblurring under 3D camera motion path and showed visually pleasing result. However, their methods do not consider 3D orientation change of the camera.

In contrast to the previous works of light field deblurring, the proposed method completely handles 6-DOF motion blur and unconstrained scene depth variation.

3 Motion Blur Formulation in Light Field

A pixel in a 4D light field has four coordinates, *i.e.* (x, y) for spatial and (u, v) for angular coordinates. A light field can be interpreted as a set of $u \times v$ multi-view images with narrow baseline, which are often called sub-aperture images [22]. Throughout this paper, a sub-aperture image is represented as $I(\mathbf{x}, \mathbf{u})$ where $\mathbf{x} = (x, y)$ and $\mathbf{u} = (u, v)$. For each sub-aperture image, the blurred image $B(\mathbf{x}, \mathbf{u})$ is the average of the sharp images $I_t(\mathbf{x}, \mathbf{u})$ during the shutter open over $[t_0, t_1]$ as follows:

$$B(\mathbf{x}, \mathbf{u}) = \int_{t_0}^{t_1} I_t(\mathbf{x}, \mathbf{u}) dt. \tag{1}$$

Following the blur model of [24, 26], we approximate all the blurred sub-aperture images by projecting a single latent image with 3D rigid motion. We choose the center-view ($\mathbf{c}$) of sub-aperture images and the middle of the shutter time (t_r) as the reference angular position and the time stamp of the latent image. With above notations, the pixel correspondence from each sub-aperture image to the latent image $I_{t_r}(\mathbf{x}, \mathbf{c})$ is expressed as follows:

$$I_t(\mathbf{x}, \mathbf{u}) = I_{t_r}(w_t(\mathbf{x}, \mathbf{u}), \mathbf{c}), \tag{2}$$

where

$$w_t(\mathbf{x}, \mathbf{u}) = \Pi_{\mathbf{c}}(\mathrm{P}_{t_r}^{\mathbf{c}}(\mathrm{P}_t^{\mathbf{u}})^{-1}\Pi_{\mathbf{u}}^{-1}(\mathbf{x}, D_t(\mathbf{x}, \mathbf{u}))). \tag{3}$$

$w_t(\mathbf{x}, \mathbf{u})$ computes the warped pixel position from $\mathbf{u}$ to $\mathbf{c}$, and from t to t_r. $\Pi_{\mathbf{c}}$, $\Pi_{\mathbf{u}}^{-1}$ are the projection and back-projection function between the image coordinate and the 3D homogeneous coordinate using the camera intrinsic parameters.

Matrices $\mathrm{P}^{\mathbf{c}}_{t_r}$ and $\mathrm{P}^{\mathbf{u}}_{t} \in SE(3)$ denote the 6-DOF camera pose at the corresponding angular position and the time stamp. $D_t(\mathbf{x}, \mathbf{u})$ is the depth map at the time stamp t.

In the proposed model, the blur operator $\Psi(\cdot)$ is defined by approximating the integral in (1) as a finite sum as follows:

$$B(\mathbf{x}, \mathbf{u}) \approx (\Psi \circ I)(\mathbf{x}, \mathbf{u}), \tag{4}$$

where

$$(\Psi \circ I)(\mathbf{x}, \mathbf{u}) = \frac{1}{M} \sum_{m=0}^{M-1} I_{t_r}(w_{t_m}(\mathbf{x}, \mathbf{u}), \mathbf{c}). \tag{5}$$

In (5), t_m is m_{th} uniformly sampled time stamp during the interval $[t_0, t_1]$.

Our goal is to formulate $(\Psi \circ I)(\mathbf{x}, \mathbf{u})$ with only center-view variables, *i.e.* $I_{t_r}(\mathbf{x}, \mathbf{c})$, $D_{t_r}(\mathbf{x}, \mathbf{c})$, and $\mathrm{P}^{\mathbf{c}}_{t_0}$. $\mathrm{P}^{\mathbf{u}}_{t_m}$ and $D_{t_m}(\mathbf{x}, \mathbf{u})$ are variables related to $\mathbf{u}$ in the warping function (5). Therefore, we parameterize $\mathrm{P}^{\mathbf{u}}_{t_m}$ and $D_{t_m}(\mathbf{x}, \mathbf{u})$ by employing center-view variables. Because the relative camera pose $\mathrm{P}^{\mathbf{c}\to\mathbf{u}}$ is fixed over time, $\mathrm{P}^{\mathbf{u}}_{t_m}$ is expressed by $\mathrm{P}^{\mathbf{c}}_{t_0}$ and $\mathrm{P}^{\mathbf{c}}_{t_1}$ as follows:

$$\mathrm{P}^{\mathbf{u}}_{t_m} = \mathrm{P}^{\mathbf{c}\to\mathbf{u}} \mathrm{P}^{\mathbf{c}}_{t_m}, \tag{6}$$

$$\mathrm{P}^{\mathbf{c}}_{t_m} = \exp(\frac{m}{M} \log(\mathrm{P}^{\mathbf{c}}_{t_1} (\mathrm{P}^{\mathbf{c}}_{t_0})^{-1})) \mathrm{P}^{\mathbf{c}}_{t_0}, \tag{7}$$

where exp and log denote the exponential and logarithmic maps between Lie group $SE(3)$ and Lie algebra $\mathfrak{se}(3)$ space [2]. To minimize the viewpoint shift of the latent image, we assume $\mathrm{P}^{\mathbf{c}}_{t_1} = (\mathrm{P}^{\mathbf{c}}_{t_0})^{-1}$ which makes $\mathrm{P}^{\mathbf{c}}_{t_m}$ an identity matrix when $t_m = t_r$. Note that we use the camera path model used in [24, 26]. However, the Bézier camera path model used in [29] can be directly applied to (7) as well. $D_{t_m}(\mathbf{x}, \mathbf{u})$ is also represented by $D_{t_r}(\mathbf{x}, \mathbf{c})$ by forward warping and interpolation.

In order to estimate all blur variables in the proposed light field blur model, we need to recover the latent variables, *i.e.* $I_{t_r}(\mathbf{x}, \mathbf{c})$, $D_{t_r}(\mathbf{x}, \mathbf{c})$, and $\mathrm{P}^{\mathbf{c}}_{t_0}$. We model an energy function as follows:

$$\begin{aligned} E = &\sum_{\mathbf{u}} \sum_{\mathbf{x}} \lambda_u \|(\Psi \circ I)(\mathbf{x}, \mathbf{u}) - B(\mathbf{x}, \mathbf{u})\|_1 \\ &+ \lambda_L \sum_{\mathbf{x}} \|\nabla I_{t_r}(\mathbf{x}, \mathbf{c})\|_2 + \lambda_D \sum_{\mathbf{x}} \|\nabla D_{t_r}(\mathbf{x}, \mathbf{c})\|_2. \end{aligned} \tag{8}$$

The data term imposes the brightness consistency between the input blurred light field and the restored light field. Notice that the L1-norm is employed in our approach as in [19], where it effectively removes the ringing artifact around object boundary and provides more robust deblurring results on large depth change. The last two terms are the total variation (TV) regularizers [1] for the latent image and the depth map, respectively.

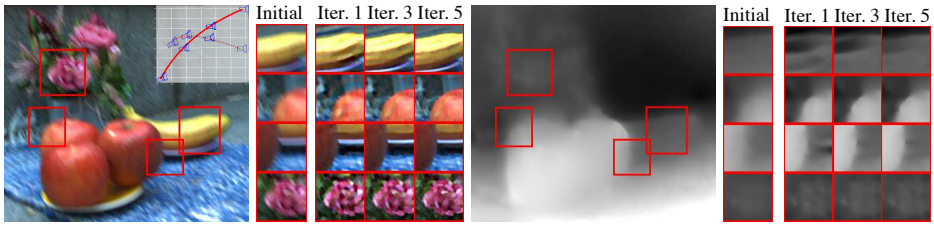


Fig. 3: Example of the iterative joint estimation. The proposed method converges in small number of iteration. (a)∼(b) Input blurred image and deblurring results by iteration. (c)∼(d) Initial blurred depth map and depth estimation results by iteration.

In our energy model, $D_{t_r}(\mathbf{x}, \mathbf{c})$ and $\mathrm{P}^{\mathbf{c}}_{t_0}$ are implicitly included in the warping function (5). The pixel-wise depth $D_{t_r}(\mathbf{x}, \mathbf{c})$ determines the scale of the motion at each pixel. At the boundary of an object where depth changes abruptly, there is a large difference of the blur kernel size between the near and farther objects. If the optimization is performed without considering this, the blur will not be removed well at the boundary of the object.

Simultaneously optimizing the three variables is complicated because the warping function (5) has severe nonlinearity. Therefore, our strategy is to optimize three latent variables in an alternating manner. We minimize one variable while the others are fixed. The optimization (8) is carried out in turn for the three variables. The L1 optimization is approximated using iterative reweighted least square (IRLS) [25]. The optimization procedure converges in small number of iterations (< 10).

An example of the iterative optimization is illustrated in Figure 3 which shows the benefit of the iterative joint estimation of sharp depth map and latent image. The initial depth map from the blurred light field is blurry as shown in Figure 3(c). However, both depth maps and latent images get sharper as the iteration continues as shown in Figure 3(d).

4 Joint Estimation of Latent Image, Camera Motion, and Depth Map

4.1 Update of the Latent Image

The proposed algorithm first updates the latent image $I_{t_r}(\mathbf{x}, \mathbf{c})$. In our data term, the blur operator (5) is simplified as the linear matrix multiplication, if $D_{t_r}(\mathbf{x}, \mathbf{c})$ and $\mathrm{P}^{\mathbf{c}}_{t_0}$ remain fixed. Updating the latent image is equivalent to minimizing (8) as follows:

$$\min_{I^{\mathbf{c}}_t} \sum_{\mathbf{u}} \|K^{\mathbf{u}} I^{\mathbf{c}}_{t_r} - B^{\mathbf{u}}\|_1 + \lambda_L \|\nabla I^{\mathbf{c}}_{t_r}\|_2. \tag{9}$$

$I^{\mathbf{c}}_{t_r}$, $B^{\mathbf{u}} \in \mathbb{R}^n$ are vectorized images and $K^{\mathbf{u}} \in \mathbb{R}^{n \times n}$ is the blur operator in square matrix form, where n is the number of pixels in the center-view sub-aperture image. TV regularization serves as a prior to the latent image with clear boundary while eliminating the ringing artifacts.

4.2 Update of the Camera Pose and Depth Map

Since (5) is a non-linear function of $D_{t_r}(\mathbf{x}, \mathbf{c})$ and $\mathrm{P}^{\mathbf{c}}_{t_0}$, it is necessary to approximate it in a linear form for efficient computation. In our approach, the blur operation (5) is approximated as a first-order expansion. Let $D_0(\mathbf{x}, \mathbf{c})$ and $\mathrm{P}^{\mathbf{c}}_0$ denote the initial variables, then (5) is approximated as follow:

$$\begin{aligned} &(\Psi \circ I)(\mathbf{x}, \mathbf{u}) \\ &= B_0(\mathbf{x}, \mathbf{u}) + \tfrac{\partial B_0}{\partial \mathbf{f}} \left(\tfrac{\partial \mathbf{f}}{\partial D_{t_r}(\mathbf{x}, \mathbf{c})} \Delta D_{t_r}(\mathbf{x}, \mathbf{c}) + \tfrac{\partial \mathbf{f}}{\partial \varepsilon_{t_0}} \varepsilon_{t_0} \right), \end{aligned} \tag{10}$$

where

$$B_0(\mathbf{x}, \mathbf{u}) = (\Psi \circ I)(\mathbf{x}, \mathbf{u})|_{D_{t_r}(\mathbf{x}, \mathbf{c}) = D_0(\mathbf{x}, \mathbf{c}), \mathrm{P}^{\mathbf{c}}_{t_0} = \mathrm{P}^{\mathbf{c}}_0}, \tag{11}$$

Note that $\mathbf{f}$ is motion flow generated by warping function, and ε_{t_0} denotes six-dimensional vector on $\mathfrak{se}(3)$. The partial derivatives related to $D_{t_r}(\mathbf{x}, \mathbf{c})$ and ε_{t_0} are given in [2].

Once it is approximated using $\Delta D_{t_r}(\mathbf{x}, \mathbf{c})$ and ε_{t_0}, (8) can be optimized using IRLS. The resulting $\Delta D_{t_r}(\mathbf{x}, \mathbf{c})$ and ε_{t_0} are incremental values for the current $D_{t_r}(\mathbf{x}, \mathbf{c})$ and $\mathrm{P}^{\mathbf{c}}_{t_0}$, respectively. They are updated as follows:

$$\begin{aligned} D_{t_r}(\mathbf{x}, \mathbf{c}) &= D_{t_r}(\mathbf{x}, \mathbf{c}) + \Delta D_{t_r}(\mathbf{x}, \mathbf{c}), \\ \mathrm{P}^{\mathbf{c}}_{t_0} &= \exp(\varepsilon_{t_0}) \mathrm{P}^{\mathbf{c}}_{t_0}, \end{aligned} \tag{12}$$

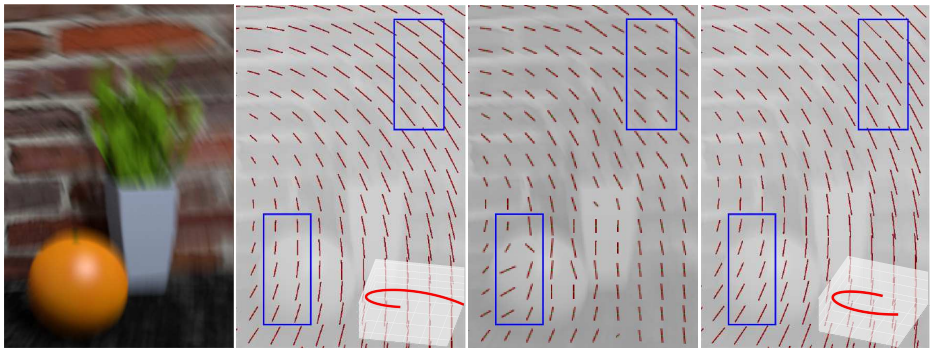
where $\mathrm{P}^{\mathbf{c}}_{t_0}$ is updated through the exponential mapping of the motion vector ε_{t_0}.

Figure 3 shows the initial latent variables and final outputs. After joint estimation, both the latent image and the depth map become clean and sharp.

The proposed blur formulation and joint estimation approach are not limited to the light field but can also be applied to images obtained from a stereo camera or general multi-view camera system. The only property of the light field we use is that sub-aperture images are equivalent to the images obtained from multi-view camera array. Note that the proposed method is not limited to a simple motion path model (moving smoothly in $\mathfrak{se}(3)$ space). More complex parametric curves, such as the Bézier curve used in the prior work [29], can be directly applied only if they are differentiable.

4.3 Initialization

Since deblurring is a highly ill-posed problem and the optimization is done in a greedy and iterative fashion, it is important to start with good initial values.



(a) (b) (c) (d)

Fig. 4: Example of camera motion initialization on a synthetic light field. (a) Blurred input light field. (b) Ground truth motion flow. (c) Sun et al. [30] (EPE = 3.05), (d) Proposed initial motion (EPE = 0.95). In (b) and (d), the linear blur kernels are approximated only using the end points of camera motion for the visualization.

First, we initialize the depth map using the input sub-aperture images of the light field. It is assumed that the camera is not moving and (8) is minimized to obtain the initial $D_{t_r}(\mathbf{x}, \mathbf{c})$. Minimizing (8) becomes a simple multi-view stereo matching problem. Figure 3(c) shows the initial depth map which exhibits fattened object boundary.

Camera motion $\mathrm{P}^{\mathbf{c}}_{t_0}$ is initialized from the local linear blur kernels and initial scene depth. We first estimate the local linear blur kernel of $B(\mathbf{x}, \mathbf{c})$ using [30]. Then, we fit the pixel coordinates moved by the linear kernel and the re-projected coordinates by the warping function as follows:

$$\min_{\mathrm{P}^{\mathbf{c}}_{t_0}} \sum_{i=1}^{N} \| w_{t_0}(\mathbf{x}_i, \mathbf{c}) - l(\mathbf{x}_i) \|_2^2, \tag{13}$$

where $\mathbf{x}_i$ is the sampled pixel position and $l(\mathbf{x}_i)$ is the point that $\mathbf{x}_i$ is moved by the end point of the linear kernel. $\mathrm{P}^{\mathbf{c}}_{t_0}$ is obtained by fitting $\mathbf{x}_i$ moved by $w_{t_0}(\cdot, \mathbf{c})$ and $l(\cdot)$. $\mathrm{P}^{\mathbf{c}}_{t_0}$ is the only variable of $w_{t_0}(\cdot, \mathbf{c})$ since the scene depth is fixed to the initial depth map. In our implementation, RANSAC is used to find the camera motion that best describes the pixel-wise linear kernels. N is the number of random samples, which is fixed to 4.

Figure 4 shows an example of camera motion initialization. It is shown that [30] underestimates the size of the motion (upper blue rectangle) and produces noisy motion where the texture is insufficient (lower blue rectangle).

5 Experimental Results

The proposed algorithm is implemented using Matlab on an Intel i7 7770K @ 4.2GHz with 16GB RAM and is evaluated for both synthetic and real light fields. Our method takes 30 minutes to deblur a single light field. Synthetic light field is generated using *Blender* [3] for qualitative as well as quantitative evaluation. It includes 6 types of camera motion for 3 different scenes in which each light field has 7×7 angular structure of 480×360 sub-aperture images. Synthetic blur is simulated by moving the camera array over a sequence of frames (≥ 40) and then by averaging the individual frames. On the other hand, real light field data is captured using Lytro Illum camera which generates 7×7 angular structure of 552×383 sub-aperture images. We generate the sub-aperture images from light field using the toolbox [4] which provides the relative camera poses between sub-aperture images. Light fields are blurred by moving camera quickly under arbitrary motion, while the scene remains static. In our implementation, we fixed most of the parameters except λ_D such that $\lambda_u = 15, \lambda_c = 1, \lambda_L = 5$. λ_D is set to a larger value for a real light field ($\lambda_D = 400$) than for synthetic data ($\lambda_D = 20$).

For quantitative evaluation of deblurring, we use both peak signal to noise ratio (PSNR) and structural similarity (SSIM). Note that PSNR and SSIM are measured by the maximum (best) ones among individual PSNR and SSIM values computed between the deblurred image and the ground truth images (along the motion path) as adopted in [21]. For comparison with light field depth estimation methods, we use the relative mean absolute error (L1-rel) defined as

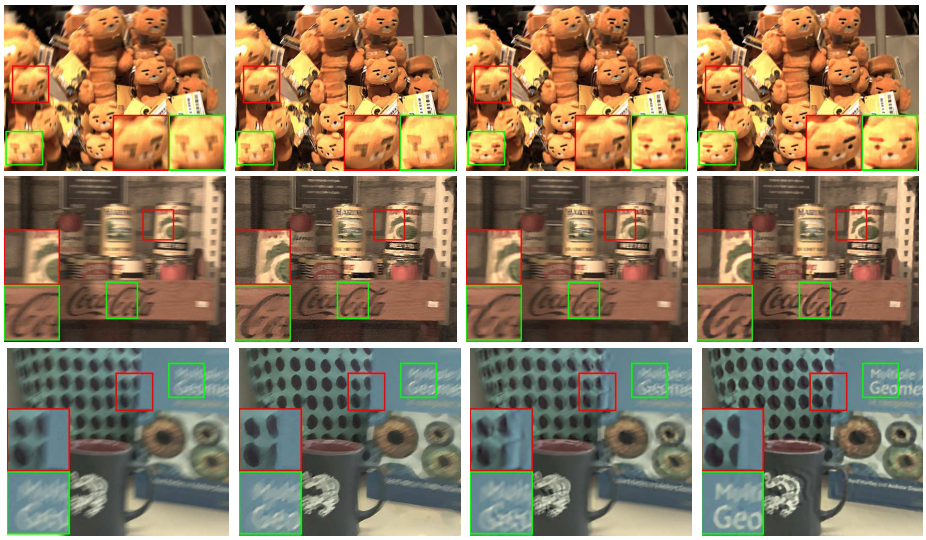
$$\text{L1-rel}(D, \hat{D}) = \frac{1}{n} \sum_i \frac{|D_i - \hat{D}_i|}{\hat{D}_i}, \tag{14}$$

which computes the relative error of the estimated depth $\hat{D}$ to the ground truth depth D. The accuracy of camera motion estimation is measured by the average end point error (EPE) to the end point of ground truth blur kernels. In our evaluation, we compute the EPE by generating an end point of blur kernel using the estimated camera motion and ground truth depth. We compare the performance of the proposed algorithm to linear blur kernel methods that directly computes the EPE between the ground truth and their pixel-wise blur kernel.

5.1 Light Field Deblurring

Real Data. Figure 5 and Figure 6 show the light field deblurring results for blurred real light field with spatially varying blur kernels. In Figure 5, the result is compared with the existing motion deblurring methods [19, 30] which utilize motion flow estimation. It is shown that the proposed algorithm reconstructs sharper latent image better than others. Note that [19, 30] show satisfactory performance only for small blur kernels.

Figure 6 shows the comparison results with the deblurring method based on the global camera motion model [14, 29]. In comparison with [29], we deblur only

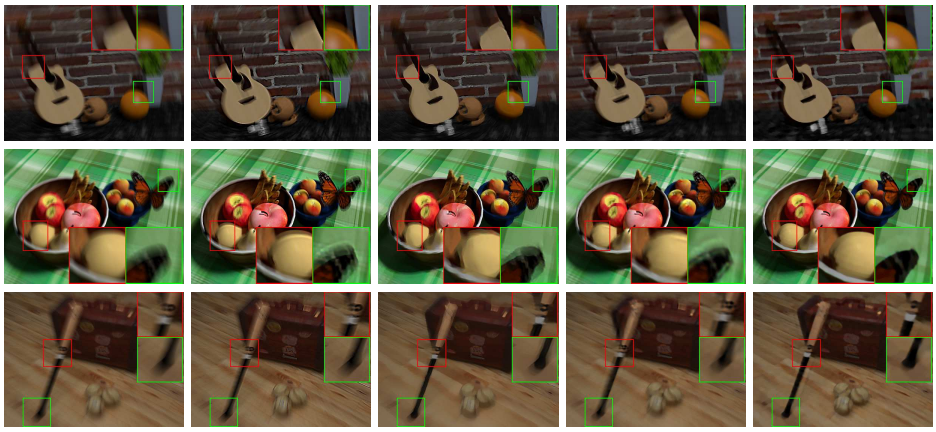


(a) (b) (c) (d)

Fig. 5: Deblurring result for real light field dataset with comparison to local linear blur kernel deblurring methods. (a) Blurred input image. (b) Result of Kim and Lee [19]. (c) Sun et al. [30]. (d) Proposed algorithm.

(a) (b) (c) (d)

Fig. 6: Deblurring result for real light field dataset with comparison to global camera motion estimation methods. (a) Blurred input image. (b) Result of Hu et al. [14]. (c) Srinivasan et al. [29]. (d) Proposed algorithm.



(a) (b) (c) (d) (e)

Fig. 7: Deblurring result for synthetic light field. (a) Blurred input light field. (b) Result of Hu et al. [14]. (c) Kim and Lee [19]. (d) Sun et al. [30]. (e) Proposed algorithm.

cropped regions shown in the yellow boxes of Figure 6(c) due to GPU memory overflow (>12GB) for larger spatial resolution.

[14] assumes the scene depth is piecewisely planar. Therefore, it cannot be generalized to arbitrary scene, yielding unsatisfactory deblurring result. [29] estimates the reasonably correct camera motion of the blurred light field while their output is less deblurred. Note that [29] can not handle the rotational camera motion which produces completely different blur kernels from translational motion. On the other hand, the proposed algorithm fully utilizes the 6-DOF camera motion and the scene depth, yielding outperforming results for the arbitrary scene.

The light field deblurring experiments with real data show that the proposed algorithm works robustly even for the hand-shake motion which does not match the proposed motion path model. The proposed algorithm showed superior deblurring performance for both natural indoor and outdoor scenes, which confirms the robustness of the proposed algorithm to noise and depth level.

Synthetic Data. The performance of the proposed algorithm is evaluated using synthetic light field dataset, as shown in Figure 7 and Table 1. The synthetic data consists of forward, rotation, in-plane translation motion and their combinations. In Figure 7, we visualize and compare the deblurring performance with existing motion flow methods [19, 30] and a camera motion method [14]. In all examples, the proposed algorithm produces sharper deblurred images than others as shown clearly in the cropped boxes.

Table 1 shows the quantitative comparison of deblurring performance by measuring PSNR and SSIM to the ground truth. It shows that the proposed

Table 1: Quantitative evaluation of deblurring on synthetic light field dataset (in PSNR and SSIM).

Methods	Forward		Rotation		Translation		Forward+Rot.		Forward+Tran.		Rot.+Tran.	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Input	20.72	0.740	20.37	0.731	21.82	0.758	19.84	0.723	20.30	0.731	19.79	0.728
Hu et al. [14]	20.00	0.716	19.42	0.704	21.42	0.745	19.18	0.701	19.43	0.699	19.24	0.711
Kim and Lee [19]	20.06	0.721	19.78	0.714	21.42	0.749	19.32	0.706	19.65	0.708	19.34	0.714
Sun et al. [30]	27.69	0.896	27.68	0.881	25.41	0.856	27.40	0.874	27.23	0.899	26.42	0.868
Proposed Method	**29.24**	**0.915**	**29.14**	**0.913**	**26.99**	**0.876**	**28.92**	**0.905**	**28.91**	**0.922**	**27.85**	**0.893**
Input (cropped)	21.01	0.758	21.19	0.746	19.39	0.698	21.73	0.758	21.67	0.782	20.50	0.745
Srinivasan et al. [29]	17.15	0.730	19.02	0.652	16.28	0.620	19.17	0.660	16.20	0.726	16.38	0.626
Proposed Method	**27.15**	**0.871**	**27.32**	**0.870**	**25.30**	**0.836**	**28.83**	**0.904**	**28.01**	**0.901**	**25.88**	**0.867**

joint estimation algorithm significantly outperforms the others. Sun et al. [30] achieves comparable performance to the proposed algorithm in which CNN is trained with MSE loss. Other algorithms achieve minor improvement from the input image because the assumed blur models are simple and inconsistent with the ground truth blur.

For the comparison with [29], we crop the each light field to 200×200 because of the GPU memory overflow. Note that we use the original setting of [29]. [29] shows lower performance than the input blurred light field due to the spatial viewpoint shift as in the output of [29]. Since the original point exists at the end point of the camera motion path in [29], the viewpoint shift occurs when the estimated 3D motion is large. It is observed that this is an additional cause to decrease PSNR and SSIM when the estimated 3D motion is different from the ground truth. The proposed algorithm estimates the latent image with ignorable viewpoint shift because the origin is located in the middle of the camera motion path.

5.2 Light Field Depth Estimation

To show the performance of light field depth estimation, we compare the proposed method with several state-of-the-art methods [7, 16, 31, 32, 34]. For comparison, all blurred sub-aperture images are independently deblurred using [30] before running their own depth estimation algorithms.

Figure 8 shows the visual comparison of estimated depth map generated by different methods, which confirms that the proposed algorithm produces significantly better depth map in terms of accuracy and completeness. Since independent deblurring of all sub-aperture images does not consider correlation between them, conventional correspondence and defocus cue do not produce reliable matching, yielding noisy depth map. Only the proposed joint estimation algorithm results in sharp and unfattened object boundary, and produces the closest result to the ground truth.

Quantitative performance comparison of depth map estimation is shown in Table 2. For three synthetic scenes with three different motion for each scene,

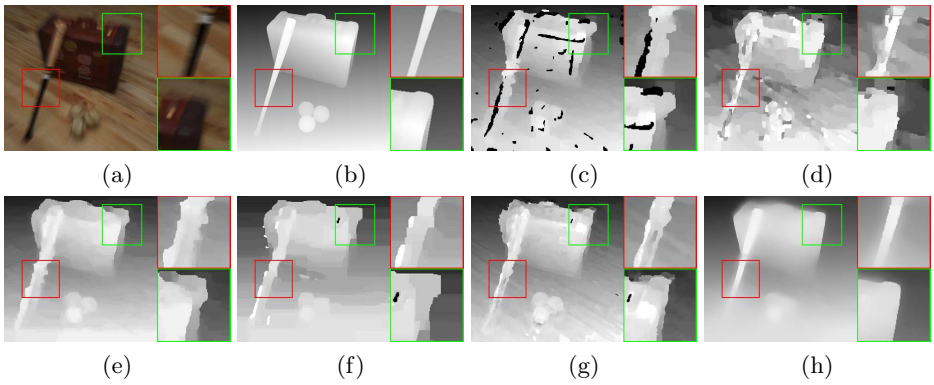


Fig. 8: Depth estimation results on blurred light field. (a) Blurred center sub-aperture image. (b) Ground truth depth. (c) Result of Jeon et al. [16]. (d) Williem and Park [34]. (e) Tao et al. [31]. (f) Wang et al. [32]. (g) Chen et al. [7]. (h) Proposed algorithm.

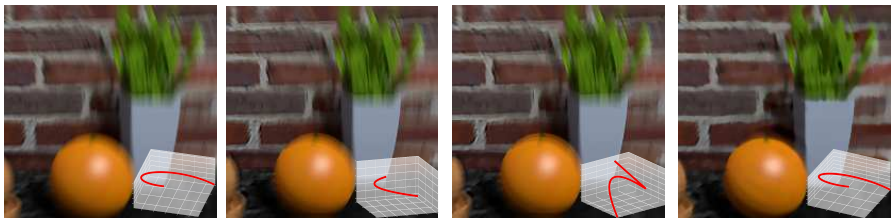
Table 2: Comparison of depth estimation (in average L1-rel error).

Methods	Forward	Rotation	Trans.	Overall
Chen et al. [7]	0.0251	0.0326	0.0331	0.0303
Tao et al. [31]	0.0251	0.0359	0.0371	0.0327
Wang et al. [32]	0.0312	0.0377	0.0400	0.0363
Jeon et al. [16]	0.0835	0.0916	0.0921	0.0891
Williem and Park [34]	0.0615	0.0895	0.0966	0.0825
Proposed Method	**0.0198**	**0.0150**	**0.0243**	**0.0197**

the average L1-rel error of the estimated depth map is computed and compared. The comparison clearly shows that the proposed method produces the lowest error in all types of camera motion. Note that the second best result is achieved by Chen et al. [7], which is relatively robust in the presence of motion blur because bilateral edge preserving filtering is employed for cost computation. The depth estimation experiment demonstrates that solving deblurring and depth estimation in a joint manner is essential.

5.3 Camera Motion Estimation

Table 3 shows the EPE of the estimated motion on synthetic light field dataset. Compared with other methods [19, 30], the proposed method improves the accuracy of the estimated motion significantly. In particular, a large gain is obtained in the rotational motion, which indicates that the rotational motion cannot be modeled accurately as a linear blur kernel used in [19, 30].



(a) (b) (c) (d)

Fig. 9: Deblurring and camera motion estimation result for synthetic light field with comparison to [29]. (a) Input light field and ground truth camera motion. (b) Result of Srinivasan et al. [29] (quadratic). (b) Srinivasan et al. [29] (cubic). (d) Proposed algorithm.

Table 3: Comparison of motion estimation (in EPE).

Methods	Forward	Rotation	Translation
Kim and Lee [19]	2.153	3.317	1.989
Sun et al. [30]	1.492	2.557	1.810
Proposed Method	**0.325**	**0.171**	**0.590**

Figure 9 shows the motion estimation results compared to the ground truth motion. Since the camera orientation changes while the camera is moving, the 6-DOF camera motion can not be recovered properly by [29]. As shown in Figure 9(b) and Figure 9(c), the deblurring results are similar to the input, because the motion can not converge to the ground truth. In contrast, the proposed algorithm converges to the ground truth 6-DOF motion and also produces the sharp deblurring result.

6 Conclusion

In this paper, we presented the novel light field deblurring algorithm that estimated latent image, sharp depth map, and camera motion jointly. Firstly, we modeled all the blurred sub-aperture images by center-view latent image using 3D warping function. Then, we developed the algorithm to initialize the 6-DOF camera motion from the local linear blur kernel and scene depth. The evaluation on both synthetic and real light field data showed that the proposed model and algorithm worked well with general camera motion and scene depth variation.

Acknowledgement. This work was supported by the Visual Turing Test project (IITP-2017-0-01780) from the Ministry of Science and ICT of Korea, and the Samsung Research Funding Center of Samsung Electronics under Project Number SRFC-IT1702-06.

References

1. Beck, A., Teboulle, M.: Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. IEEE Trans. on Image Processing **18**(11), 2419–2434 (2009)
2. Blanco, J.L.: A tutorial on se (3) transformation parameterizations and on-manifold optimization. University of Malaga, Tech. Rep **3** (2010)
3. Blender Online Community: Blender - A 3D modelling and rendering package. Blender Foundation, Blender Institute, Amsterdam (http://www.blender.org
4. Bok, Y., Jeon, H.G., Kweon, I.S.: Geometric calibration of micro-lens-based light-field cameras using line features. In: Proc. of the European Conference on Computer Vision. pp. 47–61 (2014)
5. Chan, T.F., Wong, C.K.: Total variation blind deconvolution. IEEE Trans. on Image Processing **7**(3), 370–375 (1998)
6. Chandramouli, P., Perrone, D., Favaro, P.: Light field blind deconvolution. CoRR **abs/1408.3686** (2014), http://arxiv.org/abs/1408.3686
7. Chen, C., Lin, H., Yu, Z., Bing Kang, S., Yu, J.: Light field stereo matching using bilateral statistics of surface cameras. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1518–1525 (2014)
8. Cho, S., Lee, S.: Fast motion deblurring. In: Proc. of the ACM Trans. on Graphics. vol. 28, p. 145 (2009)
9. Cho, S., Matsushita, Y., Lee, S.: Removing non-uniform motion blur from images. In: Proc. of the IEEE International Conference on Computer Vision. pp. 1–8 (2007)
10. Cho, S., Wang, J., Lee, S.: Video deblurring for hand-held cameras using patch-based synthesis. ACM Trans. on Graphics **31**(4), 64 (2012)
11. Dansereau, D.G., Eriksson, A., Leitner, J.: Richardson-lucy deblurring for moving light field cameras. Proc. of the IEEE Conference on Computer Vision and Pattern Recognition Workshop (2017)
12. Fergus, R., Singh, B., Hertzmann, A., Roweis, S.T., Freeman, W.T.: Removing camera shake from a single photograph. In: Proc. of the ACM Trans. on Graphics. vol. 25, pp. 787–794 (2006)
13. Gupta, A., Joshi, N., Zitnick, C.L., Cohen, M., Curless, B.: Single image deblurring using motion density functions. In: Proc. of the European Conference on Computer Vision. pp. 171–184 (2010)
14. Hu, Z., Xu, L., Yang, M.H.: Joint depth estimation and camera shake removal from single blurry image. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2893–2900 (2014)
15. Hu, Z., Yang, M.H.: Fast non-uniform deblurring using constrained camera pose subspace. In: Proc. of the British Machine Vision Conference. vol. 2, p. 4 (2012)
16. Jeon, H.G., Park, J., Choe, G., Park, J., Bok, Y., Tai, Y.W., Kweon, I.S.: Accurate depth map estimation from a lenslet light field camera. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1547–1555 (2015)
17. Ji, H., Wang, K.: A two-stage approach to blind spatially-varying motion deblurring. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 73–80 (2012)
18. Jin, M., Chandramouli, P., Favaro, P.: Bilayer blind deconvolution with the light field camera. In: Proc. of the IEEE International Conference on Computer Vision Workshop. pp. 10–18 (2015)
19. Kim, T.H., Lee, K.M.: Segmentation-free dynamic scene deblurring. In: Proc. of IEEE Conference on Computer Vision and Pattern Recognition. pp. 2766–2773 (2014)

20. Kim, T.H., Lee, K.M.: Generalized video deblurring for dynamic scenes. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5426–5434 (2015)
21. Köhler, R., Hirsch, M., Mohler, B., Schölkopf, B., Harmeling, S.: Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database pp. 27–40 (2012)
22. Ng, R.: Digital light field photography. Ph.D. thesis, stanford university (2006)
23. Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., Hanrahan, P.: Light field photography with a hand-held plenoptic camera. Computer Science Technical Report **2**(11), 1–11 (2005)
24. Park, H., Lee, K.M.: Joint estimation of camera pose, depth, deblurring, and super-resolution from a blurred image sequence. In: Proc. of the IEEE International Conference on Computer Vision (2017)
25. Scales, J.A., Gersztenkorn, A.: Robust methods in inverse theory. Inverse problems **4**(4), 1071 (1988)
26. Sellent, A., Rother, C., Roth, S.: Stereo video deblurring. In: Proc. of the European Conference on Computer Vision. pp. 558–575 (2016)
27. Shan, Q., Jia, J., Agarwala, A.: High-quality motion deblurring from a single image. In: ACM Trans. on Graphics. vol. 27, p. 73 (2008)
28. Snoswell, A., Singh, S.: Light field de-blurring for robotics applications. In: Australasian Conference on Robotics and Automation (2014)
29. Srinivasan, P.P., Ng, R., Ramamoorthi, R.: Light field blind motion deblurring. Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (2017)
30. Sun, J., Cao, W., Xu, Z., Ponce, J.: Learning a convolutional neural network for non-uniform motion blur removal. In: Proc. of IEEE Conference on Computer Vision and Pattern Recognition. pp. 769–777 (2015)
31. Tao, M.W., Srinivasan, P.P., Malik, J., Rusinkiewicz, S., Ramamoorthi, R.: Depth from shading, defocus, and correspondence using light-field angular coherence. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1940–1948 (2015)
32. Wang, T.C., Efros, A.A., Ramamoorthi, R.: Occlusion-aware depth estimation using light-field cameras. In: Proc. of the IEEE International Conference on Computer Vision. pp. 3487–3495 (2015)
33. Whyte, O., Sivic, J., Zisserman, A., Ponce, J.: Non-uniform deblurring for shaken images. International Journal of Computer Vision **98**(2), 168–186 (2012)
34. Williem, Park, I.K.: Robust light field depth estimation for noisy scene with occlusion. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4396–4404 (2016)
35. Wulff, J., Black, M.J.: Modeling blurred video with layers. In: Proc. of the European Conference on Computer Vision. pp. 236–252 (2014)
36. Xu, L., Jia, J.: Two-phase kernel estimation for robust motion deblurring. In: Proc. of the European Conference on Computer Vision. pp. 157–170 (2010)
37. Xu, L., Jia, J.: Depth-aware motion deblurring. In: Proc. of IEEE International Conference on Computational Photography. pp. 1–8 (2012)
38. Zheng, S., Xu, L., Jia, J.: Forward motion deblurring. In: Proc. of the IEEE International Conference on Computer Vision. pp. 1465–1472 (2013)