Depth From Motion and Optical Blur With an Unscented Kalman Filter

C. Paramanand and A. N. Rajagopalan, *Senior Member, IEEE*

***Abstract*—Space-variantly blurred images of a scene contain valuable depth information. In this paper, our objective is to recover the 3-D structure of a scene from motion blur/optical defocus. In the proposed approach, the difference of blur between two observations is used as a cue for recovering depth, within a recursive state estimation framework. For motion blur, we use an unblurred–blurred image pair. Since the relationship between the observation and the scale factor of the point spread function associated with the depth at a point is nonlinear, we propose and develop a formulation of unscented Kalman filter for depth estimation. There are no restrictions on the shape of the blur kernel. Furthermore, within the same formulation, we address a special and challenging scenario of depth from defocus with translational jitter. The effectiveness of our approach is evaluated on synthetic as well as real data, and its performance is also compared with contemporary techniques.**

***Index Terms*—Blur kernel, depth from defocus (DFD), motion blur, out-of-focus blur, unscented Kalman filter (UKF).**

I. Introduction

TRADITIONALLY, blur in images is regarded as an undesirable effect. The cause for blur can be due to relative motion between the camera and the scene or due to optical defocusing. In scenes with depth variations, the amount of induced blur varies across the image and is related to the 3-D structure of the scene. Depth estimation techniques that use blur as a cue [10] are based on measuring the extent of blur at each image point. In this paper, we develop a recursive filtering approach for estimating depth map of static scenes from motion-blurred and/or optically defocused observations.

Motion blur in images is a commonly observed phenomenon, and it occurs when there is camera jitter or when objects in a scene move while the image is being captured. A large number of approaches exist that have addressed the problem of restoring motion-blurred images [13], [17], [28], [33], [42]. When the scene is of constant depth and the camera undergoes only in-plane translations during exposure, the motion-blurred image can be related to the original reference image through convolution with a point spread function (PSF). In contrast, one needs to resort to space-variant restoration when the extent of blur varies across an image. This can arise when objects in the scene move differently, when the camera motion is not restricted to in-plane translation, or when the scene has significant depth variations [36], [41].

The fact that motion blur carries with it the depth cue has encouraged researchers to attempt to recover 3-D information from motion-blurred observations. Our focus in this paper is on depth estimation and not image restoration. In [15] and [23], this is achieved by determining the extent of smearing of edges. These methods assume that the blurring of edges is uniform and that the scene can be approximated by a set of planar patches. Using a variational approach, joint depth estimation and restoration from space-variant motion-blurred images is attempted in [11] and [30]. Favaro and Soatto [11] consider scenes in which different objects move along different directions. They estimate the motion field, the depth map, and the restored image from motion-blurred observations captured with different exposure times. The motion-blurred images are modeled as solutions of an anisotropic diffusion equation. However, the shape of the PSF is constrained to be Gaussian. In real scenarios, the shape of the PSF can be arbitrary, and the Gaussian model is appropriate only when the extent of blur is small [30], [37]. Sorel and Flusser have proposed a technique to estimate the shape and the restored image from observations that are blurred due to camera shake [30]. They use two images that are captured with different camera shakes. The PSF can be of arbitrary shape and is initially determined from the blurred observations over regions of constant depth. With this PSF, an initial estimate of the depth map is obtained using a window-based technique, and the image and the depth map are simultaneously computed iteratively.

Yet, another rich source of information of depth is optical defocusing. Depth from defocus (DFD) refers to a passive depth estimation technique, which uses optical defocus blur as a cue, and is a well-established area [4], [6], [7], [12]. In DFD, two or more images of the same scene are captured with different camera settings. The observed images undergo blurring according to the scene structure and the camera settings. The depth map is estimated by measuring the spatial extent of blur in the observations. The diffusion-based approach in [12] avoids the problem of image estimation by using the notion of relative blur. Techniques also exist that jointly estimate the focused image and the depth map [4], [9]. The work by Wang and Liang [39] addresses the interesting problem of estimating depth and motion from images that have undergone both motion blur and optical blur. However, they do not consider the variation of motion blur with depth and assume that the velocity of camera remains uniform during exposure. Favaro *et al.* [8] have proposed a technique to recover the original image and scene depth from observations that have undergone space-variant motion and optical blur simultaneously. They model the blurred observations using anisotropic diffusion

Manuscript received November 29, 2010; revised May 04, 2011 and September 26, 2011; accepted November 24, 2011. Date of publication December 14, 2011; date of current version April 18, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Wan-Chi Siu.

The authors are with the Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai 600 036, India (e-mail: paramanand@gmail.com; raju@ee.iitm.ac.in).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2011.2179664

and attempt to recover the structure and the original image. However, the model constrains the shape of the blur kernels of both the sources to be Gaussian. Yang and Schonfield have proposed a technique for estimating the focused image and depth from out-of-focus videos [43]. Their method relates the camera motion with optical blur and does not involve changing of the camera settings during image capture.

In this paper, we propose a recursive state estimation approach for depth estimation of static scenes that can be seamlessly applied to motion-blurred images as well as the defocus scenario. We estimate the relative blur between a pair of images at all locations but avoid the process of restoration. For the case of motion blur, following other works [8], [30], we consider the camera motion to be in-plane translation. Our method uses two blurred observations of a scene, wherein one is treated to be a space-variantly blurred version of the other. It is easy to capture an image with minimal camera motion when the exposure time is less. Hence, one could capture the reference image with a smaller exposure duration and higher ISO, and the blurred observation with a higher exposure duration and lesser ISO [32], [37]. Such a pair can also be obtained by a hybrid camera, which captures videos at two different frame rates [36]. In the DFD scenario, the two blurred observations of a scene are captured with different lens settings leading to the following two situations: 1) One of the images is more blurred than the other at all points [6]; and 2) the observations constitute a Near–far focused pair [12].

We represent the relative blur between the two motion/defocus-blurred observations with a PSF that can be of arbitrary shape. We relate the PSF and depth at a point in the image using a parameter called scale factor. It turns out that the relationship between the pixel intensities of the blurred image and the scale factor is nonlinear. The unscented Kalman filter (UKF) is a recursive state estimation technique, which has found a wide use for handling nonlinear transformations [18]. We treat the scale factor at each pixel as the state that is to be determined from the given observations using the UKF. We represent the state model using a discontinuity adaptive Markov random field (DAMRF) prior [22], which enforces smoothness while also preserving discontinuities in depth. Earlier works have used DAMRF in the context of image super-resolution [34] and shape from defocus [26]. Since the distribution of the DAMRF prior is non-Gaussian, we use importance sampling [24] to estimate the state moments. Based on the blurred observations, the Bayesian estimate of the state is obtained at every pixel using the UKF. The depth of the scene is obtained from the estimated scale factor.

Our formulation provides a platform for even addressing the problem of simultaneous motion and optical blur. This is a practical extension of the ideal DFD scenario in which an observation can undergo both motion blur and defocus blur together (termed as DFD with jitter). Very few techniques exist that can estimate the space-variant shift and blur between the images [5], [27]. This is very relevant within our context because most DFD techniques assume that the blurred observations are spatially registered. However, during image capture, changes in the camera position can lead to unintentional camera shakes resulting in the second image being both motion blurred and defocused with respect to the first. We also extend the proposed technique to handle this challenging situation.

For recovering depth from space-variantly blurred images, the state-of-the-art methods ([30] for motion blur and [12] for optical blur) employ a variational approach. We follow a recursive state estimation approach that involves only local computations and yet is quite accurate. Moreover, the memory requirements for the proposed UKF-based method is parsimonious as computing depth at a pixel requires data only from its neighboring pixels. Our technique can be readily implemented on digital signal processors, which have limited memory. In contrast, in iterative techniques, image intensities and gradients corresponding to all of the image pixels must be computed and stored.

As compared with our earlier work in [25], which addresses depth estimation from motion blur alone, this paper contains several significant additions. The main contributions of this paper can be summarized as follows: 1) Our method is the first to incorporate a recursive state estimation approach based on local computations for recovering depth information from blurred images; 2) we show that the same UKF-based framework can be elegantly tailored to handle both motion and optical defocus situations; 3) using sparsity constraint on the motion blur, we even address the challenging problem of DFD-based depth estimation in the presence of motion blur; 4) there are no restrictions on the form of the PSF; 5) comparisons results are given with respect to relevant prior works; and 6) we also study the performance of our algorithms in the presence of noise and geometric alignment errors.

In Section II, we discuss the motion blur model. We describe the proposed approach for the motion blur scenario in Section III. Depth estimation by recursive filtering is presented in Section IV. The scenarios of optical defocus and DFD with jitter are discussed in Sections V and VI, respectively. Experimental results are given in Section VII. We conclude with Section VIII.

II. Motion Blur Model

Consider a static 3-D scene captured by a moving camera with motion being restricted to in-plane translations. The velocity vector of the camera at time τ is denoted by $\mathbf{T}(\tau) = [T_x(\tau)\ T_y(\tau)\ 0]^T$ with respect to the coordinate system having origin at the optical center of the camera, the X and Y axes are parallel to the image plane, and the Z axis is along the optical axis. Let f denote the image of the scene captured by the camera. Assuming a pin-hole camera model, velocity $\mathbf{v}$ at point (x, y) of f is given [16] by

$$\mathbf{v}(x,y,\tau) = \begin{bmatrix} v_x(x,y,\tau) \\ v_y(x,y,\tau) \end{bmatrix} = \frac{-\nu}{d(x,y)} \begin{bmatrix} T_x(\tau) \\ T_y(\tau) \end{bmatrix} \quad (1)$$

where ν is the "focal length" of projection and $d(x, y)$ is the depth of the image point (x, y). Clearly, the velocity of points that are near to the camera will be higher than those which are farther. The blurred image g is the average of the light intensities observed by the image sensors during the exposure time. Consequently, in the blurred image, the extent of smearing of a scene point nearer to the camera is more. The averaging effect can be written as

$$g(x,y) = \frac{1}{T_e} \int_0^{T_e} f\left(x - \bar{x}(x,y,\tau), y - \bar{y}(x,y,\tau)\right) d\tau \quad (2)$$

where $\bar{x}(x,y,\tau)$ and $\bar{y}(x,y,\tau)$ denote the components of the displacement of the point (x,y) during exposure and T_e is the exposure duration. Ignoring occlusion effects, the blurred image can be related (see [30] and [36]) to the original image through a space-variant PSF as

$$\begin{aligned} g(x,y) &= f *_v h(x,y) \\ &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x-s,y-t)h(x-s,y-t;s,t)dsdt \end{aligned} \tag{3}$$

where $h(x,y,s,t)$ is the PSF and $*_v$ denotes space-variant blurring. The expression for the PSF can be obtained as

$$h(x,y,s,t) = \frac{1}{T_e}\int_0^{T_e} \delta\left(s-\bar{x}(x,y,\tau), t-\bar{y}(x,y,\tau)\right) d\tau \tag{4}$$

where δ indicates the 2-D Dirac delta function. The PSF represents the displacements of the image point (x,y) during exposure (due to camera translation) and is weighted according to the fraction of the exposure time the point stays at the displaced position (any translational shift between f and g is reflected in the PSF).

A. PSF Variation With Depth

Sorel and Flusser noted in [30] that when the camera motion is restricted to in-plane translations, the shape of the PSF $h(x,y,s,t)$ remains the same throughout the image except for a scale factor, which is related to depth. This is due to the fact that the paths traversed by the projected scene points in the image due to camera motion during exposure are similar in shape but are scaled according to depth. Consider the reference point (o_x,o_y) with depth $d(o_x,o_y)=d_o$. If the PSF at (o_x,o_y) is $h_o(o_x,o_y;s,t)$, then the PSF at any other point (x,y) is a scaled version of $h_o(o_x,o_y;s,t)$, and using (4), we get

$$h(x,y;s,t) = k^2(x,y)h_o\left(o_x,o_y;sk(x,y),tk(x,y)\right) \tag{5}$$

where $k(x,y)=(d(x,y)/d_o)$ is a scale factor (a positive real number) and represents the relative depth with respect to the depth at point (o_x,o_y).

In scenes that have objects near to the camera, blur due to in-plane camera translations will be significant. Assume that the camera moves only along the x-axis. If the total displacement of the camera during exposure is X_c, then the corresponding shift at the image point (x,y) will be $(\nu X_c/d(x,y))$ (according to the pin-hole model). Assuming a typical value of $\nu = 400$ pixels, when the depth $d(x,y) = 20$m, even if the camera moves by 1 cm, the length of the blur would be only 0.2 pixels; whereas if $d(x,y) = 20$ cm, for the same camera displacement, the blur length will be as high as 20 pixels.

III. Proposed Approach for Depth From Motion blur

In this section, we briefly discuss the framework we follow to estimate the shape of a scene from two observations such that one of the images can be space-variantly blurred to obtain the other. We consider that the reference image of the scene is captured by a still camera and the second observation is blurred due to translational motion of the camera. The reference image f and the blurred observation g are related to each other through a space-variant PSF h as $g(x,y) = f *_v h(x,y)$. The discretized model is

$$g(i,j) = f *_{v_d} h\,(i,j) + e\,(i,j)$$

where

$$\begin{aligned} f *_{v_d} h(i,j) = &\sum_{m,n} f(i-m,j-n) \\ &\times h(i-m,j-n;m,n) \end{aligned} \tag{6}$$

and e is the zero-mean Gaussian noise with variance σ_e^2.

The initial step of our algorithm is to estimate the shape of the PSF h between the two observations. Towards this end, we crop a small region of constant depth from the two images and estimate the relative blur between these regions. There are many algorithms that can be used for PSF estimation [13], [33], [37]. We employ a least squares approach, which is discussed in the Appendix. The estimated blur kernel is considered as the reference PSF $h_o(,;m,n)$ from which the blur at any other image point (i,j), which is denoted by $h(i,j;m,n)$, can be obtained from the scale factor $k(i,j)$ through (5). From the image formation model of (6), we see that the pixel intensities of the blurred observation g in a neighborhood of (i,j) depend on the intensity values of the reference image f and the space-variant PSF. The 3-D structure can be inferred from the values of k, which determine the PSF at each pixel.

The relationship between the state $k(i,j)$ and the observation $g(i,j)$ is nonlinear. For illustrating this nonlinearity, let us consider the reference kernel $h_o(,;m,n)$ as a Gaussian with standard deviation σ. The PSF $h(i,j;m,n)$ will be a Gaussian with standard deviation $(\sigma/k(i,j))$. If $k(i,j)$ is multiplied by a constant c_k, then the PSF $h(i,j;m,n)$ will have standard deviation $(\sigma/c_k k(i,j))$. From (6) we see that, a change in the value of the scale factor by c_k does not necessarily alter the value of $g(i,j)$ to $c_k g(i,j)$. We propose to use the UKF [18] for handling this nonlinearity.

The UKF works on the principle of the unscented transformation (UT). It is a variant of the Kalman filter and represents the state distribution using a set of optimally chosen sample points. The UT is a deterministic sampling approach that can be used to calculate the statistics of a random variable that undergoes a nonlinear transformation [18]. Consider the n_x-dimensional random vector $\mathbf{x}$. Let $\mathbf{F}: \Re^{n_x} \rightarrow \Re^{n_y}$ be a nonlinear function acting on $\mathbf{x}$, i.e., $\mathbf{y} = \mathbf{F}(\mathbf{x}) \in \Re^{n_y}$. Let $\overline{\mathbf{x}}$ and $\mathbf{P}_x$ denote the mean and covariance of $\mathbf{x}$, respectively. The probability distribution of $\mathbf{x}$ is captured by a set of carefully chosen samples known as sigma points. These sigma points are propagated through the nonlinear function to get the sigma points of $\mathbf{y}$. The moments of $\mathbf{y}$ can be evaluated from these sample points and weighting factors, which are given by the following equations [19]:

$$\begin{aligned} \mathbf{X}_0 &= \overline{\mathbf{x}}; \quad \mathbf{X}_i = \overline{\mathbf{x}} + \left(\sqrt{(n_x+\lambda)\mathbf{P}_x}\right)_i, \quad i = 1,\ldots,n_x \\ \mathbf{X}_i &= \overline{\mathbf{x}} - \left(\sqrt{(n_x+\lambda)\mathbf{P}_x}\right)_i, \quad i = n_x+1,\ldots,2n_x \\ w_0^{(m)} &= \frac{\lambda}{(n_x+\lambda)} \quad w_0^{(c)} = w_0^{(m)} + (1-\alpha_{\mathrm{UT}}^2+\beta_{\mathrm{UT}}) \\ w_i^{(m)} &= w_i^{(c)} = \frac{1}{2(n_x+\lambda)}, \quad i = 1,\ldots,2n_x \\ &\text{where } \lambda = \alpha_{\mathrm{UT}}^2(n_x) - n_x. \end{aligned} \tag{7}$$

Here, $\mathbf{X}_i$ denotes the ith sigma point, and $w_i^{(c)}$ and $w_i^{(m)}$ denote its weights for evaluating the mean and covariance, respectively. Including $\mathbf{X}_0$, the number of sigma points is $2n_x+1$. The term $(\sqrt{(n_x+\lambda)\mathbf{P}_x})_i$ is the ith column of the matrix square root $(\sqrt{(n_x+\lambda)\mathbf{P}_x})$ which is computed using Cholesky factorization [18]. The parameter α_{UT} determines the spread of the sigma points around the mean, and β_{UT} is used to incorporate prior knowledge of the distribution of $\mathbf{x}$ [40].

The nonlinear mapping is applied on each $\mathbf{X}_i$ to get $\mathbf{Y}_i = \mathbf{F}(\mathbf{X}_i)$, where $i = 0, 1, \ldots, 2n_x$. The mean and covariance of $\mathbf{y}$ can be estimated as

$$\overline{\mathbf{y}} = \sum_{i=0}^{2n_x} w_i^{(m)} \mathbf{Y}_i, \quad \mathbf{P}_y = \sum_{i=0}^{2n_x} w_i^{(c)} (\mathbf{Y}_i - \overline{\mathbf{y}})(\mathbf{Y}_i - \overline{\mathbf{y}})^T. \quad (8)$$

The sigma points completely capture the distribution of $\mathbf{x}$ up to the second moment. This results in the estimates of the moments of $\mathbf{y}$ being accurate up to the second order of the Taylor series expansion for any nonlinear function [18].

The statistics of the observation are obtained by propagating the sigma points using the principle of UT. The performance of the UKF has been proven to be much better when compared with the extended Kalman filter for nonlinear systems [18]. We formulate the scale factor computation as a recursive state estimation problem with $k(i,j)$ as the state of the UKF. The scale map (k evaluated at all pixels) represents the shape of the scene. The absolute depth d can be obtained from k at all pixels if the depth is known at any one image point.

IV. Recursive Depth Estimation

To facilitate recursive filtering, we locally approximate the blurring model of (6) as

$$g(i,j) = \sum_{m,n} f(i-m, j-n) h(i,j;m,n) + e(i,j). \quad (9)$$

According to this model, the PSF in the neighborhood of (i,j) is assumed to be the same as the PSF at (i,j). The blurred image intensity $g(i,j)$ is dependent on the reference image f, the reference kernel $h_o(,;m,n)$, and the scale factor $k(i,j)$. Initially, we estimate the reference PSF $h_o(,;m,n)$ using image patches of f and g that correspond to a region of constant depth. The scale factor $k(i,j)$ of this reference blur kernel is considered as the state at pixel (i,j). Its mean and variance (denoted by $\mu_{k(i,j)}$ and $\mathbf{P}_{k(i,j)}$, respectively) are propagated through the system and observation models. Initially, the moments of the state are predicted using the knowledge of previously estimated states. According to the measurement model, the observation mean and its covariance are predicted using predicted state mean and variance, $h_o(,;m,n)$, and f, respectively, through UT. Based on the observation g, the estimate of the state is updated.

A. Prediction of State Moments

For the system model, we use a DAMRF prior [22]. We use a statistical model for the prior as it does not restrict the neighborhood dependence to be linear. The DAMRF prior helps in incorporating smoothness and is also good at preserving discontinuities. The previously estimated states are related to the current state $k(i,j)$ through a DAMRF prior with a nonsymmetric half-plane support (NSHP) [35]. Let $\bar{k}_{i,j}$ denote the first-order NSHP neighbors of $k(i,j)$, i.e., $\bar{k}(i,j) = \{k(i,j-1), k(i-1,j-1), k(i-1,j), k(i-1,j+1)\}$. The conditional pdf of $k(i,j)$ takes the following form:

$$P\left(k(i,j)/\bar{k}(i,j)\right) = \frac{1}{Z}\exp\left(-\gamma \log\left(1 + \frac{\eta^2\left(k(i,j), \bar{k}(i,j)\right)}{\gamma}\right)\right) \quad (10)$$

where γ controls the convexity of the DAMRF prior [22], Z is a normalization constant, and

$$\eta^2\left(k(i,j), \bar{k}(i,j)\right) = \frac{1}{\rho^2(i,j)} \sum_{k \in \bar{k}(i,j)} \left(k(i,j) - k\right)^2. \quad (11)$$

Parameter $\rho^2(i,j)$ controls the variations of $k(i,j)$ with respect to its neighbors. The mean $\widehat{\mu}_p$ and variance $\widehat{\sigma}_p^2$ of the conditional pdf correspond to the predicted mean and variance based on the previously estimated scale factors and are denoted by $\mu_{k(i,j)/(i,j-1)}$ and $\mathbf{P}_{k(i,j)/(i,j-1)}$, respectively. The conditional density function is non-Gaussian, and it is not straightforward to estimate its mean and variance. To estimate these moments, at each pixel, we employ importance sampling, a Monte Carlo method [24].

Importance sampling is a technique for estimating the properties of a target distribution from the samples generated by a different distribution rather than the distribution of interest [24]. Consider distribution $q(z)$ from which it is easy to generate samples. The support of $q(z)$ should include the support of $P(z)$. Such density $q(z)$ is called the sampler density. The expected value of function $\phi(z)$ under distribution $P(z)$ can be written as

$$E_P\left[\phi(z)\right] = \int \phi(z)\left(\frac{P(z)}{q(z)}\right) q(z) dz = E_q\left[\phi(z)\left(\frac{P(z)}{q(z)}\right)\right]. \quad (12)$$

We predict the state moments with the Cauchy sampler function q. For an accurate estimation of the moments, the location parameter (point at which the peak of the distribution occurs) of the sampler density μ_q and the scale parameter (which specifies the width of the distribution) have to be chosen appropriately. The average of the previously estimated scale factors in the NSHP $\mu_{\bar{k}} = (1/4)(k(i,j-1) + k(i-1,j-1) + k(i-1,j) + k(i-1,j+1))$ is assigned to the location parameter μ_q. The scale parameter is chosen in such a way that the support of q is wide enough to include the support of P. The samples from the Cauchy sampler are denoted as $\{z^l\}, l = 1, 2, \ldots, L$. These samples are weighted using $w^l = (p(z^l)/q(z^l))$. The mean $\widehat{\mu}_p$ and variance $\widehat{\sigma}_p^2$ of p are computed as

$$\widehat{\mu}_p = \frac{\sum_{l=1}^{L} w_l z^{(l)}}{\sum_{l=1}^{L} w_l}, \quad \widehat{\sigma}_p^2 = \frac{\sum_{l=1}^{L} w_l \left(z^{(l)} - \widehat{\mu}_p\right)^2}{\sum_{l=1}^{L} w_l}. \quad (13)$$

These formulas hold even when the functional forms of $p(z)$ and $q(z)$ are known only up to a multiplication constant.

B. Observation Model

The blurred image intensity $g(i,j)$ depends on the scale factor $k(i,j)$ through (9). One can regard $k(i,j)$ as the state and the pixel intensity $g(i,j)$ as the observation and use (9) as the measurement model for state estimation. However, for accurately estimating the state $k(i,j)$, in addition to $g(i,j)$,

we consider the blurred intensities at points in the neighborhood of (i,j) as the observation. The scale factor $k(i,j)$ represents the PSF at (i,j) and it affects those pixels around (i,j) that correspond to the location of the peaks of the scaled PSF. Consequently, the blurred image at these locations contain information about $k(i,j)$. We scale the reference kernel $h_o(,;m,n)$ by the scale factor $\mu_{k(i,j)/(i,j-1)}$ and place it around (i,j). Depending on where the peaks of this PSF occur, we pick n_o blurred image pixels around (i,j) including $g(i,j)$ and regard it as the observation $\mathbf{g}(i,j)$. With the knowledge of f and $h_o(,;m,n)$, the nonlinear relationship between the state and the observation can be written as

$$\mathbf{g}(i,j) = H_{i,j}\left(k(i,j)\right) + e(i,j). \tag{14}$$

The nonlinear operation $H_{i,j}$ denotes both the scaling of the reference PSF by $k(i,j)$ and the generation of blurred image intensities from this scaled PSF and the reference image. We have found that $n_o = 4$ typically suffices.

C. State Update

The observation mean $\mu_{\mathbf{g}(i,j)/(i,j-1)}$ and covariance $\mathbf{P}_{\mathbf{g}(i,j)/(i,j-1)}$ are predicted from the moments of the state and the observation model using UT. In order to obtain the sigma points of the observation noise, we augment the state $k(i,j)$ with n_o-independent observation noise terms to get an n_o+1 dimensional vector $\mathbf{x}$ on which UT is applied [18]. The sigma point selection scheme [see (7)] is applied as follows:

$$\begin{aligned}
\overline{\mathbf{x}}^a_{(i,j)/(i,j-1)} &= [\widehat{\mu}_p\ 0 \dots\ \dots 0]^T \\
\mathbf{P}^a_{(i,j)/(i,j-1)} &= \operatorname{diag}\left(\widehat{\sigma}_p^2\ \sigma_e^2 \dots\ \sigma_e^2\right) \\
\mathbf{X}^a_{(i,j)/(i,j-1)} &= \Big[\overline{\mathbf{x}}^a_{(i,j)/(i,j-1)}\ \overline{\mathbf{x}}^a_{(i,j)/(i,j-1)} \\
&\qquad \pm\sqrt{(n_a+\lambda)\mathbf{P}^a_{(i,j)/(i,j-1)}}\Big] \\
&\qquad \text{for } n_a = 1,\dots n_o+1.
\end{aligned}$$

Here, $\mathbf{X}^a$ is the augmented sigma point matrix. Let $\mathbf{X}^a = [(\mathbf{X}^k)\ (\mathbf{X}^e)]^T$ where $\mathbf{X}^k$ and $\mathbf{X}^e$ contain the $2n_o+3$ sigma points of state $k(i,j)$ and measurement noise, respectively. We propagate each of these sigma points through the measurement model given in (14) to obtain the sigma points of the observation, which is denoted by $\mathbf{y}$, i.e.,

$$\mathbf{y}_l = H_{i,j}\left(x_l^k\right) + \mathbf{x}_l^e, \quad l = 1\dots 2n_o+3 \tag{15}$$

where x_l^k and $\mathbf{x}_l^e$ denote the lth columns of $\mathbf{X}^k$ and $\mathbf{X}^e$, respectively. Note that, at pixel (i,j), for each sigma point, the reference PSF $h_o(,;m,n)$ is scaled by a factor x_l^k. The sigma point of the observation $\mathbf{y}_l$ is obtained through the observation model using this scaled PSF and the intensity values of the reference image f. From the sigma points $\mathbf{y}$ and the weights, the statistics $\overline{\mathbf{y}}$ and $\mathbf{P}_y$ that correspond to $\mu_{\mathbf{g}(i,j)/(i,j-1)}$ and $\mathbf{P}_{\mathbf{g}(i,j)/(i,j-1)}$, respectively, are estimated by (8). The state-measurement cross-covariance $\mathbf{P}_{xy}$ is obtained as

$$\mathbf{P}_{xy} = \sum_{l=0}^{2n_o+3} w_l^{(c)}\left(x_l^k - \widehat{\mu}_p\right)\left(\mathbf{y}_l - \overline{\mathbf{y}}\right)^T. \tag{16}$$

Based on the observation $\mathbf{g}(i,j)$, the Bayesian estimate of the state conditional mean and its variance are obtained from the Kalman filter update equations as

$$\begin{aligned}
\mu_{k(i,j)} &= \mu_{k(i,j)/(i,j-1)} + \mathbf{K}_{(i,j)}\left(\mathbf{g}_{(i,j)} - \mu_{\mathbf{g}(i,j)/(i,j-1)}\right) \\
\mathbf{P}_{k(i,j)} &= \mathbf{P}_{k(i,j)/(i,j-1)} - \mathbf{K}_{(i,j)}\mathbf{P}_y\mathbf{K}^T_{(i,j)}
\end{aligned} \tag{17}$$

where the Kalman gain $\mathbf{K}_{(i,j)}$ is given by $\mathbf{K}_{(i,j)} = \mathbf{P}_{xy}\mathbf{P}_y^{-1}$. The updated mean $\mu_{k(i,j)}$ is regarded as the estimated scale factor $k(i,j)$. In the next step, $k(i,j+1)$ is predicted similarly, and the variance $\mathbf{P}_{k(i,j)}$ is used as the parameter $\rho^2(i,j+1)$ in the prior.

The DAMRF prior not only incorporates spatial smoothness but also enables automatic adaptation to sudden changes in depth. The state conditional distribution is tailored such that even those sample values at the tail end of the distribution are encouraged (i.e., they are not entirely penalized). The parameter ρ of the DAMRF [see (11)] is adaptively varied in accordance with the updated posterior error variance $\mathbf{P}_k$. When ρ is high, the predicted mean is at liberty to be significantly different from its neighbors. This is naturally facilitated by the peakiness of the posterior error variance near depth discontinuities (as revealed subsequently by the results in the experimental section). On the other hand, a low value of ρ tends to have the opposite effect of enforcing a smooth estimate. In the update stage, the value of the predicted state is eventually rectified based on the observation error. In summary, the proposed filtering scheme incorporates smoothness, estimates scale factors that match the observation, and promotes discontinuity preservation.

D. Color Observations

When color images are present, the intensities of the RGB channels of the blurred image can be incorporated as additional observations in our recursive filtering framework. Each channel of the blurred image g is modeled as a space-variantly blurred version of the corresponding color channel of the reference image f. The space-variant PSF is considered to be the same for all the color channels. Hence, for estimating the scale factor at a pixel, the state remains the same for all channels but we now have more observations. In the observation $\mathbf{g}(i,j)$, we stack n_o pixels from each of the three channels of the blurred image. Since the number of observations is more, state estimation can be done more accurately. For the case of depth estimation from out-of-focus blur and simultaneous blur (which are discussed later in this paper), color images are used in a similar manner.

The prediction of state mean and variance using importance sampling can be computed in $O(L)$ time [see (13)]. The generation of the state sigma points involves $((n_o+1)^3/6)$ multiplications and $((n_o+1)^2/2)$ additions for square root computation. While computing the sigma points of the observation [see (15)], for each sigma point, we need to generate an observation having n_o pixels. If the blur kernel is of size $N\times N$, this involves N^2 multiplications and additions. Since this step dominates the computation time in our algorithm, the overall time complexity per pixel is approximately $O(n_o^2N^2)$.

V. Defocus Blur

An optically defocused image g can be expressed in terms of the space-variant PSF $h(x,y,s,t)$ and the focused image f in much the same way as the motion-blurred image [see (3)] [4],

i.e., $g(x,y) = f *_v h(x,y)$. In this case, f represents the image of the scene captured in such a manner that it is in focus at all points (pin-hole equivalent). Consider two defocus-blurred images of a scene g_1 and g_2, which are captured with different focus settings. If the only variation is the change in lens aperture, then one of the blurred observations (i.e., g_2) will be more blurred than the other (i.e., g_1) at all points of the image. Consequently, the two images can be related as $g_2(x,y) = g_1 *_v h_r(x,y)$ using the notion of relative blur [6], [12]. The existence of the blur kernel h_r is not tacitly assumed but is estimated by solving a least squares constrained optimization problem (which is discussed in the Appendix).

For a point object at distance (depth) d from the lens, the blur radius in the image plane r_b is given [4] by

$$r_b = r_o v_o \left(\frac{1}{F_l} - \frac{1}{v_o} - \frac{1}{d}\right) = r_o v_o \left(\frac{1}{u} - \frac{1}{d}\right) \tag{18}$$

where r_o denotes aperture radius, v_o is the distance between the lens and image plane, F_l denotes the focal length, and u denotes the working distance.

DFD algorithms typically model the defocus kernel as Gaussian or pillbox blur [10]. However, in practical scenarios, the shape of the out-of-focus blur PSF need not assume a parametric form [20], [29]. For extracting depth from blurred images, the PSF has to be related to the scene depth. One way to achieve this is to create a lookup table relating a depth value with a PSF as part of camera calibration [6]. In this paper, we do not restrict the PSF to any functional form. However, following the approach in [31], we consider that the shape of the defocus blur kernel remains the same except for a spatial scaling. The scale factor is related to the blur radius and thereby to scene depth. For the out-of-focus blur of the pillbox model and the Gaussian model, we note that the scaling of the PSF happens inherently.

A. Depth From Relative Defocus Blur

We follow an approach similar to that of motion blur (discussed earlier in Section III) for recursively estimating DFD images i.e., the depth is estimated through the scale factor of the relative blur between the observations g_1 and g_2. Let us assume that the reference relative blur kernel h_{r_o} is known at the depth value d_o. If a scene point is away from the point of focus, the spatial extent of the blur kernel is relatively large. Conversely, the blur extent is small when the scene point is focused. Hence, we consider that the scale factor of the relative blur kernel is inversely proportional to the blur radius. Using this, the scale factor of the PSF at point (x_1, y_1) having the depth value $d(x_1, y_1)$ can be written as

$$k(x_1, y_1) = \frac{\left(\frac{1}{u} - \frac{1}{d_0}\right)}{\left(\frac{1}{u} - \frac{1}{d(x_1,y_1)}\right)}. \tag{19}$$

In the case of motion blur, the scale factor of the PSF was directly proportional to the actual depth of the point. Here, the scale factor depends on the working distance and the depth of the reference point.

B. Near–Far Focusing

While capturing a DFD pair, the working distance of the camera can be changed such that, in one image, the objects near to the camera will be more focused while in the other image, farther objects will be more focused. Consider that we have two defocus-blurred observations g_1 and g_2 captured with different working distances. Let Ω_1 and Ω_2 denote the support where the images g_1 and g_2 are more focused, respectively. In the pixel locations within Ω_1, the relationship between the observations can be written as $g_2(x,y) = g_1 *_v h_r(x,y)$, and similarly in Ω_2, $g_1(x,y) = g_2 *_v h_r(x,y)$ holds. At a pixel, since we do not know *a priori* whether an image is to be used as the reference image or as the blurred observation, we recursively estimate twice by considering each of the two images as the reference, i.e., one at a time. Depending on the estimation error, we select the correct scale factor. At pixel (i,j), g_1 is initially considered as the reference image and g_2 as the blurred observation, and the scale factor $k_1(i,j)$ is estimated. Then, g_2 is considered as the reference to estimate $k_2(i,j)$. The norm of the error term $(\mathbf{g}_{(i,j)} - \mu_{\mathbf{g}(i,j)/(i,j-1)})$ is evaluated for both cases. If the norm of the error is lesser for the case of $k_1(i,j)$ than for that of $k_2(i,j)$, then the pixel belongs to Ω_1, and $k_1(i,j)$ is regarded as the true scale factor. Otherwise, (i,j) belongs to Ω_2, and k_2 is chosen as the scale factor with a negative sign. The sign of the scale factor indicates which image is to be considered as the reference.

VI. DFD With Motion Jitter

The DFD technique discussed earlier is applicable only in the ideal case wherein the camera is kept still while capturing the observations. However, if there is camera jitter during image capture, then in addition to the shift in the position of scene points in the image, there will be camera motion blur induced as well. In this section, we consider this challenging situation of DFD. Let g_1 and g_2 denote two observations captured with different camera focus settings such that g_2 is more blurred than g_1. Using PSF h_f that denotes the relative out-of-focus blur, we can write $g_2(x,y) = g_1 *_v h_f(x,y)$. Let g_{2_s} denote the second observation captured under the influence of camera shake. Note that g_{2_s} is both motion blurred and defocus blurred with respect to g_1 (in a space-variant manner). Image g_{2_s} can then be modeled as

$$g_{2_s}(x,y) = g_2 *_v h_m(x,y) = (g_1 *_v h_f) *_v h_m(x,y) \tag{20}$$

where h_m denotes the space-variant motion blur PSF.

In planar regions, the blurring operators commute. In a region of constant depth (i.e., d_0), blur h_{r_0} between the image patches from g_1 and g_{2_s} will be a combination of motion and defocus blur kernels. i.e., $h_{r_0} = h_{m_0} * h_{f_0}$, where $*$ denotes convolution, and h_{m_0} and h_{f_0} are the motion and defocus blur kernels, respectively. The PSF at any other depth will be a convolution of the scaled motion and scaled optical blur kernels. In Section II, we saw that the motion blur scale factor at a point is the ratio of the actual depth of the point and the reference depth. In the case of optical blur, this scale factor is different as it is dependent on the depth at a point as well as the camera settings [evident from (19)]. Hence, to relate the depth and extent of blurring, the two blur sources have to be considered separately, and their shapes have to be estimated. At an image point, both the motion and optical blur scale factors are governed by its depth value. Using the shape of the motion and defocus blur kernels (estimated at a reference depth), we estimate the actual depth

through the UKF-based recursive state estimation, as discussed before.

A. Decoupling Motion Blur and Defocus Blur

The blur kernel h_{r_0} can be obtained from many combinations of motion and defocus blur kernels. For instance, the motion blur can be an impulse, and the optical blur can be h_{r_0} itself, or vice versa. To make the blur decoupling problem well posed, we assume that the motion blur is sparse in nature and that the shape of the optical blur kernel is known. The sparsity assumption is based on the fact that, during camera shake, the camera undergoes only a few translations out of all possible translations [38]. The defocus blur shape is dependent on the camera lens properties, and it can be determined before hand as part of camera calibration. We assume that the mapping between a depth value and its corresponding optical blur kernel is known *a priori* (for a particular lens settings). In our implementation, we estimate the out-of-focus blur kernel $h_{\rm def}$ from images of a planar object kept at a known depth using the blur estimation technique given in the Appendix. The optical blur kernel at any other depth for the same lens settings can be obtained by scaling the known kernel $h_{\rm def}$ [the value of the scale factor should be according to (19)]. These steps can be avoided if we assume a Gaussian form for the optical blur. In this case, the blur kernel can be directly related to the scene depth for the given lens settings.

The entropy of the blur kernel $H_{\rm etrp}$ has been used as a measure of sparsity in [38]. More the number of nonzero elements in h_{m_0}, greater will be its entropy. We follow the approach given in [38] to evaluate the entropy of a blur kernel. Let g_{1_0} and $g_{2_{s_0}}$ denote the reference and blurred image patches at depth d_0, respectively. Let $h_{f_0}(k_f)$ be the scaled version of the known defocus kernel $h_{\rm def}$ by a factor k_f. The value of k_f corresponding to the depth d_o needs to be determined. For a set of values of k_f, we blur g_{1_0} by convolving it with $h_{f_0}(k_f)$ to get an intermediate image $g_{2_0}(k_f)$. For each value of k_f, the blur between $g_{2_0}(k_f)$ and the patch $g_{2_{s_0}}$ is estimated as the motion blur kernel $h_{m_0}(k_f)$ through the technique discussed in the Appendix. Among these estimated motion blur kernels, we choose that value of k_f that not only returns a sparse motion blur kernel but also reduces the data error. Let $E_{\rm RMS}(h_{m_0}(k_f))$ denote the RMS value of the error term between the image observation patch and the output of convolution $(g_{2_{s_0}} - g_{2_0}(k_f) * h_{m_0}(k_f))$. We choose that k_f and the corresponding $h_{m_0}(k_f)$, which minimize the total cost $E_{\rm tot}(h_{m_0}(k_f))$ defined as

$$E_{\rm tot}\left(h_{m_0}(k_f)\right) = E_{\rm RMS}\left(h_{m_0}(k_f)\right) + \varepsilon H_{\rm etrp}\left(h_{m_0}(k_f)\right) \tag{21}$$

where ε is a constant to weigh the sparsity cost. When the scaled defocus blur $h_{f_0}(k_f)$ is of a larger extent than the true defocus blur kernel at the reference depth, $E_{\rm RMS}(h_{m_0}(k_f))$ does not get minimized. Consequently, in (21), those values of k_f lead to high $E_{\rm RMS}(h_{m_0}(k_f))$. When the extent of $h_{f_0}(k_f)$ is smaller than the true defocus blur, the minimization of $E_{\rm RMS}(h_{m_0}(k_f))$ leads to the motion blur kernel $h_{m_0}(k_f)$ that is not sparse. Hence, the optimum k_f approximately leads to the true defocus blur component between patches g_{1_0} and $g_{2_{s_0}}$, whereas the corresponding $h_{m_0}(k_f)$ represents the motion blur component. The value of d_0 is obtained from the optimum k_f, the lens settings, and the known value of the depth corresponding to $h_{\rm def}$. Thus, using g_{1_0} and $g_{2_{s_0}}$, we estimate the reference depth d_0, the optical blur h_{f_0} (which is defined by the optimum k_f), and the motion blur h_{m_0}. While evaluating $E_{\rm tot}(h_{m_0}(k_f))$ in (21), initially, k_f is coarsely varied to get an approximate estimate. The optimum value is found subsequently by varying k_f in fine steps.

B. Depth Estimation

We consider g_1 as the reference image and g_{2_s} as the blurred observation. At each pixel (i,j), the blurred image intensity is modeled using (9). In this case, $h_r(i,j,;)$ the blur kernel at (i,j), is regarded as being composed of motion and optical blur kernels. i.e., $h_r(i,j,;) = h_m(i,j,;) * h_f(i,j,;)$. The blur kernels $h_m(i,j,;)$ and $h_f(i,j,;)$ are scaled versions of h_{m_0} and h_{f_0}, respectively. The depth value $d(i,j)$ which caused this scaling is to be estimated. We follow an approach similar to the case of motion/optical blur. The motion blur scale factor $k(i,j)$ which denotes the relative depth is regarded as the state which is recursively estimated using UKF. Since the value of d_0 is known, the absolute depth $d(i,j)$ can be related with the state as $d(i,j) = d_0 \cdot k(i,j)$. We follow the procedure discussed in Section IV-A for the prediction of state moments. The sigma points of the state are then propagated through the observation model to get observation sigma points. For each sigma point x_l^k of the state $k(i,j)$, the blur kernel $h_r(i,j,;)$ is the convolution of the scaled motion and optical blur kernels. The motion blur component is obtained by scaling h_{m_0} by the factor x_l^k. The value of the depth corresponding to the sigma point x_l^k is $d_0 \cdot x_l^k$. The scaled defocus blur kernel that results from this depth value is chosen. The motion and optical blur kernels are convolved, and the resultant kernel is used along with the intensities of the reference image g_1 to get the sigma points of the blurred observation according to (15). The observation mean and covariance are predicted from its sigma points. Based on the pixel intensities of the blurred image g_{2_s}, the estimates of the moments of the state $k(i,j)$ at each pixel are recursively updated through the UKF [see (17)].

VII. Experimental Results

We tested the proposed approach on several synthetic and real examples for each of these cases: depth from motion blur, DFD blur, and DFD with jitter. Except in the DFD experiment in which we use the data set of [12], in all the other experiments, we use color images since they are available by default. The values of the parameters used in our algorithm were as follows: $\alpha_{\rm UT} = 1$, $\beta_{\rm UT} = 2$, $\gamma = 5.2$, and $L = 100$. The values for UT parameters $\alpha_{\rm UT}$ and $\beta_{\rm UT}$ were chosen as suggested in [40]. The observation noise level σ_e was chosen to be 2 (after some trial and error) for the camera used in our real experiments.

To implement the scaling $h(i,j;m,n) = k^2(i,j)h_o(mk(i,j), nk(i,j))$ for blur kernels defined on discrete pixel locations, we follow a resampling approach. The scale factor $k(i,j)$ is written as $k(i,j) = (n_1/n_2)$, where n_1 and n_2 are integers. The blur kernel is initially upsampled by a factor of n_2 by inserting zeros followed by low-pass filtering. The upsampled kernel is then downsampled by a factor of n_1 by averaging to get the scaled blur kernel. When $k(i,j) < 1$, the blur kernel effectively undergoes upsampling, and this can lead to numerical errors in the estimated scaled kernel. Hence, to avoid upsampling by large factors, we typically estimate the reference blur kernel in a region near to the camera (which corresponds to a large support).

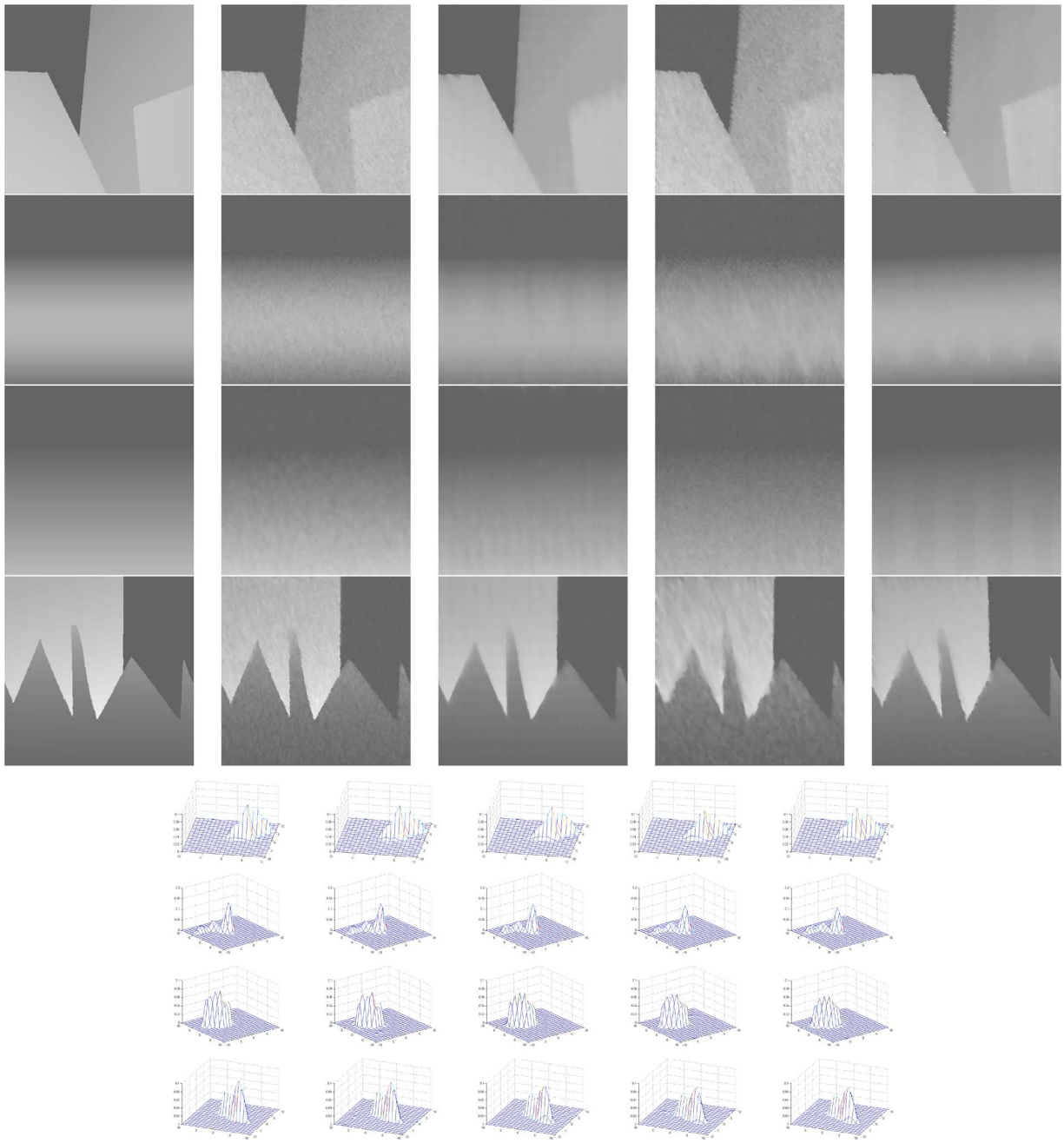

Fig. 1. Rows 1–4, first column: true depth maps; second and fourth columns: depth estimates when noise standard deviation was 10 for motion and simultaneous blur scenarios, respectively; third and fifth column: depth maps when alignment error was 2° for motion and simultaneous blur scenarios, respectively. Rows 5–8, first column: true blur kernels; second and fourth columns: estimated blur kernels when noise standard deviation was 10 for motion and simultaneous blur scenarios, respectively; third and fifth column: estimated blur kernels for motion and simultaneous blur scenarios, respectively, when misalignment was of 2°.

We initially applied the proposed approach to estimate depth from motion blur using synthetic images. We considered four different scenes, each with a different set of a reference image, a depth map, and a blur kernel (see Fig. 1). Calf, herringbone and grass (two different types) textures from [2] were used as reference images. To each set of a reference and blurred image pair, we added Gaussian noise of standard deviation 5 and 10. From image patches of these observations, we estimated the blur kernels using our PSF estimation method. The estimated blur kernels shown in the second column (rows 5–8) of Fig. 1 match closely with the true blur kernels. Using the estimated PSFs, the proposed UKF-based approach was applied on the noisy observations to determine the scale factors. We evaluated the error in estimation ERR using formula

$$\mathrm{ERR} = \sqrt{\mathrm{Avg}[((\hat{k} - k)/k)^2]},$$

where $\mathrm{Avg}[]$ denotes averaging over and $\hat{k}$ and k denote the estimated and actual scale factors, respectively. This error measure is dimensionless and has been used for depth maps in [12]. The ERR values averaged over the four different scenes were as

follows: 2.3% when no noise was added, 3.4% when the noise standard deviation was 5, and 4.3% for noise level of 10. From the second column (rows 1–4) of Fig. 1, we see that our estimates closely match the true depth values even for a noise level of standard deviation 10 (which is well within the noise limits of modern cameras).

In the following experiment, we show that even if there is incidental geometric misalignment between the observations, the proposed method can be used after compensating for the relative motion (exact compensation will not be possible due to parallax). Studies have revealed [21] that, for camera jitter, in-plane rotation is the most likely cause for misalignment. It is straightforward to perform a brute-force search to determine the angle that minimizes the sum-squared error between the warped reference image and the blurred observation. Of course, one could use more sophisticated registration techniques [14]. On the same four sets of scenes considered earlier, we performed synthetic experiments by inducing alignment errors in the blurred observation. For each case, the second observation was generated using not only translations but also by inducing rotation. From the observations, the angle of misalignment was determined by coarse-to-fine search, and the second observation was corrected by inverse warping. Using the registered pair of images, the basic blur kernel and its scale factors were estimated using the proposed approach. The resultant depth maps and the estimated blur kernels when the alignment error was 2° are shown in rows 1–4 (third column) and rows 5–8 (third column) of Fig. 1, respectively. The value of ERR averaged over these four different sets of images, blur kernels, and depth maps was found to be less than 4.6% even up to a rotation of 2°.

We next tested our algorithm for estimating scene depth from motion blur on images captured by a digital camera. In our experiments, we use a simple translating stage whose motion along different directions can be controlled. While capturing the reference image, the camera is stationary. To capture the blurred observation, the camera is translated arbitrarily along the plane perpendicular to the optical axis of the camera during exposure. Because of constrained motion in our laboratory setup, our images do not suffer from alignment errors; else, they should be first registered as discussed earlier. In our experiments, the objects in the scene were within a distance of 1 m from the camera so as to induce depth-dependent blurring effects. Note that, for accurate depth estimation, the scene should not only be well textured but should also have significant components along different directions.

The scene consisted of objects at different distances from the camera. The reference image and the blurred observations are shown in Fig. 2(a) and (b), respectively. The PSF was estimated from a small patch in the region corresponding to the green rectangular block and is shown in Fig. 2(c). We see that the shape of the estimated PSF (which depicts the camera motion) is quite unlike a Gaussian and that the peak value of the PSF is away from the center. The scale map estimated from the proposed UKF-based technique is shown in Fig. 2(d). Farther objects in the scene undergo less smearing; hence, the corresponding scale factors would be higher as correctly depicted in the recovered scale map of Fig. 2(d). We also notice that Fig. 2(d) captures the scene variations even at depth discontinuities. For instance, the thin red thread connected to the face of the horse is well separated from the background in the scale map. The values of the estimated scale factors near the discontinuity of the green clay

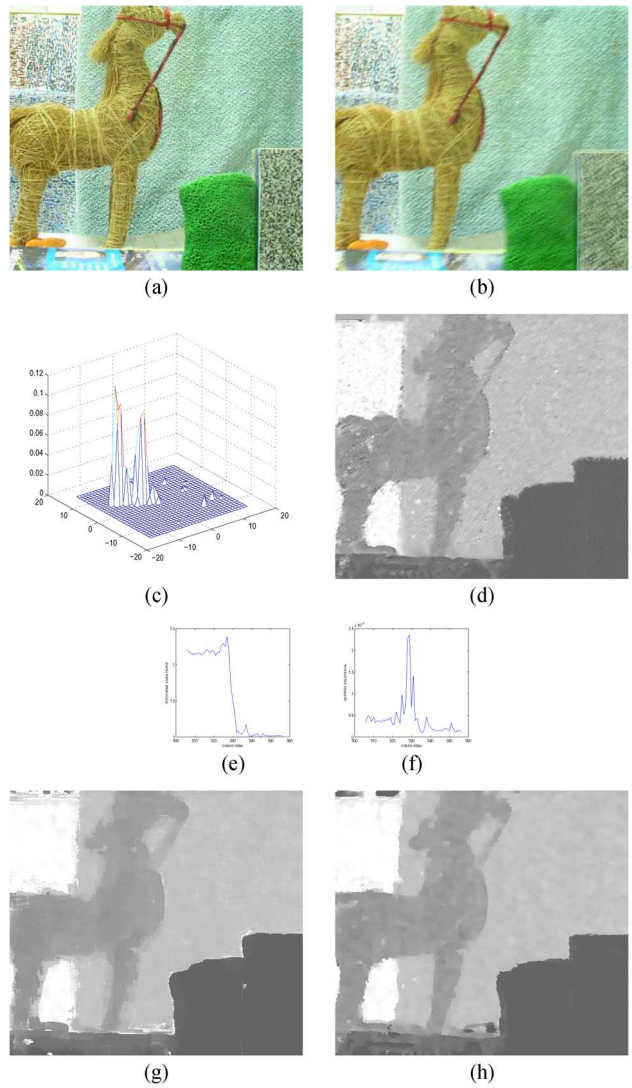

Fig. 2. (a) Reference image. (b) Blurred observation. (c) Estimated PSF. (d) Estimated structure. Cross-sectional view of (e) estimated scale factors and (f) variance. Depth map from (g) an exhaustive search algorithm and (h) MRF regularization-based method with a truncated smoothness cost.

model along one of the randomly selected rows, and the corresponding values of the updated variance are shown in Fig. 2(e) and (f), respectively. From these cross sections, we see that the peak of the variance occurs at the the depth discontinuity and matches quite well with the transition of the estimated scale factors. As discussed in Section IV-C, due to the ability of the proposed method to preserve the discontinuities, the depth estimates are well localized at the edges.

For sake of comparison, we also implemented an exhaustive search algorithm in which, at each pixel, the scale factor was varied in fine steps and that value which minimizes the RMS error (evaluated over a patch of the size of the scaled blur kernel) was selected. This approach is computationally quite involved, since at every pixel, we must generate a patch of blurred observation corresponding to the blur kernel of each scale factor. In the resultant depth map shown in Fig. 2(g), we see that the depth estimates are not localized at the discontinuities, the primary reason being the lack of a judicious prior to handle the inherent ill-posedness of the problem. In Fig. 2(h), we show the

output obtained by implementing MRF with an edge-preserving prior [3] and belief propagation for energy minimization. Although the resultant depth estimate is comparable to ours, the MRF scheme requires about 15 iterations to converge and necessitates large memory. Moreover, the computation time is an order higher compared with our technique. The elegance of our method lies in the fact that it is local and yet quite accurate.

We performed yet another experiment to assess the performance of the proposed algorithm in a real scenario. The scene shown in Fig. 3(a) consisted of several objects kept at calibrated distances from the camera. The PSF was estimated in the region corresponding to the orange rectangular block. The motion-blurred observation is shown in Fig. 3(b). The proposed method was used to obtain the scale factors, as shown in Fig. 3(c). We considered a point on the orange block as the reference point. The scale factor at this point was estimated as one (since the reference blur kernel was estimated in this region). The estimated scale factors were multiplied by the distance between the camera and the orange block to get the depth values for all the image points. While the depth of the scene varied from 17.6 (for the orange block) to 42.4 cm (for the background), the estimated depth values were less than 4% in error. The updated state variance is shown as an image in Fig. 3(d) wherein we note that the peaks of the updated variance match with the depth discontinuities. For purpose of comparison, depth estimation was also performed by the recent technique proposed in [30] using the same images. In [30], an auxiliary algorithm is used to get an initial estimate of the depth map. By assuming that the blur is space invariant in the neighborhood of a pixel, this algorithm searches for a depth label (from a range of possible depth values) that minimizes the observation error. The optimum depth label at each pixel is treated as the initial estimate of the depth map. Depth is finally estimated by minimizing a cost function that consists of the observation error and total-variation (TV) regularization terms. The depth maps from the auxiliary algorithm and the final TV-regularized estimate of the technique in [30] are shown in Fig. 3(e) and (f), respectively. Although the window-based auxiliary algorithm searches for the optimum depth value at each pixel, in the depth map of Fig. 3(e), we see that the estimates go quite wrong at some locations. For instance, in the region corresponding to the green-and-white cloth (at the bottom of the image), there are bright spots that correspond to a larger depth value, although this part of the scene was close to the camera. The effect of these errors carries over to the TV-regularized estimate in Fig. 3(f). In contrast, in our UKF-based estimate of Fig. 3(c), although there are small undulations at a few locations, the depth map is more reliable. When implemented on a computer with a 2.13-GHz processor and 2-GB RAM, the average computation time for our method was found to be about 2.9 ms per pixel.

In the following experiment (see Fig. 4), we considered a scene with significantly less texture. In addition, note the presence of shadows in some regions in the image. The blurred observation and the estimated PSF are shown in Fig. 4(b) and (c), respectively, whereas the depth estimated using our algorithm is shown in Fig. 4(d). Note that, except for a few minor artifacts, our result correctly depicts the underlying scene structure even in this difficult scenario.

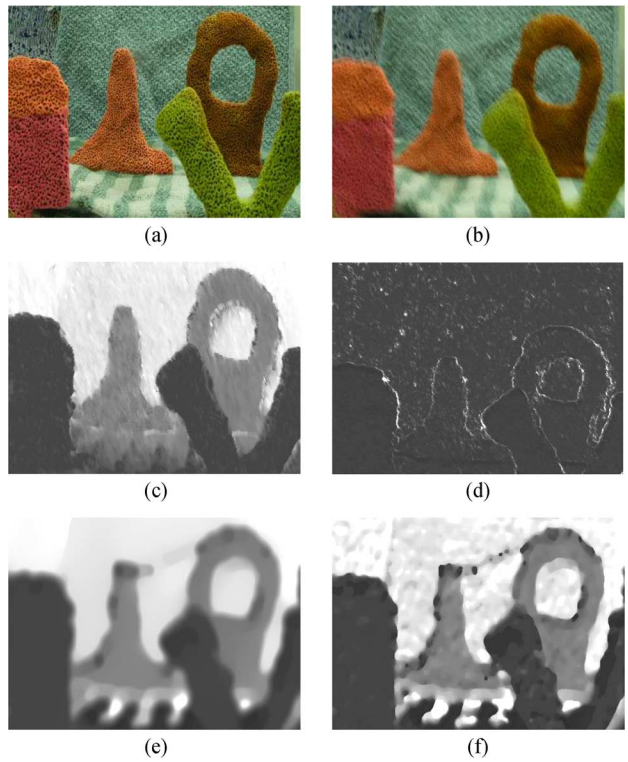

Fig. 3. (a) Reference image. (b) Blurred observation. (c) Depth map by the proposed technique. (d) Updated state variance. Depth map estimated by (e) the auxiliary method in [30] and (f) TV regularization. [30].

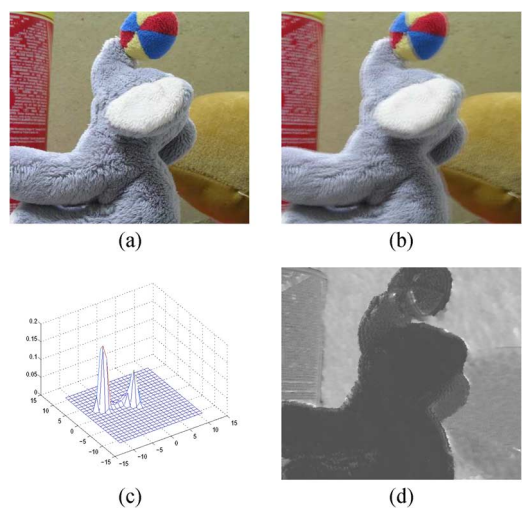

Fig. 4. (a) Reference image. (b) Blurred observation. (c) Estimated PSF. (d) Derived scale map.

The next set of experiments correspond to performance evaluation in a DFD scenario. In a synthetic experiment, an image generated by random dots was used as the pin-hole reference image of the scene. The scene depth was assumed to be of the shape of a ramp with depth values ranging from 21.5 to 28 cm [see Fig. 5(a)]. We used a Gaussian-shaped PSF to generate two blurred observations. For each image, the standard deviation of Gaussian PSF was calculated at each pixel by assuming realistic values for camera parameters. The focal length and the distance between the lens and the image plane were chosen to be of the

Fig. 5. (a) True depth. (b) Estimated depth.

order of a few centimeters for both the observations. The value of the aperture radius was different for the two observations. The resulting standard deviation of the Gaussian PSF ranged between 0.27 to 1.14 for one observation and between 0.44 to 1.83 for the other. The observations were generated by space-variantly blurring the reference image with a Gaussian PSF. The proposed method was applied to estimate the scale factors of the relative blur by using the less blurred image as the reference image. The standard deviation of the relative blur between the two images was evaluated at each pixel from the estimated scale factors. Using the relationship between the variance of Gaussian denoting the relative blur and the depth [12], the depth map was obtained [see Fig. 5(b)]. The value of ERR was found to be only 1%.

To test the performance of our algorithm when the depth map consists of a sharp vertical discontinuity, we performed a synthetic experiment in which the scene was assumed to be composed of two planes of depth levels of 22 and 26 cm [see Fig. 6(a)]. By assuming realistic camera parameters and a Gaussian form for the blur kernel, two observations corresponding to two different aperture radii were generated. The depth map estimated by the proposed method is shown in Fig. 6(b). The true and estimated relative blur values corresponding to one of the rows are plotted in Fig. 6(c). We see that the estimates from our method are quite close to the true values and adapt well at the edges. The value of the updated variance at every pixel is shown as an image in Fig. 6(d), whereas a cross-sectional view is shown in Fig. 6(e). We again note that the peak in the variance value coincides with the depth discontinuity.

Next, we demonstrate the performance of our method in a real scenario. We first show results on the images used in a recent paper on diffusion-based depth estimation in [12]. The pair of blurred observations shown in Fig. 7(a) and (b) correspond to near–far focused pair from their data set in [12]. The focus settings for the two observations were as follows: a focal length of 1.2 cm, F-number 2, near focused plane at 52 cm, and the far focused plane at 85 cm. The PSF was estimated in a small region in the image, and it was observed to be close to a Gaussian (as correctly postulated in [12]). Assuming a Gaussian form for the PSF, the signed scale factors were estimated, as discussed in Section V-B. From the estimated scale map, the standard deviation of relative blur was found at all the image points. Note that our depth map [see Fig. 7(c)] closely mimics the 3-D nature of the scene. A quantitative evaluation is not possible since the authors of [12] have not made the ground truth available.

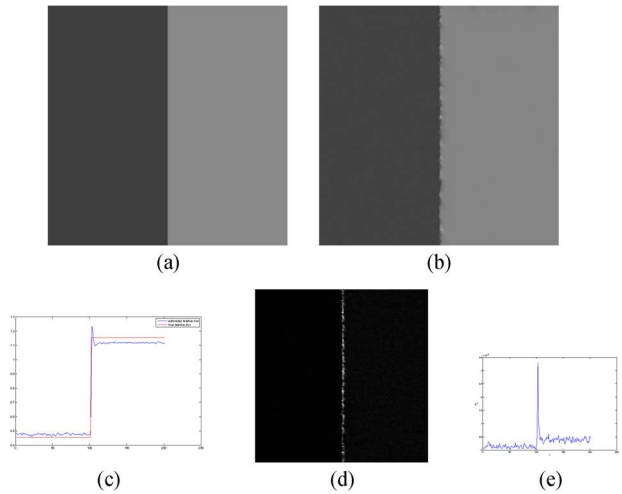

Fig. 6. (a) True depth map. (b) Estimated depth. (c) Cross section of true and estimated relative blur. (d) Variance. (e) Cross section of the variance.

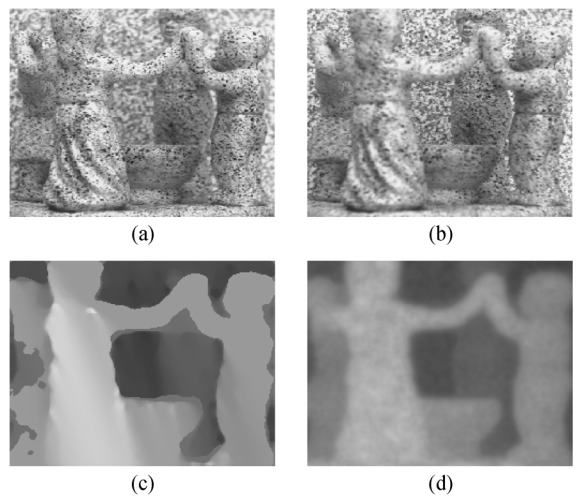

Fig. 7. (a) and (b) Blurred observations. (c) Depth estimate by the proposed UKF-based method. (d) Corresponding depth estimate by diffusion [12].

However, we do provide comparison with the diffusion algorithm [12] whose implementation is available at [1]. The corresponding depth map obtained by [12] is shown in Fig. 7(d). We see that the values of the depth estimates by our algorithm are quite comparable to the estimates from the diffusion technique, which employs a variational approach. In fact, the discontinuities are better localized with our method.

When objects in a scene are at a significant distance from the point of focus, camera lenses can lead to PSFs, which do not have a peak at the center [20], [29]. We next tested the proposed algorithm to perform DFD for such a scenario. Fig. 8(a) and (b) shows two defocus-blurred observations captured with different apertures using an Olympus C-5050Z digital camera. The working distance for both the images was 32 cm, whereas the objects in the scene were kept between 34 and 48 cm from the camera. The relative blur between the two observations was estimated using an image patch cropped from the background region (whose depth value was known). In Fig. 8(c), we see that the shape of the blur kernel is quite different from a Gaussian. We applied the proposed technique to get an estimate of the scale factors. The depth map of the scene was obtained, as shown in

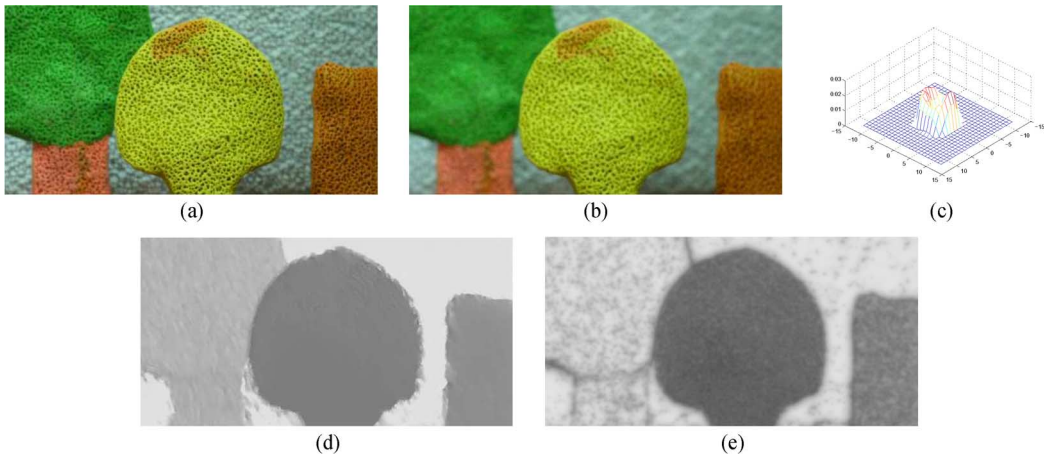

Fig. 8. (a) and (b) Blurred observations. (c) Reference PSF. Estimated shape of the scene by (d) proposed method, and (e) diffusion [12].

Fig. 8(d), using the estimated scale factors with the knowledge of the camera settings and the background depth value. In the depth map, objects near the camera are represented by lesser intensities than the farther ones. We note that the structure of the scene is correctly depicted in Fig. 8(d). When the diffusion algorithm [12] (which inherently makes a Gaussian assumption for the optical PSF) was applied on these images, depth of the scene was estimated, as shown in Fig. 8(e). Since the scene is composed of different objects which are approximately planar, one would expect the depth estimate to be piecewise constant. However, in Fig. 8(e), we observe some visual artifacts throughout the depth map. In addition, the distinction between the green tree model and the background is visibly poor in this case, as compared with our estimate in Fig. 8(d).

The last set of experiments correspond to the scenario of depth estimation in the presence of simultaneous motion and optical blur. We again considered the four sets of images, motion blur kernels and depth maps that were used in our earlier synthetic experiment for the case of motion blur alone. The defocus blur kernel was assumed to be of Gaussian form. For each scene, we initially generated two defocus-blurred observations g_1 and g_2 according to the depth maps (Fig. 1, rows 1 to 4, first column) by considering realistic camera parameters. Following the model in (20), out of the two observations, the more blurred image was subject to space-variant blurring with the motion blur PSF (according to the depth map of the scene) to get the observation denoted by g_{2_s}. For each set, Gaussian noise was added to both the observations. Using small image patches from the observations, the reference motion blur kernel h_{m_0} was estimated for a range of values of the defocus blur parameter σ_f. The optimum value of σ_f (which also indicates the estimated reference depth) and the corresponding motion blur kernel were quite close to the true values for all the four scenes even upto a noise level of 10 (see fourth column, rows 5–8 of Fig. 1). The value of the sparsity weight ε [used in (21)] was the same in all the cases. For each noise level, the proposed method was applied on the observations to estimate the depth map with the knowledge of the estimated motion blur kernel and the reference depth value. The ERR values averaged over the four different scenes were as follows: 3.5% for no noise and 4.4% and 5.49% (fourth column, rows 1 to 4 of Fig. 1) when the standard deviation of noise was 5 and 10, respectively.

We also considered the case of rotational misalignments between the observations g_1 and g_{2_s} on these four sets of scenes. As in the case of motion blur, the observations are used for depth estimation after compensating for the rotational motion. Observation g_1 was generated for each case in the same manner as in the previous experiment by assuming Gaussian kernels for the optical blur. To simulate the effect of misalignment, we rotated the pin-hole reference image while generating the observation g_{2_s}. We used a brute-force search to determine the angle of misalignment between g_1 and g_{2_s}. The observation g_{2_s} was aligned with g_1 by inverse warping. Using images patches from the registered pair, the motion blur kernel and the reference depth value were estimated by minimizing the combined cost E_{tot} (for all the four scenes). Even when the angle of misalignment was 2°, the estimated motion blur kernels (shown in the fifth column, rows 5–8 of Fig. 1) were quite acceptable. The corresponding depth maps estimated from the proposed method are shown in the fifth column, rows 1–4 of Fig. 1. The value of the error ERR averaged over the four scenes was 4.88%.

We applied the proposed approach on a real example in which there was camera shake while capturing optically defocused observations. The reference image shown in Fig. 9(a) was captured from a digital camera in aperture-priority mode. For this image, the focal length was 21.3 mm, the working distance was 200 mm, and the F-number was 8. We changed the F-number of the camera to 5 and induced camera shake while capturing the second observation shown in Fig. 9(b). The optical blur kernel shown in Fig. 9(c) was computed *a priori* between a pair of observations of a planar object kept at a distance of 24.8 cm from the camera for the same lens settings (without any camera motion). This was used as the known defocus kernel h_{def}. In the two images, we chose a patch corresponding to the green clay model for estimating the motion blur kernel h_{m_0} and the reference depth d_0. The scale factor of the defocus kernel k_f was varied from 2 to 5.3, and the corresponding motion blur kernels were estimated. The optimum value of k_f that minimizes the cost function $E_{\text{tot}}(k_f)$ was found to be $k_f = 4.7$. The estimated motion blur kernel when $k_f = 4.7$ is shown in Fig. 9(d). From the camera parameters, d_0 was determined as 20.9 cm. Using the proposed technique, the scene structure was estimated, as shown in Fig. 9(e). We observe that objects that are close to the camera are correctly indicated by lesser intensities, and the background is indicated by a higher intensity. Note that stan-

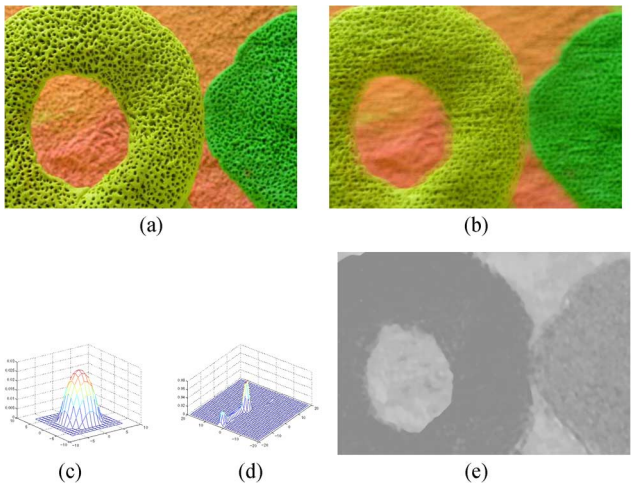

Fig. 9. (a) Reference image. (b) Simultaneously blurred observation. (c) Known optical blur. (d) Estimated motion kernel. (e) Estimated depth.

dard DFD techniques are incapable of recovering depth in such a scenario as they do not account for the motion blur kernel and its spatial variations with respect to depth.

VIII. Conclusion

We proposed a UKF-based approach for recovering the shape of a 3-D scene from its blurred observations. Our framework involves only local computations and is applicable to both motion as well as out-of-focus blur. Our method has the additional advantage that it can be applied to DFD without constraining the PSF to be Gaussian. By enforcing sparsity constraint, we also addressed the problem of depth estimation when an observation undergoes simultaneous motion and optical blur. Studies were also conducted to evaluate our approach in the presence of noise and alignment errors. For good performance, our algorithm requires textured images but this is true of most blur-based depth estimation methods. Possible extensions to this paper include considering general camera and/or object motion, and handling occlusions.

Appendix
Estimation of Relative Blur

The reference kernel h_o is estimated from the regions of constant depth in f and g using a gradient-based technique. Let f_c and g_c denote a patch of $N_p \times N_p$ pixels of f and g, respectively, corresponding to a region of constant depth. They are related to each other through the space-invariant convolution equation $g_c = f_c * h_o$. Let the kernel h_o be of size $N \times N$. The convolution equation can be expressed as

$$\overline{g_c} = F_c \overline{h_o} \tag{22}$$

where $\overline{g_c}$ is an $N_p^2 \times 1$ vector, F_c is a sparse matrix of size $N_p^2 \times N_p^2 N^2$, and $\overline{h_o}$ is of size $N_p^2 N^2 \times 1$ formed by stacking the blur kernel h_o at every pixel. We estimate the blur kernel by minimizing the cost function

$$E_{\rm ls}(h_o) = \|\overline{g_c} - F_c \overline{h_o}\|^2 = \overline{g_c}^T \overline{g_c} - 2\overline{h_o}^T F_c^T \overline{g_c} + \overline{h_o}^T F_c^T F_c \overline{h_o}. \tag{23}$$

The gradient equation is

$$\frac{\partial E_{\rm ls}(h_o)}{\partial h_o} = \frac{\partial \overline{h_o}}{\partial h_o} \frac{\partial E_{\rm ls}}{\partial \overline{h_o}} = 2 \frac{\partial \overline{h_o}}{\partial h_o} F_c^T (F_c \overline{h_o} - \overline{g_c}). \tag{24}$$

The estimate of h_o is updated using gradient descent as

$$h_o^{n+1} = h_o^n - \mu_h \frac{\partial E_{\rm ls}(h_o)}{\partial h_o}. \tag{25}$$

For effective computations, the matrix multiplication operations in (24) are implemented using convolution. We performed many synthetic experiments to check the performance of the PSF estimation technique and found that the estimates were quite accurate. In this method, since we estimate N^2 unknowns from N_p^2 observations, the estimates will be accurate provided N_P is much greater than N. In our experiments, we consider the size of image patches such that N_P is at least twice the value of N.

Acknowledgment

The authors would like to thank the anonymous reviewers for their useful comments and suggestions, and Dr. M. Sorel for executing his algorithm [30] and sending the results on some of the images from the data set.

References

[1] [Online]. Available: http://www.eps.hw.ac.uk/~pf21
[2] P. Brodatz, *Textures; A photographic album for artists and designers*. New York: Dover, 1966.
[3] A. V. Bhavsar and A. N. Rajagopalan, "Depth estimation with a practical camera," in *Proc. BMVC*, London, U.K., 2009.
[4] S. Chaudhuri and A. N. Rajagopalan, *Depth From Defocus: A Real Aperture Imaging Approach*. New York: Springer-Verlag, 1999.
[5] F. Deschenes, D. Ziou, and P. Fuchs, "A unified approach for a simultaneous and cooperative estimation of defocus blur and spatial shifts," *Image Vis. Comput.*, vol. 22, no. 1, pp. 35–57, Jan. 2004.
[6] J. Ens and P. Lawrence, "An investigation of methods for determining depth from focus," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 2, pp. 97–108, Feb. 1993.
[7] P. Favaro, "Recovering thin structures via nonlocal-means regularization with application to depth from defocus," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2010, pp. 1133–1140.
[8] P. Favaro, M. Burger, and S. Soatto, "Scene and motion reconstruction from defocused and motion-blurred images via anisotropic diffusion," in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 257–269.
[9] P. Favaro, A. Mennucci, and S. Soatto, "Observing shape from defocused images," *Int. J. Comput. Vis.*, vol. 52, no. 1, pp. 25–43, Apr. 2003.
[10] P. Favaro and S. Soatto, *3-D Shape Estimation and Image Restoration Exploiting Defocus and Motion-Blur*. New York: Springer-Verlag, 2006.
[11] P. Favaro and S. Soatto, "A variational approach to scene reconstruction and image segmentation from motion-blur cues," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2004, pp. 631–637.
[12] P. Favaro, S. Soatto, M. Burger, and S. Osher, "Shape from defocus via diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 3, pp. 518–531, Mar. 2008.
[13] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 787–794, Jul. 2006.
[14] J. Flusser, J. Boldys, and B. Zitova, "Moment forms invariant to rotation and blur in arbitrary number of dimensions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 234–246, Feb. 2003.
[15] J. S. Fox, "Range from translational motion blurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 1988, pp. 360–365.
[16] D. J. Heeger and A. D. Jepson, "Subspace methods for recovering rigid motion," *Int. J. Comput. Vis.*, vol. 7, no. 2, pp. 95–117, Jan. 1992.

[17] J. Jia, "Single image motion deblurring using transparency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2007, pp. 1141–1151.
[18] S. Julier and J. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *Proc. 11th Int. Symp. Aerosp./Defense Sens., Simul. Controls*, 1997, pp. 182–193.
[19] S. Julier, J. Uhlmann, and H. F. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Autom. Control*, vol. 45, no. 3, pp. 477–482, Mar. 2000.
[20] H. C. Lee, *Introduction to Color Imaging Science*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
[21] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2009, pp. 1964–1971.
[22] S. Z. Li, *Markov Random Field Modeling in Computer Vision*. Berlin, Germany: Springer-Verlag, 1995.
[23] H. Y. Lin and C. H. Chang, "Depth recovery from motion blurred images," in *Proc. Int. Conf. Pattern Recogn.*, 2006, pp. 135–138.
[24] D. J. C. Mackay, "Introduction to Monte Carlo methods," in *Learning in Graphical Models*, M. I. Jordan, Ed. Norwell, MA: Kluwer, 1998, pp. 175–204.
[25] C. Paramanand and A. N. Rajagopalan, "Unscented transformation for depth from motion-blur in videos," in *Proc. IEEE CVPR—Workshop Three Dimensional Information Extraction Video Analysis Mining*, 2010, pp. 38–44.
[26] K. Ramnath and A. N. Rajagopalan, "Discontinuity adaptive shape from focus," in *Proc. 31st DAGM Symp. Pattern Recogn.*, 2009, pp. 181–190.
[27] S. Seitz and S. Baker, "Filter flow," in *Proc. Int. Conf. Comput. Vis.*, 2009, pp. 143–150.
[28] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," *ACM Trans. Graph.*, vol. 27, no. 3, Aug. 2008, Art. no. 73.
[29] W. J. Smith, *Modern Optical Engineering*. Bellingham, WA: SPIE, 2007.
[30] M. Sorel 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.
[31] M. Sorel, "Multichannel blind restoration of images with space-variant degradations," Ph.D. dissertation, Charles Univ., Prague, Czech Republic, 2007.
[32] M. Sorel and F. Sroubek, "Space-variant deblurring using one blurred and one underexposed image," in *Proc. Int. Conf. Pattern Recogn.*, 2009, pp. 157–160.
[33] F. Sroubek and J. Flusser, "Multichannel blind deconvolution of spatially misaligned images," *IEEE Trans. Image Process.*, vol. 14, no. 7, pp. 874–883, Jul. 2005.
[34] K. V. Suresh and A. N. Rajagopalan, "Super-resolution of license plates in real traffic videos," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 321–331, Jun. 2007.
[35] G. R. K. S. Subrahmanyam, A. N. Rajagopalan, and R. Aravind, "A recursive filter for despeckling SAR images," *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1969–1974, Oct. 2008.
[36] Y. Tai, H. Du, M. Brown, and S. Lin, "Correction of spatially varying image and video motion blur using a hybrid camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 6, pp. 1012–1028, Jun. 2010.
[37] L. Yuan, J. Sun, L. Quan, and H. Y. Shum, "Image deblurring with blurred/noisy image pairs," *ACM Trans. Graph.*, vol. 26, no. 3, pp. 1–10, Jul. 2007.
[38] L. Yuan, J. Sun, L. Quan, and H. Y. Shum, "Blurred/non-blurred image alignment using sparseness prior," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2007, pp. 1–8.
[39] Y. F. Wang and P. Liang, "3D Shape and motion analysis from image blur and smear: a unified approach," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1998, pp. 1029–1034.
[40] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proc. IEEE Adapt. Syst. Signal Process., Commun., Control Symp.*, 2000, pp. 153–158.
[41] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce, "Non-uniform deblurring for shaken images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2010, pp. 491–498.
[42] L. Xu and J. Jia, "Two-phase kernel estimation for robust motion deblurring," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 157–170.
[43] J. Yang and D. Schonfeld, "Virtual focus and depth estimation from defocused video sequences," *IEEE Trans. Image Process.*, vol. 19, no. 3, pp. 668–679, Mar. 2010.

C. Paramanand received the B.E. degree in telecommunication engineering from R.V. College of Engineering, Bengaluru, India, in 2004 and the M.S. degree in electrical engineering from Indian Institute of Technology Madras, Chennai, India, in 2007. He is currently working toward the Ph.D. degree with the Indian Institute of Technology Madras.

A. N. Rajagopalan (SM'11) received the Ph.D. degree in electrical engineering from the Indian Institute of Technology, Bombay, India, in 1998.

During the summer of 1998, he was a Visiting Scientist with the Image Communication Laboratory, University of Erlangen, Erlangen, Germany. Between 1998 and 2000, he was with the Center for Automation Research, University of Maryland, College Park, as an Assistant Research Scientist. Since October 2000, he has been with the Indian Institute of Technology Madras, Chennai, India, and is currently a Professor with the Department of Electrical Engineering, Indian Institute of Technology Madras. In 2002 and 2005, he held visiting positions with the Center for Automation Research, University of Maryland. In 2007–2008, he was a Humboldt Fellow with the Technical University of Munich, Munich, Germany. He is a coauthor of the book *Depth from Defocus: A Real Aperture Imaging Approach* (New York: Springer, 1999). His research interests include 3-D shape from optical and motion blur, image restoration, super-resolution, gait and face, and video matting.

Dr. Rajagopalan was Program Co-Chair for the seventh Indian Conference on Computer Vision, Graphics and Image Processing and is an Area Chair for the upcoming IEEE Conference on Computer Vision and Pattern Recognition. He served as an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE from 2007–2011.