# Blur Calibration for Depth from Defocus

Fahim Mannan* and Michael S. Langer†
*School of Computer Science*
*McGill University*
*Montreal, Quebec H3A 0E9, Canada*
{*fmannan, †langer}@cim.mcgill.ca

***Abstract*—Depth from defocus based methods rely on measuring the depth dependent blur at each pixel of the image. A core component in the defocus blur estimation process is the depth variant blur kernel. This blur kernel is often approximated as a Gaussian or pillbox kernel which only works well for small amount of blur. In general the blur kernel depends on the shape of the aperture and can vary a lot with depth. For more accurate blur estimation it is necessary to precisely model the blur kernel. In this paper we present a simple and accurate approach for performing blur kernel calibration for depth from defocus. We also show how to estimate the relative blur kernel from a pair of defocused blur kernels. Our proposed approach can estimate blurs ranging from small (single pixel) to sufficiently large (e.g. $77 \times 77$ in our experiments). We also experimentally demonstrate that our relative blur estimation method can recover blur kernels for complex asymmetric coded apertures which has not been shown before.**

***Keywords*-Depth from Defocus, Point spread functions, Relative Blur, Optimization**

## I. Introduction

Defocus blur in an image depends on camera parameters such as the aperture size ($A$), focal length ($f$) and focus distance, and the depth of the scene. When the camera parameters are fixed, the blur varies as a function of depth. The central problem in Depth from Defocus (DFD) is to estimate the defocus blur at every pixel and convert that to depth estimates using the known camera parameters. Typically in DFD, two differently defocused images are used and the problem is to find the depth that produces the observed defocused images. For accurate depth estimation we need to model the way defocus blur changes with camera parameters and depth. This is done by modelling what a point light source at different depths looks like under different camera settings, that is, the point spread function (PSF).

Although the PSF can be considered to be the image of a point light source, in practice it is challenging to take images of point light sources. Furthermore there is no true point light source and for certain camera and scene configurations the point source assumption does not hold. A real point light source has a finite size and may not appear as a single point even when it is in focus (e.g. Fig. 4a).

We propose a simple approach for calibrating PSFs for different depths and camera configurations. We highlight some of the issues involved in calibration assuming the pinhole model or the thin lens camera model (aperture ratio, center of projection, moving sensor, etc). We also show how to calibrate the relative blur kernel for DFD. Our main contribution is proposing an accurate procedure for calibrating PSFs from disk images and estimating the relative blur kernel. Our PSF estimation approach is robust to noise, large blur (e.g. $77\times77$ blur kernel) and display pixel density, and yet simple and flexible. For instance, we do not require any complex priors in the optimization objective or the texture to follow a certain distribution.

The paper is organized as follows. Sec. II gives some of the necessary background for DFD. Sec. III discusses the setup and preprocessing steps required for calibration. Sec. IV presents how absolute and relative blurs are estimated. Sec. V evaluates the estimated PSFs and their relative blur kernels using synthetic and real defocused images.

## II. Background

To motivate the need for blur kernel calibration we first look at how blurred images are formed and how defocus blur changes with depth. Then we look at how the problem of DFD is modelled using depth dependent blur kernels and some of the relevant works in blur kernel estimation.

### A. Blurred Image Formation

First we consider how a point light source is imaged by a thin lens in Fig. 1. Light rays emanating from a scene point at distance $u$ from a thin lens fall on the lens and converge at distance $v$ on the sensor side of the lens. For a lens with focal length $f$, the relationship between $u$ and $v$ is given by the thin lens model as:

$$\frac{1}{u}+\frac{1}{v}=\frac{1}{f}\ . \tag{1}$$

If the imaging sensor is at distance $s$ from the lens then the imaged scene point creates a circular blur pattern of radius $r$ as shown in the figure. In general the shape of the blur pattern will depend on the shape of the aperture. For a lens with aperture $A$, the thin lens model (Eq. 1) and similar triangles from Fig. 1 give the radius of the blur in pixels:

$$\sigma=\rho r=\rho s\frac{A}{2}(\frac{1}{f}-\frac{1}{u}-\frac{1}{s}). \tag{2}$$

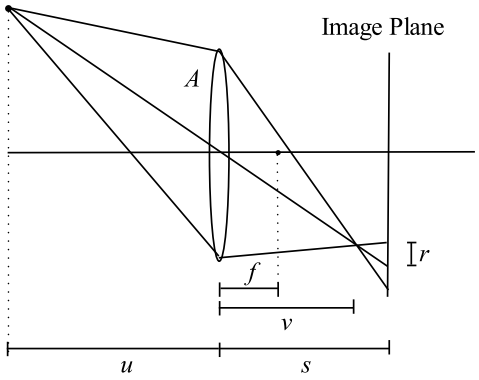



Figure 1: Defocus blur formation.

The variable $\rho$ is used to convert from physical to pixel dimension. In the rest of this paper we will use $\sigma$ to denote blur radius in pixels. Note that the blur can be positive or negative depending on which side of the focus plane a scene point resides. For circularly symmetric aperture the sign of the blur has no affect on the blurred image formation process. But for asymmetric apertures, the two images would appear slightly different because the corresponding PSFs will be flipped both horizontally and vertically.

If the scene depth is nearly constant in a local region, then an observed blurred image $i$, can be modelled as a convolution of a focused image $i_0$, with a depth-dependent point spread function (PSF) $h(\sigma)$.

$$i = i_0 * h(\sigma) \tag{3}$$

In this paper our goal is to find the depth dependent PSF $h$ (which we sometimes refer to as absolute blur) from observed blurred images $i$. For real defocused images the lens and camera sensor will produce artifacts due to diffraction and lens aberrations (e.g. chromatic aberration). Therefore an accurate PSF estimation process needs to capture the combined effect of defocus scale, diffraction and lens aberrations.

Relative blur estimation requires a pair of defocused images captured with different camera parameters. The most widely used configurations involve varying either the aperture size or the focus between the two images. We refer to these configurations as variable aperture and variable focus in the rest of the paper. The purpose of the relative blur model is to find the degree by which the sharper image is blurred to obtain the blurrier image. If the sharper image $i_S$ has blur kernel $h_S$, and the blurrier image $i_B$ has blur kernel $h_B$, then the relative blur between the two images is $h_R$ where $h_B \approx h_S * h_R$. Similar to the absolute blur PSF estimation problem the estimated relative blurs need to reconstruct the features of the blurrier PSFs.

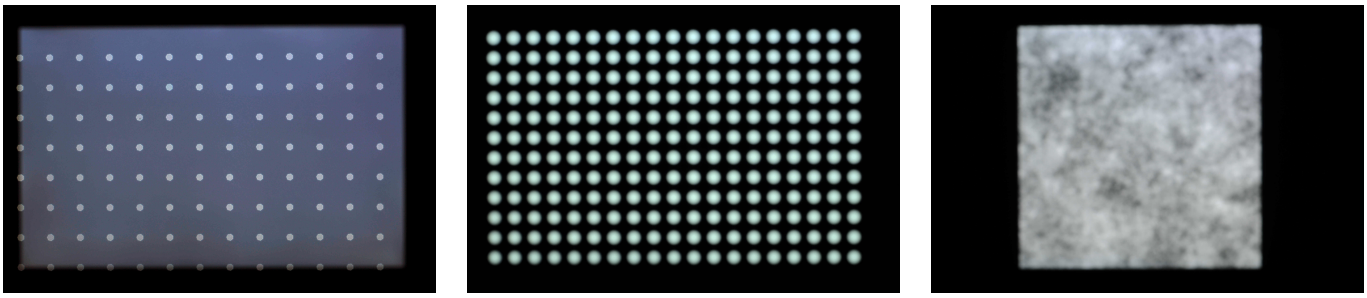
### B. Related Work

There have been several works on blur kernel estimation from images. Most of them are motivated by deblurring defocused or motion blurred images. Many are related to blind image deconvolution. In this section we only consider works that are related to blur kernel calibration i.e. very accurate blur estimation using a calibration pattern.

Most blur kernel estimation methods require some knowledge about the latent sharp image. In Joshi et al. [1] the authors rely on first estimating the latent sharp edges and then using that for PSF estimation. They also propose a calibration pattern for performing more accurate PSF estimation. Delbracio et al. [2] used the Bernoulli noise pattern for PSF estimation. A similar noise pattern was used in [3] for estimating intrinsic lens blur. An issue with using such noise patterns is that the scene and camera setup need to be such that the underlying noise pattern assumption is satisfied in the projected image. Kee et al. [4] uses disk images similar to ours but with a different objective function for estimating the intrinsic lens blur.

In the case of calibrating the depth dependent relative blur kernels, the only work known to us is by Ens and Lawrence [5]. This relative blur model has been used both in the frequency [6], [7] and in the spatial domain [5], [8]. Ens and Lawrence calibrated the relative blur kernel from two observed defocused images. As regularizers they used constraints that prefer the relative blur kernel to be in a certain family of kernels. This family includes smooth circularly symmetric kernels with zeros at the boundary. In our work the relative blur calibration problem is a special case of the absolute blur estimation problem. Our data terms consider the gradient of the observed images and the smoothness terms do not require the circularly symmetric assumption. In the case of DFD, once the relative blur kernels are calibrated for different depths, depth estimation for a pair of defocused images ($i_S$ and $i_B$) is done by looking up the relative blur kernel that minimizes: $\underset{h_R}{\operatorname{argmin}} \|i_B - i_S * h_R\|_2^2$.

In terms of applying the estimated kernels for DFD estimation, besides the relative blur method there is the Blur Equalization Technique (BET) proposed by Xian and Subbarao [9] that takes a pair of depth dependent absolute blur kernels and chooses the depth that minimizes: $\underset{h_S,h_B}{\operatorname{argmin}} \|i_S * h_B - i_B * h_S\|_2^2$. For the non-blind deconvolution model, the estimated blur kernels are used to estimate the sharp image by solving the minimization problem: $\underset{i_0}{\operatorname{argmin}} \|i_0 * h - i\|_2^2$, where $i_0$ is the latent sharp image and $i$ is the observed blurred image. By considering depth dependent blur kernels this idea can be extended to DFD [10], [11]. For more details on these different approaches to DFD and their comparison see [12].



(a) Grid of Dots with 10s exposure (b) Grid of Disks with 0.2s exposure (c) $1/f$ texture with 0.5s exposure

Figure 2: (a) and (b) are defocused images of the calibration patterns. We use the disk images for PSF estimation and the dot images for qualitative comparison. (c) defocused test image used for evaluating the DFD models. For all these images the object to sensor distance is $1.5$ m and focus distance is $0.5$ m. The captured images are of size $4288 \times 2848$ pixels but we only use the center part for our experiments.

## III. Setup and Calibration

A typical approach in DFD is to use the known camera parameters with the analytical equations for thin-lens blur formation model and use a parametric PSF for depth estimation. However real lenses do not follow the thin-lens model exactly. For example the focus distance is specified from the sensor plane rather than from the center of projection. Furthermore the PSF kernels can change with aperture shape, size and blur size. There are also other modelling assumptions that do not always hold for real images e.g. that the two defocused images are aligned and have the same average intensity. For an accurate comparison of the DFD models we need to satisfy these general assumptions. This is done by performing geometric, radiometric and PSF calibrations.

### *A. Focus Distance from Sensor Plane*

In the calibration process we need to find the pairing between depth and PSF for a given camera setting. The camera parameters that we vary are the aperture size and focus distance. In Sec. II-A we used the thin lens model and assumed the object and sensor distances to be from the center of projection which is at the center of the thin lens. However for real lenses such a center of projection does not exist. Furthermore the blur formation model assumes that the sensor is moved between taking images. But in practice the sensor to object distance is fixed and only the lens system is moved. In our experiments we used a 50 mm prime lens with focus marks on it. These focus distances indicate the distance from the sensor plane to the plane in focus [13]. In the calibration process the distance between the calibration grid and the sensor position is measured manually to avoid modelling the real lens.

### *B. Setup and Image Preprocessing*

For PSF calibration, we use a grid of disks with a known radius and spacing as the calibration image. The patches containing disks are identified and the disk centers are estimated by finding the centroid of those patches. The advantage of using disk images over a checkerboard pattern is that centroid estimation is more robust to defocus blur than corner estimation especially when the aperture is non-circular. In addition, the checkerboard is dominated by just two orientations.

A 24 inch LED display of resolution $1920 \times 1200$ is used in our work. We capture raw images using a Nikon D90 camera with a 50 mm prime-lens, under varying focus and aperture settings, and placing the camera fronto-parallel to the display at different distances. The processing pipeline includes radiometric correction of the raw images, magnification correction and alignment, and normalization of the average image intensity. We render different calibration patterns on the display as well as noise and natural image texture patterns for the DFD experiments.

Fig. 2 shows the calibration and test images that were captured in our experimental setup. We use the disk pattern in Fig.2b for PSF estimation and the textured image in Fig. 2c for depth estimation. We use images of three more textured images, two of them from the Brodatz texture library. Depth estimation accuracy for them is similar to the $1/f$ texture. The image of the grid of dots in Fig. 2a is used for qualitative comparison only.

Single pixel (or dot) images approximate the impulse function. To closely approximate the impulse function the image has to be taken from beyond a certain distance. With images of disks we do not have to strictly satisfy such distance constraint. The image capturing distance also becomes important when taking photos of the noise pattern since we want the noise distribution to be satisfied in the captured image. If the images are taken close-up then the color filter array (CFA) of the display and size of display pixel will modify the noise distribution. Furthermore images of dot patterns require long exposure time. For large blurs,

single pixel images suffer from low SNR problem and may not even be visible. Therefore it is more convenient to use disk images. Noise patterns [3], [2] also suffer from similar problems due to large blurs. Kee et al. [4] also used a disk pattern. However their pattern is applicable for small amount of blur. This is because they are estimating the intrinsic blur of the lens system (i.e. blur that is present even when the image is supposed to be in focus). The calibration pattern proposed by Joshi et al. [1] can estimate relatively large amounts of blur.

Our optimization approach is closest to [1], [5], [14]. However our optimization objective also uses the image gradient and has boundary constraints. Using our calibration approach we estimated blurs of up to size $77 \times 77$ pixels (Fig. 3). For the camera settings used in the depth estimation experiments the largest blur kernel is of size $51 \times 51$.

*Radiometric Correction:* We take images of the LED display with different textures rendered on it. The display has certain radiometric characteristics and the image formed on the sensor plane also has its own characteristics that depend on the camera parameters. For different positions these characteristics can also change to some extent. As a result, a uniform scene will appear non-uniform in the captured image. Most DFD models do not take this into account and so the captured images need to be pre-processed before applying any DFD model. In our experiments we estimate the combined effect of the display and the camera's radiometric properties. For this we render a uniform color on the display and capture images of it for the camera parameters we are calibrating for. Then a quadric surface is fit to the image with its center and curvature estimated from the observed image. The model used in this work is:

$$\begin{aligned} R &= (x - x_0)^2 + (y - y_0)^2 \\ V &= a + bR + cR^2 + dR^3 + eR^4. \end{aligned} \tag{4}$$

This is fit using least squares with the ceres-solver software [15]. The color filter array on the monitor and on the camera sensor can produce undesirable Moiré patterns. In our experiments we found that using a robust penalty function to account for Moiré patterns does not significantly change the quadric surface parameters. Estimating the center of the quadric results in a better fit (in terms of reduction in variance in the corrected image). For numerical stability the data points need to be centered and scaled.

*Magnification Correction and Alignment:* Images taken with different focus settings will have a difference in magnification. DFD methods assume that the same pixel from a pair of images corresponds to the same scene point. Watanabe and Nayar in [16] used telecentric optics to keep the magnification factor constant between the two differently focused images. However most consumer lenses are not telecentric. As a result the pair of defocused images need to be registered before applying any blur estimation algorithm. For magnification correction we find an affine transformation between the disk centers for two different camera settings.

## IV. PSF and Relative Blur Estimation

### A. Blur PSF Estimation

After radiometric correction of the calibration image, 25 disk patches are extracted from the center of the image and averaged. Then the latent sharp disk image is created based on the projected disk center distance. The absolute PSF is estimated by taking a sharp and a blur image pair and solving the following Quadratic Programming (QP) problem.

$$\begin{aligned} \underset{h}{\operatorname{argmin}} & \sum_{j=1}^{n} \lambda_j \|f_j * (i_S * h - i_B)\|_2^2 \\ & + \lambda_{n+1}\|\nabla h\|_2^2 + \lambda_{n+2}\|R \circ h\|^2 \\ \text{subject to } & \|h\|_1 = 1\,,\ h \geq 0. \end{aligned} \tag{5}$$

In the above optimization problem, $i_B$ is the observed blurred image and $i_S$ is the sharp image. $h$ is the PSF kernel that is to be estimated. $f_j$ is a filter that is applied to the images. In the experiments, we use $f_1 = \delta$, $f_2 = G_x$, and $f_3 = G_y$, where $G_*$ is the spatial derivative of a Gaussian in the horizontal and vertical directions. The matrix $R$ – in the element-wise product with the kernel $h$ – is a spatial regularization matrix which in this case is a parabola to ensure that the kernel goes to zero near the edge. The constraints ensure that the kernel is non-negative and preserves the mean intensity after convolution. The optimization function is similar to the one proposed by Ens and Lawrence except in this case we formulate the problem in 2D and in the filter space with explicit non-negativity and unity constraints. The convolution operation and derivative of the kernel operators can be expressed using a convolution matrix [17] and the optimization problem can be solved using off-the-shelf QP solvers (in our case Matlab's `quadprog`).

Estimated PSFs are shown in Fig. 4 along with their corresponding dot images. Since we use quadratic cost on the gradient, it does not suppress small noise. It is possible to use a second optimization stage consisting of iterative shrinkage and thresholding to obtain less noisy and sharper PSFs. However in our experiments we use simple median filtering to get rid of most of the noise in the estimated PSF. Compared to Joshi et al. [1], we use both the original images and their gradients. We also have a compactness constraint similar to [5], [14]. Ens and Lawrence assumed a circularly symmetric kernel and formulated a 1D kernel estimation problem. Like Joshi et al. they only considered the reconstruction error of the image.

### B. Relative Blur PSF Estimation

For the relative blur PSF estimation we take the absolute PSFs and use Eq. 5 by assigning the sharper and blurrier PSFs to $i_S$ and $i_B$ respectively. Here $\lambda_{n+2} = 0$ to relax the compactness constraint for the relative blur kernel. For more

robustness, the corresponding defocused disk images are used along with the absolute PSFs. We can also add defocus blurred images of textures to further improve relative blur estimation. However we found the PSFs and disk pairs to be sufficient. Adding additional images is equivalent to adding the convolution matrices together.

## V. PSF Evaluation

The PSF estimation method is evaluated qualitatively using images of a single pixel and quantitatively using different DFD models. Relative blur estimation accuracy is evaluated using the PSF reconstruction error and also depth estimation accuracy.

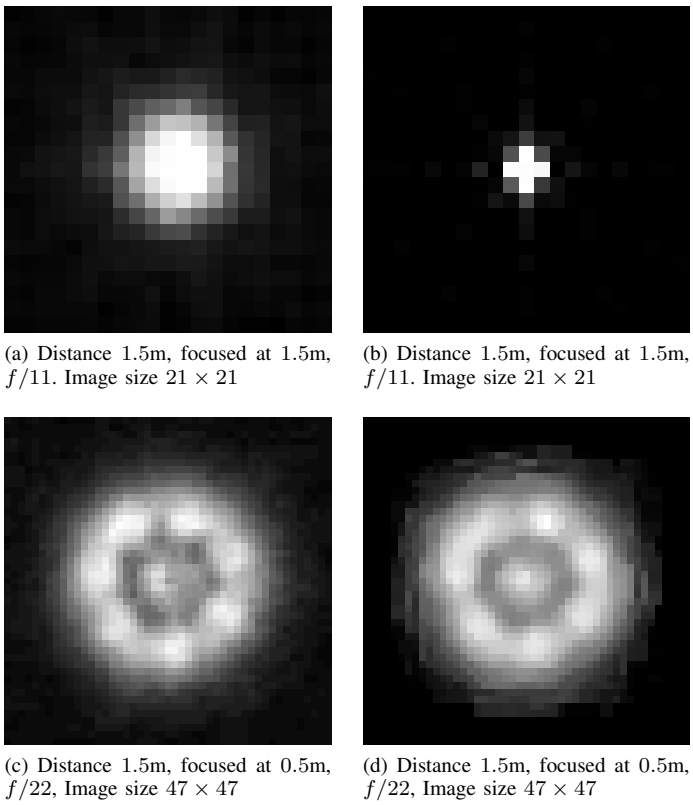
### A. Absolute PSF Estimation

Fig. 3 shows an example of the absolute blur estimation process. Our method only requires a single defocused disk image as shown in Fig. 3a. The true sharp image of the disk is estimated from the size of the projected disk grid. This is because the radius of the disks is a known fraction of the distance between disk centers. Using Eq. 5 we get an estimated PSF as shown in Fig. 3c which is similar to the corresponding single pixel observed image shown in Fig. 3d.

Fig. 4 shows some more comparisons between observed single pixel image and estimated PSFs. Fig. 4a shows the observed image of a real point light source that is in focus. Since the point source is in focus we would expect the image i.e. the PSF to be a point. But due to the finite size of the point source we do not see a point PSF. On the contrary, our calibration disk based PSF estimation process can overcome such limitations and estimate a PSF (Fig. 4b) that is closer to the true PSF. Fig. 4c shows an example where a defocused image is taken with a very small aperture. The small size of the aperture creates diffraction effects which is captured in the PSF estimated from the defocused disk image (Fig. 4d).

### B. Relative Blur PSF Estimation

In Fig.5 we show examples of relative blur estimation using the coded apertures proposed in Zhou et al. [11], pillbox, and estimated absolute blur PSFs. For the synthetic apertures, we take the pair of apertures and simulate the variable focus configuration with focus distances $0.7$ m and $1.22$ m, $f/11$, and $\rho = 180$ pixels-per-mm. The samples correspond to inverse depths $1.6$ D and $0.6$ D. The estimated blurred PSFs (right-most column) are obtained by convolving the sharper PSFs (left-most column) with the estimated relative blurs (3rd column). We can see that the estimated blurrier PSFs are reasonably close to the true blurrier PSFs (2nd column). For instance for the coded apertures the relative blur PSFs capture the hole in the aperture, orientation and boundary of the hole correctly. For the real PSF (last row of Fig. 5), the shape of the aperture-stop and the diffraction effects are also captured accurately. A Gaussian approximation or circularly symmetric constraint would not be able to model such complex shapes.




(a) Distance $1.5$m, focused at $1.5$m, $f/11$. Image size $21 \times 21$

(b) Distance $1.5$m, focused at $1.5$m, $f/11$. Image size $21 \times 21$

(c) Distance $1.5$m, focused at $0.5$m, $f/22$, Image size $47 \times 47$

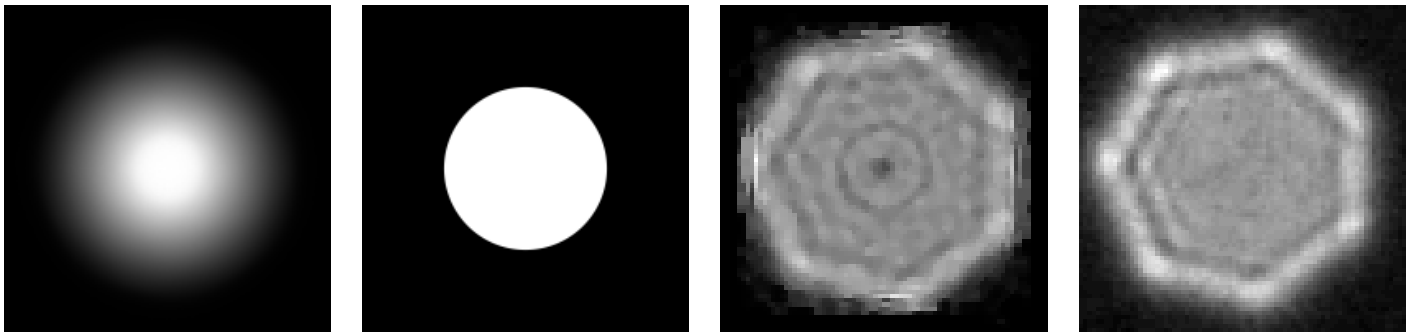
(d) Distance $1.5$m, focused at $0.5$m, $f/22$, Image size $47 \times 47$

Figure 4: (a) and (c) are examples of PSFs extracted from a single pixel on the computer monitor (i.e. similar to Fig. 2a). (b) and (d) show the corresponding estimated PSFs from a calibration grid of disks (i.e. similar to Fig. 2b). For (a) and (b) the camera is focused on the object and in (c) and (d) the camera is defocused. When the PSF is a delta function (i.e. (a) and (b) where the camera is focused on the object), the estimation process finds a sharper PSF than the observed single pixel image. Diffraction effects such as the valley in (c) are also captured in the estimated PSF in (d).

### C. DFD using estimated PSFs

In this section we evaluate the quality of the estimated PSFs and the relative blur kernels using DFD with real and synthetically defocused images. For the real experiments, we capture images of fronto-parallel textures for different object-to-sensor distances and camera settings. The images are captured simultaneously with the calibration pattern discussed in the previous section. This allows us to find the corresponding PSFs for the defocused images. For this experiment we use the variable focus configuration with the camera settings from the previous section.

We use $27$ object-to-sensor distances ranging between $0.61$ m and $1.5$ m spaced uniformly (roughly) in inverse



(a) Observed blurred disk image (b) Estimated sharp disk (c) Estimated PSF from (a) and (b) (d) Observed pixel (green channel)

Figure 3: Example of PSF estimation from observed blurred disk image. (a) Observed disk ($199 \times 199$ pixels), (b) estimated sharp disk image based on the projected size of the disk grid, (c) PSF estimated using Eq. 5, and (d) image of a single pixel (green channel). Object to sensor distance is $1.5$ m and focus distance $0.5$ m, and f-number $f/11$. The size of the PSF kernel is $77 \times 77$. Note that diffraction effects (e.g.ringing, brighter corners, etc.) as well as the aperture-stop's shape are also captured in the estimated PSF.

depth space. We choose uniform subdivision in inverse depth space because the blur radius changes linearly with inverse depth (recall Eq. 2). In practice, the relationship is approximately linear because we are moving the lens instead of the sensor. The captured textured images go through the same pre-processing steps as the calibration images, namely – radiometric correction, scaling and alignment, and intensity normalization by the mean intensity.

For the synthetic experiments, we use the coded aperture pair proposed in [11]. Using the aperture templates we generate a set of PSF kernels of different sizes (using Eq. 2) and orientation (based on the sign of the blur), and their corresponding relative blur kernels. The camera parameters are the same as in the relative blur experiment. Similar to the real experiment, the scene is considered to be within $0.61$ m to $1.5$ m and therefore extends on both sides of the focal planes. A $512 \times 512$ image of $1/f$ noise texture is synthetically blurred with the PSF pair that is being used for evaluation. This is followed by adding additive Gaussian noise $\sigma_n = 2\%$ to the blurred images.

In all the experiments, depth estimation is performed by choosing the appropriate PSFs for every depth hypothesis and evaluating the model cost. The per-pixel cost is then averaged over a finite window and the depth label is chosen to be the one that minimizes the cost at every pixel.

For evaluating the relative blur kernel estimation accuracy, we consider Zhou et al's coded aperture pair [11]. Fig. 5 showed a couple of the example aperture pairs for this case. Fig. 6a shows the depth estimation accuracy using the estimated relative blur kernel and the deconvolution method from [11] with the ground-truth PSF pairs.

Fig. 6b shows an example of depth estimation with real defocused images and the estimated relative blurs and the absolute blurs (BET). In both cases the true depth is within two sigma of the estimated mean. In [12] we use the estimated absolute and relative blurs to evaluate different DFD models.

## VI. Conclusion

In this paper we presented a simple and robust approach for absolute blur and relative blur kernel estimation. Estimated absolute blurs were qualitatively compared with corresponding single pixel images. We showed that our approach is able to estimate blur kernels ranging from single pixel to reasonably large kernels (e.g. $77 \times 77$). We also showed results for relative blur kernel estimation. To our knowledge besides the work by Ens and Lawrence there has not been any work on relative blur kernel estimation. Furthermore Ens and Lawrence assumes circularly symmetric relative blur kernels but ours is more flexible and we have demonstrated its effectiveness using complex coded aperture pair [11] as well as conventional aperture. We avoided issues with real lens modelling by measuring distance from the sensor plane and performing calibration for each of those distances. We have experimentally showed that our estimated PSFs and relative blurs can be used for depth from defocused.


## Acknowledgements

We would like to thank Tao Wei for help with an early version of camera calibration. This work was supported by grants from the Natural Sciences and Engineering Research Council of Canada (NSERC). Computations were performed on the HPC platform Guillimin from McGill University, managed by Calcul Qubec and Compute Canada. The operation of this compute cluster is funded by the Canada Foundation for Innovation (CFI), NanoQubec, RMGA and the Fonds de recherche du Qubec - Nature et technologies (FRQ-NT).

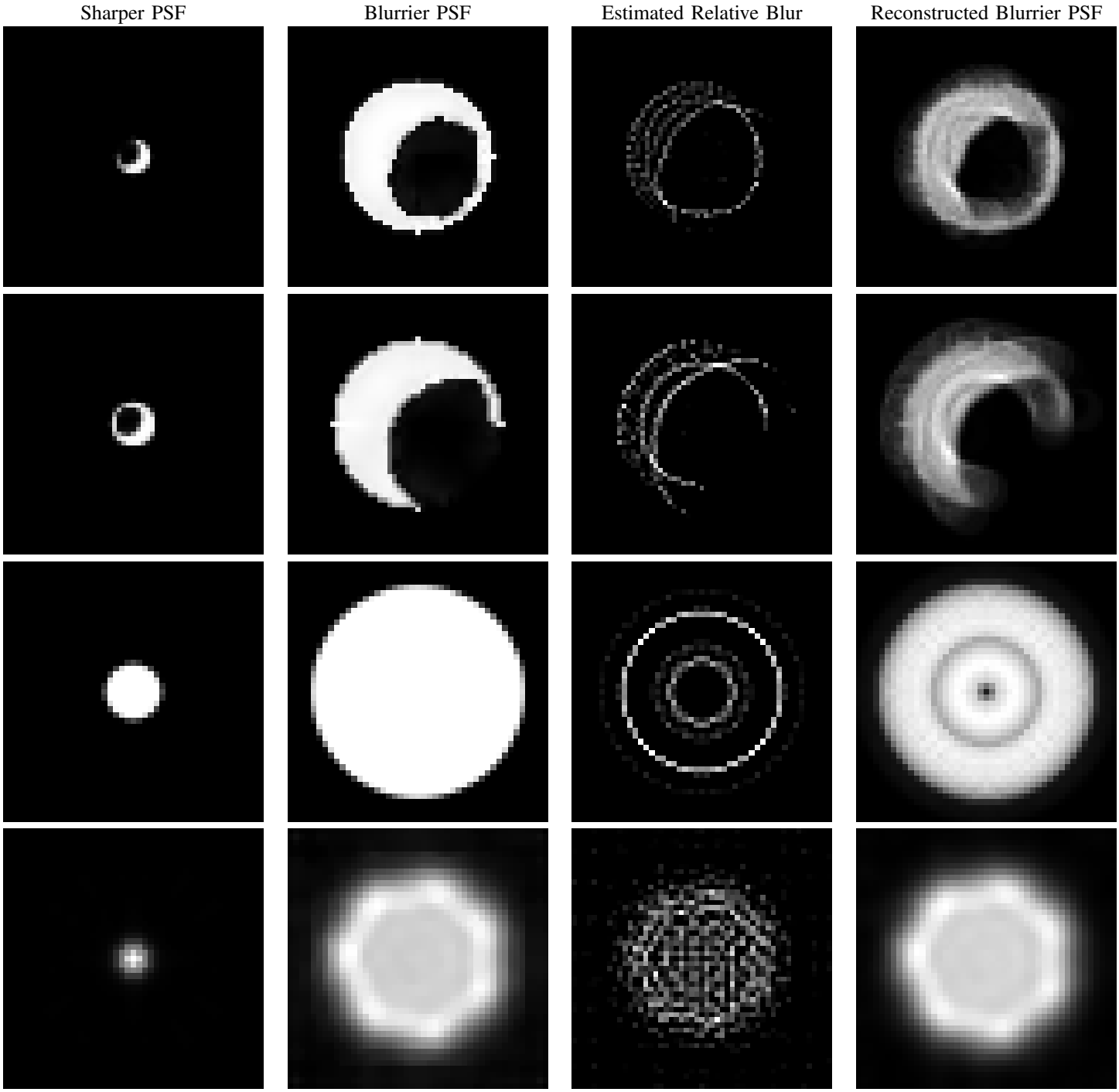
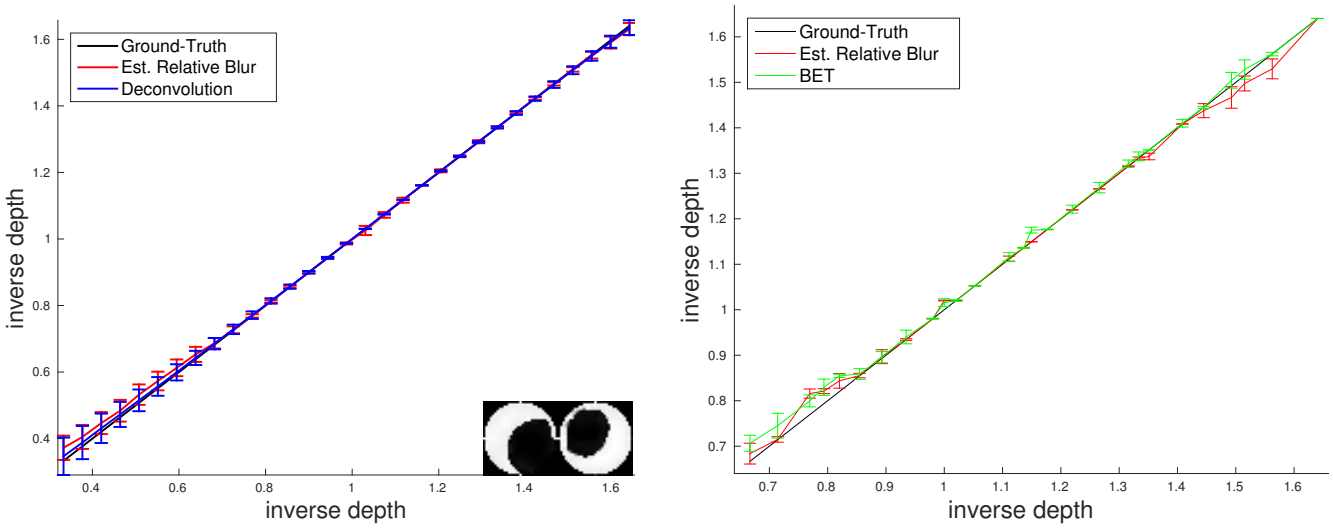



Figure 5: Examples of relative blur estimated from coded aperture [11] (first two rows), pillbox (third row), and real PSF (last row), and reconstructing the blurrier PSF from the sharper PSF using the estimated relative blur. Top row corresponds to inverse depth $1.6$ D and the rest to $0.6$ D with variable focus distances $0.7$ m and $1.22$ m and $f/11$. The reconstructed PSFs correctly capture the open and closed shape of the coded apertures. The corresponding depth estimation is shown in Fig. 6a (coded aperture) and Fig. 6b (real PSF). More examples can be found at http://cim.mcgill.ca/~fmannan/relblur.html




(a) Zhou et al's PSF (synthetic), $f/11$

(b) Estimated PSF real defocus, $f/11$

Figure 6: Examples of depth estimation using the coded aperture pair from [11] and estimated PSFs, with the same camera and scene configuration as Fig. 5. a) Shows that the relative blur estimated from the coded aperture gives similar results to the deconvolution based method with ground-truth PSFs [11]. (b) shows that the estimated PSFs and their relative blur can recover depth under most cases.

## REFERENCES


[1] N. Joshi, R. Szeliski, and D. Kriegman, "Psf estimation using sharp edge prediction," in *CVPR*, June 2008, pp. 1–8.

[2] M. Delbracio, P. Musé, A. Almansa, and J.-M. Morel, "The non-parametric sub-pixel local point spread function estimation is a well posed problem," *IJCV*, vol. 96, no. 2, pp. 175–194, 2012.

[3] A. Mosleh, P. Green, E. Onzon, I. Begin, and J. Pierre Langlois, "Camera intrinsic blur kernel estimation: A reliable framework," in *CVPR*, 2015, pp. 4961–4968.

[4] E. Kee, S. Paris, S. Chen, and J. Wang, "Modeling and removing spatially-varying optical blur," in *ICCP*, April 2011, pp. 1–8.

[5] J. Ens and P. Lawrence, "An investigation of methods for determining depth from focus," *PAMI*, vol. 15, no. 2, pp. 97–108, 1993.

[6] A. P. Pentland, "A new sense for depth of field," *PAMI*, vol. 9, pp. 523–531, July 1987.

[7] M. Subbarao, "Parallel depth recovery by changing camera parameters," in *ICCV*, dec 1988, pp. 149–155.

[8] M. Subbarao and G. Surya, "Depth from defocus: A spatial domain approach," *IJCV*, vol. 13, no. 3, pp. 271–294, 1994.

[9] T. Xian and M. Subbarao, "Depth-from-defocus: Blur equalization technique," *SPIE*, vol. 6382, 2006.

[10] A. Levin, R. Fergus, F. Durand, and W. T. Freeman, "Image and depth from a conventional camera with a coded aperture," *ACM Trans. Graph.*, vol. 26, no. 3, p. 70, 2007.

[11] C. Zhou, S. Lin, and S. Nayar, "Coded Aperture Pairs for Depth from Defocus and Defocus Deblurring," *IJCV*, vol. 93, no. 1, p. 53, May 2011.

[12] F. Mannan and M. S. Langer, "What is a good model for depth from defocus?" in *CRV*, 2016.

[13] R. Kingslake, *Optics in Photography*, ser. SPIE press monographs. SPIE Optical Engineering Press, 1992.

[14] S. M. Seitz and S. Baker, "Filter flow," in *ICCV*, 29 2009-oct. 2 2009, pp. 143 –150.

[15] S. Agarwal, K. Mierle, and Others, "Ceres solver," http://ceres-solver.org.

[16] M. Watanabe and S. Nayar, "Telecentric optics for focus analysis," *PAMI*, vol. 19, no. 12, pp. 1360 –1365, dec 1997.

[17] P. Hansen, J. Nagy, and D. O'Leary, *Deblurring Images*. Society for Industrial and Applied Mathematics, 2006.