PSF Estimation using Sharp Edge Prediction

Neel Joshi[†] Richard Szeliski[*] David J. Kriegman[†]

[†]University of California, San Diego [*]Microsoft Research

Abstract

Image blur is caused by a number of factors such as motion, defocus, capturing light over the non-zero area of the aperture and pixel, the presence of anti-aliasing filters on a camera sensor, and limited sensor resolution. We present an algorithm that estimates non-parametric, spatially-varying blur functions (i.e., point-spread functions or PSFs) at sub-pixel resolution from a single image. Our method handles blur due to defocus, slight camera motion, and inherent aspects of the imaging system. Our algorithm can be used to measure blur due to limited sensor resolution by estimating a sub-pixel, super-resolved PSF even for in-focus images. It operates by predicting a "sharp" version of a blurry input image and uses the two images to solve for a PSF. We handle the cases where the scene content is unknown and also where a known printed calibration target is placed in the scene. Our method is completely automatic, fast, and produces accurate results.

1. Introduction

Image blur is introduced in a number of stages in a camera. The most common sources of image blur are motion, defocus, and aspects inherent to the camera, such as pixel size, sensor resolution, and the presence of anti-aliasing filters on the sensor.

When blur is undesirable, one can deblur an image using a deconvolution method, which requires accurate knowledge of the blur kernel. In applications where blur is desirable and essential, such as shape from defocus, it is still necessary to recover the shape and size of the spatially varying blur kernel.

Recovering a PSF from a single blurred image is an inherently ill-posed problem due to the loss of information during blurring. The observed blurred image provides only a partial constraint on the solution, as there are many combinations of PSFs and "sharp" images that can be convolved to match the observed blurred image.

Prior knowledge about the image or kernel can disambiguate the potential solutions. Early work in this area significantly constrained the form of the kernel [6], while more recently, researchers have put constraints on the underlying sharp image [3]. In our work, we take the latter approach; however, instead of using statistical priors, we leverage our prior assumption more directly. Specifically, we present an algorithm for estimating regions of a sharp image from a blurry input—if one can estimate the sharp image, recovering the blur kernel is possible.

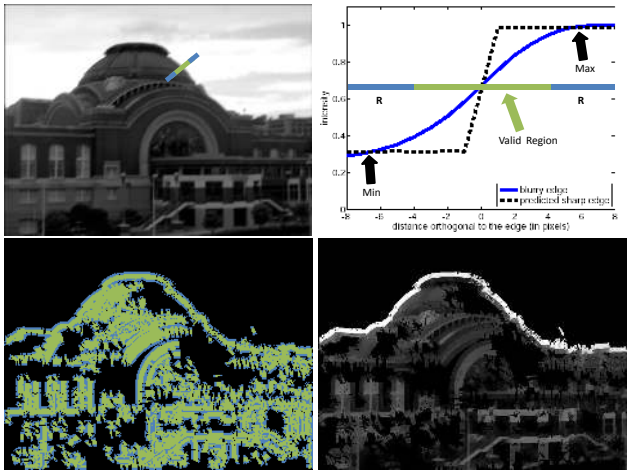


Figure 1. Sharp Edge Prediction. A blurry image (top left) and the 1D profile normal to an edge (top right, blue line). We predict a sharp edge (top right, dashed line) by propagating the max and min values along the edge profile. The algorithm uses predicted and observed values to solve for a PSF. Only observed pixels within a radius R are used. (bottom left) Predicted pixels are blue and valid observed pixels are green. (bottom right) The predicted values.

The key insight of our work is that with certain types of image blur, the location of image features such as edges are detectable even if the feature strength is weakened. When the scene content is unknown, we detect edges and predict the underlying sharp edges that created the blurred observations, under the assumption that detected edge was a step edge before blurring. Each pair of predicted and blurred edges gives information about a radial profile of the PSF. If an image has edges spanning all orientations, the blurred and predicted sharp image contain enough information to solve for a general two-dimensional PSF.

For situations where the scene content can be controlled, we have designed a printed calibration target whose image is automatically aligned with a known representation of the target. We then use this pair to solve for an accurate PSF.

Our method has several advantages over previous approaches: it measures the entire PSF of an image system from world to image, it is fast and accurate, and it can solve for spatially varying PSFs at sub-pixel resolution using only a single image.

We show results for both unknown scenes and images of our calibration target. We present deconvolution results using the recovered PSFs to validate the blur kernels and show a synthetic experiment to further evaluate the method.

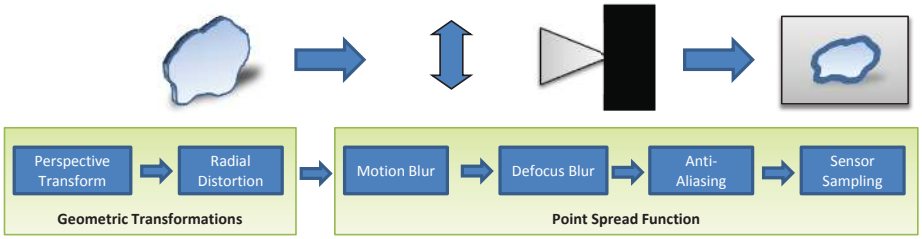


Figure 2. Image Formation Model. The imaging model consists of two geometric transforms as well as blur induced by motion, defocus, sensor anti-aliasing, and finite-area sensor sampling. We solve for an estimate of the continuous point-spread function at each discretely sampled (potentially blurry and noisy) pixel.

We also show that by solving for spatially varying, per-color channel PSFs combined with per-channel radial distortion corrections, we can remove chromatic aberrations artifacts.

2. Related Work

The problem of blur kernel estimation and more generally blind deconvolution is a longstanding problem in computer vision and image processing. The entire body of previous work in this area is beyond what can be covered here. For a more in depth study of much of the earlier work in blur estimation, we refer the reader to the survey paper by Kundur and Hatzinakos [6].

In the computer vision literature, classical shape-from-defocus [10] addresses PSF estimation using a parametric model for blur that is either a "pillbox" or 2D Gaussian function with a single parameter for the PSF size, i.e., focal length or kernel radius. For more complex blurs, such as motion blur, many recent single-image estimation techniques model blurs as a collections of 1D or 2D box blurs and use segmentation techniques to handle multiple motions [7, 4, 2]. Shan *et al*. [12], on the other hand, use a low-parameter model to remove motion blur due to an object translating and rigidly rotating about an axis parallel to the camera's optical axis. In contrast with this previous work, we do not use a parametric model for the PSF and solve for spatially varying kernels without performing any explicit segmentation.

There is significantly less work in the area of single image blur estimation using non-parametric kernels. The work by Fergus *et al*. [3] is perhaps the most notable method of this type. Fergus *et al*. use natural image statistics to derive an image prior that is used in a variational Bayes formulation. In contrast, we leverage prior assumptions on images to directly predict the underlying sharp image. We consider our approach complementary to that of Fergus *et al*., as our method excels at accurately computing smaller kernels, and it can be used for lens and sensor characterization. Their method is not as well suited to these applications, but excels at computing large kernels due to complex camera motion, which is outside the scope of our work.

Our work is conceptually most similar to slant-edge calibration [11, 1]. These methods recover 1D blur profiles by imaging a slanted edge feature and finding the 1D kernel normal to the edge profile that gives rise to the blurred observations of the known step edge. Reichenbach *et al*. [11] note that one can combine several 1D sections to estimate a 2D PSF. We take a similar approach philosophically to slant-edge techniques, with three major differences: we extend the method to *directly* solve for 2D PSFs, we solve for spatially varying PSFs, and we present a blind approach where the underlying step edge is not know a priori.

A related area is modulation transfer function (MTF) estimation for lenses that uses images of random dot patterns [8]. In theory, infinitesimal dot patterns are useful for PSF estimation, but in practice, it is not possible to create such a pattern. In contrast, creating sharp step edges is relatively easy and thus generally preferable [11]. An additional advantage of our work relative to using dot patterns is that by using a grid-like structure with regular, detectable corner features, we can compute a radial distortion correction in addition to estimating PSFs.

3. Image Formation Model

We now give a brief overview of relevant imaging and optics concepts needed for PSF estimation. As illustrated in Figure 2, the imaging model consists of two geometric transforms: a perspective transform (used when photographing a known planar calibration target) and a radial distortion. There are several sources of blur induced by motion, defocus, sensor anti-aliasing, and pixel sampling area (fill factor and active sensing area shape). We model all blur as a convolution along the image plane and account for depth dependent defocus blur and 3D motion blur by allowing for the PSF to be spatially varying.

Our method estimates a discretely sampled version of the continuous PSF by either matching the sampling to the image resolution (which is useful for estimating large blur kernels) or using a sub-pixel sampling grid to estimate a detailed PSF, which captures effects such as anti-aliasing of the sensor and allows us to do more accurate image restoration. In addition, by computing a sub-pixel PSF, we can perform single-image super-resolution by deconvolving up-sampled images with the recovered PSF.

Geometric Transformations: The world to image transformation consists of a perspective transform and a

radial distortion. With the blind method, we ignore the perspective transform and operate in image coordinates.

With the non-blind method, where we photograph a known calibration target, we model the perspective transformation as a 2D homography to map known feature locations F^k on the grid pattern to detected feature points from the image F^d. We use a standard model for radial distortion: $(F'_x, F'_y)^T = (F_x, F_y)^T(a_0 + a_1 r^2(x,y) + a_2 r^4(x,y))$, where $r(x,y) = \sqrt{F_x^2 + F_y^2}$ is the radius relative to the image center.

Given a radial distortion function $R(F)$ and warp function which applies a homography $H(F)$, the full alignment process is $F^d = R(H(F^k))$. We compute the parameters that minimize the L_2 norm of the residual $||F^d - R(H(F^k))||^2$. Computing these parameters cannot be done simultaneously in closed form. However, the problem is bilinear, and thus we solve for the parameters using an iterative approach.

Modeling the Discrete Point-Spread Function: The equation for the observed image B is a convolution of a kernel K and a potentially higher resolution sharp image I, plus additive Gaussian white noise, whose result is potentially down-sampled:

$$B = D(I \otimes K) + N, \tag{1}$$

where $N \sim \mathcal{N}(0, \sigma^2)$. $D(I)$ down-samples an image by point-sampling $I_L(m,n) = I(sm, sn)$ at a sampling rate s for integer pixel coordinates (m,n). In our formulation, the kernel K models all blurring effects, which are potentially spatially varying and wavelength dependent.

4. Sharp Image Estimation

The blurring process is formulated as an invertible linear system, which models the blurry image as the convolution of a sharp image with the imaging system's PSF. Thus, if we know the original sharp image, recovering the kernel is straightforward. The key contribution of our work is a reliable and widely applicable method for predicting a sharp image from a single blurry image. In the following section, we present our methods for predicting the sharp image. In Section 5, we discuss how to formulate and solve the invertible linear system to recover the PSF. In the following discussion, we consider images to be single channel or grayscale; in Section 6, we discuss color images.

4.1. Blind Estimation

For blind sharp image prediction, we assume blur is due to a PSF with a single mode (or peak), such that when an image is blurred, the ability to localize a previously sharp edge is unchanged; however, the strength and profile of the edge is changed, as illustrated in Figure 1. Thus, by localizing blurred edges and predicting sharp edge profiles, locally estimating a sharp image is possible.

We assume that all observed blurred edges result from convolving an ideal step edge with the unknown kernel. Our algorithm finds the location and orientation of edges in the blurred image using a sub-pixel difference of Gaussians edge detector. It then predicts an ideal sharp edge by finding the local maximum and minimum pixel values, in a robust way, along the edge profile and propagates these values from pixels on each side of an edge to the sub-pixel edge location. The pixel on the edge itself is colored according to the weighted average of the maximum and minimum values according to the distance of the sub-pixel location to the pixel center, which is a simple form of anti-aliasing (see Figure 1).

To find the maximum value, our algorithm marches along the edge normal, sampling the image looking for a local maximum using hysteresis. Specifically, the maximum location is the first pixel that is less than 90% (as opposed to strictly less than) of the previous value. Once this value and location are identified, we store the "maximum" value as the mean of all values along the edge profile that are within 10% of the initial maximum value. An analogous approach is used for the minimum.

Since we can only reliably predict values near edges, we only use observed pixels within a radius of the predicted sharp values. These locations are stored as valid pixels in a mask, which is used when solving for the PSF, as discussed in Section 5. At the end of the prediction process, we have a partially estimated sharp image, as shown in Figure 1.

4.2. Non-Blind Estimation

For non-blind sharp edge prediction, we want to compute the PSF given that we know the sharp image. Since we anticipate using this technique in a controlled lab setup, we designed a special calibration pattern for this purpose (Figure 3). We take an image of this pattern and align the known grid pattern to the image to get the sharp/blurry pair needed to compute the PSF accurately. The grid has corner (checkerboard) features so that it can be automatically detected and aligned, and it also has sharp step edges equally distributed at all orientations within a tiled pattern, so that it provides edges that capture every radial slice of the PSF. (Alternatively, we can say that the calibration patterns provides measurable frequencies at all orientations.) Furthermore, we represent the grid in mathematical form (the curved segments are 90° arcs), which gives us a very precise definition for the grid, which is advantageous for performing alignment.

For non-blind prediction, we continue to assume that kernel has no more than a single peak. Thus even when the pattern is blurred, we can detect corners on the grid with a sub-pixel corner detector. Because our corners are actually balanced checkerboard crossings (radially symmetric), they do not suffer from "shrinkage" (displacement) due to blur-

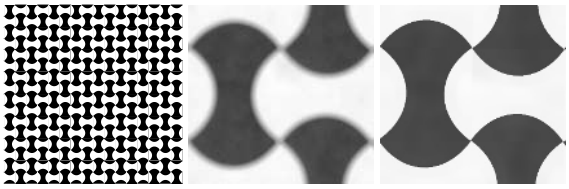


Figure 3. Non-Blind Estimation. (left) The tiled calibration pattern, (middle) cropped section of an image of a printed version of the grid, and (right) the corresponding cropped part of the known grid warped and shaded to match the image of the grid.

ring. Once corners are found, the ground truth pattern is aligned to the acquired image. To obtain an accurate alignment, we correct for both geometric and radiometric aspects of the imaging system.

We perform geometric alignment using the corrections discussed in Section 3. We fit a homography and radial distortion correction to match the known feature locations on the grid pattern to corners detected with sub-pixel precision on the acquired (blurry) image of the printed grid.

We also must account for the lighting and shading in the image of the grid. We do this by first aligning the known grid to the image. Then, for each edge location (as known from mathematical form of the ground truth grid pattern), the algorithm finds the maximum and minimum values on the edge profile and propagates them just as in the non-blind approach. We shade the grid for pixels within the blur radius of each edge. By performing the shading operation, our algorithm has corrected for shading, lighting, and radial intensity falloff. Figure 3 shows the results of the geometric warp and shading transfer.

5. PSF Estimation

Once the sharp image is predicted, we estimate the PSF as the kernel that, when convolved with the sharp image, produces the blurred input image. We formulate the estimation using a Bayesian framework solved using a *maximum a posteriori* (MAP) technique. In MAP estimation, one tries to find the most likely estimate for the blur kernel K given the sharp image I and the observed blurred image B, using the known image formation model and noise level.

We express this as a maximization over the probability distribution of the posterior using Bayes' rule. The result is minimization of a sum of negative log likelihoods $L(.)$:

$$P(K|B) = P(B|K)P(K)/P(B) \quad (2)$$

$$\underset{K}{\operatorname{argmax}}\, P(K|B) = \underset{K}{\operatorname{argmin}}\, L(B|K) + L(K). \quad (3)$$

The problem is now reduced to defining the negative log likelihood terms. Given the image formation model (Equation 1), the data term is:

$$L(B|K) = ||M(B) - M(I \otimes K)||^2/\sigma^2. \quad (4)$$

(The downsampling term D in (1) will be incorporated in Section 5.1.) $M(.)$ is a masking function such that this term is only evaluated for "known" pixels in B, i.e., those pixels that result from the convolution of K with properly estimated pixels I, which form a band around each edge point, as described in Section 4.1.

The remaining negative log likelihood term, $L(K)$, models prior assumptions on the blur kernel and regularizes the solution. We use a smoothness prior and a non-negativity constraint. The smoothness prior penalizes large gradients and thus biases kernel values to take on values similar to their neighbors: $L_s(K) = \lambda\gamma||\nabla K||^2$. λ controls the weight of the smoothness penalty, and $\gamma = (2R+1)^2$ normalizes for the kernel area (R is the kernel radius). Since the kernel should sum to one (as blur kernels are energy conserving) the individual values decrease with increased R. This factor is needed to keep the relative magnitude of kernel gradient values on par with the data term values regardless of kernel size.

We minimizing the following error function:

$$L = ||M(B) - M(I \otimes K)||^2/\sigma^2 + \lambda\gamma||\nabla K||^2, \quad (5)$$

subject to $K_i \geq 0$, to solve for the PSF using non-negative linear least squares using a projective gradient Newton's method. We currently estimate the noise level σ using a technique similar to that of Liu *et al*. [9], and we have empirically found $\lambda = 2$ to work well.

5.1. Computing a Super-Resolved PSF

By taking advantage of sub-pixel edge detection for blind prediction and sub-pixel corner detection for non-blind prediction, we can estimate a super-resolved blur kernel by predicting a sharp image at a higher resolution than the observed image.

For the blind method, in the process of estimating the sharp image, it is necessary to rasterize the predicted sharp edge-profile back onto a pixel grid. By rasterizing the sub-pixel sharp-edge profile onto an up-sampled grid, we can estimate a super-resolved sharp image. In addition, at the actual identified edge location (as before), the pixel color is a weighted average of the minimum and maximum, where the weighting reflects the sub-pixel edge location on the grid.

For the non-blind method, we also must rasterize the grid pattern at a some desired resolution. Since we detect corners at sub-pixel precision, the geometric alignment is computed with sub-pixel precision. Using the mathematical description of our grid, we can choose any upsampled resolution when rasterizing the predicted sharp image. We also perform anti-aliasing, as described in Section 4.2.

To solve for the PSF using the super-resolved predicted sharp image I_H and the observed (vectorized) blurry image b, we modify Equation 4 to include a down-sampling function according to our image model (Equation 1). We consider $\hat{b}_H = A_H k_H$ to be super-resolved sharp image blurred by the super-resolved kernel k_H, where A_H is the matrix form of I_H. Equation 4 is then $||b - DA_H k_H||^2$ (we

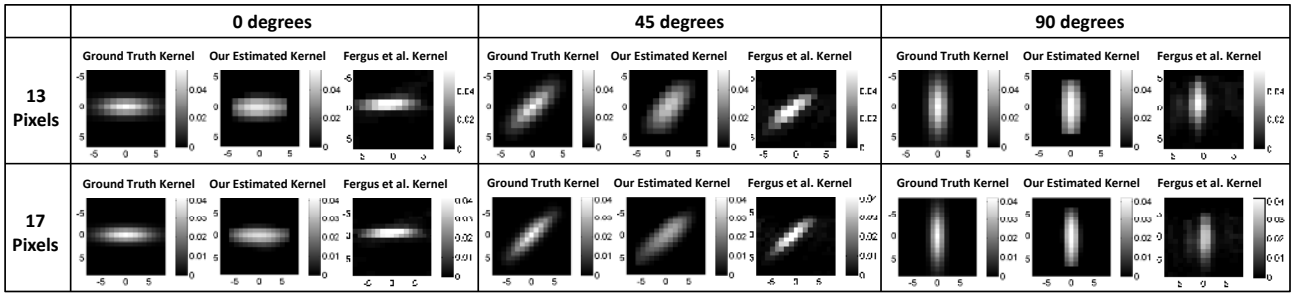


Figure 4. Recovering Blur Kernels of Different Sizes and Orientations. We convolved the sharp original version of the image shown in Figure 1 with kernels of 13 and 17 pixels for three different orientations. Each set is a side by side comparisons of the ground truth (left), our recovered kernel (middle), and the result of running Fergus *et al*.'s [3] method (right).

have left out the masking function for readability). D is a matrix reflecting the down-sampling function: $\hat{B}_L(m,n) = \hat{B}_H(sm,sn)$.

5.2. Computing a Spatially Varying PSF

Computing a spatially varying PSF is straightforward given our formulation—we simply perform the MAP estimation process described in the previous section for sub-windows of the image. The process operates on any size sub-window as long as enough edges at different orientations are present in that window. In the limit, we could compute a PSF for every pixel using sliding windows. We have found, in practice, that such a dense solution is not necessary, as the PSF tends to vary spatially relatively slowly.

Our method requires enough edges to be present at most orientations. When using the entire image, this is not usually an issue; however, when using smaller windows, the edge content may under-constrain the PSF solution. We have a simple test that avoids this problem. We ensure that (a) the number of valid pixels in the mask described in Equation 4 is greater than the number of unknowns in the kernel, and (b) we compute a histogram of 10 degree bins of the detected edges orientations and ensure that each bin contains at least 100 edges. When this check fails, we do not compute a kernel for that window.

6. Chromatic Aberration

In the previous sections, we did not explicitly address solving for PSFs for color images. To handle color, one could convert the image to grayscale. In many cases this is sufficient; however, it is more accurate to solve for a PSF for each color channel. This need arises when chromatic aberration effects are apparent.

Due to the wavelength-dependent variation of the index of refraction of glass, the focal length of a lens varies continually with wavelength. This property causes longitudinal chromatic aberration (blur/shifts along the optical axis), which implies that the focal depth, and thus amount of defocus, is wavelength dependent. It also causes lateral chromatic aberration (blur/shifts perpendicular to the optical axis). We refer the reader to the paper by Kang [5] for a more detailed discussion of these artifacts.

By solving for a PSF per color channel, we can model the longitudinal aberrations; we use a per-color channel radial distortion correction to handle the lateral distortions. We correct for lateral distortions by first performing edge detection on each color channel independently and only keeping edges that are detected within 5 pixels of each other in R, G, and B. We then compute a radial correction to align the R and B edges to the G edges and then perform blind sharp image prediction.

To correct for any residual radial shifts, we use the green edge locations for all color channels so that all color bands have sharp edges predicted at the same locations. One could perform this last step without correcting radial distortion first and allow the shifts to be entirely modeled within the PSF; however, we have found the two stage approach is better, as it removes some aberration artifacts even when there is not enough edge information to compute a PSF, and by removing the majority of the shift first, we can solve for smaller kernels.

If we have access to RAW camera images, we can compute more accurate per-channel PSFs by accounting for the Bayer pattern sampling during PSF computation instead of using the demosaicked color values. We solve for a PSF at the original image resolution, which is 2x the resolution for each color channel and use the point sampling function discussed in Section 3, where the sampling is shifted according to the appropriate Bayer sample location.

7. Results

To validate our blind prediction method, we synthetically blurred a sharp image with oriented Gaussian kernels of 13 and 17 pixels in diameter for three different orientations, added Gaussian white noise with standard-deviation 0.01 (where 0=black and 1=white), and then estimated the blur kernel using our blind method. Figure 4, shows a comparison of the ground truth kernels, our recovered kernels, and the result of running Fergus *et al*.'s method. Our blind algorithm recovers the size and shape of each kernel accurately.

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

Figure 5. Defocus and Slight Motion-Blur. (a) The original blurred image and (b) the deconvolved output using our recovered PSF. (c–d) Zoomed-in versions of the original and deconvolved image respectively. (e) The kernel recovered using the method of Fergus *et al*. [3] and (f) our recovered kernel.

In Figures 5 and 7, we show results for estimating kernels for images with real, unknown blurs, where there is both defocus and camera motion blur. Our method predicts slightly asymmetric disk-like kernels that are consistent with defocus and slight motion blur.

To qualitatively validate these kernels, we deconvolve the input images using the Lucy-Richardson algorithm. We chose this over other methods as it produces results with a good balance of sharpness and noise reduction. Furthermore, the method is less forgiving than some newer methods, which allows for better validation. (Deconvolution with a incorrect kernel leads to increased ringing artifacts, as shown in Figure 6). Our resulting deconvolved images are significantly sharper and show relatively minimal ringing artifacts, which indicates that the kernels are accurate.

In Figure 5, we also compare our recovered kernel to a result from running Fergus *et al*.'s code. The kernel obtained by their method has more noise than ours, does not have a shape consistent with defocus blur, and the size of the non-zero area of the kernel does not match the amount of blur seen in the input image. Fergus *et al*.'s method took 21 minutes, while ours took 2.5 seconds for the original resolution and 9.5 seconds at 2x super-resolution. Our method is significantly faster as its running time scales with the number of edges and kernel size, while the Fergus *et al*. method is a multi-resolution approach whose speed scales with *image size* and kernel size. Our method is a couple seconds faster when using regular least-squares instead of a non-negative version; however, more smoothing is needed to suppress large negative values. Thus we prefer to enforce non-negativity as it produces sharper PSFs.

Figure 8 displays an image with camera motion blur. Our recovered kernel correctly shows the diagonal motion blur that is apparent in the input image. The deconvolved image is much sharper with minimal ringing.

Figure 6. Kernel Size and Orientation. Image deconvolved with (left) our kernel, (middle) our kernel scaled 20% larger, and (right) our kernel rotated by 45°. The middle and right images have more ringing (most apparent at the bottom of the word "Leicester").

In Figure 9, we show super-resolution results where we have taken a sharp image, bicubically down-sampled it by 4x, and then solved for a 4x super-resolved kernel from the down-sampled input. We compare the original full resolution image to a bicubically up-sampled version of the low-resolution image and to the upsampled image deconvolved with our recovered kernel. The deconvolution results show a sharpening and recovery of high-frequency texture that is consistent with the full resolution images.

Figure 10 shows results for our calibration grid captured with an 11 mega-pixel Canon 1Ds using a Canon EF 28-200mm f3.5-5.6 lens at two apertures and focal lengths. For each image, we computed spatially varying PSFs by computing kernels for non-overlapping 220-pixel (the size of one grid tile) windows across the image at 2x resolution, i.e., two times the Bayer sampling resolution. Each PSF is displayed according to the location of its corresponding image window. The recovered PSFs show some interesting properties. The PSFs should be images of the aperture, and some kernels do show the shape of the aperture, which we know from the lens specifications to have 6 blades. They

Figure 7. Defocus and Slight Motion-Blur. (top left) The original blurred image and (top right) the deconvolved output with the recovered kernel displayed in the top right of the image (the kernel has been enlarged by 10x for display). (bottom row) Zoomed-in versions of the original and deconvolved image, respectively.

Figure 8. Motion Blur. (top row) The original blurred image (left) and the deconvolved output (right) with the recovered kernel displayed in the top right of the image (the kernel has been enlarged by 10x for display). (bottom row) Zoomed-in versions of the original and deconvolved image, respectively.

also show "donut" artifacts that can occur at some settings with lower-quality lenses. Perspective distortion across the image plane and vignetting (clipping of the aperture) by the lens barrel are also visible. For comparison we imaged back-lit pinholes at the same camera settings. Imaging pinholes to measure PSFs has some inherent problems due to the pinhole actually being a disk and not an infinitesimal point and due to diffraction; however, these images validate our recovered PSFs.

We also acquired a very sharply focused image, so that we could measure sub-pixel blur. Figure 11 shows an image of our grid from a 6 mega-pixel Canon 1D, using a high-quality Canon EF 135mm f/2L lens. We show recovered PSFs at 1x, 2x, 8x, and 16x sub-pixel sampling. The PSFs using higher sub-pixel resolution show an interesting structure that results from a combination of diffraction, lens imperfections, and sensor anti-aliasing and sampling.

Figure 12 shows a result for performing blind chromatic

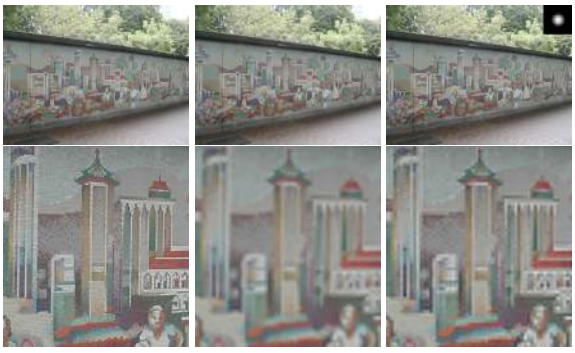
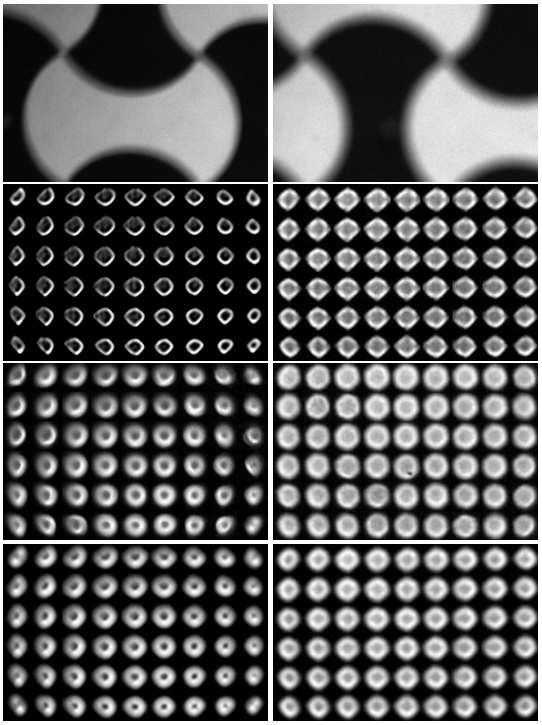


Figure 9. 4x Super-Resolution. (left) The original image and zoom-in, (middle) the original image bi-cubically downsampled and re-upsampled by 4x and zoom-in, (right) the upsampled image deconvolved using the recovered 4x super-resolved kernel (displayed in the top right of the image–the kernel has been enlarged by 10x for display) and a zoom-in on the bottom.



(a) 150mm f5.6 (b) 145mm f10

Figure 10. Different Apertures and Focal Lengths. (first row) Cropped portions of the observed blurred images, (second row) recovered spatially varying PSFs (green channel only), (third row) images of pinholes at the same depths and settings, and (fourth row) our recovered PSFs convolved with a disk the size of the pinhole. For (a) each PSF is 33×33 pixels and (b) they are 41×41 pixels. The PSFs reflect the shape of the aperture and show perspective distortion and vignetting across the image plane.

aberration correction for a JPEG image from a Canon S60 using a 5.8mm focal length at f8. After performing radial distortion correction and piecewise deconvolution using the spatially varying PSF, the aberration artifacts are significantly reduced. Figure 13 shows chromatic aberration correction for our non-blind method.

To view full resolution versions of our results, including additional examples, visit http://vision.ucsd.edu/kriegman-grp/research/psf_estimation/.

8. Discussion and Future Work

We have shown how to recover spatially varying PSFs at sub-pixel precision that capture blur due to motion, defocus, and intrinsic camera properties. Our method is fast, straightforward to implement, and predicts kernels accurately for a wide variety of images. Nevertheless, our method does have some limitations, and there are several avenues for future work.

The primary limitation of our method is that we can only solve for kernels with a single peak. This limitation is due to our reliance on an edge detector to find a single location for every blurred edge. In the case of a multi-peaked kernel, our method will incorrectly interpret the "ghost" copies

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

Figure 12. Blind Chromatic Aberration. (a) Recovered spatially varying PSFs for red, green, and blue shown as a color image. PSFs are only computed where there are enough edges observed. (b) The original image, (c) after radial correction and deconvolution the aberrations are significantly reduced, and (d–e) zoomed-in versions and intensity profiles for (b–c).

of edges as independent edges. While we have shown that single-peaked kernels model many commonly occurring cases of blur, we would like to extend our method to handle multi-modal kernels. One option is to group each stronger edge with its weaker ghost edges using contour matching. Once the ghost edges are identified, we could perform sharp edge prediction only for the primary edges.

As each sharp edge profile gives information about a radial slice of the PSF, it is necessary for an image, or image window, to have edges (or at least high-frequency content) at most orientations. If some orientations are lacking, our regularization terms can compensate; however, there is a breaking point, and there may not always be enough edge information to properly compute a PSF. In these cases, a low parameter kernel model may be more appropriate, but our sharp image prediction could still be be used to improve more traditional parametric kernel estimation procedures. We also plan to try using robust least squares to compensate for erroneous edge detections or profile fits.

Lastly, we would like to characterize more lenses and cameras. We would like to build a database that the vision and photography community could contribute to by using our pattern and code to take their own measurements.

9. Acknowledgements

We would like to thank the anonymous reviewers for their comments. This work was partially completed while the first author was an intern at Microsoft Research.

References

[1] P. D. Burns and D. Williams. Using slanted edge analysis for color registration measurement. In *IS&T PICS Conference*, pages 51–53. Society for Imaging Science and Technology, 1999.

[2] S. Cho, Y. Matsushita, and S. Lee. Removing non-uniform motion blur from images. In *ICCV 2007*, pages 1–8, 14-21 Oct. 2007.

[3] R. Fegus et al. Removing camera shake from a single photograph. *ACM Transactions on Graphics*, 27(3):787–794, August 2006.

[4] J. Jia. Single image motion deblurring using transparency. In *CVPR '07*, pages 1–8, 17-22 June 2007.

[5] S. B. Kang. Automatic removal of chromatic aberration from a single image. In *CVPR '07*, pages 1–8, 17-22 June 2007.

[6] D. Kundur and D. Hatzinakos. Blind image deconvolution. *SPMag*, 13(3):43–64, May 1996.

[7] A. Levin. Blind motion deblurring using image statistics. In *Advances in Neural Information Processing Systems*. MIT Press, 2006.

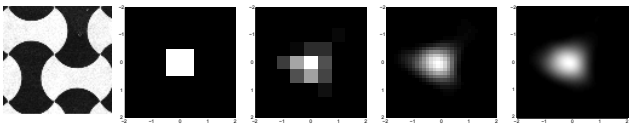
[8] E. Levy, D. Peles, M. Opher-Lipson, and S. Lipson. Modulation transfer function of a lens measured with a random target method. *Applied Optics*, 38:679–683, Feb. 1999.

[9] C. Liu, W. T. Freeman, R. Szeliski, and S. B. Kang. Noise estimation from a single image. In *CVPR '06*, volume 2, pages 901–908, New York, NY, June 2006.

[10] S. Nayar, M. Watanabe, and M. Noguchi. Real-time focus range sensor. In *Fifth International Conference on Computer Vision (ICCV'95)*, pages 995–1001, Cambridge, Massachusetts, June 1995.

[11] S. E. Reichenbach, S. K. Park, and R. Narayanswamy. Characterizing digital image acquisition devices. *Optical Engineering*, 30(2):170–177, February 1991.

[12] Q. Shan, W. Xiong, and J. Jia. Rotational motion deblurring of a rigid object from a single image. In *ICCV 2007*, pages 1–8, 14-21 Oct. 2007.



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

Figure 11. Sub-Pixel PSFs. (a) Cropped section of a sharp image of our grid, (b) PSF (green channel only) at the Bayer resolution (1x), (b) 2x, (c), 8x, and (d) 16x sub-pixel sampling. The sub-pixel PSFs show blur resulting from a combination of diffraction, lens imperfections, and sensor anti-aliasing and sampling.

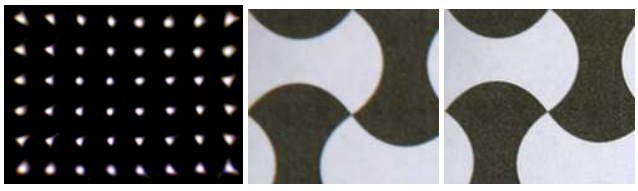


Figure 13. Chromatic Aberration. (left) The recovered spatially varying PSFs for red, green, and blue shown as a color image. The red and blue fringing is reflected in the PSF image and the PSFs are larger towards the edge of the image and spread along the direction orthogonal to the optical axis. (middle) Zoom-in on the input image. (right) After radial correction and deconvolution the aberrations are significantly reduced.