Focused Image Recovery from Two Defocused Images Recorded With Different Camera Settings

Murali Subbarao Tse-Chung Wei Gopal Surya

Department of Electrical Engineering

State University of New York

Stony Brook, NY 11794-2350

Phone: (516) 632-8405. email: murali@sbee.sunysb.edu

Abstract

Two new methods are presented for recovering the focused image of an object from only two blurred images recorded with different camera parameter settings. The camera parameters include lens position, focal length, and aperture diameter. First a blur parameter σ is estimated using one of our two recently proposed depth-from-defocus methods. Then one of the two blurred images is deconvolved to recover the focused image. The first method is based on a recently proposed Spatial Domain Convolution/Deconvolution Transform. This method requires only the knowledge of σ of the camera's point spread function (PSF). It does not require information about the actual form of the camera's PSF. The second method, in contrast to the first, requires full knowledge of the form of the PSF. As part of the second method, we present a calibration procedure for estimating the camera's PSF for different values of the blur parameter σ. In the second method, the focused image is obtained through deconvolution in the Fourier Domain using the Wiener filter. For both methods, results of experiments on actual defocused images recorded by a CCD camera are given. The first method requires much less computation than the second method. The first method gives satisfactory results for up to medium levels of blur and the second method gives good results for up to relatively high levels of blur.

Index Terms

Defocus, image restoration, inverse filtering, spatial domain deconvolution

1 Introduction

In machine vision, early processing tasks such as edge-detection, image segmentation, stereo matching, etc. are easier for focused images than for defocused images of three-dimensional (3D) scenes. However, the image of a 3D scene recorded by a camera is in general defocused due to limited depth-of-field of the camera. Autofocusing can be used to focus the camera onto a desired target object. But, in the resulting image, only the target object and those objects at the same distance as the target object will be focused. All other objects at distances other than that of the target object will be blurred. The objects will be blurred by different degrees depending on their distance from the camera. The amount of blur also depends on camera parameters such as lens position with respect to the image detector, focal length of the lens, and diameter of the camera aperture. In this paper, we address the problem of recovering the focused image of a scene from its defocused images.

We recently proposed two new methods for estimating the distance of objects in a scene [14, 15] using image defocus information. In these methods, two defocused images of the scene are recorded simultaneously with different camera parameter settings. The defocused images are then processed to obtain the distance of objects in the scene in small image regions. In this process, first a blur parameter σ which is a measure of the spread of the camera's point spread function (PSF) was estimated as an intermediate step. In this paper we present two methods for using the same blur parameter σ for recovering the focused images of objects in the scene from their blurred images. The main contributions of this paper are summarized below.

The first method of focused image recovery is based on a new spatial domain convolution/deconvolution transform (S transform) proposed in [13]. This method uses only the blur parameter σ which is a measure of the spread of the camera's PSF. In particular, the method does not require a knowledge of the the exact form of the camera PSF. The second method, in contrast to the first, requires complete information about the form of the camera PSF. For most practical camera systems, the camera PSF cannot be characterized with adequate accuracy using simple mathematical models such as Gaussian or cylindrical

functions. A better model is obtained by measuring experimentally the actual PSF of the camera for different degrees of image blur and using the measured data. This however requires camera calibration. An alternative but usually a more difficult solution is to derive and use a more accurate mathematical model for the PSF based on diffraction, lens aberrations, and characteristics of the various camera components such as the optical system, image sensor elements, frame grabber, etc. As part of the second method, we present a camera calibration procedure for measuring the camera PSF for various degrees of image blur. The calibration procedure is based on recording and processing the images of blurred step edges. In the second method, the focused image is obtained through a deconvolution operation in the Fourier domain using the Wiener filter.

For both methods of recovering the focused image, results of experiments on an actual camera system are presented. The results of the first method are compared with the results obtained using two commonly used PSF models– cylindrical based on geometric optics, and a 2D Gaussian. The results of the second method are compared with simulation results. A subjective evaluation of the results leads to the following conclusions. The first method performs better and is much faster than the methods based on simple PSF models. The focused image recovery is good for up to medium levels of image blur (upto an effective blur circle radius of about 5 pixels). The performance of the second method is comparable to the simulation results. The simulation results represent the best attainable when all noise, except quantization noise, is absent. The second method gives good results upto relatively high levels of blur (upto an effective blur circle radius of about 10 pixels). Overall the second method gives better results than the first, but it requires estimation of the camera's PSF through calibration and is computationally several times (about 4 in practice) more expensive.

In the next section we summarize the two methods for estimating the blur parameter σ. In the subsequent sections we describe methods for recovering the focused image using the blur parameter and experimental details.

2 Estimation of Blur Parameter σ

The blur parameter σ is a measure of the spread of the camera PSF. For a circularly symmetric PSF denoted by $h(x, y)$ it is defined as

$$\sigma^2 = \int_{\infty}^{\infty} \int_{\infty}^{\infty} (x^2 + y^2)\ h(x, y)\ dx\ dy \tag{1}$$

For a PSF model based on paraxial geometric optics, it can be shown that the blur parameter σ is proportional to the blur circle radius. If R is the blur circle radius, then $\sigma = R/\sqrt{2}$. For a PSF model based on a 2D Gaussian function, σ is the standard deviation of the distribution of the 2D Gaussian function.

We recently proposed two depth-from-defocus methods– DFD1F [14] and STM [15]. In both these methods, the blur parameter σ is first estimated and then the object distance is estimated based on σ. In this paper we will not provide details of these methods, but summarize below some relevant results.

In addition to object distance, the blur parameter depends on other camera parameters shown in Figure 1. The parameters include– the distance between the lens and the image detector denoted by s, the focal length f of the lens, and the diameter D of the camera aperture. We denote a particular setting for these camera parameters by $\mathbf{e} = (s, f, D)$. Both DFD1F and STM require at least two images, say $g_1(x, y)$ and $g_2(x, y)$, recorded with different camera parameter settings, say $\mathbf{e_1} = (s_1, f_1, D_1)$ and $\mathbf{e_2} = (s_2, f_2, D_2)$ respectively, such that at least one, but possibly two or all three, of the camera parameters are different, i.e. $s_1 \neq s_2$ or $f_1 \neq f_2$, or $D_1 \neq D_2$. DFD1F and STM also require a knowledge of the values of the camera parameters $\mathbf{e_1}$ and $\mathbf{e_2}$ (or a related camera constant which can be determined through calibration). Using the two blurred images g_1, g_2, the camera settings (or related camera constants) $\mathbf{e_1}$ and $\mathbf{e_2}$, and some camera calibration data related to the camera PSF, both DFD1F and STM methods estimate the blur parameter σ. A Fourier domain method is used in DFD1F whereas a spatial domain method is used in STM. The methods are general in that no specific model is used for the camera PSF, such as a 2D Gaussian or a cylindrical function.

Both DFD1F and STM have been successfully implemented on a prototype camera system named SPARCS. Experimental results on estimating σ have yielded a root-mean-square (RMS) error of about 3.7% for DFD1F and about 2.3% for STM. One estimate of σ

can be obtained in each image region of size as small as 48×48 pixels. By estimating σ in small overlapping image regions, the scene depth-map can be obtained.

In the following sections we describe two methods for using the blur parameter σ thus estimated (using DFD1F or STM) to recover the focused image of the scene.

3 Spatial Domain Approach

In this section we describe the spatial domain method for recovering the focused image of a 3D scene from a defocused image for which the blur parameter σ has been estimated using either DFD1F or STM [15]. The recovery is done through deconvolution of the defocused image using a new Spatial-Domain Convolution/Deconvolution Transform (S Transform) [13]. The transform itself is general and applicable to n-dimensional continuous and discrete signals for the case of arbitrary order polynomials. However, a special case of the general transform will be used in this section. First we summarize the S-Transform Convolution and Deconvolution formulas that are applicable here and then discuss their application for recovering the focused image.

3.1 S Transform

Let $f(x, y)$ be an image which is a two variable cubic polynomial in a small neighborhood, defined by

$$f(x, y) = \sum_{m=0}^{3} \sum_{n=0}^{3-m} a_{m,n} x^m y^n \tag{2}$$

where $a_{m,n}$ are the polynomial coefficients [3]. Let $h(x, y)$ be the PSF of a camera. The moment $h_{m,n}$ of the PSF is defined by

$$h_{m,n} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^m y^n h(x, y)\, dxdy \tag{3}$$

Let $g(x, y)$ be the blurred image obtained by convolving the focused image $f(x, y)$ with the PSF $h(x, y)$. Then we have

$$g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x - \zeta, y - \eta) h(\zeta, \eta)\, d\zeta d\eta \tag{4}$$

By substituting the Taylor series expansion of f in the above relation and simplifying, the following relation can be obtained:

$$g(x,y) = \sum_{0 \leq m+n \leq 3} \frac{(-1)^{m+n}}{m!n!} f^{m,n}(x,y) h_{m,n} \tag{5}$$

Equation (5) expresses the convolution of a function $f(x,y)$ with another function $h(x,y)$ as a summation involving the derivatives of $f(x,y)$ and moments of $h(x,y)$. This corresponds to the *forward S-Transform*. If the PSF $h(x,y)$ is circularly symmetric (which is largely true for most camera systems) then it can be shown that

$$h_{0,1} = h_{1,0} = h_{1,1} = h_{0,3} = h_{3,0} = h_{2,1} = h_{1,2} = 0 \text{ and } h_{2,0} = h_{0,2} \tag{6}$$

Also, by definition, for the PSF of a camera,

$$h_{0,0} = 1 \tag{7}$$

Using these results Equation (5) can be expressed as

$$g(x,y) = f(x,y) + \frac{h_{2,0}}{2} \bigtriangledown^2 f(x,y) \tag{8}$$

where $\bigtriangledown^2$ is the Laplacian operator. Taking the Laplacian on both sides of the above equation and noting that 4-th and higher order derivatives of f are zero as f is a cubic polynomial, we obtain

$$\bigtriangledown^2 g(x,y) = \bigtriangledown^2 f(x,y) \tag{9}$$

Substituting the above equation in Equation (8) and rearranging terms we obtain

$$f(x,y) = g(x,y) - \frac{h_{2,0}}{2} \bigtriangledown^2 g(x,y) \tag{10}$$

Equation (10) is a deconvolution formula. It expresses the original function (focused image) $f(x,y)$ in terms of the convolved function (blurred image) $g(x,y)$, its (i.e. g's) derivatives, and the moments of the point spread function $h(x,y)$. In the general case this corresponds to *Inverse S-Transform* [13].

Using the definitions of the moments of h and the definition of the blur parameter σ of h, we have $h_{2,0} = h_{0,2} = \sigma^2/2$, and therefore the above deconvolution formula can be written as

$$f(x,y) = g(x,y) - \frac{\sigma^2}{4} \bigtriangledown^2 g(x,y) \tag{11}$$

The above equation suggests a method for recovering the focused image $f(x, y)$ from the blurred image $g(x, y)$ and the blur parameter σ. Note that the above equation has been derived under the following assumptions (i) the focused image $f(x, y)$ is modeled by a cubic polynomial (as in Eq. 2) in a small (3×3 pixels in our implementation) image neighborhood, and (ii) the PSF $h(x, y)$ is circularly symmetric. These two assumptions are good approximations in practical applications and yield useful results.

3.2 Advantages

Equation (11) is similar in form to the previously known result that a sharper image can be obtained from a blurred image by subtracting a constant times the Laplacian of the blurred image from the original blurred image [11]. However that result is valid only for a diffusion model of blurring where the PSF is restricted to be a Gaussian. In comparison, our deconvolution formula is valid for all PSFs that are circularly symmetric including a Gaussian. Therefore, the previously known result is a special case of our deconvolution. Further, the restriction on the circular symmetry of the PSF can be removed if desired in our method of deconvolution using a more general version of the S-Transform [13]. Such generalization is not possible for the previously known result. In our deconvolution method, the focused image can be generalized to be an arbitrarily high order polynomial although such a generalization does not seem useful in practical applications that we know.

The main advantages of this method are (i) the quality of the focused image obtained (as we shall see in the discussion on experimental results), (ii) computational complexity, and (iii) the *locality* of the computations. *Simplicity* of the computational algorithm is another characteristic of this method. Given the blur parameter σ, at each pixel, estimation of the focused image involves the following operations (a) estimation of the Laplacian which can be implemented with a few integer addition operations (8 in our implementation), (b) floating point multiplication of the estimated Laplacian with $\sigma^2/4$, and (c) one integer operation corresponding to the subtraction in Eq. (11). For comparison purposes in the following sections, let us say that these computations are roughly equivalent to 4 floating point operations. Therefore, for an $N \times N$ image, about $4N^2$ floating point operations are required. All operations are *local* in that only a small image region is involved (3×3

in our implementation). Therefore the method can be easily implemented on a parallel computation hardware.

Next we describe the camera system on which this method of focused image recovery was implemented, and then we describe the experiments.

3.3 Camera System

All our experiments were performed on a camera system named StonyBrook Passive Autofocusing and Ranging Camera System (SPARCS). SPARCS consists of a SONY XC-77 CCD camera and an Olympus 35-70 mm motorized lens. Images from the camera are captured by a frame grabber board (Quickcapture DT2953 of Data Translation) residing in an IBM PS/2 (model 70) personal computer. The captured images are processed in the PS/2.

The lens system consists of multiple lenses and focusing is done by moving the front lens forward and backward. The lens can be moved under computer control using a stepper motor. The stepper motor has 97 steps, numbered 0 to 96. Step number 0 corresponds to focusing an object at distance infinity and step number 96 corresponds to focusing a nearby object, at a distance of about 55cm from the lens.

There is a one-to-one relation between the lens position specified by the step number of the stepper motor and the distance of an object that would be in best focus for that lens position. Based on this relationship, we often find it convenient to specify distances of objects in terms of lens step number rather than in units of length such as meter. For example, when the "distance" of an object is specified as step number n, it means that the object is at such a distance D_0 that it would be in best focus when the lens is moved to step number n.

3.4 Experiments

A set of experiments is described in Section 5 where the blur parameter σ is first estimated from two blurred images and then the focused image is recovered. In this section we describe experiments where σ is assumed to be given.

A poster with printed characters was placed at a distance of step 70 (about 80 cms) from the camera. The focused image is shown in Figure 3. The camera lens was moved to different positions (steps 70,60,50,40,30 and 20) to obtain images with different degrees of blur. The images are shown in figures 4a to 9a. The corresponding blur parameters (σs) for these images were roughly 2.2, 2.8, 3.5, 4.7, 6.0 and 7.2 pixels. These images were deblurred using equation (11). The results are shown in Figures 4d-9d. We see that the results are satisfactory for small to moderate levels of blur corresponding to about $\sigma = 3.5$ pixels. This corresponds to about 20 lens steps or a blur circle radius of about 5 pixels.

In order to evaluate the above results through comparison, two standard techniques were used to obtain focused images. The first technique was to use a two-dimensional Gaussian model for the camera PSF. The spread parameter of the Gaussian function was taken to be equal to the blur parameter σ, and therefore the PSF was:

$$h_b(x,y) \;=\; \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{12}$$

The plots of the PSF for two values of σ corresponding to about 2.7 pixels and 5.3 pixels are shown in Figure 2.

The focused image was obtained using the Wiener filter [11] specified in the Fourier domain by:

$$M(\omega,\nu) = \frac{1}{H(\omega,\nu)} \frac{|H(\omega,\nu)|^2}{|H(\omega,\nu)|^2 + \Gamma} \tag{13}$$

where $H(\omega,\nu)$ is the Fourier Transform of the PSF and Γ is the noise-to-signal power density ratio. In our experiments Γ was approximated by a constant. The constant was determined empirically through several trials so as to yield best results. Let $g(x,y)$ be the blurred image, and $\hat{f}(x,y)$ be the restored focused image. Let their corresponding Fourier Transforms be $G(\omega,\nu)$ and $\hat{F}(\omega,\nu)$ respectively. Then the restored image, according to Wiener filtering is

$$\hat{F}(\omega,\nu) = G(\omega,\nu)M(\omega,\nu). \tag{14}$$

By taking the inverse Fourier Transform of $\hat{F}(\omega,\nu)$, we can obtain the restored image $\hat{f}(x,y)$.

The results are shown in Figures 4c-9c. We see that for small values of σ (about 3.5 pixels), the Gaussian model performs well, but not as good as the previous method (Figs. 4d-9d). In addition to the quality of the focused image that is obtained, this method has

three important disadvantages. The first is computational complexity. For a given σ, first one needs to compute the the OTF $H(\omega, \nu)$, and then the Weiner filter $M(\omega, \nu)$. It is possible to precompute and store $M(\omega, \nu)$ for later usage for different values of σ. But this would require large storage space. After $M(\omega, \nu)$ has been obtained for a given σ, we need to compute $G(\omega, \nu)$ from $g(x, y)$ using FFT algorithm, multiply $M(\omega, \nu)$ with $G(\omega, \nu)$ to obtain $\hat{F}(\omega, \nu)$, and then compute the inverse Fourier transform of $\hat{F}(\omega, \nu)$. The complexity of the FFT algorithm is $O(N^2 log N)$ for an $N \times N$ image. Roughly, at least $(2N^2 + 2N^2 log_2 N)$ floating point operations are involved. For $N = 128$ used in our experiments, the number of computations is at least $16N^2$. In comparison, the number of computations in the previous case was $4N^2$. Therefore, this method is at least 4 times slower than the previous method. The second disadvantage of this method is that the computations are not local because of the computation of the Fourier transform of the entire image. The third disadvantage is the estimation of the noise parameter Γ.

In the second standard technique of focused image recovery, the PSF was modeled by a cylindrical function based on paraxial geometric optics:

$$h_a(x, y) = \begin{cases} \frac{1}{\pi R^2} & \text{if } x^2 + y^2 \leq R^2 \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

where R is the radius of the blur circle. The spread parameter σ corresponding to the above PSF can be shown to be related to the radius R by the relation $R = \sqrt{2}\sigma$. The plots of the PSF for two values of σ of about 2.7 pixels and 5.3 pixels are shown in Figure 2. With a knowledge of the blur parameter σ, it is thus possible to use equation (15) and generate the entire cylindrical PSF. The focused image was again obtained using the Wiener filter mentioned earlier, but this time using the cylindrical PSF.

In computing the Wiener filter, computation of the discrete cylindrical PSF at the border of the corresponding blur circle involves some approximations. The value of a pixel which lies only partially in the blur circle should be proportional to the area of overlap between the pixel and the blur circle. Violation of this rule leads to large errors in the restored image, especially for small blur circles. In our implementation, the areas of partial overlap were computed by resampling the ideal PSF at a higher rate (about 16 times), calculating the PSF by ignoring the pixels whose center did not lie within the blur circle, and then downsampling by adding the pixel values in 16 x 16 non-overlapping regions.

The results of this case are shown in Figures 4b-9b for different degrees of blur. The images exhibit "ripples" around the border between the background and the characters. Once again we see that the results are not as good as for the S transform method. For low levels of blur (upto about $R = 5$ pixels) Gaussian model gives better results than the cylindrical PSF, and for higher levels of blur (R greater than about 5 pixels) the cylindrical PSF gives better results than the Gaussian PSF.

In addition to the quality of the final result, the relative disadvantages of this method in comparison with the S transform method are same as those for the Gaussian PSF model.

4 Second Method

In the second method, the blur parameter σ is used to first determine the complete PSF. In practice, the PSF is determined by using σ as an index into a prestored table that specifies the complete PSF for different values of σ. In theory, however, the PSF may be determined by substituting σ into a mathematical expression that models the actual camera PSF. Since it is difficult to obtain a sufficiently accurate mathematical model for the PSF, we use a prestored table to determine the complete PSF. After obtaining the complete PSF, Wiener filter is used to compute the focused image. First we describe a method of obtaining the prestored table through a calibration procedure.

4.1 Camera calibration for PSF

Theoretically, the PSF of a camera can be obtained from the image of a point light source. However, in practice, it is difficult to create an ideal point light source that is incoherent and polychromatic. Therefore the standard practice in camera design is to estimate the PSF from the image of an edge.

Let $f(x, y)$ be a step edge along the y-axis on the image plane. Let a be the image intensity to the left of the y-axis and b be the height of the step. The image can be expressed as

$$f(x, y) = a + b\, u(x) \tag{16}$$

where $u(x)$ is the standard *unit step function*. If $g(x,y)$ is the observed image and $h(x,y)$ is the camera's PSF then we have,

$$g(x,y) = h(x,y) * f(x,y) \tag{17}$$

where * denotes the convolution operation.

Now consider the derivative of g along the gradient direction. Since differentiation and convolution commute, we have

$$\begin{aligned} \frac{\partial g}{\partial x} &= h(x,y) * \frac{\partial f}{\partial x} && (18) \\ &= h(x,y) * b\,\delta(x) && (19) \end{aligned}$$

where $\delta(x)$ is the *dirac delta* function along the x axis. The above expression can be simplified to obtain

$$\frac{\partial g}{\partial x} = b\,\theta(x) \tag{20}$$

where $\theta(x)$ is the *line spread function* of the camera defined by

$$\theta(x) = \int_{-\infty}^{\infty} h(x,y)dy \tag{21}$$

For any PSF $h(x,y)$ of a lossless camera, by definition, we have

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,y)\,dx\,dy = 1 \tag{22}$$

Therefore we obtain

$$\int_{-\infty}^{\infty} \frac{\partial g(x,y)}{\partial x}dx = b \tag{23}$$

Therefore, given the observed image $g(x,y)$ of a blurred step edge, we can obtain the line spread function $\theta(x)$ from the expression

$$\theta(x) = \frac{\frac{\partial g}{\partial x}}{\int_{-\infty}^{\infty}\frac{\partial g}{\partial x}dx} \tag{24}$$

After obtaining the line spread function $\theta(x)$, the next step is to obtain the PSF or its Fourier Transform, which is known as the Optical Transfer Function (OTF). Here we outline two methods of obtaining the OTF, one assuming the separability of the OTF and another using Inverse Abel Transform.

4.1.1 Separable OTF

Let the Fourier Transforms of the PSF $h(x, y)$ and LSF $\theta(x)$ be $H(\omega, \nu)$ and $\Phi(\omega)$ respectively. Then we have [11]

$$\Phi(\omega) = H(\omega, 0) \tag{25}$$

If the camera has a circular aperture then the PSF is circularly symmetric. If the PSF is circularly symmetric (and real), then the OTF is also circularly symmetric (and real), i.e. $H(\omega, \nu)$ is also circularly symmetric. Therefore we get

$$H(\omega, \nu) = \Phi(\sqrt{\omega^2 + \nu^2}) \tag{26}$$

Once we have the Fourier Transform of the LSF, $\Phi(\omega)$, we can calculate $H(\omega, \nu)$ for any values of ω and ν. However, in practice where digital images are involved, $\sqrt{\omega^2 + \nu^2}$ may have non integer values, and we may have to interpolate $\Phi(\omega)$ to obtain $H(\omega, \nu)$. Due to the nature of $\Phi(\omega)$, linear interpolation did not yield good results in our experiments. Therefore interpolation was avoided by assuming that the OTF to be separable, i.e. $H(\omega, \nu) = H(\omega, 0)H(0, \nu) = \Phi(\omega)\Phi(\nu)$. A more accurate method, however, is to use to the Inverse Abel Transform.

4.1.2 Inverse Abel Transform

In the case of a circularly symmetric PSF $h_1(r)$, the PSF can be obtained from its LSF $\theta(x)$ directly using the Inverse Abel Transform [5] :

$$h(r) = -\frac{1}{\pi} \int_r^\infty \frac{\theta'(x)}{\sqrt{x^2 - r^2}} dx \tag{27}$$

where $\theta'(x)$ is the derivative of LSF $\theta(x)$. Note that $h(x, y) = h_1(r)$ if $r = \sqrt{x^2 + y^2}$. In our implementation the above integral was evaluated using a numerical integration technique.

After obtaining $H(\omega, \nu)$, the final step in restoration is to use equations (13) and (14) and obtain the restored image.

4.2 Calibration Experiments

All experiments were performed using the SPARCS camera system. Black and white stripes of paper were pasted on a cardboard to create a step discontinuity in reflectance along a straight line. The step edge was placed at such a distance (about 80 cms) from the camera that it was in best focus when the lens position was step 70. The lens was then moved to 20 different positions corresponding to step numbers $0, 5, 10 \cdots 90, 95$. At each lens position, the image of the step edge was recorded, thus obtaining a sequence of blurred edges with different degrees of blur. Twelve of these images are shown in Figure 10. The difference between the actual lens position and the reference lens position of 70 is a measure of image blur. Therefore, an image blur of +20 steps corresponds to an image recorded at lens position of step 50 and an image blur of -20 steps corresponds to an image recorded at lens position of step 90. The size of each image was 80×200.

In our experiments, the step edge was placed vertically and therefore the image intensity was almost a constant along columns and the gradient direction was along the rows. To reduce electronic noise, each image was cut into 16 horizontal strips of size 5×200 and in each strip, the image intensity was integrated (summed) along columns. Thus each strip was reduced to just one image row. In each row, the first derivative was computed by simply taking the difference of gray values of adjacent pixels. Then the approximate location of the edge was computed in each row by finding the first moment of the derivative, i.e., if $\overline{i}$ is the column number where the edge is located, and $g_x(i)$ is the image derivative at column i, then

$$\overline{i} = \frac{\sum_{i=1}^{i=200} i g_x(i)}{\sum_{i=1}^{i=200} g_x(i)} \tag{28}$$

The following step was included to reduce the effects of noise further. Each row was traversed on either side of position $\overline{i}$ until a pixel was reached where either $g_x(i)$ was zero or its sign changed. All the pixels between this pixel (where for the first time, g_x became zero or its sign changed) and the pixel at the row's end were set to zero. We found this noise cleaning step to be very important in our experiments. A small non-zero value of image derivative caused by noise at pixels far away from the position of the edge affects the estimation of the blur parameter σ considerably.

From the noise-cleaned $g_x(i)$, the line spread function was computed as

$$\theta(i) = \frac{g_x(i)}{\sum_{i=1}^{i=200} g_x(i)} \tag{29}$$

Eight LSFs corresponding to different degrees of blur are plotted in Figure 11. It can be seen that, as the blur increases the LSF function becomes more flat and spread out. The location of the edge $\bar{i}$ was then recomputed using equation (28). The spread or second central moment of the LSF, σ_l was computed from

$$\sigma_l = \sqrt{\sum_{i=1}^{200} (i - \bar{i})^2 \theta(i)} \tag{30}$$

The computed values of σ_l for adjacent strips were found to differ by only about 2 percent. The average $\overline{\sigma_l}$ was computed over all the strips. It can be shown that σ_l is related to the blur parameter σ by $\sigma = \sqrt{2}\sigma_l$. The effective blur circle radius R is related to σ by $R = \sqrt{2}\sigma$. The values of R computed using the relation $R = 2\sigma_l$ for different step edges are shown in Figure 13. Figure 13 also shows the value of R predicted by ideal paraxial geometric optics. The values of R obtained for a horizontal step edge are also plotted in the figure. The values for the vertical and horizontal edges are in close agreement except for very low degrees of blur. This minor discrepancy may be due to the asymmetric (rectangular) shape of the CCD pixels (13×11 microns for our camera).

The PSF's were obtained from the LSFs using the inverse Abel Transform. Cross sections of the PSFs thus obtained corresponding to the LSFs in Figure 11 are shown in Figure 12.

4.3 Experimental Results

Using the calibration procedure described in the previous section, the PSFs and the corresponding OTFs were precomputed for different values of the blur parameter σ. These results were prestored in a lookup table indexed by σ. The OTF data $H(\omega, \nu)$ in this table was used to restore blurred images using the Wiener filter $M(\omega, \nu)$. Figures 4e-9e show the results of restoration using the separability assumption for the OTF and Figures 4f-9f are the results for the case where the inverse Abel transform was used to compute the PSF from the LSF. Both these results are better than the other results in Figures 4 (b,c,d) - 9 (b,c,d).

The method using the inverse Abel transform is better than all the other methods. We find that the results in this case are good even for highly blurred images. For example, the images in Figures 8a and 9a are severely blurred corresponding to 40 and 50 steps of blur or σ equal to about 6.0 and 7.2 pixels respectively. It is impossible for humans to recognize the characters in these images. However, in the restored images shown in Figures 8f and 9f respectively, many of the characters are easily recognizable.

In order to compare the above results with the best obtainable results, the restoration method which uses the inverse Abel transform was tested on computer simulated image data. Two sets of blurred images were obtained by convolving an original image with a Cylindrical and a Gaussian functions. The only noise in the simulated images was quantization noise. The blurred images were then restored using the Wiener Filter. The results are shown in Figures 14 and 15. We see that these results are only somewhat better but not much better than the results on actual data in Figures 4f-9f. This indicates that our method of camera calibration for the PSF is reliable.

The main advantage of this method is that the quality of the restored image is the best in comparison with all other methods. It gives good results for even highly blurred images. It has two main disadvantages. First, it requires extensive calibration work as described earlier. Second, the computational complexity is the same as that for the Weiner filter method discussed earlier. For an $N \times N$ image, it requires at least $2N^2 + 2N^2 log_2 N$ floating point operations as compared with $4N^2$ floating point operations for the method based on spatial domain deconvolution. Therefore, for an image of size 128×128, this method is at least 4 times slower than the method based on spatial domain deconvolution. Another disadvantage is that it requires the estimation of the noise parameter Γ for the Wiener filter.

5 Experiments with unknown σ and 3D object

In the experiments described earlier, the blur parameter σ of a blurred image was taken to be known. We now present a set of experiments where σ is unknown. It is first estimated using one of the two depth-from-defocus methods proposed by us recently [15]. Then, of the two

blurred images, the one that is less blurred is deconvolved to recover the focused image. Results are presented for both the first method based on spatial-domain deconvolution and the second method which uses inverse Abel transform.

The results are shown in Figures 16a-d. The first image in Fig. 16a is the focused image of an object recorded by the camera. The object was placed at a distance of step 14 (about 2.5 meters) from the camera. Two images of the object were recorded with two different lens positions–steps 40 and 70 (see Fig. 16a). The blur parameter σ was estimated using the depth-from-defocus method proposed in [15]. It was found to be about 5.5 pixels. Using this, the results of restoring the image recorded at lens step 40 is shown in Fig. 16a. Similar experiments were done by placing the object at distances steps 36, 56, and 76 corresponding to 1.31, 0.9 and 0.66 meters from the camera. In each of these cases, the focused image, the two recorded image at steps 40 and 70, and the restored images are shown in Figs. b-d. The blur parameters in the three cases were about 1.79, 1.24, and 2.35 pixels respectively. In the last two cases, the images recorded at lens step 70 was less blurred than the the one recorded at step 40. Therefore the image recorded at lens step 70 was used in the restoration.

In another experiment, a 3D scene was created by placing three planar objects at three different distances. Two images of the objects were recorded at lens steps 40 and 70. These images are shown in Figure 17. It can be seen that different image regions are blurred by different degrees. The image was divided into 9 regions of size 128 x 128 pixels. In each region the blur parameter σ was estimated and the image in the region was restored. The nine different estimated values of σ are 3.84, 4.76, 4.76, 0.054, 0.15, 0.46 (for image with lens step 40) and -2.65, -2.55 and -2.55 (for image with lens step 70) respectively. The different restored regions were combined to yield an image, where the entire image looks focused. Figure 17 shows the results using both the first and second methods of restoration. Currently each region can be as small as 48 x 48 pixels, which is a small region in the entire field of view of 640 x 480 pixels.

In the next experiment, a planar object with posters was placed inclined to the optical axis. The nearest end of the object was about 50 cms from the camera and the farthest end was about 120 cms. The blurred images of the object acquired with lens steps 40 and 70 are shown in Figure 18(a) and (b). The images were divided into non-overlapping regions

of 64 x 64 pixels and a depth estimate was obtained for each region. The different regions were then restored separately as before and combined to yield the restored images as shown in Figure 18(c) and (d). The restored images appear better than either of the blurred images. However there are some blocking artifacts, which are due to the "wrap around" problem of the FFT algorithm and the finite filter size in the case of the S-Transform method.

6 Conclusion

The focused image of an object can be recovered using two defocused images recorded with different camera parameter settings. The same two images can used to estimate the depth of the object using a depth-from-defocus method proposed by us [14, 15]. For a 3D scene where the depth variation is small in image regions of size about 64×64, each image region can be processed separately and the results can be combined to obtain both a focused image of the entire scene and a rough depth-map of the scene. If, in each image region, at least one of the two recorded defocused images is blurred only moderately or less ($\sigma <= 3.5$ pixels), then the focused image can be recovered very fast (computational complexity of $O(N^2)$ for an $N \times N$ image) using the new spatial domain deconvolution method described here. In most practical applications of machine vision, the camera parameter setting can be arranged so that this condition holds, i.e. in each image region at most only one of the two recorded defocused images is severely blurred ($\sigma > 3.5$ pixels). In those cases where this condition does not hold, the second method which uses the inverse Abel transform can be used to recover the focused image. This method requires camera calibration for the PSF and is several times more computationally intensive than the first method above. The methods in this paper can be used as part of a 3D machine vision system to obtain focused images from blurred images for further processing such as edge detection, stereo matching, and image segmentation.

References

[1] J. D. Gaskill, *Linear Systems, Fourier Transforms, and Optics,* John Wiley & Sons, New York, 1978.

[2] P. Grossman, "Depth from focus", *Pattern Recognition Letters* 5, pp. 63–69, Jan. 1987.

[3] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision,* Addison-Wesley Publishing Company, 1992, Ch. 8.

[4] B. K. P. Horn, "Focusing", Artificial Intelligence Memo No. 160, MIT, 1968.

[5] B. K. P. Horn, *Robot Vision,* McGraw-Hill Book Company, 1986, page 143.

[6] J. Ens and P. Lawrence, "A Matrix Based Method for Determining Depth from Focus", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 1991.

[7] E. Krotkov, "Focusing", *International Journal of Computer Vision*, 1, 223-237, 1987.

[8] S. K. Nayar, "Shape from Focus System" *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Champaign, Illinois, pp 302-308 June 1992.

[9] P. Meer and I. Weiss, *Smoothed differentiation filters for images*, Tech. Report No. CS-TR-2194, Center for Automation Research, University of Maryland, College Park, MD 20742-3411.

[10] A. P. Pentland, "A new sense for depth of field", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No. 4, pp. 523–531.

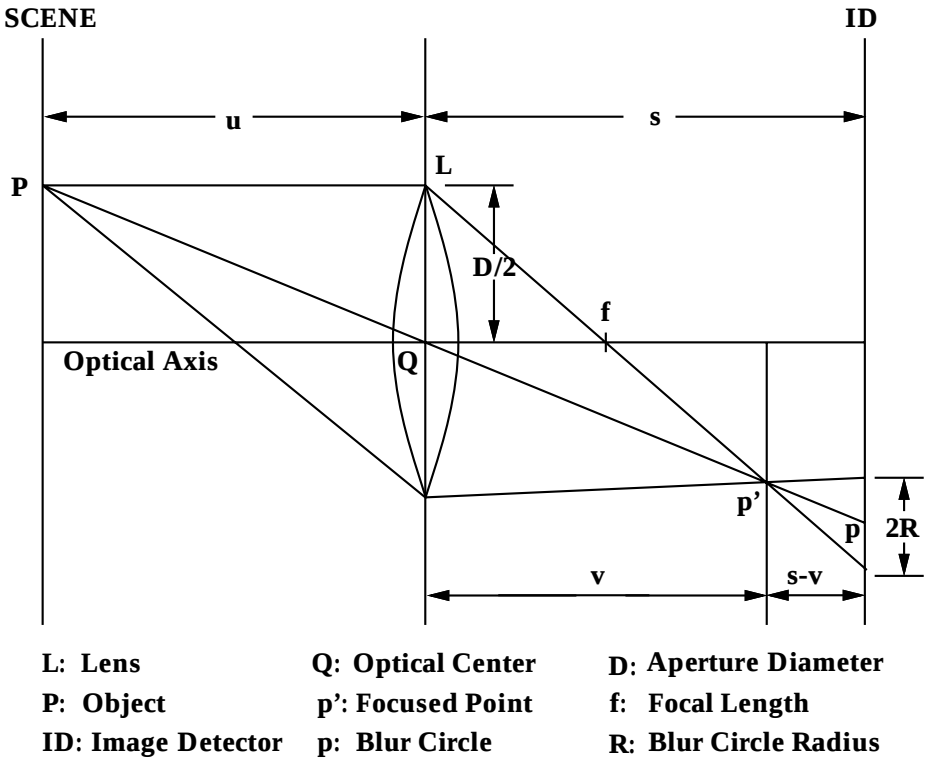
[11] A. Rosenfeld, and A. C. Kak, *Digital Picture Processing*, Vol. I . Academic Press, 1982.

[12] M. Subbarao, and G. Natarajan, "Depth recovery from blurred edges", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Ann Arbor, Michigan, pp. 498-503, June 1988.

[13] M. Subbarao, "Spatial-Domain Convolution/Deconvolution Transform ", Tech. Report No. 91.07.03, Computer Vision Laboratory, Dept. of Electrical Engineering, State University of New York, Stony Brook, NY 11794-2350.

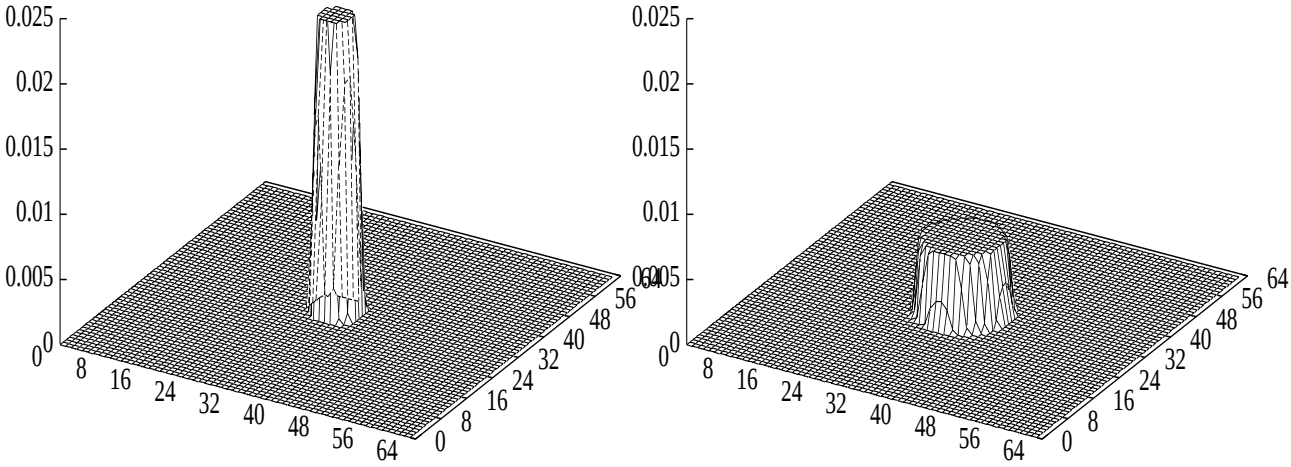
[14] T. Wei, *Three Dimensional Machine Vision Using Image Defocus*, Ph.D Thesis, Dept. of Electrical Engg., State University of New York at Stony Brook, Dec. 1994.

[15] G. Surya, *Three Dimensional Scene Recovery from Image Defocus*, Ph.D Thesis, Dept. of Electrical Engg., State University of New York at Stony Brook, Dec. 1994.

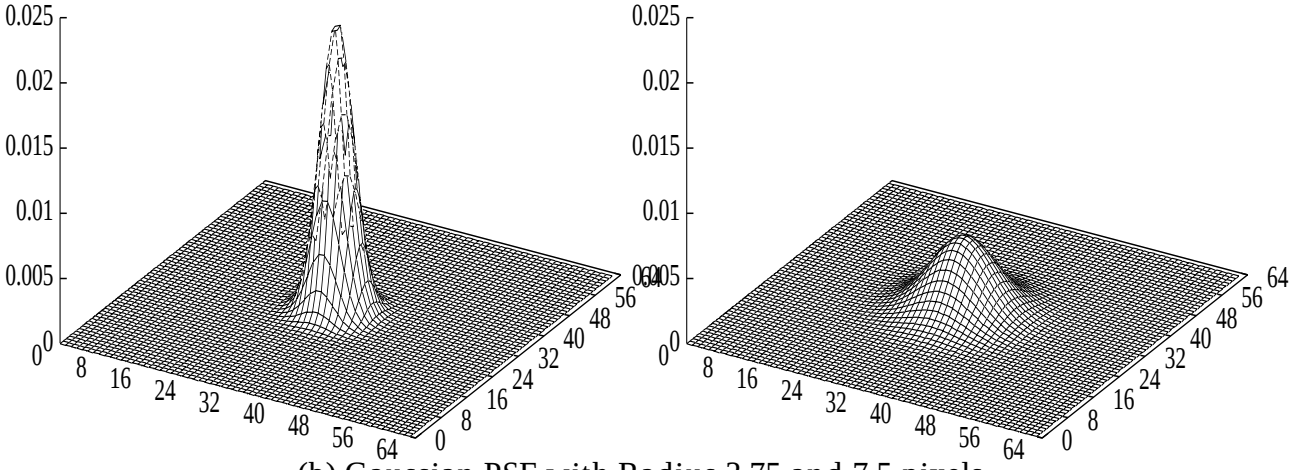


L: Lens **Q: Optical Center** **D: Aperture Diameter**
P: Object **p': Focused Point** **f: Focal Length**
ID: Image Detector **p: Blur Circle** **R: Blur Circle Radius**

Fig. 1 Image Formation in a Convex Lens



(a) Geometric Optics PSF with Radius 3.75 and 7.5 pixels



(b) Gaussian PSF with Radius 3.75 and 7.5 pixels

Fig. 2 PSF

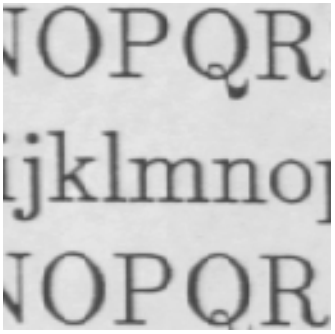
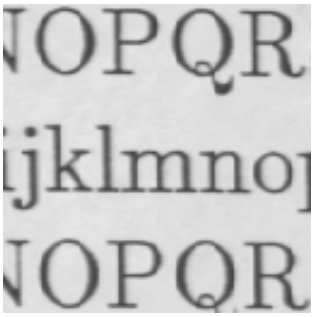


Fig. 3 Focused Image for Character



(a) Blurred Image

(b) Restored by Geometric PSF Model

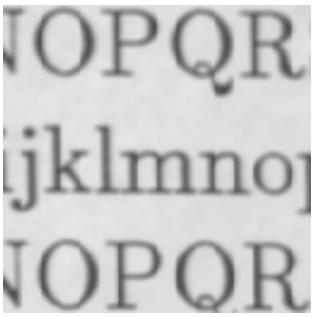
(c) Restored by Gaussian PSF Model

(d) Restored by S-Transform

(e) Restored by Separable MTF Model

(f) Restored using Actual PSF (Abel Transform)

Fig. 4 Restoration with 0 Step of Blur



(a) Blurred Image

(b) Restored by Geometric PSF Model

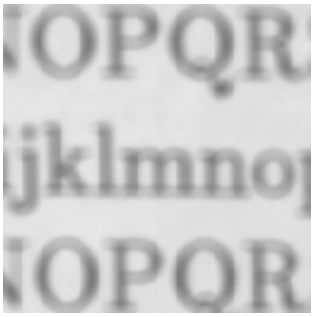
(c) Restored by Gaussian PSF Model

(d) Restored by S-Transform

(e) Restored by Separable MTF Model

(f) Restored using Actual PSF (Abel Transform)

Fig. 5 Restoration with 10 Steps of Blur



(a) Blurred Image

(b) Restored by Geometric PSF Model

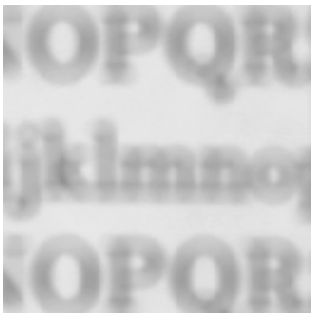
(c) Restored by Gaussian PSF Model

(d) Restored by S-Transform

(e) Restored by Separable MTF Model

(f) Restored using Actual PSF (Abel Transform)

Fig. 6 Restoration with 20 Steps of Blur



(a) Blurred Image

(b) Restored by Geometric PSF Model

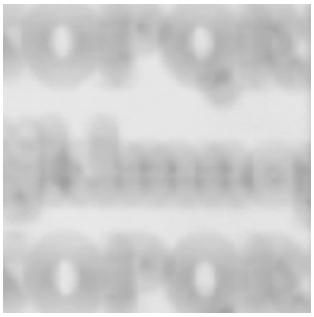
(c) Restored by Gaussian PSF Model

(d) Restored by S-Transform

(e) Restored by Separable MTF Model

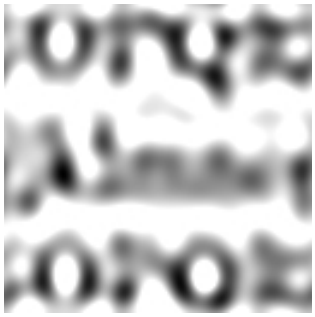
(f) Restored using Actual PSF (Abel Transform)

Fig. 7 Restoration with 30 Steps of Blur



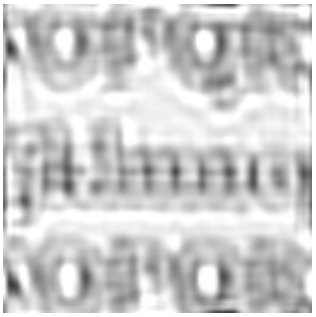
(a) Blurred Image

(b) Restored by Geometric PSF Model

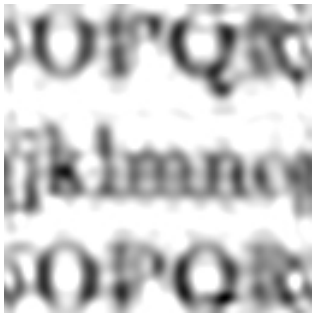


(c) Restored by Gaussian PSF Model

(d) Restored by S-Transform

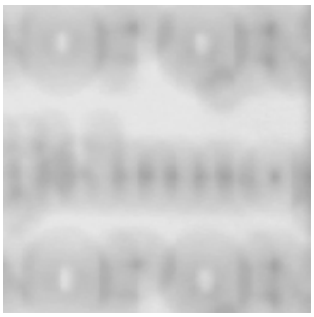


(e) Restored by Separable MTF Model



(f) Restored using Actual PSF (Abel Transform)

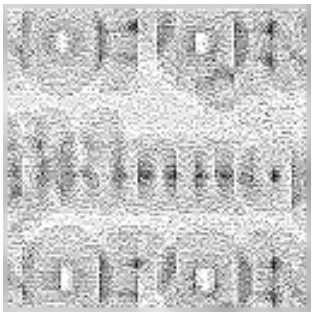
Fig. 8 Restoration with 40 Steps of Blur



(a) Blurred Image

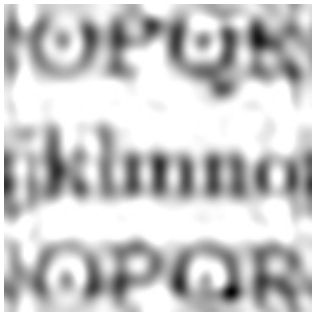
(b) Restored by Geometric PSF Model

(c) Restored by Gaussian PSF Model



(d) Restored by S-Transform

(e) Restored by Separable MTF Model



(f) Restored using Actual PSF (Abel Transform)

Fig. 9 Restoration with 50 Steps of Blur

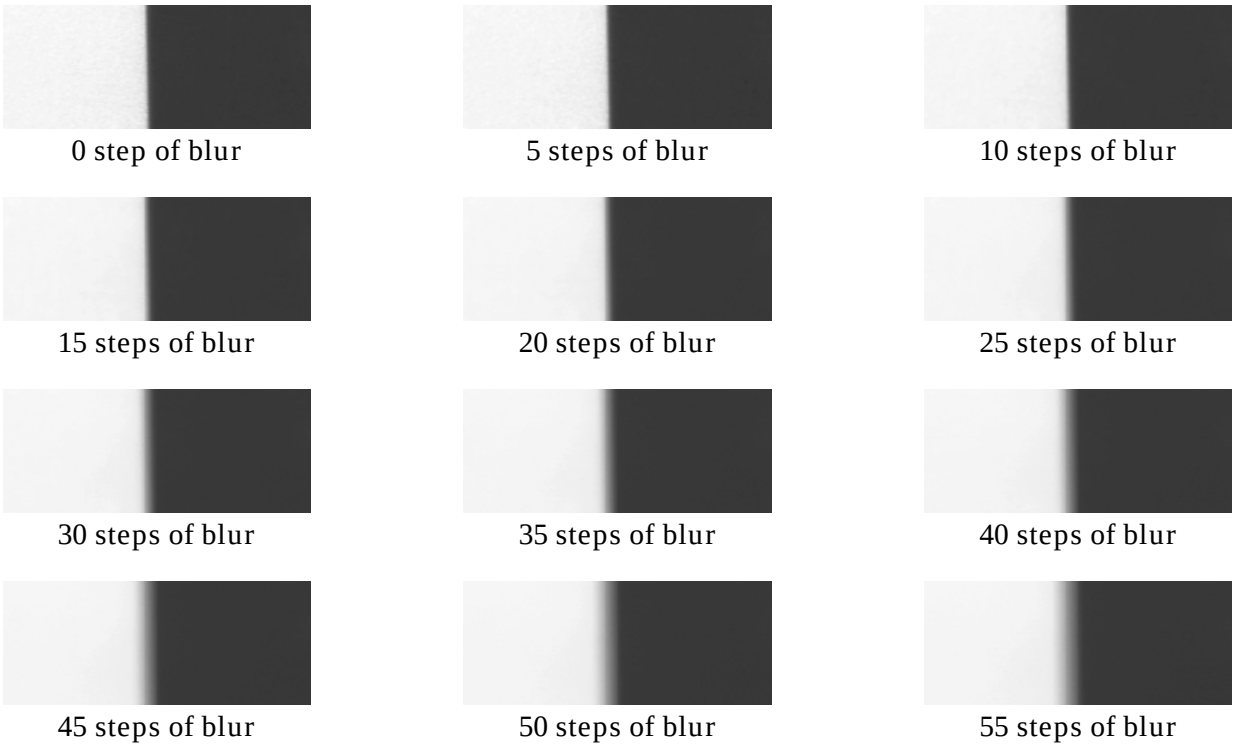


Fig. 10 Step Edges for Calibration

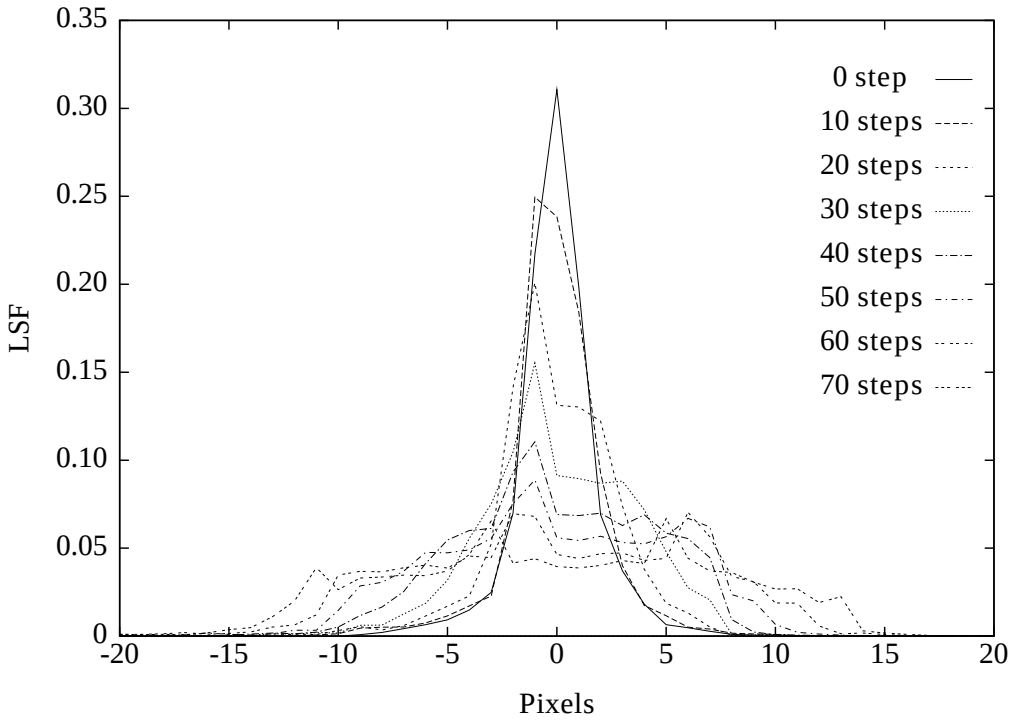


Fig. 11 LSF from Step Edges

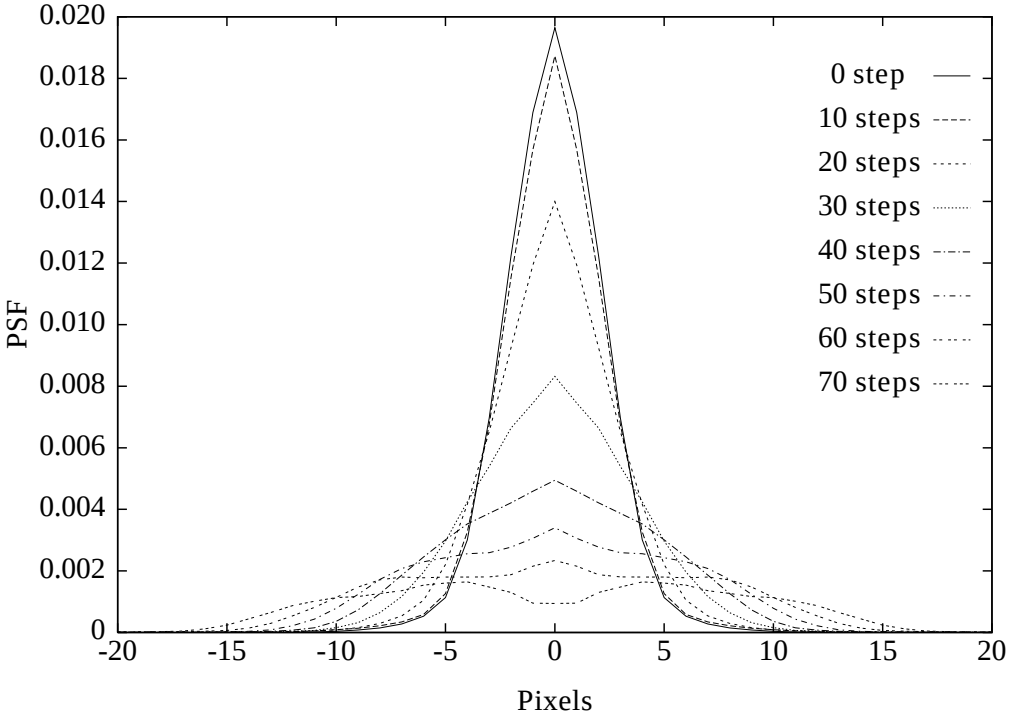


Fig. 12 PSF by Inverse Abel Transform

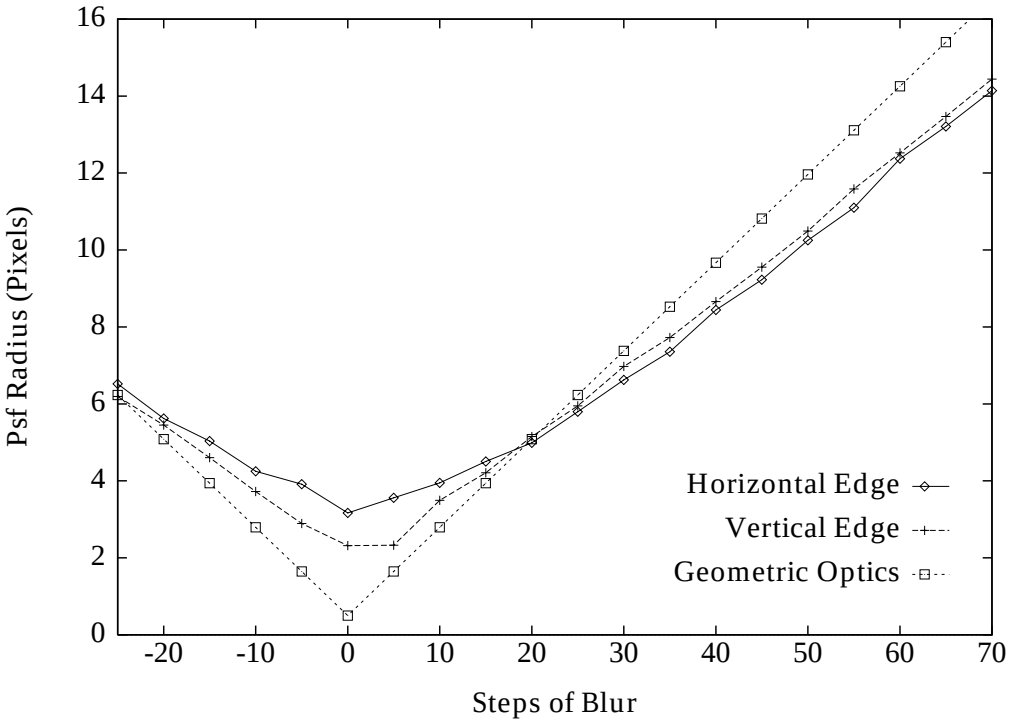
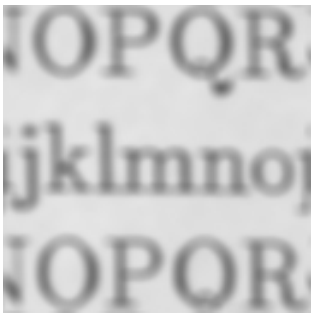
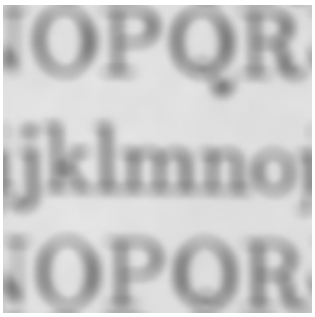


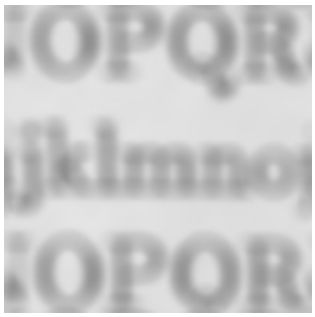
Fig. 13 PSF Radius from Step Edges



Blurred (0 step)



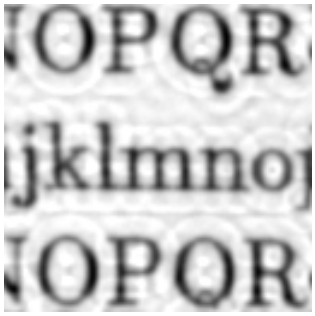
Blurred (10 steps)



Blurred (20 steps)

Restored (0 step)
(a)

Restored (10 steps)
(b)



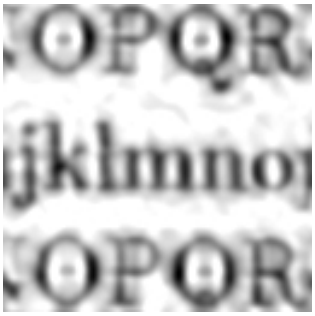
Restored (20 steps)
(c)

Blurred (30 steps)

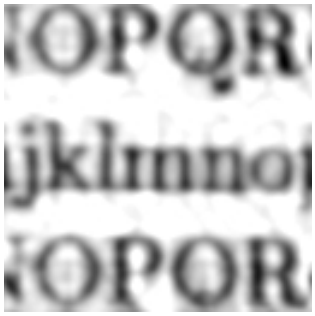
Blurred (40 steps)

Blurred (50 steps)

Restored (30 steps)
(d)

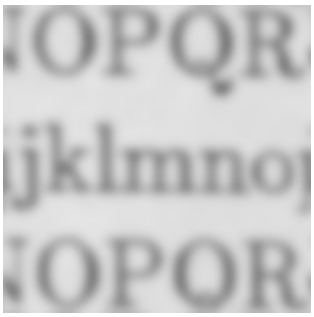


Restored (40 steps)
(e)

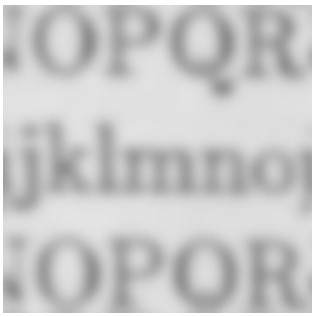


Restored (50 steps)
(f)

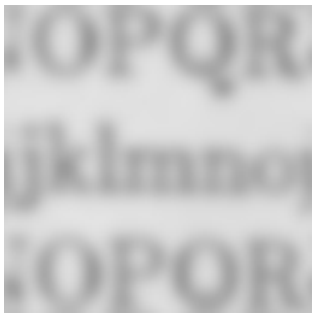
Fig. 14 Simulation with Geometric Optics PSF



Blurred (0 step)



Blurred (10 steps)



Blurred (20 steps)

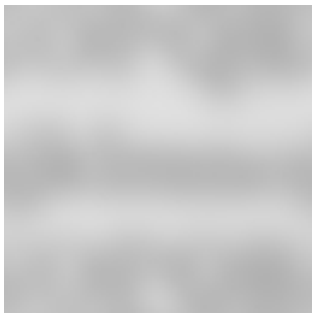
Restored (0 step)
(a)

Restored (10 steps)
(b)

Restored (20 steps)
(c)

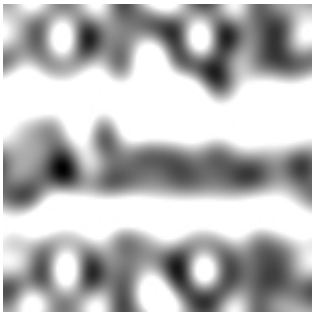
Blurred (30 steps)

Blurred (40 steps)

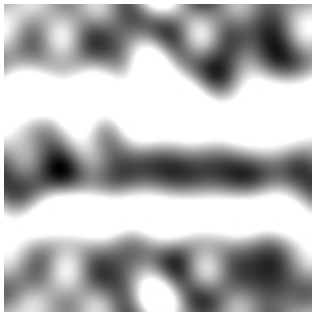


Blurred (50 steps)

Restored (30 steps)
(d)

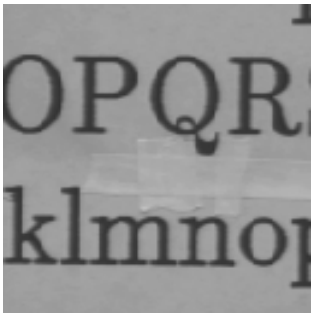


Restored (40 steps)
(e)

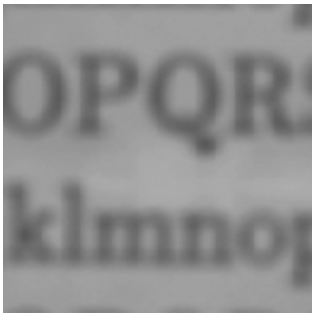


Restored (50 steps)
(f)

Fig. 15 Simulation with Gaussian PSF



Focused Image
(Focus at Step 14)



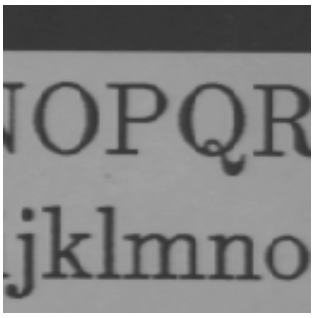
Blurred Image
(Lens at Step 40)

Blurred Image
(Lens at Step 70)

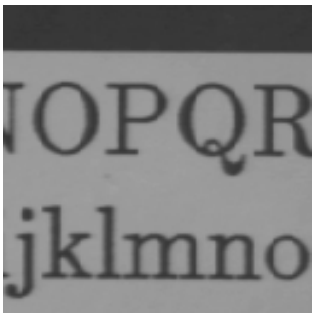
Restored by
S-Transform

Restored using Actual
PSF (Abel Transform)

Fig. 16(a) Depth Estimation with Restoration for Step 14

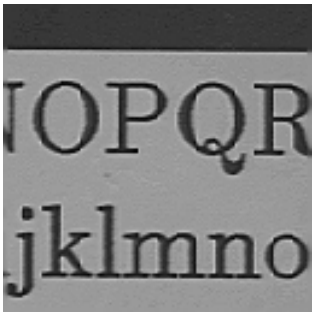


Focused Image
(Focus at Step 36)



Blurred Image
(Lens at Step 40)

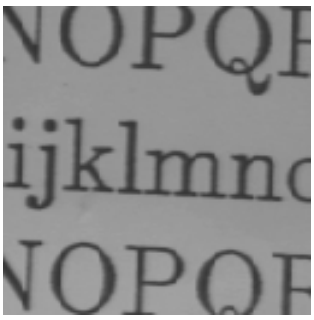
Blurred Image
(Lens at Step 70)



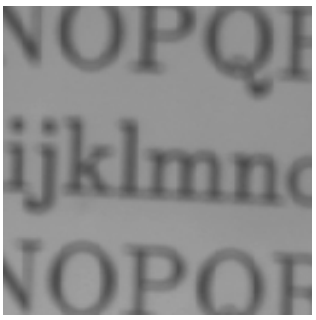
Restored by
S-Transform

Restored using Actual
PSF (Abel Transform)

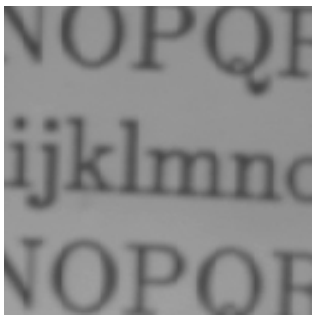
Fig. 16(b) Depth Estimation with Restoration for Step 36



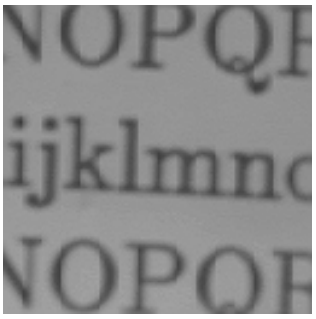
Focused Image
(Focus at Step 56)



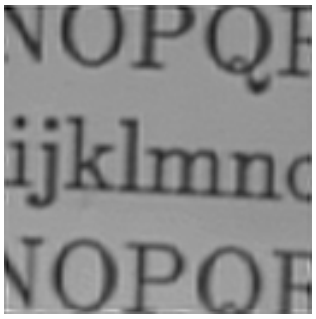
Blurred Image
(Lens at Step 40)



Blurred Image
(Lens at Step 70)

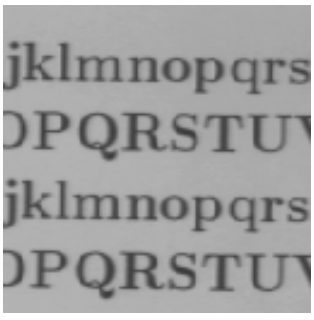


Restored by
S-Transform

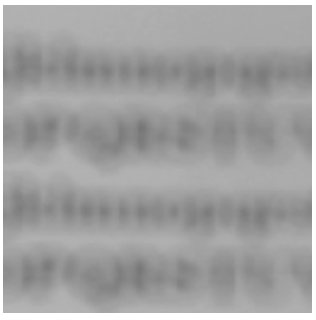


Restored using Actual
PSF (Abel Transform)

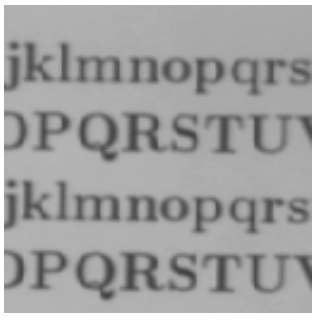
Fig. 16(c) Depth Estimation with Restoration for Step 56



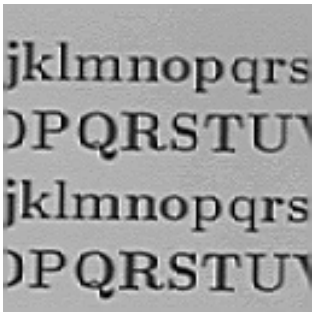
Focused Image
(Focus at Step 76)



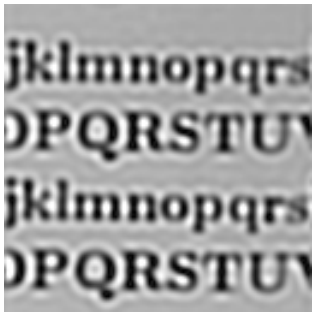
Blurred Image
(Lens at Step 40)



Blurred Image
(Lens at Step 70)



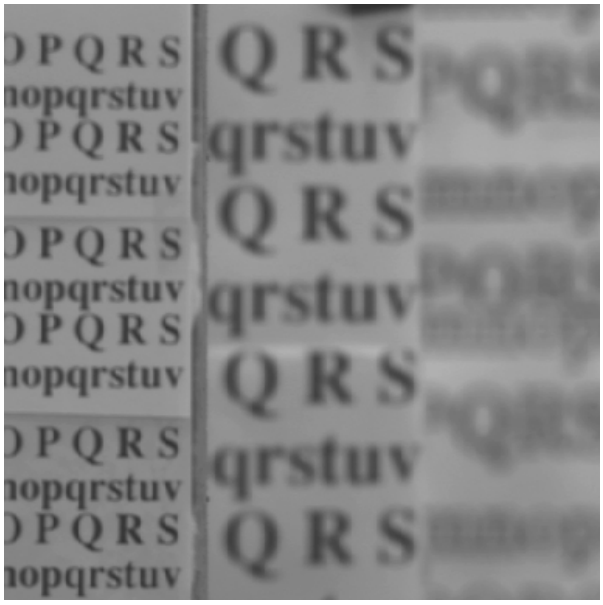
Restored by
S-Transform



Restored using Actual
PSF (Abel Transform)

Fig. 16(d) Depth Estimation with Restoration for Step 76

(a) Blurred Image
(Lens Step 40)



(b) Blurred Image
(Lens Step 70)

(c) Restored by
S-Transform

(d) Restored using Actual
PSF (Abel Transform)

Fig. 17 Depth Estimation with Restoration for 3-D Object