Accurate Recovery of Three-Dimensional Shape from Image Focus

Murali Subbarao Tae Choi

State University of New York, Department of Electrical Engineering

Stony Brook, New York 11794-2350

ABSTRACT

A new shape-from-focus method is described which is based on a new concept named Focused Image Surface *(FIS). FIS of an object is defined as the surface formed by the set of points at which the object points are focused by a camera lens. According to paraxial-geometric optics, there is a one-to-one correspondence between the shape of an object and the shape of its FIS. Therefore, the problem of shape recovery can be posed as the problem of determining the shape of the FIS. From the shape of FIS the shape of the object is easily obtained. In this paper the shape of the FIS is determined by searching for a shape which maximizes a focus measure. In contrast with previous literature where the focus measure is computed over the planar image detector of the camera, here the focus measure is computed over the FIS. This results in more accurate shape recovery than the traditional methods. Also, using FIS, a more accurate focused image can be reconstructed from a sequence of images than is possible with traditional methods. The new method has been implemented on an actual camera system, and the results of shape recovery and focused image reconstruction are presented.*

Index Terms : *shape-from-focus,Focused Image Surface,paraxial-geometric optics,focus measure,camera parameters.*

1 INTRODUCTION

The image of a scene formed by an optical system such as a lens contains both *photometric* and *geometric* information about the scene. Brightness or radiance and color of objects in the scene are part of photometric information whereas distance and shape of objects are part of geometric information. Recovering this information from a set of images sensed by a camera is an important problem in computer vision. Shape-From-Focus (SFF) methods provide one solution to the problem.

For an aberration-free convex lens, (i) the radiance at a point in the scene is proportional to the irradiance at its *focused image* [3], and (ii) the position of the point in the scene and the position of its focused image are related by the *lens formula*

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

where f is the focal length, u is the distance of the object from the lens plane, and v is the distance of the focused image from the lens plane (see Figure 1). Given the irradiance and the position of the focused image of a point, its radiance and position in the scene are uniquely determined. In fact the positions of a point-object and its image are *interchangeable,* i.e. the image of the image is the object itself. Now, if we think of an object surface in front of the lens to be comprised of a set of points, then the focused images of these points define another surface behind the lens (see Figure 1). We define this surface to be the *Focused Image Surface* (FIS) and the image irradiance on this surface to be the *focused image.* There is a *one to one correspondence* between FIS and the object surface. The geometry (i.e. the shape) and the radiance distribution of the object surface are uniquely determined by the FIS and the focused image.

In this paper we are concerned with the principles and computational methods for recovering the geometry and the radiance of an object from its *sensed image.* Note that a sensed image is in general quite different from the focused image of an object. In computer vision, the sensors are usually planar image detectors such as CCD arrays. Therefore, for

curved objects, only some parts of the image will be focused whereas other parts will be blurred. A sensed image will be the focused image only when the shape of the sensor and the shape of FIS match.

In traditional SFF methods (e.g. [2, 4, 5, 6, 8, 10]) a sequence of images are obtained by continuously varying one or both of the following camera parameters: (i) distance between the lens and image detector, and (ii) the focal length. For each image in the sequence, a sharpness measure or focus measure is computed at each pixel using a small (about 16×16) image neighborhood around the pixel. At each pixel, that image frame among the image sequence which gives a maximum sharpness measure is determined. The grey level (which is proportional to image irradiance) of the pixel in the image frame thus determined is taken to be the grey level of the focused image for that pixel. The camera parameter values for this image frame are used to compute the distance of the object point corresponding to the pixel. A simple measure of sharpness of an image is its grey level variance. Measures based on the energy of derivatives of images are however better suited [10].

The traditional SFF methods do not yield accurate shape or depth-map of objects. The main reason for this is that a focus measure is defined and computed over image frames sensed by planar image detectors. The focus measure at each pixel in an image frame is computed using a small window around the pixel. This corresponds to a piecewise constant approximation of the object shape in the window (see Figure 3a). Because of this approximation, the focused image reconstructed from the image sequence will be an approximation to the actual focused image.

The fundamental contribution of this paper is the idea that focus measures should be computed over the FIS using pixels lying on the FIS in the image sequence rather than over image frames where the pixels lie on a plane. Maximization of focus measures computed over FIS avoids the piecewise constant approximation of object shape found in the traditional SFF methods. The computational implementation of this idea involves two steps.

The first step is essentially to estimate an approximate FIS using one of the traditional SFF methods. The second step is to refine this approximate estimate by searching for an FIS shape which maximizes a focus measure computed over pixels lying on the FIS. The search is local and therefore computationally efficient. At present, our implementation corresponds to a piecewise planar (or linear) approximation of object shape as opposed to piecewise constant approximation. However, our implementation algorithm can be easily extended to higher order approximation at the cost of additional computation.

Our SFF algorithm has been implemented on a prototype camera system named Stonybrook Passive Autofocusing and Ranging Camera System (SPARCS). A brief description of SPARCS architecture is included. A number of experiments were carried out using SPARCS to evaluate our SFF algorithm. The experiments and their results are described. The experimental results show that our algorithm performs well.

In this paper we are mainly concerned with SFF methods which give dense and accurate depth-maps, and which do not require a detailed knowledge of the camera characteristics. These methods require a sequence of image frames (about 10 to 30) recorded with different camera parameter settings. However, there are methods [1, 7, 11, 12] which do not require a sequence of images, but only a few (about 2 or 3) acquired with different camera parameter values. These methods are very fast (about 10 times), but less accurate (their best performance gives a Root Mean Square (RMS) error which is twice that of the SFF methods). These fast methods are known as Depth-from-Defocus (DFD) methods whereas the SFF methods considered here are known as Depth-from-Focus (DFF) methods. Clearly, DFD methods can be used first to obtain a rough estimate of shape and then DFF or SFF methods can be used to refine the rough estimate to obtain a more accurate estimate of shape.

We first consider the case of recording the image sequence by moving the image detector (or lens) along the optical axis of the lens. The results of this case can be easily extended

to that of obtaining the image sequence by adjusting the focal length of the lens. When the image detector of a camera is moved from one end to the other, typically the focus measure in an image window gradually increases, reaches a maximum at the FIS, and then decreases gradually thereafter. The problem then is to find the image detector position at which the focus measure is a maximum. This is essentially a search of the image detector position space.

2 RELATION BETWEEN OBJECT SURFACE AND FIS

Figure 2 shows a right handed Object Space Coordinate System (OSCS) (X, Y, Z) with its origin O at the optical center of a convex lens L. The visible surface of any object in the object space (scene) can be expressed as $Z = Z(X, Y)$. We assume that

$$Z(X, Y) \geq f \tag{2}$$

because any object point closer than focal length f will produce only a virtual image but no real image.

The image formed by a convex lens is inverted with respect to the object. Therefore, for convenience, we define a left handed Image Space Coordinate System (ISCS) (x, y, z) (see Fig. 2) with its origin at O. The axes of the ISCS point in the direction exactly opposite to that of the corresponding axes of the OSCS. For any point (X, Y, Z) in OSCS, let its focused image in the ISCS be (x, y, z). The points (X, Y, Z) and (x, y, z) form a conjugate pair of points. Using the properties of similar triangles, it is easy to show that

$$\frac{x}{z} = \frac{X}{Z} \quad and \quad \frac{y}{z} = \frac{Y}{Z}. \tag{3}$$

For an object surface $Z(X, Y)$, the corresponding Focused Image Surface or FIS can be denoted by $z(x, y)$. It follows from the lens formula (1) that

$$\frac{1}{f} = \frac{1}{Z(X, Y)} + \frac{1}{z(x, y)}. \tag{4}$$

The above relation can be used to obtain the shape of an object from its FIS. It is helpful to recall here a few well-known results. A point at $Z = \infty$ comes to focus at $z = f$, a point at $Z = 2f$ comes to focus at $z = 2f$, and a point at $Z = f$ comes to focus at $z = \infty$.

Although the above relation between the shape of an object $Z(X, Y)$ and its FIS $z(x, y)$ is quite simple, the following theorem is particularly interesting.

Theorem 1 *The Focused Image Surface of a planar object is also planar.*

Proof Let

$$Z = PX + QY + R \tag{5}$$

be a planar object where P and Q are respectively the slopes of the surface along X and Y axes respectively and R is the intercept along the Z axis. We have

$$Z - PX - QY = R \tag{6}$$

$$\Longrightarrow 1 - P\frac{X}{Z} - Q\frac{Y}{Z} = \frac{R}{Z}. \tag{7}$$

$$\tag{8}$$

Using relations (3),(4), the above equation can be written as

$$1 - P\frac{x}{z} - Q\frac{y}{z} = R\left(\frac{1}{f} - \frac{1}{z}\right). \tag{9}$$

Rearranging terms in the above relation, we obtain

$$z = px + qy + r \tag{10}$$

where

$$c = \frac{f}{R - f}, \;\; p = -cP \;\; q = -cQ \;\; and \;\; r = cR. \tag{11}$$

Therefore we have shown that the FIS of a planar object $Z(X, Y) = PX + QY + R$ is also planar and is given by $z(x, y) = px + qy + r$. The relation between the object surface parameters P, Q, R and FIS parameters is given by Eq. (11).

The fact that the FIS of a planar object is also planar can be used to conclude the following: (i) the FIS of a polyhedral object is also polyhedral (with the same number of planar faces), (ii) the FIS of a straight line is also a straight line, and (iii) the FIS of a surface which can be generated by sweeping a straight line in 3D object space is also a surface which can be generated by sweeping a straight line in 3D image space. The first result follows directly from the theorem. The second result can be proved by noting that the intersection of two planes in object space is a straight line and the FIS of the straight line is the intersection of the FISs of the two intersecting planes which are themselves planar. The third result follows from the second result. A consequence of the third result is that the FIS of a cone is a "distorted cone", and the FIS of a cylinder is a "distorted cylinder".

An important significance of the above theorem in SFF methods is that if the shape of an object can be approximated well by a piecewise planar surface then the shape of the corresponding FIS can also be approximated well by a piecewise planar surface.

3 FOCUS MEASURE

There are many focus measures which perform well when used in SFF methods [4, 6, 10]. Any one of these could be used in our algorithm described next. Two simple examples of reasonably good focus measures are grey level variance and energy of Laplacian of the image. In our implementation we have chosen the energy of Laplacian of the image. It is computed by first applying the Laplacian operator, squaring the resulting values at every pixel, and then summing the values in the window of interest.

Before the focus measure of an image is calculated, the image is first normalized with respect to brightness. This is done by dividing the grey level of each pixel by the mean grey level of the whole image.

4 SFF ALGORITHM

Conceptually, our Shape-from-Focus (SFF) algorithm can be described as follows. The image detector is first moved to $z = z_0$. A sequence of images $g(i, j, k)$ are recorded by moving the image detector to positions $z_i = z_0 + (i-1)\delta$ where δ is a small displacement, for $i = 1, 2, \cdots, I$, $j = 1, 2, \cdots, J$, and $k = 1, 2, \cdots, K$. Usually, $z_0 = f$. J and K are the number of rows and columns respectively in each image frame and I is the number of image frames (see Figure 3b). We can think of this image sequence as an image volume. In this image volume, our problem is to find the set of pixels which lie on the focused image surface (FIS) of the object. For surfaces with a slope of up to about 1.0, for any given row j and column k, there is only one pixel which lies on the FIS. The image frame number i to which this pixel belongs depends on (j, k) and therefore it can be expressed as a function $i(j, k)$. The grey level of this pixel is $g(i(j, k), j, k)$. The relation between the row number j and the y coordinate is $y = (j - j_c)d$ where $j_c = J/2$ is the row index of the center row and d is the distance between two rows of pixels on the image detector array. Similarly, the relation between the x coordinate and the column index k is $x = (k - k_c)d$ where $k_c = K/2$ is the column index of the center column and d is the distance between two columns of pixels on the image detector array.

The shape of the FIS can be determined from the function $i(j, k)$ which gives the frame number of the pixel lying on the FIS for any given (j, k). The focused image $F(j, k)$ of the object is obtained from the image sequence and the function $i(j, k)$ as

$$F(j, k) = g(i(j, k), j, k) \tag{12}$$

In order to find the function $i(j, k)$ which specifies the FIS, we use the fact that the focus measure of $F(j, k)$ (or $g(i(j, k), j, k)$) is a maximum over all possible functions. Since a search for a function is computationally expensive, a two phase procedure is used in our implementation.

In the first phase of the algorithm, a rough estimate of FIS is estimated using a traditional

SFF method as follows. A small set (about 10) of N image frames g_n at regular intervals of I/N are selected from the original image sequence g_i. For each selected image frame, focus measures are computed in small field-of-views or image windows of size $M \times M$ (value of M varies from 3 to 21). For a given field of view, the corresponding image windows in different image frames shift and change magnification from one image frame to the other. Ideally, this should be taken into account, but in practice this effect may be neglected as the shift and magnification changes are usually small.

Usually overlapping windows are used at intervals of about $M/4$. Non-overlapping windows may also be used. For each window (or field-of-view), the image frame number for which the focus measure is a maximum over all the image frames is determined. The image frame numbers thus found gives a very rough estimate of the FIS. This estimate can be improved through a local interpolation scheme. For example, a low (quadratic or higher) order polynomial can be fitted to a few data points (focus measure as a function of image frame number) around the image frame with the maximum focus measure, and the location of the local maximum of the polynomial can be taken as the improved estimate of the focus location. Usually a quadratic or a cubic polynomial fit gives satisfactory results. A Gaussian function fit is used by some researchers[6]. Having obtained a rough estimate of the FIS, an approximate estimate of the slope of the FIS can be obtained by computing partial derivatives (or finite differences) along x and y directions.

In the second phase of our algorithm, the initial estimate of FIS is refined as follows. In this phase, the entire original image sequence g_i containing I image frames is used. For every window in which the FIS was estimated in the first step, a small cubic volume (about the size of $M \times M \times M$) image space is considered in the image sequence. The volume is centered at the initial estimate of FIS in that window. Now in this volume, a search is made for a planar surface which is closest to the actual FIS by maximizing the focus measure computed over the planar surface. The initial estimates of position and orientation of the FIS are used as starting values during the search. A brute-force or a simple gradient ascent search can be used. In our implementation, in order to increase robustness against noise,

the focus measure was summed over the estimated FIS and two more surfaces parallel to it where one was about one image frame closer to camera and the other was about one image frame away from the camera.

An outline of the algorithm implemented by us can be summarized as follows.

• In every field of view, do the steps below.

• Read as input the initial estimates of the position and orientation parameters of the FIS. Let i_0 be the initial estimate of position, and p_0, q_0 be the initial estimates of the slopes of the FIS along the x and y axes respectively. Then the initial estimate of the FIS is described by

$$i(j,k) = i_0 + p_0 j + q_0 k \tag{13}$$

• Read as input the size of the search space for position and orientation. Let the search space for position be the range $i_{min} = i_0 - \delta i$, to $i_{max} = i_0 + \delta i$, for x-slope be the range $p_{min} = p_0 - \delta p$, to $p_{max} = p_0 + \delta p$, and for y-slope be the range $q_{min} = q_0 - \delta q$, to $q_{max} = q_0 + \delta q$. If $i_{min} <= 1$ set $i_{min} = 1$ and if $i_{max} >= I$ set $i_{max} = I$. Take the value of maximum allowable slope $\mathrm{MAX}_S LOPE for the surface along each axis to be about 1.0. If \mathrm{p}_{min} <= -MAX_S LOPE$ set $i_{min} = -MAX_S LOPE$ and if $i_{max} >= MAX_S LOPE$ set $i_{max} = MAX_S LOPE$. Similarly, if $q_{min} <= -MAX_S LOPE$ set $q_{min} = -MAX_S LOPE$ and if $q_{max} >= MAX_S LOPE$ set $q_{max} = MAX_S LOPE$. Read as input the searching step sizes $\delta_i, \delta_p, \delta_q$ for i, p, q, respectively.

Read image window size 2S+1. •

sum = 0;

$\max_s um = 0$;

$ip = i0; pp = p0; qp = q0$

$for \mathrm{i}_m = i_{min}$ to i_{max} in steps of δ_i do /* search position space

for $p_m = p_{min}$ to p_{max} in steps of δ_p do /* search x-slope

for $q_m = q_{min}$ to q_{max} in steps of δ_q do /*search y-slope

/* compute focus measure on a candidate FIS */

for $j = -S$ to S do

for $k = -S$ to S do

l= $i_m + j * p_m + k * q_m$

sum += $\sum_{m=l-1}^{m=l+1} Focus_Measure[m][j][k]; endendif(sum > max_sum) then ip = im; pp = pm; qp = qm;$

$endendendip, pp, qp, are the position and slopes of the estimated FIS. In the above algorithm, Focus_Measure[m][j$

The estimated FIS is described by i(j,k) = ip+ pp*j + qp*k in the image window being processed. At the cost of additional computation, it is also possible to search and obtain piece-wise quadratic and higher order approximations to the FIS.

5 SPARCS

The SFF method described here was implemented on a camera system named Stonybrook Passive Autofocusing and Ranging Camera System (SPARCS). SPARCS was built by us in our laboratory over the last few years. A block diagram of the system is shown in Figure 5. SPARCS consists of a SONY XC-77 CCD camera and an Olympus 35-70mm motorized lens. Images from the camera are captured by a frame grabber board (Quickcapture DT2953 of Data Translation). The frame grabber board resides in an IBM PS/2 (model 70) personal computer. The images taken by the frame grabber are processed in the PS/2 computer.

The lens system consists of multiple lenses and focusing is done by moving the front lens forward and backward. To facilitate computer control of the lens movement there is a stepper motor with 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 50cm from the lens. The motor is controlled by a microprocessor, which can communicate with the IBM PS/2 through a digital I/O board (Contec mPIO24/24). Pictures taken by the camera can be displayed in real time on a color monitor (SONY

PVM-1342 Q). The images acquired and stored in the IBM PS/2 can be transferred to a SUN workstation. The camera settings used in the experiments were: focal length = 35mm, aperture diameter = 35/4 mm, and camera gain control = +6dB.

6 EXPERIMENTS

The SFF algorithm described above and the corresponding traditional method were implemented on the SPARCS camera system. The methods were similar in all respects except that the focus measure was computed over estimated FIS in the case of SFF whereas in the traditional method the focus measure was computed over image frames in image windows. These implementations were used to compare the improvement obtained by our SFF algorithm in comparison with the traditional method.

Here we present the results for two objects: (i) a slanted planar object (Figure 6), and (ii) a cone object of length about 79 inches and base diameter of about 15 inches (Figure 7). It is found that the the improvement in accuracy for the slanted planar object is marginal where it is significant for the cone object. This indicates that FIS improves the accuracy substantially in the case of curved objects.

The illumination for the two objects was about 600 lux. Image size was 256×256. In order to reduce electronic noise, for a fixed lens position five image frames were time averaged. The image sequence contained 97 image frames, one for each lens step position of the stepper motor. The absolute displacement between two consecutive image frames was about 0.03 mm and the distance between pixels was about 0.013 mm. The window size for computing focus measures was 21×21.

An initial estimate of FIS was obtained by computing the sum of the square of image Laplacian for 9 image frames equally spaced apart (about 10 frames apart) in the original image sequence. The position of the maximum focus measure was first improved by a quadratic interpolation scheme using three points centered at the maximum point. A typical plot of the 9 values along with the position of maximum position obtained using

interpolation is shown in Figure 4.

Figure 6 shows the results for the slanted planar object. Figs. 6(a) to 6(d) show the image frames recorded when the lens position was at motor steps 20, 40, 60, and 80. In each of these frames, only one part of the image is focused whereas the other parts are blurred by varying degrees. This is particularly noticeable in Fig. 6(d) where the closer part of the object on the left is focused whereas the blur increases gradually towards right as the object distance increases. The shape or depth-map recovered by our SFF algorithm is shown in Fig. 6(e). Here both a 3D surface plot and a smaller side view of the plot are shown for clarity. The results in this case are close to the actual shape except in regions where there is insufficient contrast. The reconstructed focused image of the object is shown in Fig. 6(f). We see that all parts of the image are in sharp focus.

Figure 7 is similar to Figure 6 except that the results in this case are for the cone object. In Fig. 7(e) we see that the recovered shape of the cone has a blunt tip rather than a sharp tip. This is due to the piece-wise planar approximation. Except in areas where there is insufficient grey level variance, the shape recovered is good. Figure 7(f) shows the reconstructed focused image of the cone object. In comparison with the image frames shown in Fig. 7(a) to 7(d), the reconstructed image appears focused everywhere.

7 Comparison with Traditional Method

In order to compare our SFF method with the traditional method we carried out experiments on both real data and simulated data. Simulated data was necessary in comparing the results of the two methods with accurate ground truth. For real data, we did not have adequate facilities to accurately measure the ground truth.

Figures 8 and 9 show the results on real data for the slanted planar object obtained using the traditional method and our SFF method respectively. In these figures, it is difficult to visually compare the two results. Since the accurate ground truth was not available, we

fitted planar surfaces to the data (we knew that the object was planar but did not accurately know its position and orientation) using a least-square error minimization approach. Then the root-mean-square (RMS) error was computed between the fitted planes and the data. The results are shown in Table 1. We see that the RMS error for the traditional method is 0.40 lens steps (out of 97 steps) whereas it is 0.37 lens steps for the traditional method. Therefore the improvement in accuracy in this case is marginal.

Figures 12 and 13 show the results on real data for the cone object. As expected, the tip of the cone is sharper in the case of our SFF method whereas it is blunted in the case of the traditional method. Except this difference the two results appear to be almost the same. The actual cone object was made of cardboard and it was distorted. Therefore we could not fit a cone to the data in order to compare the accuracy of the two methods as we did above in the case of the planar object.

In order to do a rigorous quantitative comparison of our SFF method with the traditional method we did the following experiment on simulation data. A camera simulation software named Active Vision Simulator (AVS) [13] was used to generate 97 images of a cone object corresponding to 97 lens positions. A paraxial geometric optics model was used for image formation in the computation of the blurred images. For the simulation software, the input camera parameters (focal length, aperture, pixel size, etc.) were set to be the values of our actual camera. Figure 16 shows the input depth-map of the cone. Figures 14 and 15 show the results of the traditional method and our SFF method respectively. The RMS errors between these two results and the ground truth in Figure 16 were computed. For the traditional method the RMS error was 2.22 lens steps out of 97 steps whereas for our SFF method the RMS error was 1.41 lens steps (out of 97 steps). Therefore, in this case, the traditional method has about 1.5 times the error for our SFF method.

The relationship between the reciprocal of the object distance 1/u versus the lens step number is almost linear and can be expressed as

$$1/u = ax + b \tag{14}$$

where x specifies lens position. For our camera, the lens position is specified in terms

of a motor step number where each step corresponds to a displacement of about 0.03mm. The RMS errors mentioned above are for the lens position and it gives a good indication of the performance of the method for application in camera systems. In order to compute the error in terms of object distance, we have to consider the error differentials in Eq. (14).

$$\mid \delta(1/u) \mid = a \mid \delta x \mid \tag{15}$$

$$\Rightarrow \mid \frac{\delta u}{u} \mid = a \mid \delta x \mid u \tag{16}$$

$$\Rightarrow \mid \delta u \mid = a \mid \delta x \mid u^2. \tag{17}$$

From the above relations we see that the relative(percentage) error $\mid \frac{\delta u}{u} \mid$ in actual distance u increases linearly with distance, and the absolute error $\mid \delta u \mid$ in actual distance increases quadratically with distance. For our camera, using a Depth from Focus method[10] the constants were found to be a= 0.0172 and b= -0.1143.

Setting $\mid \delta x \mid$ to be the RMS error of 2.22 steps for the traditional method and 1.41 steps for the new method respectively, a plot of relative error $\mid \frac{\delta u}{u} \mid$ is shown in Figure 10 and a plot of the absolute error is shown in Figure 11.

In Figure 10 we see that for the new method the percentage error in distance at 0.6 meter is about 0.xx% and increases linearly to about y.y% at 5 meter distance. This compares well with the resulting error of about 0.xx% at 0.6 meter and increasing linearly to about y.y% at 5 meter distance that is obtained with the traditional method.

Figure 11 shows that for the new method, absolute error increases quadratically from x.xx mm at 0.6 meter to about xxx.xx mm at 5 meter distance. The corresponding numbers for the traditional method are x.x mm at 0.6 meter and about xxx.xx mm at 5 meter distance.

8 CONCLUSION

We have described a new Shape-from-Focus method based on maximizing focus measures computed directly over the Focused Image Surface (FIS) of objects. This method corresponds to piece-wise planar approximation of the shape of objects as opposed to piece-wise

constant approximation adopted by SFF methods in the previous literature. We have experimentally demonstrated the effectiveness of our SFF method on real-world examples. The experiments indicate that our SFF method gives much more accurate results for curved objects than the comparable traditional method. Our method here can be extended to obtain piece-wise quadratic and higher order approximations to FIS at the cost of additional computation.

9 ACKNOWLEDGEMENTS

The support of this research in part by the National Science Foundation and Olympus Corporation is gratefully acknowledged.

References

[1] J. Enns 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.

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

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

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

[5] G. Ligthart and F. Groen, "A Comparison of Different Autofocus Algorithms", *International Conference on Pattern Recognition*, pp. 597-600,1982.

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

[7] 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, July 1987.

[8] J. F. Schlag, A. C. Sanderson, C. P. Neuman, and F. C. Wimberly, "Implementation of automatic focusing algorithms for a computer vision system with camera control", CMU-RI-TR-83-14, Robotics Institute, Carnegie-Mellon University, 1983.

[9] M. Subbarao, "Parallel depth recovery by changing camera parameters", *Second International Conference on Computer Vision,* Florida, USA, pp. 149-155, December 1988.

[10] M. Subbarao, T. Choi, and A. Nikzad, "Focusing Techniques", *OE/BOSTON '92, SPIE conference Proceedings*, Vol. 1823, Boston, November 1992.

[11] M. Subbarao, and G. Surya, "Application of Spatial-Domain Convolution/ Deconvolution Transform for Determining Distance from Image Defocus", *OE/BOSTON '92, SPIE conference Proceedings*, Vol. 1822, Boston, November 1992.

[12] M. Subbarao, and T. Wei, "Depth from Defocus and Rapid Autofocusing: A Practical Approach", Tech. Report No. 92.01.17, CVL, Dept. of EE, SUNY, Stony Brook, NY 11794-2350, 1992. (An abridged version of this appears in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Champaign, Illinois, pp.773-776 , June 15-18,1992.

[13] M. Subbarao, and M. C. Lu, "Computer Modeling and Simulation of Camera Defocus", Tech. Report No. 92.01.16, CVL, Dept. of EE, SUNY, Stony Brook, NY 11794-2350, 1992.

Figure 1: Image Formation in a Convex Lens.

Figure 2: Object and Image Coordinate Systems.

Figure 3: Direct Fitting on the FIS(Focused Image Surface).

Figure 4: Focus Measure for a target pixel (j,k) and Polynomial Interpolation using a second order polynomial.

Figure 5: SPARCS.

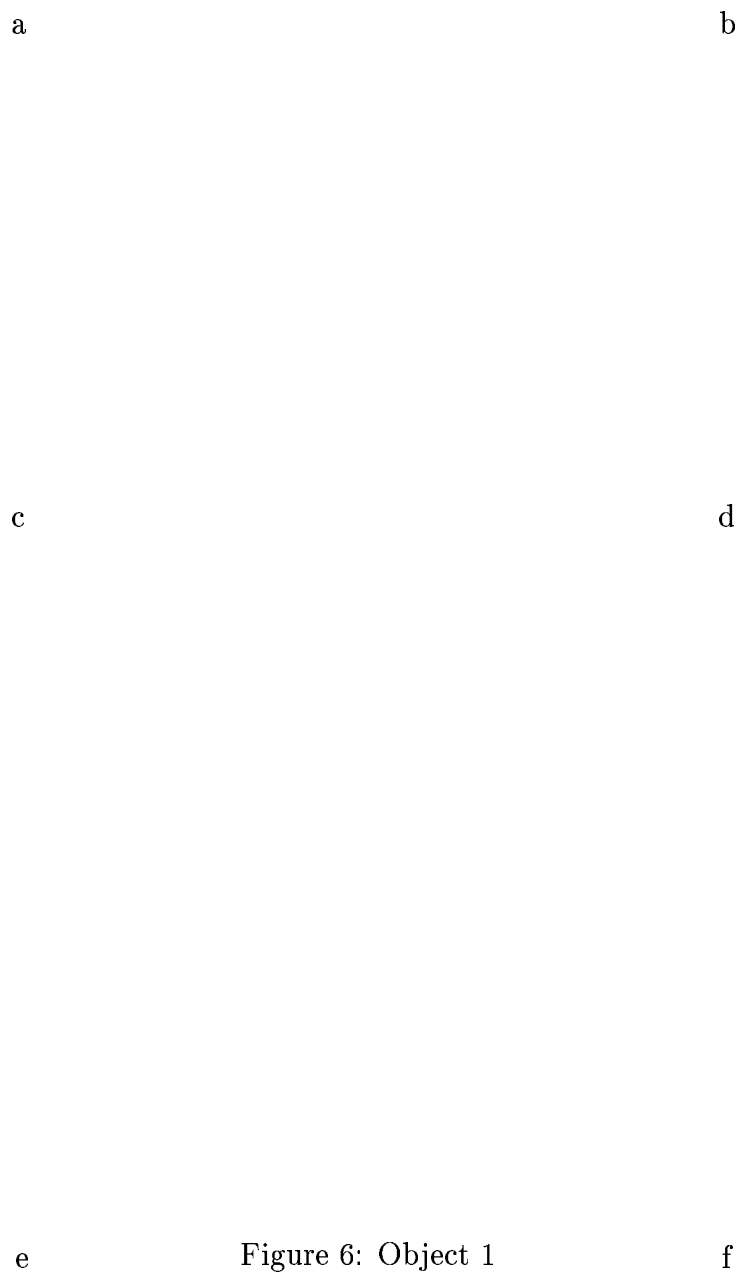

Figure 6: Object 1

a

b

c

d

e

f

Figure 7: Object 2

Figure 8:

Figure 9:

Figure 10:

Figure 11:

Figure 12:

Figure 13:

Figure 14:

Figure 15:

Figure 16:

Figure 17: