Noise Sensitivity Analysis of Depth-from-Defocus by a Spatial-Domain Approach

Murali Subbarao   Jenn-Kwei Tyan
Department of Electrical Engineering,
State University of New York
Stony Brook, NY 11794-2350
email: murali@sbee.sunysb.edu , jktyan@sbee.sunysb.edu

ABSTRACT

Depth-from-Defocus (DFD) using the Spatial-Domain Convolution/Deconvolution Transform Method (STM) is a useful technique for 3D vision. STM involves simple local operations in the spatial domain on only two images recorded with different camera parameters (e.g. by changing lens position or changing aperture diameter). In this paper we provide a theoretical treatment of the noise sensitivity analysis of STM and verify the theoretical results with experiments. This fills an important gap in the current research literature wherein the noise sensitivity analysis of STM is limited to experimental observations. Given the image and noise characteristics, here we derive an expression for the Root Mean Square (RMS) error in lens lens position for focusing an object. This RMS error is useful in estimating the uncertainty in depth obtained by STM. We present the results of computer simulation experiments for different noise levels. The experiments validate the theoretical results.

Keywords: shape-from-focus, depth-from-defocus, image defocus, autofocusing, noise sensitivity.

1 Introduction

Three-dimensional (3D) shape recovery through image defocus analysis has been called Depth-from-Defocus (DFD). In DFD methods, unlike Depth-from-Focus (DFF) methods, it is not required to focus an object in order to find the distance of the object. DFD has been investigated by many researchers.[3–7] In image defocus analysis or DFD, a few images (about 2-3) of a 3D scene are acquired by a camera with different degrees of defocus. The distance of an object is directly estimated from a measure of the level of defocus in the images. DFD methods do not involve searching for camera parameters (e.g. lens position or focal length) in contrast with Depth-from-Focus methods which require a large number of images recorded with different camera parameters, and searching for the parameters that result in the sharpest focused image.[8–11]

In DFD, a method based on a new *Spatial Domain Convolution/Deconvolution Transform* was proposed by Subbarao and Surya.[7,14] This method, named S Transform Method or STM, requires only two images taken with different camera parameters such as lens position, focal length and aperture diameter. Both images can be arbitrarily blurred and neither of them needs to be a focused image. STM is faster in comparison with DFF methods, but less accurate. Earlier, we have provided a theoretical and experimental treatment of noise sensitivity of DFF methods.[11] In this paper we address the noise sensitivity of the DFD approach based on STM. A theoretical treatment of this problem is presented, and a method is developed for estimating the Root-Mean Square (RMS) error for STM. In the existing literature, noise sensitivity of DFD methods is limited to experimental observations only. In this context, the theory developed in this paper is new, and it facilitates estimating errors under various levels of noise and image contrast (as measured by its Laplacian). In order to verify our theory easily and precisely, we

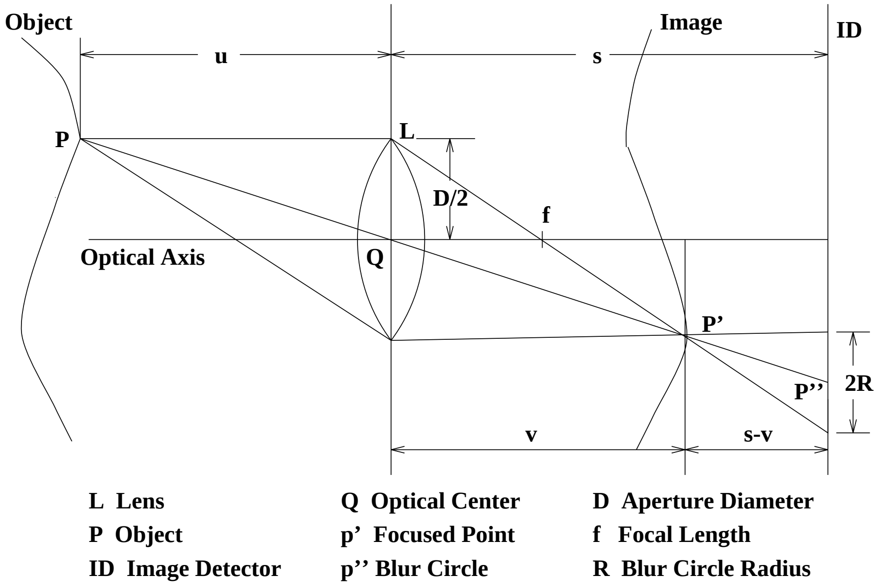

Figure 1: Image Formation in a Convex Lens

use sampled images of a blurred step edge and a cubic polynomial function in simulation experiments. In our analysis, random noise is assumed to be additive, zero-mean, and uncorrelated. Our analysis shows that the RMS error of STM in terms of the focused lens position of an object is linearly related to the noise standard deviation.

Next section describes the camera model. Section 3 provides a summary of STM approach to DFD. The noise sensitivity analysis of STM is presented in Section 4, and experimental results in Section 5.

2 Camera Model and Defocused Image

The image formation in a camera with variable camera parameters (s, f, D) is shown in Fig. 1. Here s specifies the lens position, f the focal length, and D the aperture diameter. The camera parameter setting is denoted by $\mathbf{e_i}$ where $\mathbf{e_i} = (s_i, f_i, D_i)$ is a vector representing the i-th camera parameter setting. The relationship between the focused lens position v, the focal length of the lens f, and the distance of the object u, is given by the well-known lens formula

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

According to paraxial geometric optics, it can be shown using the lens formula and similar triangles in Fig. 1 that the radius R of the blur circle is

$$R = \frac{Ds}{2}\left(\frac{1}{f} - \frac{1}{u} - \frac{1}{s}\right) \tag{2}$$

where u is the object distance. Image magnification varies with the distance s between the lens and the image detector in Fig. 1. Therefore the magnification is normalized corresponding to some standard distance s_0. After this normalization, the blur circle radius R' becomes

$$R' = \frac{Ds_0}{2}\left(\frac{1}{f} - \frac{1}{u} - \frac{1}{s}\right) \tag{3}$$

The blur circle corresponds to the Point Spread Function (PSF) of the camera. Let the PSF be denoted by $h(x,y)$. If the object surface is approximated to be planar and parallel to the image detector plane in a small image region, then the observed image $g(x,y)$ is the convolution of the focused image $f(x,y)$ with the PSF $h(x,y)$. Denoting the convolution operation by $*$, we can write:

$$g(x,y) = h(x,y) * f(x,y) \tag{4}$$

Circularly symmetric PSFs are parameterized in terms of a *spread parameter* σ defined by

$$\sigma^2 \;=\; \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x^2+y^2)\, h(x,y)\, dx\; dy \tag{5}$$

It can be shown that σ is related to the blur circle radius R' by

$$\sigma = \frac{R'}{\sqrt{2}} \tag{6}$$

For two camera parameter settings $\mathbf{e}_1, \mathbf{e}_2$, we obtain from Eqs. (3) and (6):

$$\sigma_1 = \frac{D_1 s_0}{2\sqrt{2}} \left(\frac{1}{f_1} - \frac{1}{u} - \frac{1}{s_1} \right) \quad , \quad \sigma_2 = \frac{D_2 s_0}{2\sqrt{2}} \left(\frac{1}{f_2} - \frac{1}{u} - \frac{1}{s_2} \right) \tag{7}$$

Eliminating $1/u$ from the above two relations we obtain

$$\sigma_1 = \alpha\sigma_2 + \beta \quad \text{where} \;\; \alpha = \frac{D_1}{D_2} \;\; , \;\; \beta = \frac{D_1 s_0}{2\sqrt{2}} \left(\frac{1}{f_1} - \frac{1}{f_2} + \frac{1}{s_2} - \frac{1}{s_1} \right) \tag{8}$$

3 STM: A Spatial Domain Approach to Depth-from-Defocus

In this section we provide a very brief summary of STM. Full details of this method can be found in.[7,12] A new spatial-domain convolution/deconvolution transform (S Transform) is defined in.[14] The definition of the transform for the general case is quite complicated. However, a special case of the transform suffices for image defocus analysis. This case turns out to be particularly simple.

For two-dimensional images, under a local cubic polynomial model, the transform is defined as follows. If a focused image f is blurred by convolution with a circularly symmetric PSF h to result in the blurred image g, then g is the *forward S transform* of f with respect to the kernel function h, and it is given by:

$$g(x,y) = f(x,y) + \frac{\sigma^2}{4} \bigtriangledown^2 f(x,y) \tag{9}$$

where $\bigtriangledown^2$ is the *Laplacian* operator.

The *inverse S transform* of g with respect to the moment vector $(1, \sigma^2/2)$ is equal to f and it is defined as

$$f(x,y) \;=\; g(x,y) - \frac{\sigma^2}{4} \bigtriangledown^2 g(x,y) \tag{10}$$

Therefore, if two blurred images g_1 and g_2 are acquired with different camera parameter settings $\mathbf{e}_1$ and $\mathbf{e}_2$ corresponding to blur parameters σ_1 and σ_2, we obtain

$$f(x,y) \;=\; g_1(x,y) - \frac{\sigma_1^2}{4} \bigtriangledown^2 g_1(x,y) \quad , \quad f(x,y) \;=\; g_2(x,y) - \frac{\sigma_2^2}{4} \bigtriangledown^2 g_2(x,y) \tag{11}$$

Eliminating $f(x,y)$ from the above two relations, and using Eq. (8), and the fact that $\bigtriangledown^2 g_1 = \bigtriangledown^2 g_2$ (which can be proved using Eq. (9) where $f(x,y)$ is a cubic polynomial), we obtain

$$(\alpha^2 - 1)\sigma_2^2 + 2\alpha\beta\sigma_2 + \beta^2 = \frac{8(g_1 - g_2)}{\bigtriangledown^2 g_1 + \bigtriangledown^2 g_2} \tag{12}$$

where α and β are as defined in the previous section.

In the above equation σ_2 is the only unknown. The equation is quadratic and therefore σ_2 is easily obtained by solving it. The two solutions result in a two-fold ambiguity. Methods for resolving this ambiguity are discussed in.[5,7,12,13] From the solution for σ_2, the distance u of the object is obtained from Eq. (7).

The above discussion illustrates the conceptual feasibility of determining distance from two defocused images. Repeating the above procedure in all image neighborhoods, the depth-map of an entire scene can be obtained from only two blurred images in parallel.

4 Noise Sensitivity Analysis of STM

In this section we derive expressions for the *expected value* (mean) and *variance* of the focused lens position v in the lens formula (Fig. 1) for STM. The mean and variance will be expressed in terms of the camera parameters $\mathbf{e}_i$ and the noise variance σ_n^2. These expressions are useful in computing the root mean square (RMS) error for STM in estimating the focused lens position v due to noise. We start the derivation from the right hand side of Eq. (12).

Let g_k be the blurred noise free discrete images, η_k be the additive noise for $k = 1, 2$. The two noisy blurred digital images recorded by the camera are

$$g_{\eta_k} = g_k + \eta_k \quad \text{for } k = 1, 2 \tag{13}$$

The noise η_k at different pixels are assumed to be independent, identically distributed (i.i.d.) random variables with zero mean and standard deviation σ_n.

In STM the assumption of local cubic polynomial model for the focused image is relaxed by using a set of discrete image smoothing filters. The recorded images are convolved with a smoothing kernel L_o in this case as

$$g'_{\eta_k} = g_{\eta_k} * L_o \tag{14}$$

where $*$ denotes convolution. Another filter L_2 is used for estimating the second order image derivatives for computing the Laplacian of the image. Denoting the image Laplacian by g''_{η_k}, we have

$$g''_{\eta_k} = g_{\eta_k} * L_2 \tag{15}$$

Using Eqs. (13), (14) and (15), we can rewrite the right hand side of Eq. (12) as

$$\begin{aligned} G &= 8\frac{g'_{\eta_1} - g'_{\eta_2}}{g''_{\eta_1} + g''_{\eta_2}} && (16) \\ &= 8\frac{(g_1 * L_o - g_2 * L_o) + (\eta_1 * L_o - \eta_2 * L_o)}{(g_1 * L_2 + g_2 * L_2) + (\eta_1 * L_2 + \eta_2 * L_2)} && (17) \end{aligned}$$

In the above equation, there are two components in both the numerator and the denominator where one component depends on image signal only (but not noise) and the other component on noise only. Let P and Q denote the numerator and the denominator respectively in the above equation, and let $P = \overline{P} + P'$ and $Q = \overline{Q} + Q'$, where

$$\begin{aligned}
\overline{P} &= g_1 * L_o - g_2 * L_o && (18)\\
P' &= \eta_1 * L_o - \eta_2 * L_o && (19)\\
\overline{Q} &= g_1 * L_2 + g_2 * L_2 && (20)\\
Q' &= \eta_1 * L_2 + \eta_2 * L_2 && (21)
\end{aligned}$$

Now we can write

$$\begin{aligned}
G &= 8\frac{P}{Q} = \frac{\overline{P}+P'}{\overline{Q}+Q'}\\
&= 8\frac{\overline{P}}{\overline{Q}}\left(1+\frac{P'}{\overline{P}}\right)\left(1+\frac{Q'}{\overline{Q}}\right)^{-1} && (22)
\end{aligned}$$

Assuming $\overline{Q} \gg Q'$ and $\overline{P}\,\overline{Q} \gg P'\,Q'$ (which will be true when the signal-to-noise ratio is sufficiently large), we obtain

$$G \approx 8\frac{\overline{P}}{\overline{Q}}\left(1+\frac{P'}{\overline{P}}-\frac{Q'}{\overline{Q}}\right) \qquad (23)$$

Note that we cannot assume that $\overline{P} \gg P'$ because, the two images may have similar blur level although they correspond to quite different camera parameter settings, and therefore $\overline{P}$ may be close to zero.

Now the expected value of G is

$$E\{G\} = 8\frac{\overline{P}}{\overline{Q}} + \frac{8}{\overline{Q}}E\{P'\} - \frac{8\overline{P}}{\overline{Q}^2}E\{Q'\} \qquad (24)$$

Since we assume η_k to be zero mean, the last two terms of the above equation will vanish. (Note that the expectation operator E is linear and commutes with summation.) Therefore we obtain

$$E\{G\} = 8\frac{\overline{P}}{\overline{Q}} \qquad (25)$$

This result shows that the expected value $E\{G\}$ depends only on the signal, but not noise.

Now we consider the variance of G. Using Eqs (23) and (25), and noting that $E\{P'\} = E\{Q'\} = 0$, we obtain

$$\begin{aligned}
Var\{G\} &= E\{G^2\} - (E\{G\})^2\\
&= E\{64\left(\frac{\overline{P}}{\overline{Q}}+\frac{P'}{\overline{Q}}-\frac{\overline{P}Q'}{\overline{Q}^2}\right)^2\} - \left(8\frac{\overline{P}}{\overline{Q}}\right)^2\\
&= \frac{64}{\overline{Q}^2}E\{P'^2\} + \frac{64\overline{P}^2}{\overline{Q}^4}E\{Q'^2\} - \frac{128\overline{P}}{\overline{Q}^3}E\{P'Q'\} && (26)
\end{aligned}$$

In the Appendix, it is shown that the term $E\{P'Q'\}$ vanishes, and the term $E\{P'^2\}$ and $E\{Q'^2\}$ will depend on the standard deviation σ_n of the noise and the coefficients of the filters. Denoting the coefficients of L_0 filter by $a_0(i,j)$ and the coefficients of L_2 filter by $a_2(i,j)$, we obtain

$$Var\{G\} \; = \; \mathcal{A}\sigma_n^2 \tag{27}$$

where

$$\mathcal{A} \; = \; \frac{128}{\overline{Q}^2}\sum_{i,j}^{M} a_0^2(i,j) + \frac{128\overline{P}^2}{\overline{Q}^4}\sum_{i,j}^{M} a_2^2(i,j) \tag{28}$$

We see that $\mathcal{A}$ is independent of noise and therefore $Var(G)$ is proportional to noise variance σ_n^2.

Next we consider the left hand side of Eq (12). We limit our analysis to the case where the aperture diameter is not changed (i.e. $D_1 = D_2$), but the lens position and/or the focal length are changed (i.e. $f_1 \neq f_2$ and/or $s_1 \neq s_2$). In this case, the quadratic equation reduces to a linear equation because $\alpha = 1.0$. Therefore we get the unique solution:

$$\sigma_2 = \frac{G - \beta^2}{2\beta} \tag{29}$$

From Eq (7) and the lens formula, we derive an approximate linear expression for focused lens position v:

$$\begin{aligned} \sigma_2 \; &= \; \frac{Ds_0}{2\sqrt{2}}\left[\frac{1}{f_2} - \frac{1}{u} - \frac{1}{s_2}\right] \\ &= \; \frac{Ds_0}{2\sqrt{2}}\left[\frac{1}{v} - \frac{1}{s_2}\right] \\ &\approx \; \frac{D}{2\sqrt{2}s_0}(s_2 - v) \end{aligned} \tag{30}$$

where we assumed $s_2 v \approx s_0^2$. This approximation is valid for most camera systems in machine vision.

Now substituting for σ_2 using the above equation into Eq. (29), we obtain

$$\begin{aligned} v \; &= \; -\frac{2\sqrt{2}s_0}{D}\left(\frac{G}{2\beta} - \frac{\beta}{2} - \frac{Ds_2}{2\sqrt{2}s_0}\right) \\ &= \; \mathcal{B}G + \mathcal{C} \end{aligned} \tag{31}$$

where

$$\mathcal{B} = -\frac{\sqrt{2}s_0}{\beta D} \quad \text{and} \quad \mathcal{C} = s_2 + \frac{\sqrt{2}s_0\beta}{D} \tag{32}$$

$\mathcal{B}$ and $\mathcal{C}$ depend only on the camera parameters. Thus, the expected value and the variance of v are obtained by combining Eqs. (27) and (31) as:

$$E\{v\} = 8\mathcal{B}\frac{\overline{P}}{\overline{Q}} + \mathcal{C} \quad \text{and} \quad Var\{v\} = \mathcal{A}\mathcal{B}^2\sigma_n^2 \tag{33}$$

The above equations show that the standard deviation of focused lens position v is linearly related to the standard deviation of noise. Given the noise standard deviation σ_n, the camera parameters, and the defocused images, we can directly compute the RMS error for STM using the above formula. In addition, we can use this equation to select pixels with good signal-to-noise ratio in order to obtain reliable depth estimates.

5 Experiments

The validity of Eq. (33) for the expected value and variance of the focused lens position v was verified through simulation experiments. The experiments were carried out for two objects, one a step edge and another a cubic polynomial. The blurred images of the test objects were obtained by simulating a camera system similar to the one used in the original implementation of STM reported in.[7] The parameters of the camera system were– focal length 35 mm, F-number 4, and pixel (CCD) size 0.013 mm X 0.013 mm. The distance s (see Fig. 1) between the lens and the image detector was assumed to be varied by a stepper motor with each step corresponding to a displacement of 0.030 mm. The distance s and the focused lens position v are expressed in terms of the step number of the stepper motor with step 0 corresponding a distance of focal length f. Therefore when the lens was at step 0, objects at infinity were focused on the image detector and at step 90 objects at about 49 cm were focused. For convenience, we specify the distance of objects in terms of the lens step number for which it is in focus.

In the first experiment, we used a step function as an input object (Fig. 2). A vertically oriented step edge is defined as

$$f(x,y) = \begin{cases} A & \text{for } x > 0 \\ \frac{A+B}{2} & \text{for } x = 0 \\ B & \text{for } x < 0 \end{cases} \tag{34}$$

where A and B represent the uniform brightness for two regions. We chose $A = 176$ and $B = 80$ and an image of size 64×64 in our experiment. The edge is a vertical line located in the middle of the image. For a given distance of this object from the camera, the corresponding blur circle radius R can be calculated using the camera parameters. Next the blurred image sensed by the camera can be computed by convolving the focused image with the point spread function corresponding to a blur circle radius of R. In the case of a step edge, as an alternative to numerical convolution, an analytical expression can be obtained for the blurred image and the expression can be evaluated and sampled at discrete points. This latter approach was followed in our experiments. It can be shown[1] that the defocused image of the step edge defined above will be[1]:

$$g(x,y) = B + (A-B)\left[\frac{1}{2} + \frac{1}{\pi}\left(sin^{-1}\left(\frac{x}{R}\right) + \frac{x}{R}\sqrt{1-\left(\frac{x}{R}\right)^2}\right)\right] \quad \text{for } |x| < R \tag{35}$$

where R is the radius of the blur circle. Two defocused images g_1 and g_2 were computed for two different lens positions of step number 10 and 40 respectively (all other camera parameters were left unchanged). A zero-mean Gaussian random noise was then added to both defocused images g_1 and g_2.

According to Eq. (29) we only need to compute the value of σ_2 at one pixel (x,y) in the image to obtain an estimate of the distance. This pixel should be chosen such that it is blurred in both g_1 and g_2 and the signal to noise ratio is sufficiently high in both g_1 and g_2 as required by Eq. (23). Also the images should be such that their Laplacians should be roughly the same in order for the local cubic polynomial model assumption used by STM to be applicable. Further the degree of blur should not be too high in either of the two images (the blur circle diameter should not exceed about 12 pixels) so that the error due to the image overlap problem[7] remains low. In the experiments, when one of the image was blurred too much, then a third image g_3 with a lower level of blur was used in place of the highly blurred image. More specifically, the images g_1 and g_2 were computed for lens position of step 10 and 40 respectively. When the focused position v predicted by these two images was more than step 50, then g_1 was considered to be blurred too much and therefore a third image g_3 corresponding to lens position at step 70 was computed and used. Fig. 2 shows three defocused images for lens positions at steps 10, 40 and 70, respectively when the object focused position was at step 0. The corresponding widths of the blur edge for the three images were 6, 24 and 40 pixels respectively (Fig. 4).

We compute G in Eq. (12) from two defocused images, and the camera constant β is computed from a knowledge of the camera parameters. Then an estimate of the focused lens position v is calculated from Eq. (31). This procedure for estimating distance v is repeated 20 times at a given noise level and camera parameter setting. The experimental mean and standard deviation of v are calculated from these 20 trials. The theoretical mean and standard deviation of v is computed based on Eq. (33). Table 1, 2 and 3 show the results of the experimental and theoretical computations at three noise levels for various distances of the object (from focused position of step 0 to step 90). These tables also show for each case the signal-to-noise ratio defined by

$$SNR \;=\; 10\,\log_{10}\left(\frac{\overline{Q}^2}{\frac{1}{N}\sum_{i=0}^{N} Q'^2_i}\right) \tag{36}$$

where $\overline{Q}$ and Q'_i are as defined in Eqs (20) and (21). A comparison of results in the tables shows that the experimental and theoretical results are in close agreement thus verifying the theory. In particular, as predicted by theory, we see that the standard deviation of v (i.e. the RMS error of focused lens position) is linearly related to noise standard deviation.

In the second experiment, an object having an image brightness corresponding to a cubic polynomial was used. The procedure was similar to the first experiment. The coefficients of the cubic polynomial were taken to be those that modeled the step edge in the first experiment. The coefficients were obtained by using a least squares error fitting technique.[2] The resulting image was

$$f(x,y) = a_3x^3 + a_2x^2 + a_1x + a_0 \tag{37}$$

where $a_3 = -0.003$, $a_2 = 0.313$, $a_1 = -5.957$ and $a_0 = 101.031$. In this case, by convolving $f(x,y)$ with the PSF of the camera corresponding to a blur circle of radius R, it can be shown that the blurred image is given by

$$g(x,y) = a_3x^3 + a_2x^2 + \left(\frac{3R^2}{4}a_3 + a_1\right)x + \frac{R^2}{4}a_2 + a_0 \tag{38}$$

The blurred images g_1 and g_2 are computed by discrete sampling of the above function for two different blur circle radii R_1 and R_2 respectively. These are then filtered with L_0 and L_2 filters to obtain smoothed images and their Laplacians. In this experiment, unlike the previous one, almost any pixel can be selected for computing v, and a third image is not needed. The results are good even for highly blurred cases. However, as mentioned before, we use the third image just to reduce the variance of v. Three sampled images computed for lens positions at step 10, 40 and 70, when the object is focused at step 90, are shown in Fig. 3. The blur circle radii corresponding to these images are respectively 22, 13 and 5 pixels respectively. Fig. 5 shows their gray-level profile. Table 4,5 and 6 show the experimental and theoretical results. Once again, we see that the results are in a good agreement.

6 Conclusion

Eq. (33) provides a method for estimating the uncertainty in the focused position v for an object as a function of the camera parameters, noise level, and image signal. This can be used to select those pixels that yield relaible estimates of depth of objects and ignore the unreliable pixels in the application of STM. As expected, the reliability of pixels increases with increasing value of image Laplacian at those pixels but decreases with increase in noise standard deviation. If an object is planar, or has a known shape form (e.g. spherical), then Eq. (33) can be used to combine the depth information provided by different pixels in some optimal manner to infer the actual shape parameters of the object.

Acknowledgement: The support of this research in part by a grant from Olympus Optical Co. is gratefully acknowledged.

7 Appendix A

In this section we discuss the derivation of Eq (27) by simplifying Eq (26). Let the smoothing filter L_0 and the Laplacian filter L_2 both be of size $(2M+1)$x$(2M+1)$. Let the coefficients of the two filters be $a_0(i,j)$ and $a_2(i,j)$ respectively. Also, let $\sum_{i,j}^{M}$ denote $\sum_{i=-M}^{M}\sum_{j=-M}^{M}$. Then filtering with L_0 and L_2 are convolution operations defined respectively by

$$\begin{aligned} \eta_k * L_o &= \sum_{i,j}^{M} a_0(i,j)\eta_k(m-i,n-j) && (39) \\ \eta_k * L_2 &= \sum_{i,j}^{M} a_2(i,j)\eta_k(m-i,n-j) && (40) \end{aligned}$$

There are three terms– $E\{P'^2\}$, $E\{Q'^2\}$ and $E\{P'Q'\}$– that need to be discussed in Eq (26). We start from the term $E\{P'^2\}$. Using Eqs. (19, 21) we obtain

$$\begin{aligned} E\{P'^2\} &= E\{(\eta_1 * L_o - \eta_2 * L_o)^2\} \\ &= E\{(\eta_1 * L_o)^2\} + E\{(\eta_2 * L_o)^2\} - 2E\{(\eta_1 * L_o)(\eta_2 * L_o)\} && (41) \end{aligned}$$

In the above equation, the first term can be written as

$$E\{(\eta_1 * L_o)^2\} = \sum_{i_1,j_1}^{M}\sum_{i_2,j_2}^{M} a_0(i_1,j_1)a_0(i_2,j_2)E\{\eta_1(m-i_1,n-j_1)\eta_1(m-i_2,n-j_2)\} \tag{42}$$

If $i_1 \neq i_2$ or $j_1 \neq j_2$, then, since noise in different pixels are independent and zero mean,

$$E\{\eta_1(m+i_1,n+j_1)\eta_1(m+i_2,n+j_2)\} = E\{\eta_1(m+i_1,n+j_1)\}E\{\eta_1(m+i_2,n+j_2)\} = 0 \tag{43}$$

However, if $i_1 = i_2$ and $j_1 = j_2$, then

$$E\{\eta^2(m-i_1,n-j_1)\} \;=\; \sigma_n^2 \tag{44}$$

Therefore, we get

$$E\{(\eta_1 * L_o)^2\} = \sum_{i,j}^{M} a_0^2(i,j)\sigma_n^2 \tag{45}$$

Similarly, for the second term $E\{(\eta_2 * L_o)^2\}$ in Eq (41), we obtain the same result as above. For the remaining term we have

$$2E\{(\eta_1 * L_o)(\eta_2 * L_o)\} = 2\,(E\{\eta_1\} * L_o)\,(E\{\eta_2\} * L_o) = 0 \tag{46}$$

Thus, we obtain

$$E\{P'^2\} = 2\sum_{i,j}^{M} a_0^2(i,j)\sigma_n^2 \tag{47}$$

The derivation for $E\{Q'^2\}$ is similar to that for $E\{P'^2\}$ above, and therefore we have

$$E\{Q'^2\} = 2\sum_{i,j}^{M} a_2^2(i,j)\sigma_n^2 \tag{48}$$

Now consider the term $E\{P'Q'\}$,

$$\begin{aligned} E\{P'Q'\} &= E\{(\eta_1 * L_o - \eta_2 * L_o)(\eta_1 * L_2 + \eta_2 * L_2)\} \\ &= E\{(\eta_1 * L_o)(\eta_1 * L_2)\} + E\{(\eta_1 * L_o)(\eta_2 * L_2)\} \\ &\quad -E\{(\eta_2 * L_o)(\eta_1 * L_2)\} - E\{(\eta_2 * L_o)(\eta_2 * L_2)\} \end{aligned} \tag{49}$$

Using arguments similar to those in simplifying Eqs. (42) and (46), we obtain

$$E\{(\eta_1 * L_o)(\eta_1 * L_2)\} = E\{(\eta_2 * L_o)(\eta_2 * L_2)\} = \sum_{i,j}^{M} a_0(i,j)a_2(i,j)\sigma_n^2 \tag{50}$$

as well as

$$E\{(\eta_1 * L_o)(\eta_2 * L_2)\} = E\{(\eta_2 * L_o)(\eta_1 * L_2)\} = 0 \tag{51}$$

All the terms cancel out or vanish and result in $E\{P'Q'\}$ to be zero. Therefore, combining all the results from Eqs. (47, 48 and 49), we can rewrite Eq (26) as

$$Var\{G\} = \frac{128}{\overline{Q}^2}\sigma_n^2\sum_{i,j}^{M} a_0^2(i,j) + \frac{128\overline{P}^2}{\overline{Q}^4}\sigma_n^2\sum_{i,j}^{M} a_2^2(i,j) \tag{52}$$

8 REFERENCES

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

[2] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical Recipes in C: the art of scientific computing*, 2nd ed., Cambridge University Press, 1992.

[3] 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.

[4] 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.

[5] M. Subbarao, and T. Wei, "Depth from Defocus and Rapid Autofocusing: A Practical Approach", Proceedings of IEEE Computer Society conference on Computer Vision and Pattern Recognition, June 1992, pp. 773-776.

[6] M. Subbarao, "Parallel Depth Recovery by Changing Camera Parameters", *Second International Conference on Computer Vision*, Florida, USA, pp. 149-155, December 1988.

[7] M. Subbarao and G. Surya, "Depth from Defocus: A Spatial Domain Approach", *International Journal of Computer Vision*, 13, 3, pp. 271-294 (1994).

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

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

[10] M. Subbarao, T. Choi, and A. Nikzad, "Focusing Techniques", *Journal of Optical Engineering*, Vol. 32 No. 11, pp. 2824-2836, November 1993.

[11] M. Subbarao, and J.K. Tyan, "The Optimal Focus Measure for Passive Autofocusing and Depth-from-Focus", *Proceedings of SPIE conference on Videometrics IV* , Philadelphia, Oct 1995.

[12] G. Surya, "Three-dimensional scene recovery from image defocus", Ph.D. Thesis, Dept. of Electrical Engg., SUNY at Stony Brook, 1994.

[13] T. Wei, "Three-dimensional machine vision using image defocus",

Ph.D. Thesis, Dept. of Electrical Engg., SUNY at Stony Brook, 1994.

[14] 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.

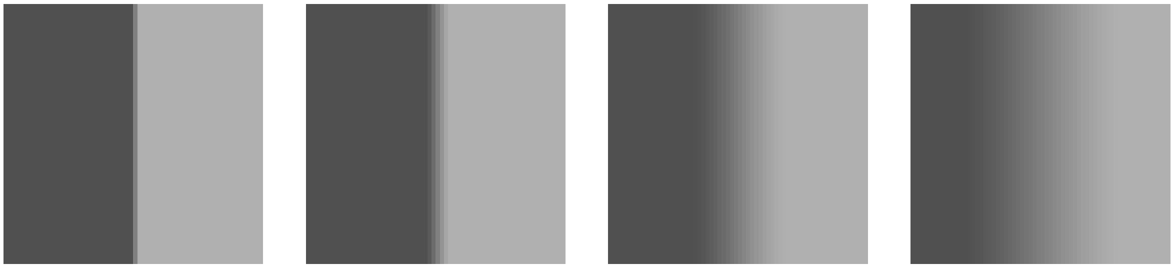

Figure 2: Step edge focused at step 0 and defocused images at lens steps 10,40,70

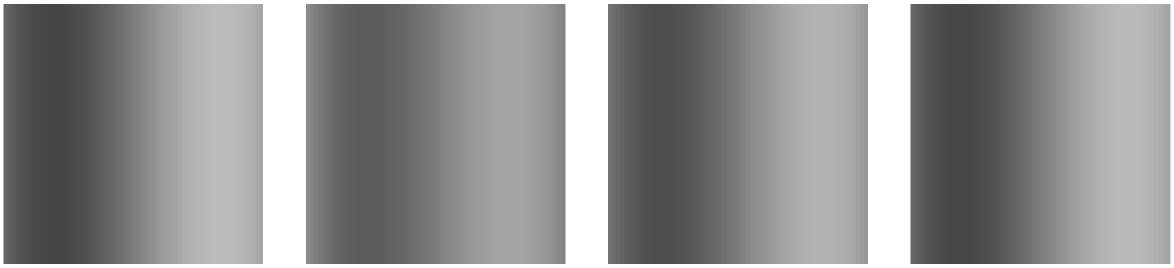

Figure 3: Cubic polynomial focused at step 90 and defocused images at lens steps 10,40,70

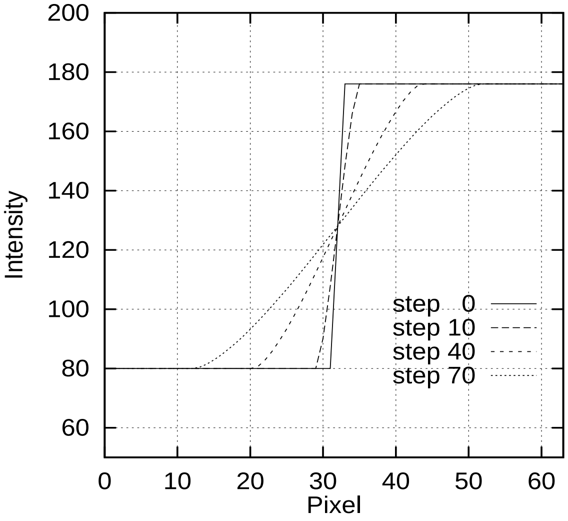

Figure 4: Gray-level in edge image

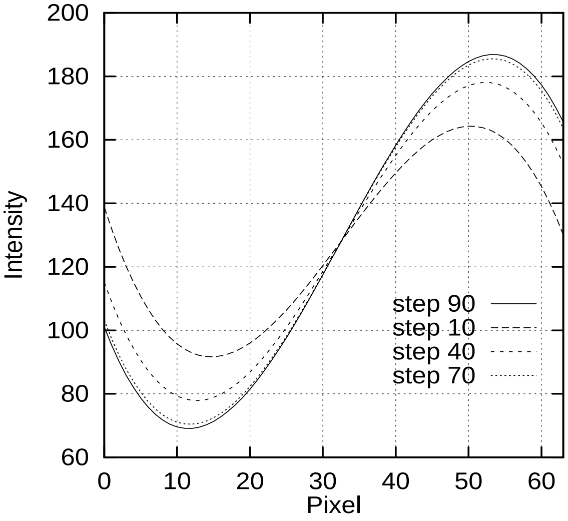

Figure 5: Gray-level in polynomial image

Object	Lap. img	**Theo.**	**Exp.**	**Diff**	**Theo.**	**Exp.**	**Diff**
Distance	SNR (dB)	E{v}	E{v}	δ	Std{v}	Std{v}	δ
0	28.47	-0.80	-0.80	0.00	0.78	0.98	-0.20
10	33.54	9.25	9.24	0.01	0.34	0.48	-0.14
20	33.29	19.72	19.74	-0.02	0.41	0.52	-0.11
30	34.35	30.21	30.15	0.06	0.29	0.32	-0.03
40	30.82	40.49	40.48	0.01	0.42	0.48	-0.06
50	31.84	51.09	51.08	0.01	0.41	0.35	0.06
60	32.12	60.16	60.11	0.05	0.30	0.25	0.05
70	29.49	69.08	69.12	-0.04	0.41	0.41	0.00
80	28.78	79.78	79.77	0.01	0.57	0.65	-0.08
90	29.94	89.05	89.31	-0.26	0.65	0.87	-0.22

Table 1: Noise std 1.0

Object	Lap. img	**Theo.**	**Exp.**	**Diff**	**Theo.**	**Exp.**	**Diff**
Distance	SNR (dB)	E{v}	E{v}	δ	Std{v}	Std{v}	δ
0	22.45	-0.80	-0.89	0.09	1.56	2.03	-0.47
10	27.52	9.25	9.21	0.04	0.68	0.99	-0.31
20	27.28	19.72	19.76	-0.04	0.83	1.05	-0.22
30	28.33	30.21	30.10	0.11	0.58	0.64	-0.06
40	24.80	40.49	40.49	0.00	0.84	0.98	-0.14
50	25.82	51.09	51.05	0.04	0.82	0.77	0.05
60	26.10	60.16	60.07	0.09	0.61	0.58	0.03
70	23.47	69.08	69.18	-0.10	0.82	0.81	0.01
80	22.76	79.78	79.92	-0.14	1.14	1.29	-0.15
90	23.92	89.05	89.64	-0.59	1.29	1.69	-0.40

Table 2: Noise std 2.0

Object	lap. img	**Theo.**	**Exp.**	**Diff**	**Theo.**	**Exp.**	**Diff**
Distance	SNR (dB)	E{v}	E{v}	δ	Std{v}	Std{v}	δ
0	18.93	-0.80	-1.07	0.27	2.34	3.05	-0.71
10	23.99	9.25	9.16	0.09	1.02	1.52	-0.50
20	23.75	19.72	19.76	-0.04	1.24	1.60	-0.36
30	24.81	30.21	30.04	0.17	0.88	0.97	-0.09
40	21.28	40.49	40.52	-0.03	1.26	1.50	-0.24
50	22.29	51.09	51.02	0.07	1.23	1.02	0.21
60	22.57	60.16	60.02	0.14	0.92	0.75	0.17
70	19.94	69.08	69.25	-0.17	1.24	1.24	0.00
80	19.24	79.78	80.34	-0.56	1.72	1.98	-0.26
90	20.39	89.05	90.04	-0.99	1.94	2.75	-0.81

Table 3: Noise std 3.0

Object	Lap. img	**Theo.**	**Exp.**	**Diff**	**Theo.**	**Exp.**	**Diff**
Distance	SNR (dB)	E{v}	E{v}	δ	Std{v}	Std{v}	δ
0	30.98	-0.98	-1.09	0.11	0.61	0.74	-0.13
10	31.84	9.62	9.52	0.10	0.49	0.51	-0.02
20	30.79	20.05	20.09	-0.04	0.42	0.42	0.00
30	29.44	30.29	30.43	-0.14	0.43	0.48	-0.05
40	29.56	40.38	40.54	-0.16	0.49	0.56	-0.07
50	30.42	50.30	50.36	-0.06	0.60	0.64	-0.04
60	32.11	60.04	60.10	-0.06	0.44	0.44	0.00
70	29.04	69.64	69.71	-0.07	0.50	0.51	-0.01
80	27.31	79.08	79.22	-0.14	0.60	0.71	-0.11
90	28.88	89.37	89.63	-0.26	0.72	0.91	-0.19

Table 4: Noise std 0.3

Object	Lap. img	**Theo.**	**Exp.**	**Diff**	**Theo.**	**Exp.**	**Diff**
Distance	SNR (dB)	E{v}	E{v}	δ	Std{v}	Std{v}	δ
0	24.96	-0.98	-1.24	0.26	1.22	1.49	-0.27
10	25.82	9.62	9.39	0.23	0.99	1.04	-0.05
20	24.77	20.05	20.15	-0.10	0.85	0.84	0.01
30	23.42	30.29	30.59	-0.30	0.86	0.95	-0.09
40	23.53	40.38	40.75	-0.37	0.99	1.14	-0.15
50	24.39	50.30	50.48	-0.18	1.21	1.31	-0.10
60	26.09	60.04	60.18	-0.14	0.89	0.89	-0.00
70	23.02	69.64	69.81	-0.17	1.01	1.03	-0.02
80	21.28	79.08	79.44	-0.36	1.21	1.41	-0.20
90	22.86	89.37	89.73	-0.36	1.44	1.79	-0.35

Table 5: Noise std 0.6

Object	Lap. img	**Theo.**	**Exp.**	**Diff**	**Theo.**	**Exp.**	**Diff**
Distance	SNR (db)	E{v}	E{v}	δ	Std{v}	Std{v}	δ
0	21.44	-0.98	-1.44	0.46	1.83	2.21	-0.38
10	22.30	9.62	9.24	0.38	1.48	1.59	-0.11
20	21.25	20.05	20.19	-0.14	1.27	1.27	0.00
30	19.89	30.29	30.77	-0.48	1.28	1.48	-0.20
40	20.01	40.38	40.98	-0.60	1.48	1.75	-0.27
50	20.87	50.30	50.65	-0.35	1.81	2.00	-0.19
60	22.56	60.04	60.26	-0.22	1.34	1.34	-0.00
70	19.49	69.64	69.95	-0.31	1.51	1.57	-0.06
80	17.76	79.08	79.73	-0.65	1.81	2.14	-0.33
90	19.33	89.37	90.07	-0.70	2.17	2.59	-0.42

Table 6: Noise std 0.9