VISION

THIRD QUARTER 2000 www.sme.org/mva VOL. 16, NO. 3

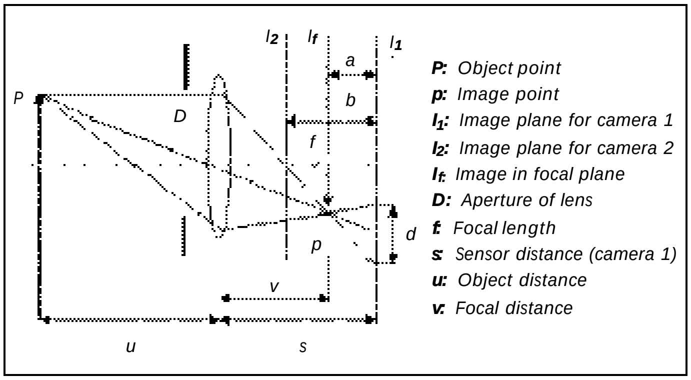
Real-Time 3D Estimation Using Depth from Defocus

Ovidiu Ghita and **Paul Whelan,** Vision Systems Laboratory, School of Electronic Engineering, Dublin City University (Dublin, Ireland)

Recovering the depth information of a scene is one of the most important tasks in machine vision. Depth information plays a key role in machine vision and has a strong relationship with the real world in robotic applications. Three-dimensional information can be obtained in various ways. Several 3D vision systems have been developed to solve a specific task, while others are more general and, consequently, more complex. Among other approaches for 3D recovery, *depth from defocus* (*DFD*) techniques have recently attracted a great deal of interest. Originally developed by Krotkov [3] and Pentland [4], the DFD method uses the direct relationship between the depth, camera parameters, and degree of blurring in several images (in the current implementation only two are used). In contrast with other techniques, such as stereo or motion parallax where solving the correspondence between different features is a major disadvantage, depth from defocus relies only on simple local algorithms; however, these methods are complementary. Stereo and motion parallax methods are used for outdoor scenes where the depth discontinuities are important, while DFD performs better for indoor scenes where the target is situated nearby. Another popular method used in 3D estimation is based on triangulation. In terms of precision, methods based on triangulation appear to perform better, but the major drawback is the amount of computation involved. Some speed improvements have been obtained using gray or color-coded patterns. Ideally, the number of independent colored stripes should be large and geometrically very dense, but in this case, the color-structured pattern is difficult to manufacture. Also, different reflection properties of the object's surface can introduce some errors in 3D estimation (when the color of the stripe is the same as the color of the object's surface). An interesting method to generate a color-structured pattern is proposed by Chen et al. [2]. They proposed a method to design a pattern that has strong contrast at the borders of any two adjacent stripes. The correlation between any two segments of a consecutive sequence of light stripes should be as small as possible to minimize the mismatch.

This article addresses the implementation of a real-time 3D sensor based on depth from defocusing. As was previously mentioned, this method requires only two images acquired using different focal settings. This method performs badly if the scene's texture does not provide high frequencies. A practical solution for this problem is to project a structured light on the scene. In this case, the scene will have a dominant frequency for texture. Xiong and Shafer [10] propose a novel approach to determine dense and accurate depth estimation based on maximal resemblance estimation. This implementation uses a large bank of filters with a different window size tuned for all dominant texture frequencies. Using a large bank of filters makes this approach unsuitable for real-time implementation. Subbarao and Surya [7] pro-

PUBLISHED BY THE MACHINE VISION ASSOCIATION OF THE SOCIETY OF MANUFACTURING ENGINEERS



1. Camera geometry and image formation.

posed the *Spatial-Domain Convolution/Deconvolution Transform* (S Transform) to estimate the depth using an analysis in frequency domain. This implementation does not perform as well as those mentioned previously. Watanabe and Nayar [9] proposed a small bank of broadband rational filters that are able to handle arbitrary textures. This implementation is simple and performs reasonably well, even with weak textures. This approach represents an improvement but still fails when the scene is textureless. Therefore, considering the aforementioned aspects, for this present implementation, the optimal solution is using structured (active) illumination. An important problem is determining the illumination pattern. Nayar et al. [6] proposed a method for optimization in the Fourier domain. The optimal pattern maximizes the sensitivity of the focus measure to enhance the high spatial resolution. Keeping in mind that the CCD sensor can be approximated with an array of square elements (cells), the optimal pattern is a rectangular spatial grid (chessboard). The next step is tuning this filter with the CCD parameters (the distance between two adjacent cells).

The *reversed projection blurring* (RPB) model, used by Asada et al. [1], is a technique used by ray-tracing algorithms, generally in computer graphics. This model uses photometric properties of occluding edges when the object's surface behind the nearer object is partially observed. Therefore, the blurring model using convolution becomes inconsistent around the occluding edges. To compensate for this problem, they use the radiance of the near and far surfaces, and then the occluded region is mapped. In this implementation, the occluded region is assigned to be equal to that from a nearer side of the depth discontinuity; this assumption is proven to be correct in most of the situations.

Theoretical Approach of DFD

Depth from defocus means calculating the depth of the scene in the image from the degree of image blurring. Let *P* be a point that belongs to an object's surface and *p* be the focused point refracted by the lens. The relationship between the object distance *u*, focal length *f*, and image formation distance *v* is given by the lens law.

$$\frac{1}{u}+\frac{1}{v}=\frac{1}{f} \qquad (1)$$

Figure 1 shows the optical settings and basic image formation geometry for a convex lens.

If the CCD sensor is not placed in the focal plane, the image is distributed over a circular patch on the sensing element. The diameter of the blur circle *d* is given by the use of similar triangles.

$$\frac{D/2}{v}=\frac{d/2}{s-v}\Rightarrow d=Ds\left(\frac{1}{v}-\frac{1}{s}\right)\Rightarrow$$

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

The blurring effect is seen as a convolution between the focused image and blurring function.

$$I(x,y)=\iint I_0(u,v)h(x-u,y-v)dudv \qquad (3)$$

where I_0 is the focused image and *h* is the blurring function.

The blurring function, also known as the *point spread function (PSF)*, can be approximated by the following expression:

$$h_p(x,y)=\begin{cases}\frac{4}{\pi d^2} & if \quad x^2+y^2\le\frac{d^2}{4}\\ 0 & otherwise\end{cases} \qquad (4)$$

where h_p is called the pillbox function and can be seen as a cone of light emerging from the lens with the point of the cone in focal plane. If the sensor plane is shifted from the focal plane, then it cuts the cone in a circle with the diameter *d*.

If, within this circle, the brightness is not uniform, the PSF is better approximated by a 2D Gaussian function (Pentland [4]).

$$h(x,y)=\frac{1}{2\pi\sigma^2}e^{-\frac{x^1+y^1}{2\sigma^1}} \qquad (5)$$

where σ is the standard deviation of the distribution of the 2D Gaussian.

In practice, we can assume that the brightness is constant over a region of the image projected onto CCD element, then the result is an invariant shift from the focal plane. The blurring is better modeled by the 2D Gaussian than the blur circle (another advantage is that the

Fourier transform of a Gaussian is also a Gaussian). If the brightness is uniform over a small region of the image (this assumption approximates the practical case very well), σ is proportional to *d*.

$$\sigma = kd \tag{6}$$

where *k* is a constant of proportionality characteristic for every camera and can be determined from a previous camera calibration.

Unless we know a priori information about the scene, one image is not enough to estimate the depth [see Eq. (2) where *d* and *u* are unknown]. Therefore, a minimum of two images acquired with different camera settings are necessary. Clearly, there are two distinct options: either the aperture *D* is maintained constant and the sensor position *s* is modified [6], or the sensor is fixed and the aperture is changing when the images are taken [3,7]. The first case has an important advantage because it does not require any user intervention while the images are acquired, but unfortunately, different magnification is caused by focusing. An effective solution was proposed by Watanabe and Nayar [9] by using telecentric lens. They suggest an optical solution to obtain constant magnification. It is well known that when using *telecentric* optics the magnification remains constant despite the focus changes. Most of the popular commercial lenses can be transformed to telecentric only by adding a small, extra aperture. The aperture will be placed in the front focal plane of the lens. Using telecentric lens, they demonstrate that the magnification changes can be reduced to as low as 0.03%. Because the aperture has to be small, the only drawback of this approach is the severe reduction in brightness. Therefore, to compensate for this, a brighter source of illumination has to be used. The second possible implementation is not hampered by this issue, but the depth estimation is by far not as precise.

Estimating Depth of the Scene

Depth information can be estimated by taking a small number of images under different camera or optical settings. Because the PSF is a circularly symmetrical function, the relationship between the focused and defocused images is illustrated by the next expression (Subbarao and Surya [7]).

$$f(x,y) = g(x,y) - \frac{\sigma^2}{4}\nabla^2 g(x,y) \tag{7}$$

where *f* is the focused image, *g* is the defocused image, σ is the standard deviation for PSF and σ^2, the Laplacian operator. Equation (7) represents the deconvolution formula. If two images g_1 and g_2 are taken under different camera settings and the term *f* (*x*,*y*) from the first equation is replaced in the second equation, the result is a simple expression.

$$g_1(x,y) - g_2(x,y) = \frac{1}{4}(\sigma_1^2 - \sigma_2^2)\nabla^2 g,$$
$$\nabla^2 g = \frac{\nabla^2 g_1 + \nabla^2 g_2}{2} \tag{8}$$

From Eq. (8), it can be observed that no terms depend on the scene's texture frequency. Furthermore, the depth can be estimated using the difference between the standard deviation of the near-focused image (g_1) and far-focused image (g_2). The use of the Laplacian as a focus operator is very convenient because it has a simple kernel, but the depth map resulting from Eq. (8) is accurate only if the depth discontinuities in the scene are important. Also, if the scene has only a weak texture, the depth estimation is poor. Certainly, to obtain a dense and robust depth map, a more sophisticated approach for modeling PSF has to be developed.

Nevertheless, the focus operator plays an important role in the depth estimation stage. Therefore, the goal of this article is to study the accuracy of depth estimation when used by different operators.
Because the defocus function (PSF) acts like a low-pass filter, the focus operator has to perform inverse filtering. The next step is determining depth from two images. The simplest solution is to use the ratio between the defocus function of the near and far-focused images. Nayar et al. [6] proposed a normalized ratio M/P that is a monotonic and bounded function.

$$\frac{M}{P} = \frac{g_1(x,y) - g_2(x,y)}{g_1(x,y) + g_2(x,y)}$$
$$= \frac{H(p,q,\sigma_1) - H(p,q,\sigma_2)}{H(p,q,\sigma_1) + H(p,q,\sigma_2)} \tag{9}$$

where *H* is the Fourier transform of the PSF and σ_1 is the standard deviation of the near-focused image (σ_2 is the standard deviation for the far-focused image).

Active Illumination

If the scene is highly textured, the depth estimation will be precise and reliable. Unfortunately, if the scene has a weak texture or is textureless (like a blank sheet of plain paper), the depth recovery is far from accurate. An effective and relatively simple solution is based on the use of structured (active) light. Initially, the solution (suggested by Pentland et al. [5] and later Nayar et al. [6]) was to develop a symmetrical pattern as a rectangular spatial grid optimized for a specific type of camera. Therefore, the illumination pattern has a sin-

0	-1	0	-1	-1	-1	0.55	-1	0.55
-1	4	-1	-1	8	-1	-1	1.8	-1
0	-1	0	-1	-1	-1	0.55	-1	0.55
	(a)			(b)			(c)	

2. Focus operator kernels: (a) Laplacian (4), (b) Laplacian (8), and (c) rational operator (3 × 3).

-0.143	0.1986	-0.1056	-0.07133	-0.1056	-0.1986	-0.143
-0.1986	-0.1927	0.01795	0.07296	0.01795	-0.1927	-0.1986
-0.1056	0.01795	0.2843	0.4601	0.2843	0.01795	-0.1056
-0.07133	0.07296	0.4601	0.6449	0.4601	0.07296	-0.07133
-0.1056	0.01795	0.2843	0.4601	0.2843	0.01795	-0.1056
-0.1986	-0.1927	0.01795	0.07296	0.01795	-0.1927	-0.1986
-0.143	-0.1986	-0.1056	-0.07133	-0.1056	-0.1986	-0.143

3. Focus operator kernel of a rational operator (7 × 7).

gle, dominant frequency in direct correlation with the pattern's arrangement for transparent and opaque regions. When the structured light is projected onto the scene, the spectrum will have the same dominant frequency.

The resulting pattern is very dense and rotational symmetrical to obtain spatial invariance. A problem caused by using a dense spatial pattern is the reduction in illumination caused by the filter's opaque regions, thus a very powerful source of light is required. Nevertheless, a very precise pattern is difficult to fabricate, and our testing concluded that this issue is not as restrictive as it seems. For the current implementation, a simple stripes grid (10 lines/mm) used in Moire contour detection was used.

Focus Operator

The goal of this operator is determining the defocus function (σ) by inverse filtering near and far-focused images. Our efforts in this article were concentrated in evaluating the efficiency of different focus operators. Because the blur circle is only uniform for small regions, the kernel of the focus operator has to be small to preserve locality, but on the other hand, the windowing introduces supplementary errors. Xiong and Shafer [11] proposed a solution to select the window size for Gabor filters. They used a simple criterion when the window size is selected to be as small as possible, while the error caused by noise and windowing is smaller than a pre-set value. Aside from window size, every focus operator must be rotationally symmetric and must not respond to any DC component (a DC component can mean a change in image brightness). This condition is satisfied if the sum of all elements of the focus operator is equal to zero.

Watanabe and Nayar [8] suggested an approach based on the use of rational filters. They proposed a method to compute a set of broadband rational operators. The first operator performs prefiltering (for removing DC components) and then another three operators are involved in depth estimation. Finally, the depth errors caused by spurious frequencies are minimized by applying a smoothing operator.

This article investigates the performance of Laplacian (4 and 8 neighborhood) and rational operators (3 × 3 and 7 × 7 kernels). The 3 × 3 operators are shown in *Figure 2* and followed by the 7 × 7 operator in *Figure 3*.

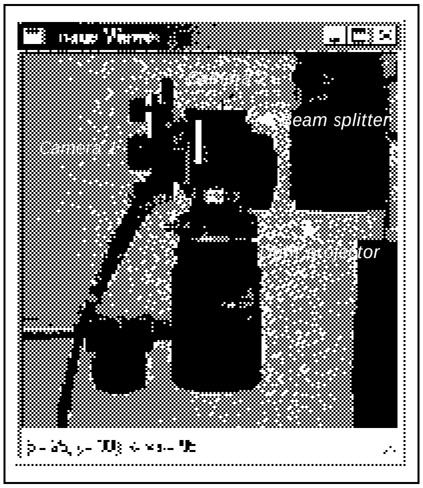
Because the image is discrete, the focus operator will introduce errors (apart from those caused by windowing). Furthermore, supplementary errors are caused by misalignment between the cells of the CCD sensor and illumination pattern. To minimize the abovementioned problems, a post-filtering operator is used after the focus operator is applied to near and far-focused images.

Physical Implementation

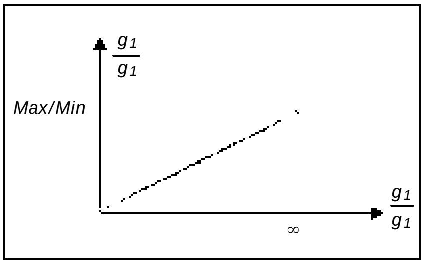
The main goal of this implementation is to build a real-time depth estimator. Thus, the near and far-focused images have to be acquired in the same time. For this purpose, two OFG VISIONplus-AT framegrabbers were used. The scene is imaged using an AF MICRO NIKKOR 60 mm F 2.8 (Nikon). A 22 mm beam splitter cube is placed between the NIKKOR lens and the sensing equipment (CCD sensors). Then, the near and far-focused images are acquired using two low-cost 256 × 256 CCD sensors, VVL 1011C, from VLSI Vision Ltd. (Nashua, NH). These sensors are precisely placed to ensure that one will acquire the near or far-focused images. The physical displacement between these sensors is approximately 0.8 mm.

The structured light is projected onto the scene using an MP-1000 Moire projector with MGP-10 Moire gratings (stripes grid with density of 10 lines/mm). The lens attached to the projector is the same type used to image the scene. All sensing equipment required by this implementation is low cost, and furthermore, the calibration procedure is relatively simple. The set up involved in this present implementation is described in *Figure 4*.

When the images are acquired, a few operations are necessary to determine the scene's depth map. For the sake of computation efficiency, the depth is estimated directly from g_1 and g_2 using a precomputed look-up table. This function is not bounded, but this is not a major drawback. A simple solution of avoiding the case when g_2 is equal with zero is to add a small constant value to g_1 and g_2. As we mentioned before, this function can be evaluated using the ratio (g_1-g_2)



4. 3D sensor and its principal components.



5. Defocus function.

/ (g_1+g_2). The defocus function (*Figure 5*) is bounded in this case, but for this implementation, while the depth is investigated within a small range (0-9 cm), it was proven not to be sensitive enough. The flowchart illustrated in *Figure 6* describes the main operations. The implementation presented in the figure computes the depth map (256 × 256) in approximately 95 ms on a Pentium 133 MHz (the time required by graphical interface is not included).

Camera Calibration

A calibration procedure is proposed that contains two important stages. The first stage is obtaining a precise alignment between the near and far-focused CCD sensors, while the second stage carries out a pixel-by-pixel gain calibration. To obtain a precise spatial alignment between the CCD sensors, we propose a gray-level rectangular grid pattern as a calibration pattern. This pattern is illustrated in *Figure 7*. The pixel-by-pixel gain calibration is applied to compensate for the offset and errors caused by the optical and sensing equipment.

Experiments and Results

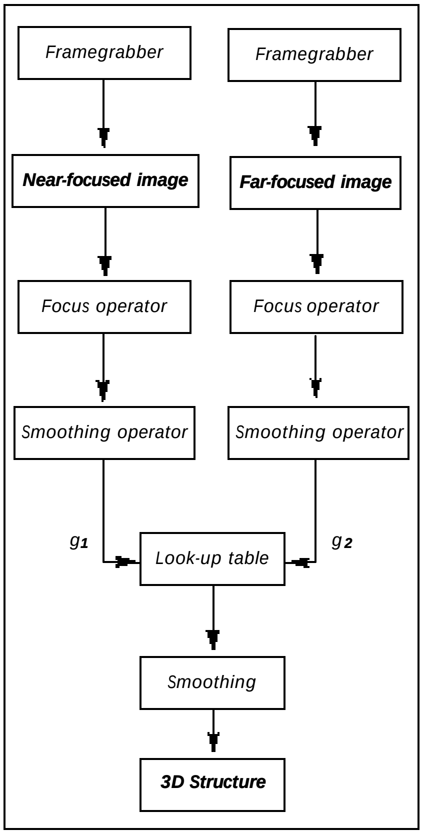
To verify the efficiency of this range sensor, it was tested on several indoor scenes. First, this sensor was tested on simple targets such as planar surfaces, then on scenes with a complex scenario. *Figure 8* shows the depth recovery for two planar objects situated at different distances in front of the sensor.

Figure 9 shows the depth map for a slanted planar object, and *Figure 10* shows a more complex scene containing LEGO® objects with different shapes and a large scale of colors.

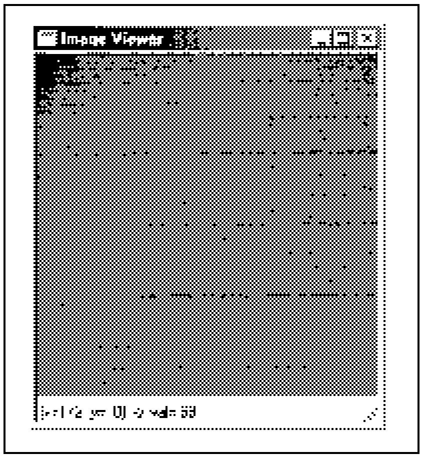
The accuracy is estimated when the sensor is placed at a distance of 86 cm from the baseline of the workspace. For these scenes, the lowest accuracy is 3.4% normalized in agreement with the distance from the sensor. This accuracy is reported for both textured and textureless nonspecular objects. We tried to identify an optimal kernel for the focus operator. As was mentioned earlier, four focus operators were used. The best results, with respect to the gain, were obtained for a 7 × 7 rational operator, but the depth estimation is not very linear. The results were more linear when the Laplacian (4) and the 3 × 3 rational were used as a focus operator, but the discontinuities in depth were not as well recovered. A trade-off between gain and linearity was given by Laplacian (8).

Conclusion

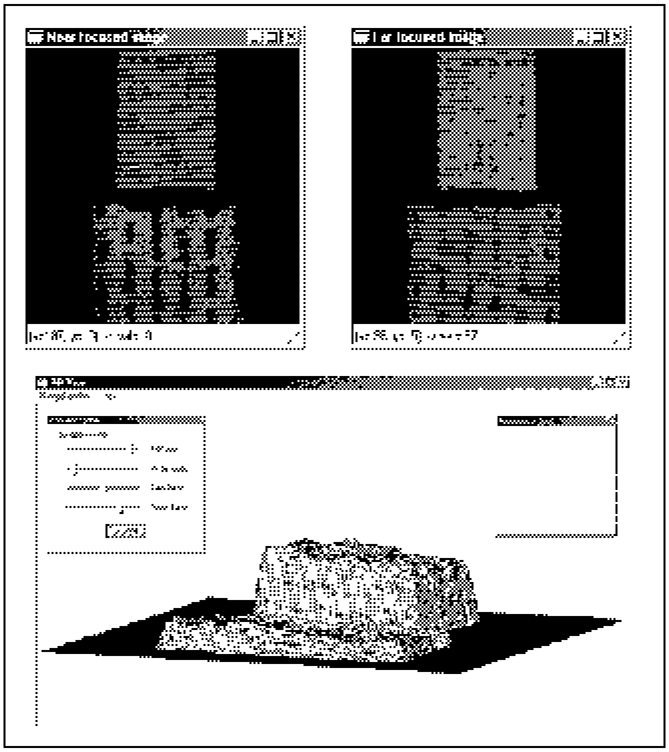
This article presented the implementation of a real-time depth sensor. In comparison to stereo technique, the DFD method does not suffer from the correspondence problem. Furthermore, the DFD approach is not affected by occlusion or missing parts; it can therefore be used as a ranging method for various applica-



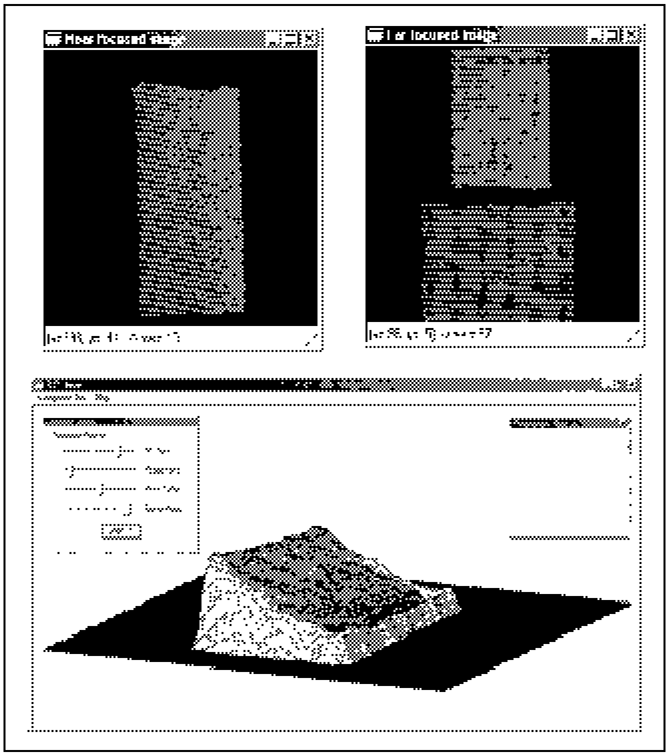
6. Data flow during the computation process.



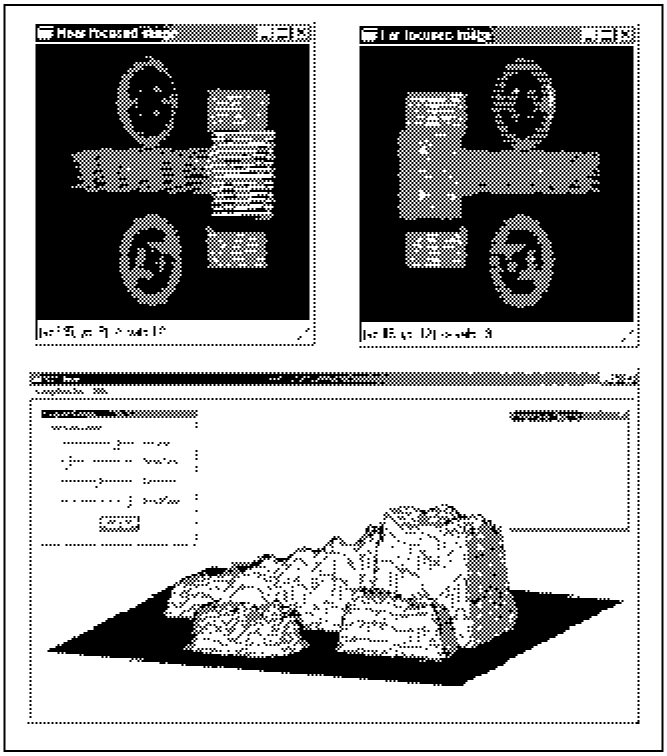
7. Calibration pattern.



8. Near and far-focused image and depth estimation for two planar objects situated at a different distance from sensor.



9. Near and far-focused images and depth recovery for a scene containing a slanted planar object.



10. Near and far-focused images and depth recovery for a scene containing various LEGO® objects.

tions. The consistency between theory and experimental results has indicated that our implementation is an attractive solution to estimate the depth quickly and accurately.

In contrast to other implementations based on defocusing where the depth range is relatively large, we proposed a solution to estimate depth within a small range (between 0-9 cm). Furthermore, this present approach has another advantage over other implementations suggested by Pentland et al. [5] and Nayar et al. [6] because it does not contain any sensitive equipment to movements or vibrations; therefore, it can easily be involved in robotics applications.

Because DFD methods perform poorly for textureless objects, the active illumination was identified as being the key issue for this implementation. The depth estimation can be further improved by using a camera with higher resolution and redesigning the illumination pattern and focus operator.

Acknowledgments

This research was supported by Motorola BV (Dublin, Ireland).

References

References for this article can be found at www.sme.org/mva.