Occlusion Analysis of Spatiotemporal Images for Surface Reconstruction

Takayuki YASUNO and Satoshi SUZUKI
NTT Human Interface Laboratories
Nippon Telegraph and Telephone Corporation
1-2356, Take, Yokosuka-shi, Kanagawa 238-03, Japan
e-mail: yasuno@nttcvg.ntt.jp

Abstract

This paper presents a method that extracts the surfaces of static objects that occlude other objects from a spatiotemporal image captured with straight-line camera motion. We propose the concept of occlusion types and show that the occlusion types are restricted to only eight patterns. Furthermore, we show occlusion type pairs contain information that confirms the existence of surfaces. Occlusion information gives strong cues for segmentation and representation. The method can estimate not only the 3D positions of edge points but also the surfaces bounded by the edge points. We show that combinations of occlusion types contain information that can confirm surface existence. The method is tested successfully on real images by reconstructing flat and curved surfaces.

1 Introduction

Computer vision techniques address the reconstruction of the 3D world to realize navigation and scene recognition. Many reconstruction techniques have been proposed. One early work was the interpretation of line drawings [1]. Line and vertex labeling methods have been proposed, but their targets are restricted and the results were ambiguous [2, 3]. Many stereo vision systems have been proposed, but because only two views are used, ambiguous interpretations are created [4, 5]. Because it uses multiple views, the spatiotemporal image analysis method [6, 7] is attractive. It can avoid the problems of pixel correspondence that plague stereo analysis. The spatiotemporal image is a sequence of successive images taken with a camera. Because the spatiotemporal image contains an abundance of information, reconstruction can be extraordinarily precise. If a camera is moved along a straight horizontal line, the trajectories of feature points appear as line segments on the horizontal slices (called the epipolar plane images) of the spatiotemporal image. The distances from the camera to the feature points can be determined from the inclination of the trajectories [8]. Therefore, epipolar plane image analysis can compute the three dimensional positions of object features easily. The existing spatiotemporal image analysis methods produce only three dimensional coordinates of feature points such as edges. Moreover, these methods yield only information about the feature points; the existence of surfaces is not treated. Therefore surface reconstruction requires an additional process.

One interesting approach is to use occlusion detection to reconstruct 3D surfaces. Occlusion detection has been studied to extract the qualitative information that indicates on which side of an edge a surface belongs [9, 10]. Occlusion detection by stereo analysis [10] and occlusion/disocclusion detection in velocity fields [9] have been studied. Geiger *et al.* showed the validity of discontinuities

BMVC 1993 doi:10.5244/C.7.55

for binocular stereo [11]. Little *et al.* detected direct evidence for occlusion in stereo and motion [12]. They indicated that labeled discontinuities would greatly simplify segmentation and recognition. However, no existing technique achieves complete three dimensional surface reconstruction. Multiple-base line stereo, for example, fails to distinguish occluding objects from occluded objects [13].

This paper extracts the surfaces of a static occluding object from a spatiotemporal image taken under straight-line camera motion. We propose the concept of occlusion types and show that the occlusion types are restricted to only eight patterns. Furthermore, we show occlusion type pairs contain information that confirms the existence of surfaces. Occlusion information gives strong cues for segmentation and representation. Our method can estimate not only the 3D positions of edge points, but also the surfaces between the edge points. Details of the proposed method are described in Sections 3. In Section 4, our method is tested successfully on synthesized images as well as real images.

2 3D feature points extraction from spatiotemporal images

In this section, we describe about usual spatiotemporal image analysis methods. In the spatiotemporal image taken by a camera moving along a straight horizontal line at constant velocity, the distance from the camera to an edge point is indicated by the slant of the edge point trajectory on the horizontal slice plane (called the epipolar-plane image or EPI) of the spatiotemporal image. Figure 1 (a) shows an example of a spatiotemporal image. Figure 1 (b) shows an EPI of the spatiotemporal image. The slant of the trajectory of an edge point close to the camera is less than that of a feature point far from the camera. The distance from the edge point to the camera can be determined from the slant of the edge point trajectory and the camera speed.

The global coordinates X-Y-Z are defined as shown in Figure 2. Here, the view point o moves on the X axis at a constant speed, and the optical axis is parallel to the Z axis. An object's image is projected onto the plane; $Z = F$, where F denotes the focal length of the camera. u-v denotes the projection plane. The spatiotemporal image is an accumulation of projection planes u-v captured from different view points. Point (x, y, z) in global coordinates is projected as $p_i(u, v)$ on the projection plane. Let u' be the velocity of $p_i(u, v)$, which corresponds to the slant of its trajectory on the EPI. The global coordinates of point (x, y, z) can be written as

$$(x, y, z) = \left(-\frac{ud'}{u'} + d, -\frac{vd'}{u'}, -\frac{Fd'}{u'}\right), \tag{1}$$

where d and d' are the view point's position and velocity, respectively. Equation (1) can be rewritten as follows:

$$(u, v) = \left(\frac{F}{z}(x - d), \frac{F}{z}y\right). \tag{2}$$

Equation (2) indicates that u is a linear function of d and v is constant for a fixed 3D point (x, y, z). That is, $p_i(u, v)$ draws a straight line on the EPI. Therefore, the three dimensional position of an object's feature points is calculated by line detection on the EPI using edge detection and the Hough transformation [14].

Figures 3 show detected edge point trajectories on the EPI . Figure 4 shows the reconstructed three dimensional edges of some leaves. As seen from this figure, ordinary spatiotemporal image analysis fails to reconstruct the surfaces.

3 Surface extraction using the occlusion types principle

In this section, we describe about the occlusion types principle and surface extraction algorithm using the principle.

3.1 Occlusion types

An edge point trajectory that hides another trajectory as shown in Figure 5, and is called an occlusion. Occlusions can be classified by the types of trajectory crossing points, which are called the occlusion types (abbreviated OT).

There are 16 probable patterns as shown in Figure 6. They represent the combinations of visible and invisible edge point trajectories at the cross point of any two (or three) edge point trajectories. In this figure, the camera moves from left to right and the time axis runs from top to bottom. The solid lines are visible edge point trajectories (tracks) and dotted lines are hidden edge point trajectories. The shaded regions are surfaces (the front occluding surface is dark gray and the rear occluding surface is light gray). We assume that each track (edge point) either bounds a surface or splits a surface. Pattern (1), in which all tracks are visible before and after the occlusion, is impossible, because no surface is attached to either track. If two tracks cross, the one closest to the camera must hide the other. From this property, Patterns (2), (9), and (10) are impossible. The heavily shaded areas in the figure show the surface(s) of the third track which crosses the others at the same point. Patterns (12) and (14) are impossible, because pattern (12) contains impossible pattern (9) or (10). Pattern (14) contains impossible pattern (2) or (10). Furthermore, pattern (6) and (11) are impossible because the third track occludes a track that is closer to the camera. Thus the occlusion types are restricted to the eight named patterns shown in Figure 6.

If more than two tracks cross, we can extract all occlusion types for all the combinations of track pairs. The occlusion types along a track are accumulated to recover surface information.

Pattern (3) is called Type-T. Pattern (4) is called Type-Λ. In this pattern, both tracks have been occluded by the third track. Pattern (5) is called Type-y. Pattern (7) is called Type-/. In this pattern, one track has already been occluded by another and has not yet reappeared. Pattern (8) is called Type-down. Pattern (13) is called Type-v. Pattern (15) is called Type-up. Patterns (8), (13) and (15) are variants of patterns (3), (5) and (7), and show the occlusion of the third track. Pattern (16) is called Type-hidden.

3.2 Occlusion type pairs

A surface appears as an EPI belt bordered by at least one track. Surface occlusion can be detected by examining the OT pairs of the neighboring cross points along each track as shown in Figure 7. Typical OT combinations include the combinations y-T, /-y, /-T, v-Λ, etc., as shown in the figures. Note that the OT order is important, the y-T combination means that the y occlusion occurs earlier than the

T occlusion. Table 1 shows that only 40 of the 64 pairs are feasible. The feasible pairs are denoted by 'O' while impossible pairs are identified by 'X'. Feasibility means that the OT combination indicates a logically feasible occlusion.

Among all possible occlusion pairs, only 16 combinations guarantee or suggest surface existence. For example, if the two neighboring OTs on a track e_0 form the y-T combination, it guarantees that a surface exists between the two edge points forming the tracks e_1 and e_2 as shown in Figure 7 (a). This is because edge points e_1 and e_2 are closer to the camera than edge point e_0 and the surface between edges points e_1 and e_2 occludes the surface that includes edge point e_0. Similarly, we can see that the remaining 15 pairs guarantee or suggest surface existence as shown in Table 1.

In this table, "Guarantee:n-m" means that the OT combination guarantees a surface exists between edge points n and m. "Suggests:n-m" means that the OT combination suggests a surface exists between edge points n and m. This suggestion can be confirmed by extracting the string of OT combinations along the occluding edge point trajectory. In Figure 7 (b) the string is y-/,/-T. If a valid OT combination is formed by the first and last OT, surface existence is guaranteed. The OT combination from this string is y-T which guarantees surface existence.

3.3 Surface reconstruction algorithm

Our algorithm recovers the surfaces of a 3D object as follows:

1. Edge point trajectories are detected from each EPI using edge detection and the Hough transformation. Furthermore, 3D coordinates of edge points are recovered.

2. Let e_{ij} be the j th crossing point on trajectory l_i, and let $E(j)$ be the number of edge points on the line segment between e_{ij} and e_{ij+1}. If the rate of $E(j)$ to the length of the line segment between e_{ij} and e_{ij+1} is smaller than threshold T, the line segment is regarded as being hidden. Otherwise, this line-segment is regarded as being visible. Next, for the occlusion type of each crossing point of each trajectory is determined as described in Section 3.1. Labeled OTs on an EPI are shown in Figure 8.

3. Horizontal surfaces between edges are estimated from OT combinations. If we detect an OT combination that guarantees surface existence between two edge points as described in Section 3.2, the 3D points E_i and E_j corresponding to the edges are connected by a 3D horizontal line segment (labeled S_{ij}) as shown in Figure 9 .

4. Vertical surfaces between edges are estimated from the OTs. If two edge points on successive EPIs have the same occlusion type and they are close, the 3D points E_i and E_k corresponding to these edges are connected by a 3D vertical line segment (labeled S_{ik}) .

4 Experiments

In this section, our method is tested successfully on synthesized images as well as real images, and the threshold is decided with the occlusion type principle.

4.1 Synthesized images

We synthesized an image to confirm the validity of this method. The image consisted of tall rectangular plates standing in front of a vertical lattice wall as shown in Figure 10 (a). We made an image sequence of this scene assuming a horizontally moving camera. Edges were detected from the synthesized spatiotemporal image. This spatiotemporal image was generated from 200 images, each of which was 255 pixels × 200 pixels in size. In this case, the EPI was generated as a horizontal cross section of the spatiotemporal image, because the camera motion was parallel to the image scanning lines. The three dimensional coordinates of edges were determined using Canny's edge detection method [15] and the Hough transformation. Figure 10 (b) shows the reconstructed surfaces. In this figure, the plates in front of the wall were reconstructed. However, the surfaces on the back wall were not reconstructed, because they did not occlude any other object.

4.2 Real images

Our method was tested using real images. We made the experimental system shown in Figure 11. Image sequences were taken by sliding the camera sideways and storing the captured images in a digital video tape recorder. We captured a sequence of images of a scene in which rectangular plates were placed in front of a lattice wall (the same arrangement as the synthesized image test), see Figure 12 (a). After 200 images were stored, they were transferred to a computer. Edges were detected by Canny's method. Next, all detected edges were calibrated with the camera parameters which had been measured beforehand [16]. After calibration, the tracks became line-segments which were then estimated by Hough transformation. Figure 12 (b) shows the reconstructed surfaces. As in the synthesized image test, all surfaces except the back wall were reconstructed.

Furthermore, we tested the method by applying it to a scene consisting of leaves in front of a book shelf, as shown in Figure 13 (a). The reconstructed leaves are shown in Figure 13 (b). Our method can represent curved surfaces by approximating them with many 3D line segments.

4.3 Parameter decision in occlusion analysis

We can decide the value of the threshold with the occlusion type principle. As described in Section 3.3, the threshold T determines which occlusion types are observed. When the threshold T is low, impossible OTs appear on EPIs. Let $OTrate$ be the ratio of possible OTs to the number of crossing points of edge trajectories on an EPI.

$$OTrate = \frac{OT_{possible}}{C}, \tag{3}$$

where $OT_{possible}$ is the total number of possible OTs and C is the total number of crossing points of edge trajectories on an EPI. The relation between $OTrate$ and threshold T is shown in Figure 14. In this graph, when T is 0, $OTrate$ is always 0 because all edge points are visible; an impossibility. As T grows from 0 to 1, $OTrate$ rapidly increase to reach an initial peak value. When T is 1, $OTrate$ is always 1 because all edge points are invisible; another impossibility. OTs are correctly determined at the initial peak $OTrate$ value. For OT decision, therefore, we can set T to the value that yields the first peak value of $OTrate$.

5 Conclusion

This paper has proposed a new technique for reconstructing the surfaces of static objects that occlude other objects from a spatiotemporal image captured with straight-line camera motion. We proposed the concept of occlusion types and showed that the occlusion types are restricted to only eight patterns. Furthermore, we showed occlusion type pairs contain information that confirms the existence of surfaces. Occlusion information gives strong cues for segmentation and representation. Our method can estimate not only the 3D positions of edge points, but also the surfaces between the edge points. The more occlusion there is, the more robust against noise our method becomes. This is because occlusion is a guarantee of surface existence. Our method was tested successfully with synthesized data as well as real data. The tests showed that our method can reconstruct flat and curved surfaces. The threshold parameter for occlusion type detection is decided using the occlusion types principle. Occlusion does not always occur when we need it. However, by projecting vertical strips onto the object and its background from a fixed point, our method could reconstruct all occluding surfaces.

Acknowledgments

The authors would like to thank Drs. Takaya Endo, Sadami Kurihara and Kazunari Nakane of NTT Human Interface Laboratories for their encouragement and helpful advice.

References

[1] L. G. Roberts. Machine perception of three-dimensional solids. In J. T. Tippett et al., editor, *Optical and Electro-Optical Information Processing*, pp. 159–197. MIT Press, Cambridge, Massachusetts, 1965.

[2] D. A. Huffman. Impossible objects as nonsense sentences. In B. Meltzer ann D. M. Michie, editor, *Machine Intelligence 6*, pp. 295–323. Edinburgh Univ. Press, 1971.

[3] D. L. Waltz. Generationg semantic descriptions from drawings of scenes with shadows. In P. H. Winston, editor, *The Psychology of Computer Vision*, chapter 3. McGraw-Hill Book Co., New York, 1972.

[4] M. D. Levine, D. A. O'Handley, and G. M. Yagi. Computer determination of depth maps. *Computer Graphics and Image Processing*, Vol. 2, No. 2, pp. 131–150, 1973.

[5] D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, Vol. 194, No. 4262, pp. 283–287, 1976.

[6] R. C. Bolles, H. H Baker, and D. H. Marimont. Epipolar-plane image analysis : An approach to determining structure from motion. *International Journal of Computer Vision*, Vol. 1, No. 1, pp. 7–55, 1987.

[7] T. Yasuno and T. Hamano. Three-dimensional reconstruction using homocentric spherical spatiotemporal image analysis. *Journal of Visual Communication and Image Representation*, Vol. 2, No. 4, pp. 365–372, 1991.

[8] M. Yamamoto. Determining 3-D structure of scene from image sequences obtained by horizontal and vertical moving camera. *Lect Notes Comput Sci*, Vol. 301, pp. 458–467, 1988.

[9] Shou-Ling Peng and Ge'rard Medioni. Spatio-temporal analysis for velocity estimation of contours in an image sequence with occlusion. In *Proceedings of International Conference of Pattern Recognition*, pp. 236–241. IEEE Computer Society, 1988.

[10] Peng-Seng Toh and Andrew K. Forrest. Occlusion detection in early vision. In *Proceedings of International Conference of Computer Vision*, pp. 126–132. IEEE Computer Society, 1990.

Former / Latter	(3)Type-T	(5)Type-y	(7)Type-/	(4)Type-Λ	(8)Type-down	(15)Type-up	(13)Type-v	(16)Type-hidden
(3)Type-T	X	O Guarantee 1-4	O Suggests 1-4	X	O Suggests 1-4	O unknown	O Guarantee 2-4	O unknown
(5)Type-y	O No	X	X	O No	X	X	X	X
(7)Type-/	X	O Suggests 1-4	O Suggests 1-4	X	O Suggests 1-4	O unknown	O Suggests 2-4	O unknown
(4)Type-Λ	X	O Guarantee 1-5	O Suggests 1-4	X	O Suggests 1-5	O unknown	O Suggests 1-4	O unknown
(8)Type-down	X	O unknown	O unknown	X	O unknown	O unknown	O unknown	O unknown
(15)Type-up	X	O Suggests 1-4	O Suggests 1-4	X	O Suggests 1-4	O unknown	O Suggests 1-4	O unknown
(13)Type-v	O No	X	X	O No	X	X	X	X
(16)Type-hidden	X	O unknown	O unknown	X	O unknown	O unknown	O unknown	O unknown

Table 1: All occlusion type pairs (OTP). The feasible pairs are denoted by 'O' while impossible pairs are identified by 'X'. "Guarantee:n-m" means that the OTP guarantees a surface exists between edge points n and m. "Suggests:n-m" means that the OTP suggests a surface exists between edge points n and m. "No" means that there is no surface. "unknown" means that surface existence can not be decided.

[11] D. Geiger, B. Landondorf, and A. Yuille. Occlusion and binocular stereo. *Lecture Notes Computer Science*, Vol. 588, pp. 425–433, 1992.

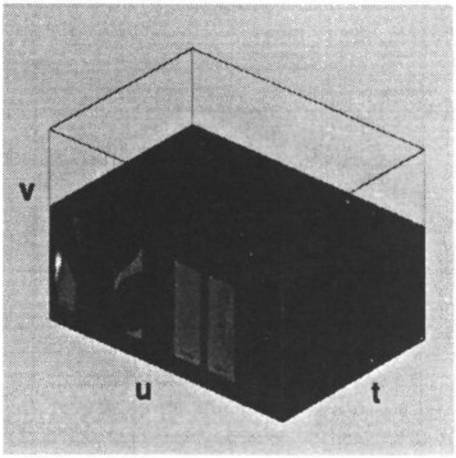
[12] James J Little and Walter E Gillett. Direct evidence for occlusion in stereo and motion. *Image and Vision Computing*, Vol. 8, No. 4, pp. 328–340, November 1990.

[13] T. Kanade, M. Okutomi, and T. Nakahara. A multiple-baseline stereo method. In *Proceedings of Image Understanding Workshop*, pp. 409–426, January 1992.

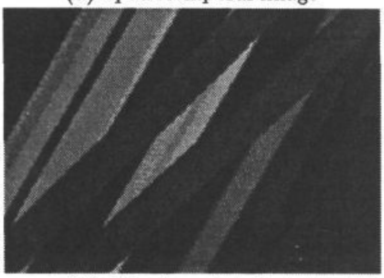
[14] P. V. C. Hough. Method and means for recognizing complex patterns. U. S. patent 3069654, 1962.

[15] John Canny. A computational approach to edge detection,. *IEEE Trans. PAMI*, Vol. 8, No. 6, pp. 679–698, Nov 1986.

[16] J. Weng, P. Cohen, and M. Herniou. Calibration of stereo cameras using a non-linear distortion model. In *Proceedings of Computer Vision and Pattern Recognition*, pp. 246–253. IEEE Computer Society, 1990.



(a) Spatiotemporal image



(b) Epipolar plane image

Figure 1: A spatiotemporal image. (The u and the v axes mean coordinates on the projection plane and the t axis means the time.) and an epipolar plane image.

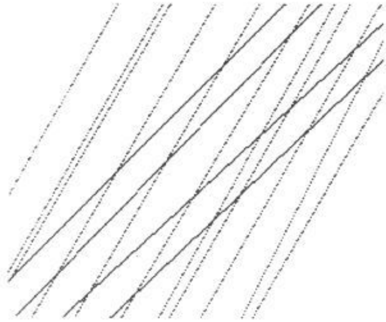


Figure 3: Edge point tracks extracted from edges by Hough transformation.

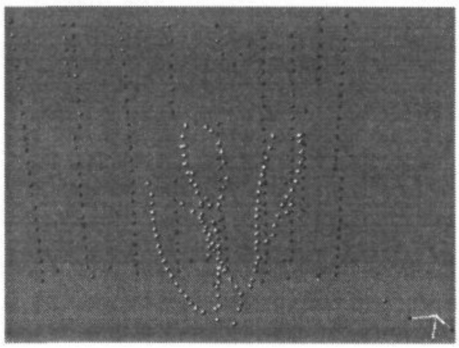


Figure 4: Reconstructed 3D edges.

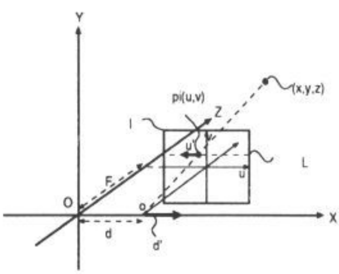


Figure 2: The global coordinates.

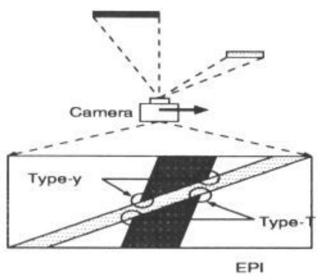


Figure 5: Occlusion types on EPIs.

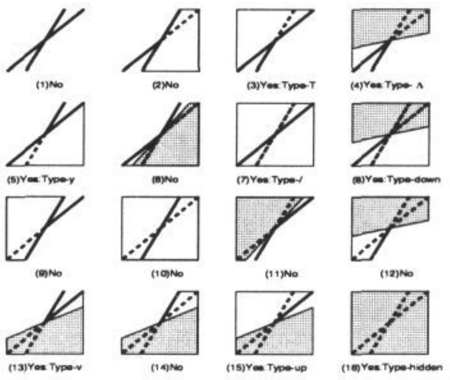


Figure 6: All occlusion types composed by two edge point trajectories.

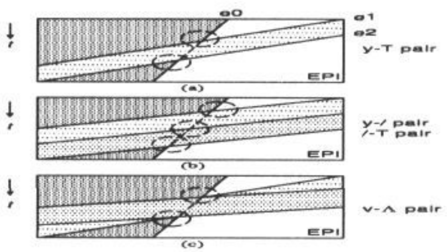


Figure 7: Neighboring occlusion type pairs.

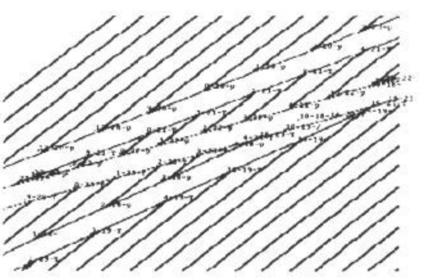


Figure 8: Labeled cross point of edge tracks. Each cross point of tracks is labeled such as "n-m-c": n and m mean line numbers and c means an occlusion type.

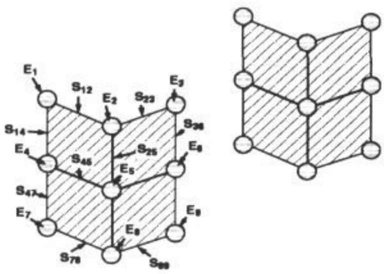
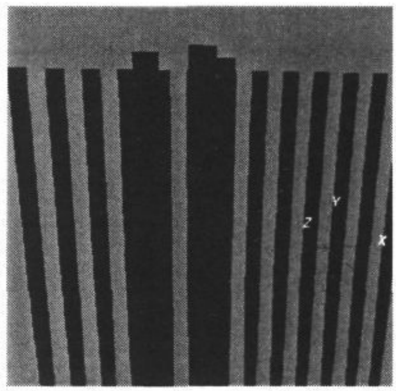
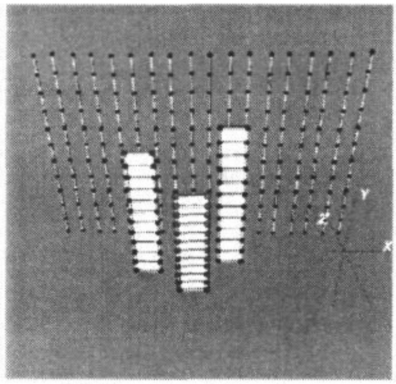


Figure 9: Connection of edges.



(a) A synthesized lattice scene



(b) Reconstructed surfaces

Figure 10: 3D surface reconstruction of a synthesized lattice scene.

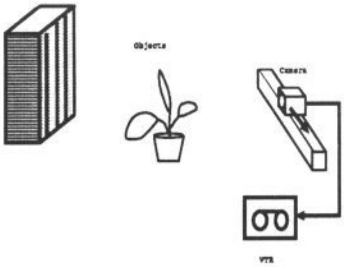
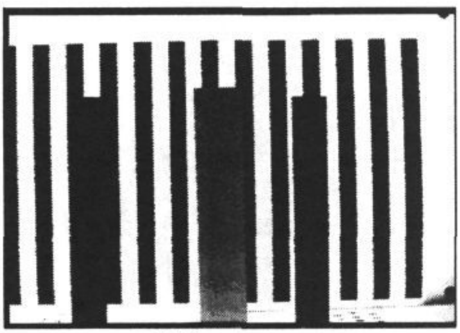
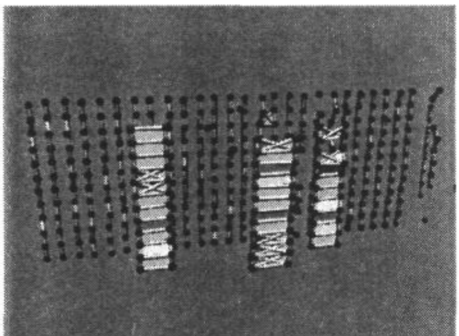


Figure 11: Our experimental system.



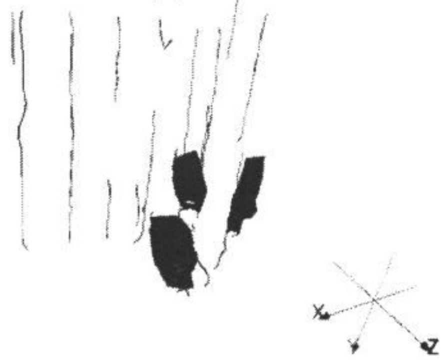
(a) Input scene



(b) Reconstructed 3D surface

Figure 12: 3D reconstruction of a real lattice scene.

(a) Input scene



(b) Recontructed leaves

Figure 13: 3D surface reconstruction of leaves in front of a book shelf.

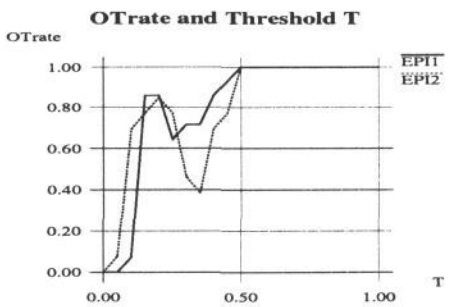


Figure 14: The relation between $OTrate$ and threshold T. EPIn means $OTrate$ of the nth EPI.