NISTIR 4669

An Integration-Based Method for Depth Estimation

Daniel Raviv
Florida Atlantic University
The Robotics Center and
The Electrical Engineering Department
Boca Raton, FL 33431
and
Sensory Intelligence Group
Robot Systems Division

*U.S. DEPARTMENT OF COMMERCE
National Institute of Standards
and Technology
Robot Systems Division
Bldg. 220 Rm. B124
Gaithersburg, MD 20899

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
John W. Lyons, Director

NIST

NISTIR 4669

An Integration-Based Method for Depth Estimation

Daniel Raviv
Florida Atlantic University
The Robotics Center and
The Electrical Engineering Department
Boca Raton, FL 33431
Sensory Intelligence Group and
Robot Systems Division

*U.S. DEPARTMENT OF COMMERCE
National Institute of Standards
and Technology
Robot Systems Division
Bldg. 220 Rm. B124
Gaithersburg, MD 20899

September 1991

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary

NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
John W. Lyons, Director

AN INTEGRATION-BASED METHOD FOR DEPTH ESTIMATION

Daniel Raviv

Robotics Center and Department of Electrical Engineering
Florida Atlantic University
Boca Raton, Florida 33431
and
Robot Systems Division
National Institute of Standards and Technology (NIST)
Gaithersburg, Maryland 20899

ABSTRACT

A closed-form, integration-based, and massively-parallel algorithm for determining depth of points in 3-D using one moving camera is presented. It is based on analyzing a sequence of images that result from a known rectilinear motion of a camera (with no rotation) in a stationary environment. A traceable point in an image sequence is reconstructed using an integration operation (no differentiation operator is involved).

The method arose from two simple observations:

(1) Stationary points in the 3-D scene appear to move away from the focus of expansion (FOE).

(2) The distance of a point in 3-D space from the camera motion-axis is the same at all instants of time.

Any visible moving point in the image can be processed independently of, and concurrently with, any other point. Laboratory results for the case where the optical axis is parallel to the motion axis show an error of less than 0.6% in absolute distance.

1. INTRODUCTION

This paper presents a new, robust, integration-based, and massively parallel algorithm for determining depth of points in 3-D using one moving camera. It is based on analyzing a sequence of images that result from a known rectilinear motion of a camera (with no rotation). A traceable point in the image sequence is reconstructed using an integration operation.

The method arose from two simple observations:

(1) Stationary points in the 3-D scene appear to move away from the focus of expansion (FOE) (or toward the focus of contraction).

(2) The distance of a point in 3-D space from the camera motion axis is the same at all instants of time.

Observation (1) provides the concurrent processing property since each image radial line can be processed independently. Using the Θ-Φ rather than the X-Y image-plane coordinate system, the algorithm becomes very simple (constant ϕ corresponds to a radial line that emerges from the FOE; constant θ corresponds to a circle in the image plane whose center is the FOE). Tracking a point in the image sequence is directional (i.e., along constant ϕ line) and thus computationally inexpensive. In fact, each point on a constant ϕ line can be processed on a separate processor. Assuming that θ and $d\theta/dt$ of a point in the image plane are known as well as the speed of the camera, then the location of the corresponding point in space can be explicitly calculated. In the proposed method $d\theta/dt$ is <u>not</u> measured, but indirectly calculated using an <u>integration</u> operation. One of the major advantages of this technique is that the integration operator "smoothes out" errors caused by unfocused camera, camera's noise, (x-y) to (θ-ϕ) image conversion error, inexact edge location, non-ideal motion of the camera, etc.). Also, the method is more appropriate for structured environments.

Using this method the distance of a traceable point from the axis of motion of the camera can be obtained. Given this distance and the location of the point in the image plane, the location of the point in 3-D in camera coordinates can be explicitly found.

In a set of experiments we obtained the distance of a point from the camera's axis-of-motion (which in our case, is also the optical axis of the camera). An accuracy of 99.4% has been achieved from a sequence of 40 images. We anticipate significantly smaller error when the camera's optical axis is perpendiculr to the motion axis.

Related work deals with depth estimation from two consecutive images, [3-6] from a larger sequence of images [1,2,7-9,12], using mainly optical flow [3-9,12,20,21], temporal and spatial brightness gradients [1,10,11,18], correlation-based methods [20] and epipolar analysis [2].

2. THE $\Theta - \phi$ DOMAIN

During a rectilinear motion (with no rotation), points in the image plane move away from the FOE (Figure 1). Based on this observation we use for our method an angular Θ-ϕ (rather than X-Y) image plane. Figures 2 and 3 show the chosen coordinate system and the definition of the angles θ and ϕ : Constant ϕ corresponds to a radial line that emerges from the FOE, and constant θ corresponds to a circle whose center is the FOE. Clearly, a point (x,y) in the (X-Y) plane can be transformed to a (θ-ϕ) point in the (Θ - Φ) plane and vice versa.

We assume that the camera moves in a known rectilinear motion. Using the Θ - Φ coordinate system, any point in the image plane moves along a constant ϕ line. This fact provides concurrent processing capability (each constant ϕ line can be processed separately and independently of any other ϕ line). Also each point on a constant ϕ line can be processed independently of other points that lie on the same line. As will be shown later, the expression for the distance of a point in space from the camera pinhole is independent of ϕ. A constant ϕ line can be processed as a 1D image. Figure 4 shows an example of the time evolution of a 1-D image.

3. THE 3-D RECONSTRUCTION ALGORITHM

Using the Θ - Φ coordinate system we show a simple way to find the 3-D location of a point in space. Along a constant ϕ line, for small enough changes in time and continuous speed $V(t)$ of the camera, the following calculations hold for the point P (Figure 5). Assume a pinhole model of the camera, then:

$$\Delta\ell = V(t)\Delta t \sin\theta(t) \tag{1}$$

$$\Delta\ell = R(t) \tan\Delta\theta(t) \tag{2}$$

From (1) and (2) and for $\Delta t \to 0$ we get (note that $\tan\Delta\theta(t) \approx \Delta\theta(t)$ for $\Delta\theta(t) \to 0$):

$$\frac{V(t)}{R(t)} = \frac{\frac{d\theta(t)}{dt}}{\sin\theta(t)} \tag{3}$$

With a known $V(t)$ and measured $\theta(t)$ and $d\theta(t)/dt$, $R(t)$ can be calculated from Equation (3). Now let us consider Figure 6. During rectilinear motion the distance d is constant at all times. By substituting

$$R(t) = \frac{d}{\sin\theta(t)} \tag{4}$$

in (3) we get

$$\frac{V(t)}{d} = \frac{d\theta(t)/dt}{\sin^2\theta(t)} \tag{5}$$

or

$$\frac{d\theta(t)}{dt} = \frac{V(t)}{d} \sin^2\theta(t) \tag{6}$$

Integrating $d\theta(t)/dt$ with respect to time yields:

$$\theta(t_2) - \theta(t_1) = \frac{1}{d}\int_{t_1}^{t_2} V(t)\sin^2\theta(t)\,dt \tag{7}$$

and thus

$$d = \frac{\int_{t_1}^{t_2} V(t)\sin^2\theta(t)\,dt}{\theta(t_2) - \theta(t_1)} \tag{8}$$

For a moving camera the expression for d in equation (8) is based on integration and is the same for all $t_1, t_2 (t_1 \neq t_2)$. Given d, θ, and ϕ, the 3-D location of a point can be explicitly calculated. By combining Equations (6) and (8) an integration-based expression for $d\theta(t)/dt$ is obtained:

$$\frac{d\theta(t)}{dt} = V(t)\sin^2\theta(t)\,\frac{\theta(t_2)-\theta(t_1)}{\int_{t_1}^{t_2} V(t)\sin^2\theta(t)\,dt} \tag{9}$$

Note that the ratio

$$\frac{\theta(t_2)-\theta(t_1)}{\int_{t_1}^{t_2} V(t)\sin^2\theta(t)\,dt} \tag{10}$$

in Equations (8) and (9) can be computed independently of the time instant for which the value of $d\theta(t)/dt$ is desired.

For the special case where the speed of the camera is constant, i.e., $V(t)=V$

$$d = V\,\frac{\int_{t_1}^{t_2} \sin^2\theta(t)\,dt}{\theta(t_2)-\theta(t_1)} \tag{11}$$

and

$$\frac{d\theta\,(t)}{dt} = \sin^2\theta(t)\,\frac{\theta(t_2)-\theta(t_1)}{\int_{t_1}^{t_2} \sin^2\theta(t)\,dt} \tag{12}$$

The latter result is independent of V.

The relations (8) and (9) can also be expressed as time independent expressions. By substituting $dx = V(t)\,dt$ in equations (8) and (9) we get:

$$d = \frac{\int_{x_1}^{x_2} \sin^2\theta(x)\,dx}{\theta(x_2)-\theta(x_1)} \tag{13}$$

and

$$\frac{d\theta(x)}{dx} = \frac{\sin^2\theta(x)}{d} \tag{14}$$

where x is the location of the pinhole point along the path of motion, and $\theta(x)$ is the angle θ of the point p at location x.

4. EXPERIMENTAL RESULTS

A set of experiments have been conducted to test the proposed method. First we describe the environment, then explain the experiments and finally show the results.

4.1. Set Up (Refer to Figure 7)

A CCD video camera is attached to an IBM 7565 gantry robot which has six degrees of freedom. The optical axis of the camera and its direction of motion coincide. The camera, with a field of view of about $\pm 12°$ was manually focused on the object at its initial position only. It moves in such a way that its distance from the object varies from 1125.2 mm to 299.7 mm. Dark objects placed on a white background are used in this experiment for better contrast. The images are processed on a PC-based vision system and the relevant image parameters (in our case, the edges) are extracted to subpixel accuracy. Since the algorithm is parallel and each ϕ line can be analyzed independently we chose for this experiment $\phi=0$ and $\phi=\Pi$ lines only (Figure 3). Similar processing could be applied to other values of the angle ϕ as well.

4.2. Finding Edges in an Image

Due to image digitization, focusing problems, etc. edges in the image are not sharp. For example, a 1-D image may look like Fig. 8a.

For the pixels near an estimated edge we used polynomial approximation and a brightness threshold to estimate the edge location. Refer to Fig. 8b.

4.3. Moving Along the Optical Axis

In order to assure that the axis of motion coincide with the optical axis, the robot was moved toward and away from a circular shaped dark object along the X axis (Figure 7). At each position we detected edges of a dark circle relative to the center of the image. The other five degrees of freedom of the robot were used to change the position and orientation of the camera such that, finally, at all positions of the camera along the motion axis, the circle appeared in the middle of the image.

4.4. Pixel Location Conversion

For the algorithm to work we converted $(x\text{-}y)$ to $(\theta\text{-}\phi)$ at each pixel. As mentioned earlier, we used $\phi=0$ and $\phi=\pi$ radial lines. The conversion resulted in a look-up table which is pictorially shown in Figure 9. d_{pixel} is the pixel number from the central pixel.

4.5. Measurement of Distances

We showed our system a new object and computed distances from the axis of motion to a point on the object. In this experiment the optical axis is parallel to the direction of motion. Forty images were taken every 0.635 mm and processed. The initial range from the focal point to the object was 972.8 mm. We used (13) to estimated the distance "d". The following table summarizes the results. For integration purposes we assumed first order polynomials between two consecutive points, i.e. $\theta(x)$ between two consecutive measurements of θ estimated to be on a straight line connecting the points $(X_i, .\theta(x_i))$ and $(X_{i+1},\theta(X_{i+1}))$.

In Table 1 we show the measurements errors. As mentioned earlier, a total of 40 measurements were taken when grouping the measurement into 36 sets of 5 images each, that is images 1,2,3,4,5

images 2,3,4,5,6

.

.

.

and images 36,37,38,39,40,

and computing the distance "d" from (13) for each set, there were different results for "d". When averaging the results obtained from the 36 sets, the average error was 2.08%. The average of absolute error of "d" was 12.82%. One of the 36 sets resulted in 0.13% error.

Similar computations were done for 10,20,30,38, and 40 images in each set. Note that for a set of 40 images the error was 0.56%.

TABLE 1: RESULTS

# of images	Camera location increment in mm	#of images' sets	Average error of d in %	Average of absolute error of d in %	Minimum absolute error of d in %
5	0.635	36	2.08	12.82	0.13
10	0.635	31	0.71	8.77	0.62
20	0.635	21	3.38	4.97	0.01
30	0.635	11	2.67	2.67	0.18
38	0.635	3	0.32	0.60	0.42
40	0.635	1	0.56	0.56	0.56

5. CONCLUSIONS

An algorithm for depth estimation of a traceable point has been presented. In Θ-Φ coordinates all traceable points in the image can be concurrently reconstructed. Due to the integration process the points whose 3-

D locations are desired do not have to be reliably traced. Errors caused by an unfocused camera, camera noise, (X-Y) to (θ-ϕ) conversion, non-ideal motion of the camera, etc. are "smoothed" by the reconstruction algorithm. The method may also work for object features such as centroids of 2-D objects. Centroids can be more reliably traced than visible feature points. The location of the FOE is assumed to be known. In many practical cases this is not the case. However, the FOE can be obtained using methods as described by Jain [16], Nagahdaripour et. al. [17], Vitoria Lobo at el. [19], or by using an inertial navigation system. An average absolute error in distance measurements of 0.56% for 40 images has been obtained in the case of a camera optical axis which is parallel to the motion direction. Better results may be obtained with an improved calibration process, a focused camera, higher camera resolution, better robot, wider range of θ, largers θ's (in particular θ near 90°), more sampled images, improved edge detection methods and better numerical integration methods.

6. ACKNOWLEDGMENTS

The author would like to thank H. Yakali for his help in performing the experiments and M. Herman for very useful discussions and comments..

REFERENCES

[1] J. Heel, "Direct Estimation of Structure and Motion from Multiple Frames," AI Memo 1190, MIT Artificial Intelligence Laboratory, March 1990.

[2] H. H. Baker and R. C. Bolles, "Generalizing Epipolar-plane Image Analysis on the Spatiotemporal Surface," International Journal of Computer Vision, 3, pp. 33.-49, 1989.

[3] G. Adiv, "Determining Three-Dimensional Motion and Structure from Optical Flow Generated by Several Moving Objects," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-7, No. 4, pp. 384-401, 1985.

[4] J. Weng, T. S. Huang and N. Ahuja, "3-D Motion Estimation, Understanding, and Prediction from Noisy Image Sequences," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-9, No. 3, pp. 370-389, 1987.

[5] R. Y. Tsai and T. S. Huang, "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surface," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-6, No. 1, pp. 13-26, 1984.

[6] J. Q. Fang and T. S. Huang, "Some Experiments on Estimating the 3-D Motion Parameters of a Rigid Body from Two Consecutive Image Frames,' IEEE Trans. Pattern Analysis Machine Intelligence, Vol. PAMI-6, pp. 545-554, 1984.

[7] T. J. Broida and R. Chellappa, "Estimation of Object Motion Parameters from Noisy Images," IEEE Trans. Pattern Analysis Machine Intelligence, Vol. PAMI-8(1), pp. 90-99, 1986.

[8] S. Bharwani, E. Riseman and A. Hanson, "Refinement of Environment Depth Maps over Multiple Frames," In Proceedings of the Workshop on Motion: Representation and Analysis, Charleston, S. C., pp. 73-80, 1986.

[9] L. Matthies, T. Kanade and R. Szeliski, "Kalman Filter-based Algorithms for Estimating Depth from Image Sequences," International Journal of Computer Vision, 3, pp. 209-236, 1989.

[10] B. K. P. Horn and E. J. Weldon, Jr., "Direct Methods for Recovering Motion," International Journal of Computer Vision, 2, pp. 51-76, 1988.

[11] S. Negahdaripour and B. K. P. Horn, "Direct Passive Navigation" IEEE Trans. on Pattern Analysis Machine Intelligence, Vol. PAMI-9(1), pp. 168-176, 1987.

[12] B. Shahraray and K. B. Michael, "Robust Depth Estimation from Optical Flow," Second International Conference on Computer Vision, pp. 641-650, 1988.

[13] A. M. Waxman and S. S. Sinha, "Dynamic Stereo: Passive Ranging to Moving Objects from Relative Image Flows," IEEE Trans. on Pattern Analysis Machine Intelligence, Vol. PAMI-8, No. 4, pp. 406-412, 1986.

[14] C. Tomasi and T. Kanade, Shape and Motion without Depth," Proceedings of Image Understanding Workshop, pp. 258-270, 1990.

[15] J. Aloimonos, "Active Vision," DARPA Image Understanding Workshop, pp. 552-573, 1987.

[16] R. Jain, "Direct Computation of the Focus Expansion," IEEE Trans. on Pattern Analysis Machine Intelligence, Vol. PAMI-5, No. 2, 1983.

[17] S. Negahdaripour and B. K. P. Horn, "A Direct Method for Locating the Focus of Expansion," Computer Vision, Graphics and Image Processing, 46, pp. 303-326, 1989.

[18] K. Skifstad and R. Jain, "Range Estimation from Intensity Gradient Analysis," Machine Vision and Applications, 2, pp. 81-102, 1989.

[19] N. Vitoria Lobo and J. K. Tsotsos, "Using Colinear Points to Compute Egomotion and Detect Nonrigidity", Proceedings of the IEEE Computer Vision and Pattern Recognition Conference'91, Hawaii, 1991.

[20] J. S. Albus and T. H. Hong, "Motion, Depth and Image Flow," Proceedings of the IEEE Conference on Robotics and Automation, 1990.

[21] R. Rangachar, T. Hong, M. Herman and J. Lupo, "Real Time Implementation of a Differential Range Finder," Proceedings SPIE Real-Time Image Processing II, Vol. 1295, 1990.

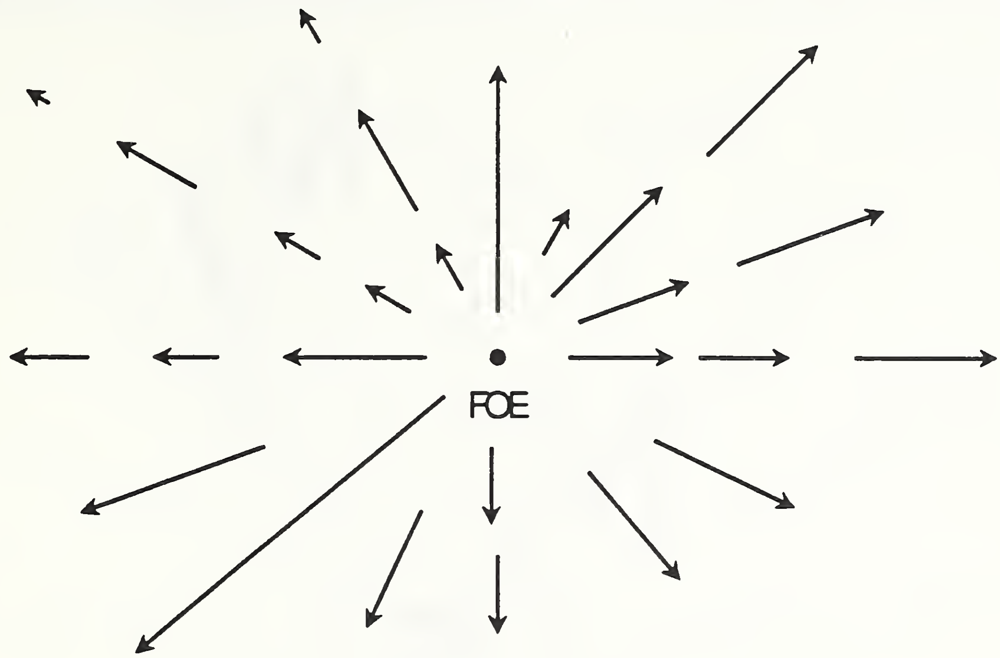

Figure 1. Optical Flow Relative to the Focus of Expansion (FOE).

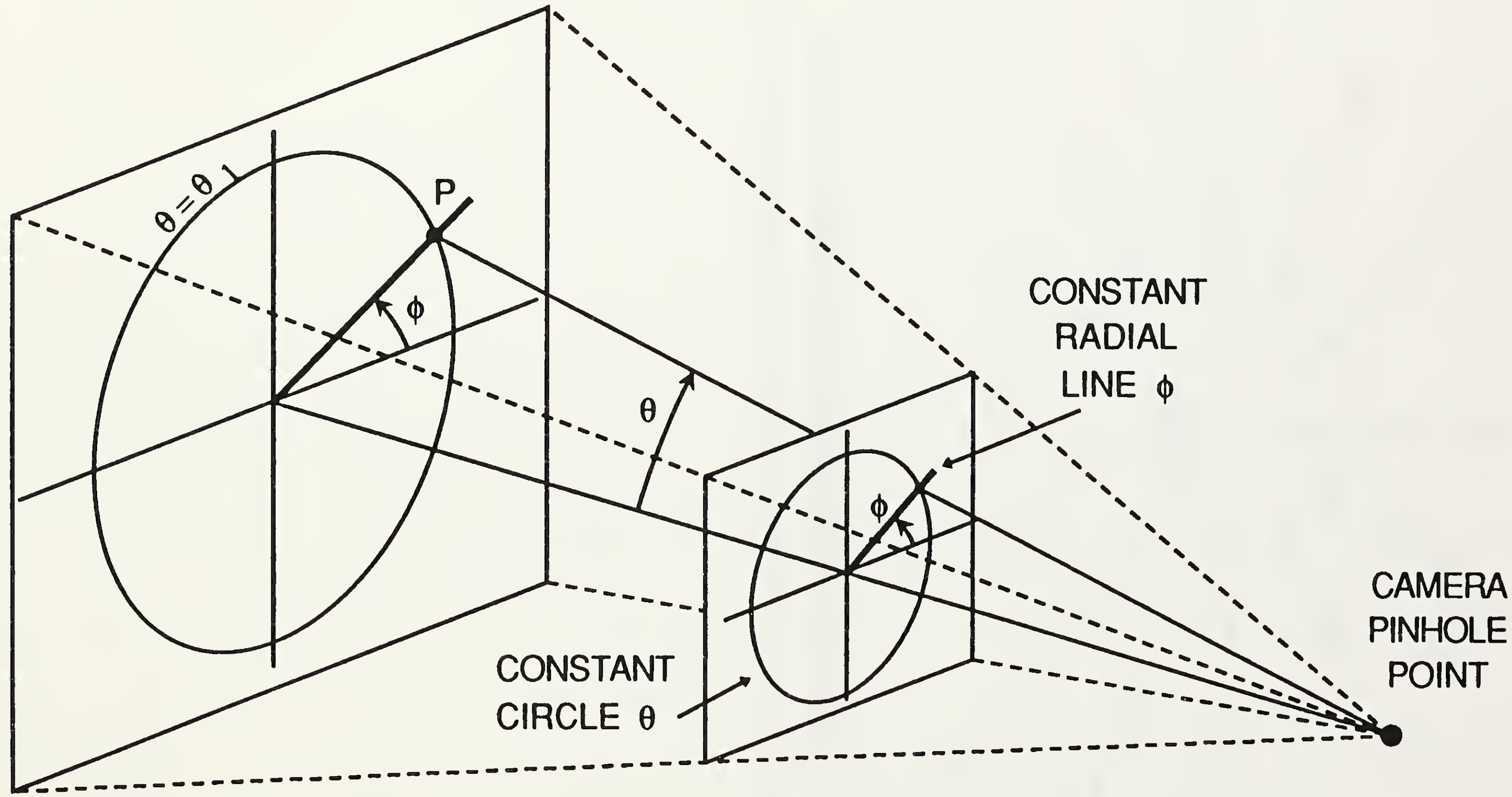

Figure 2. θ–ϕ Coordinate System

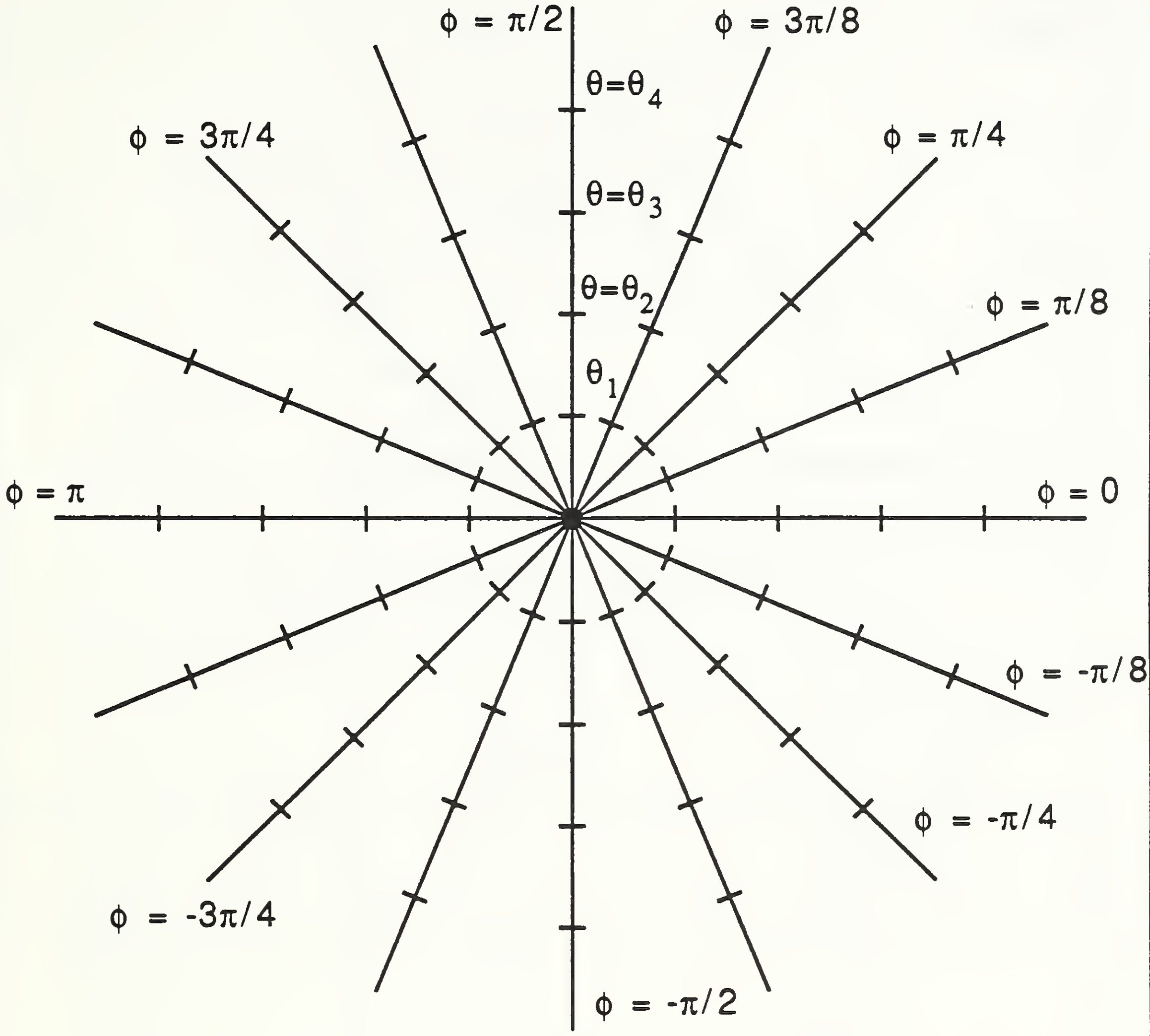

Figure 3. θ-φ Image Plane.

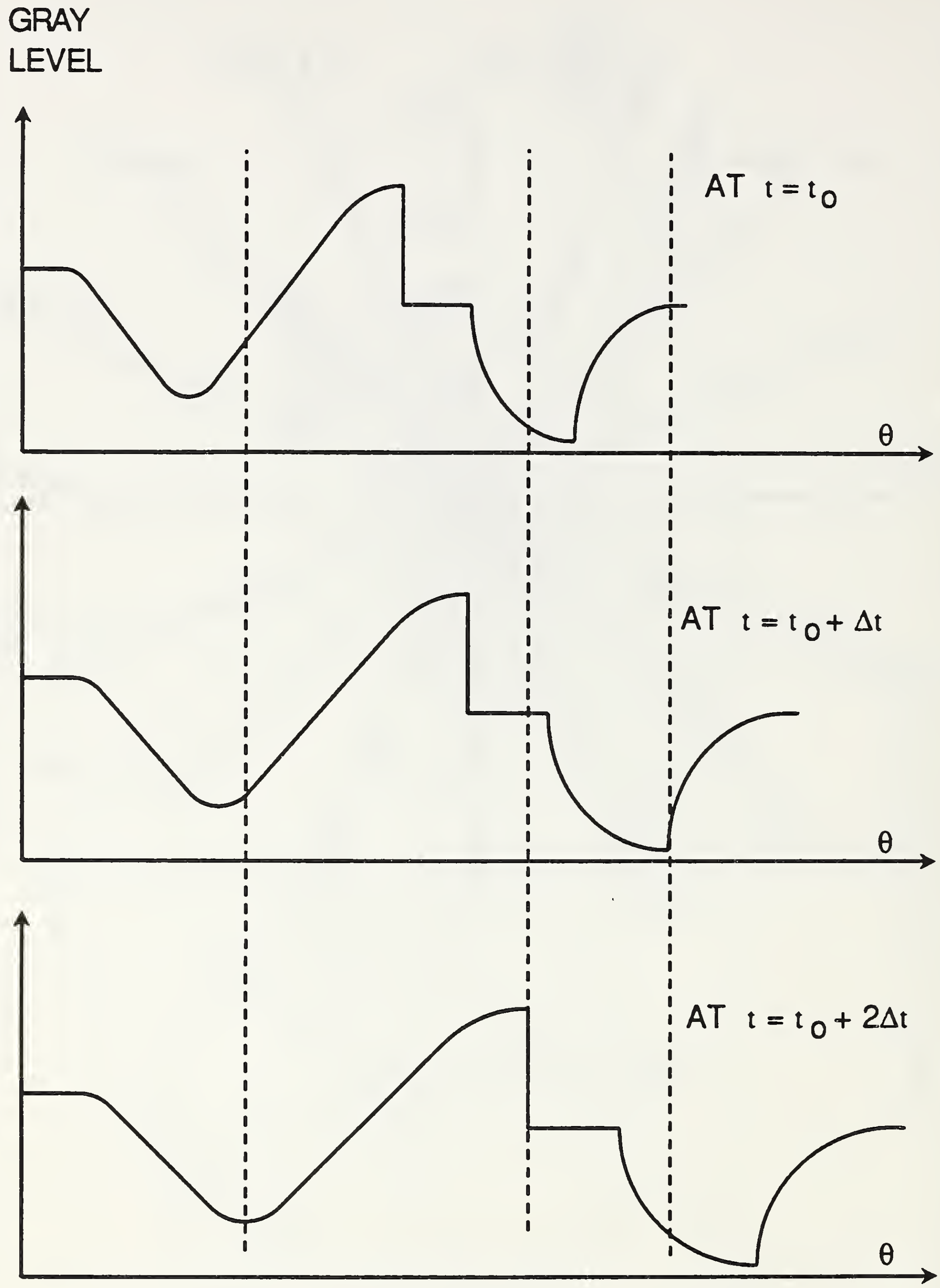

Figure 4. 1-D Image evolution.

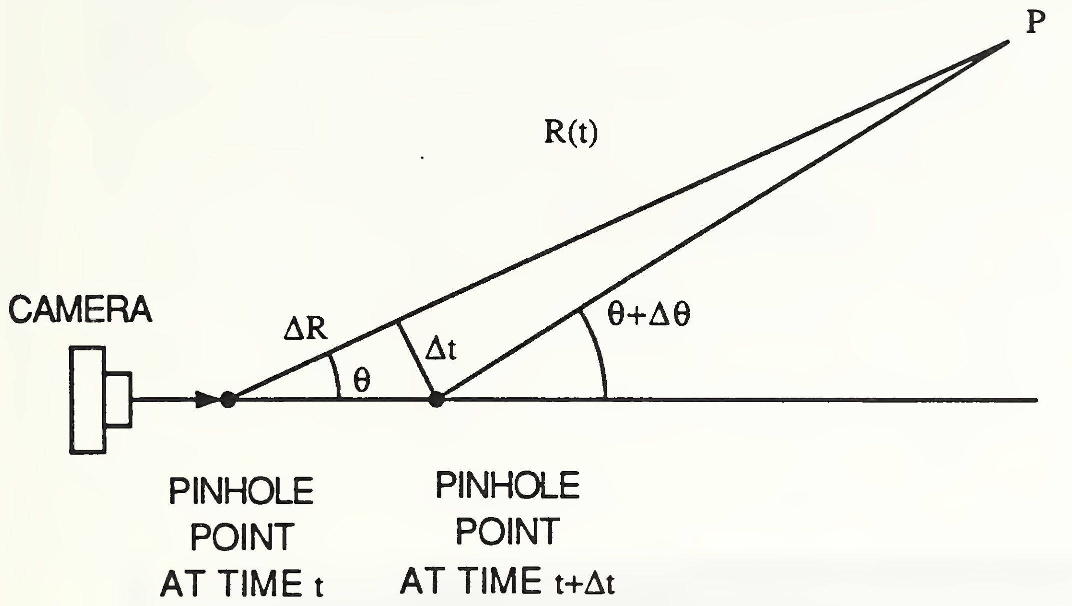

Figure 5. Geometry for constant ϕ line.

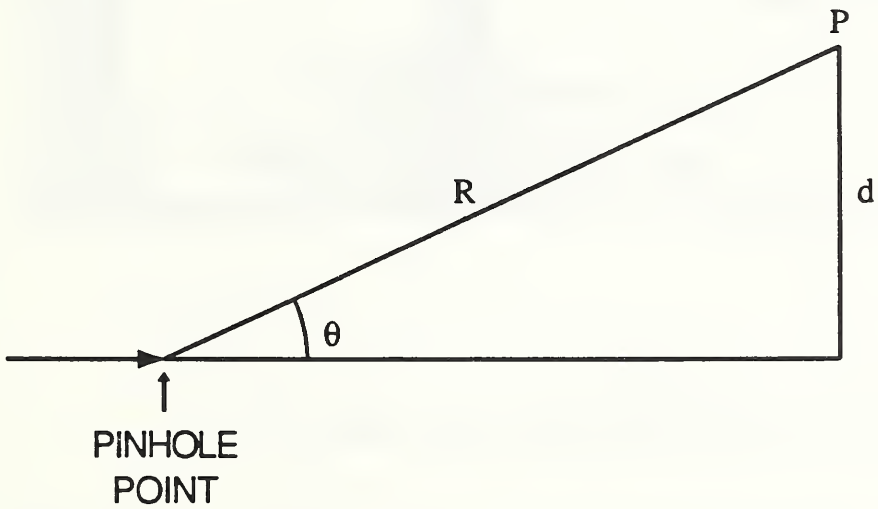

Figure 6. Basic Geometry.

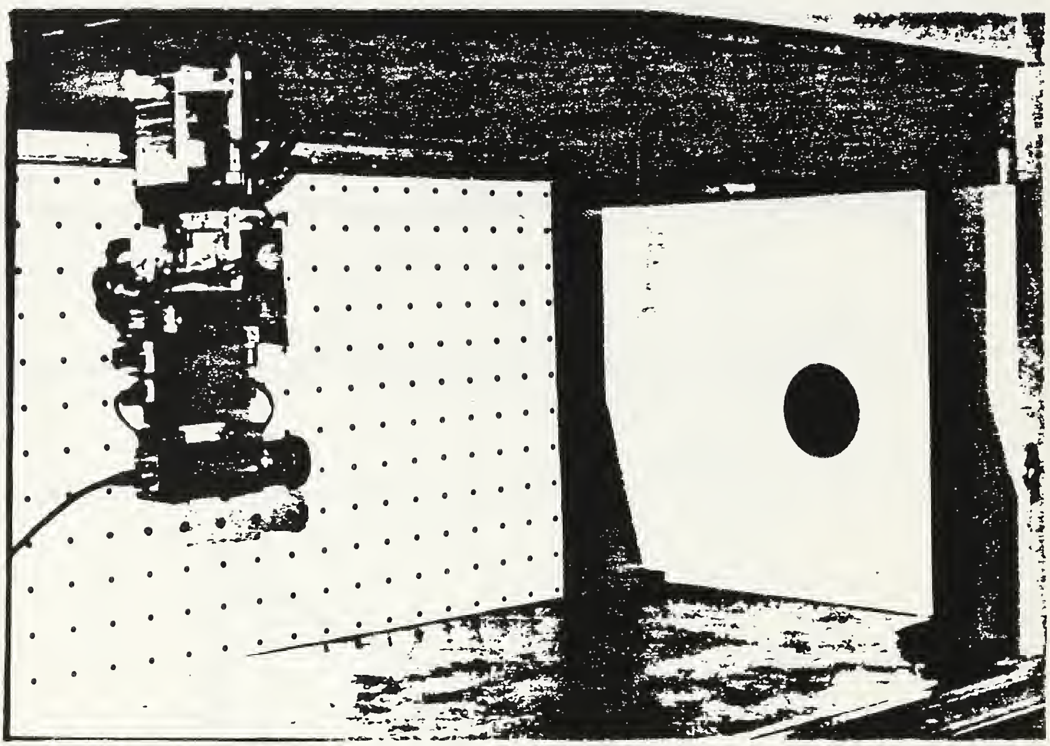

Figure 7. A Camera Mounted on the IBM 7565 Robot.

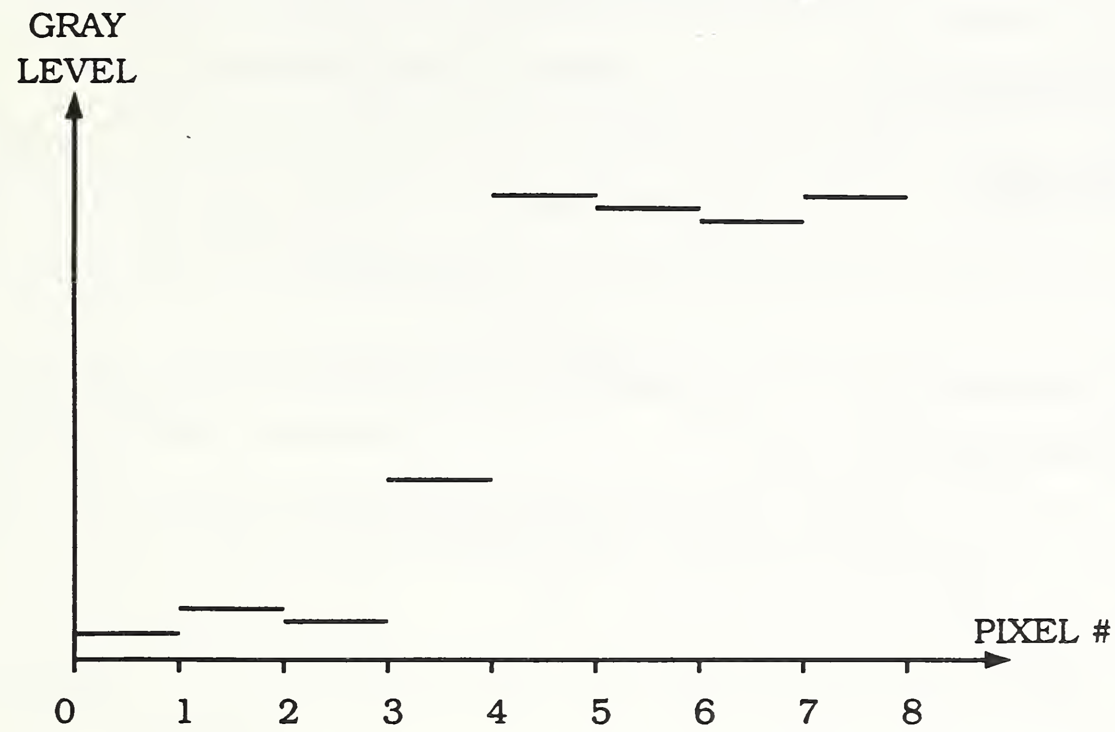

Figure 8(a). Gray Level of 1-D Image.

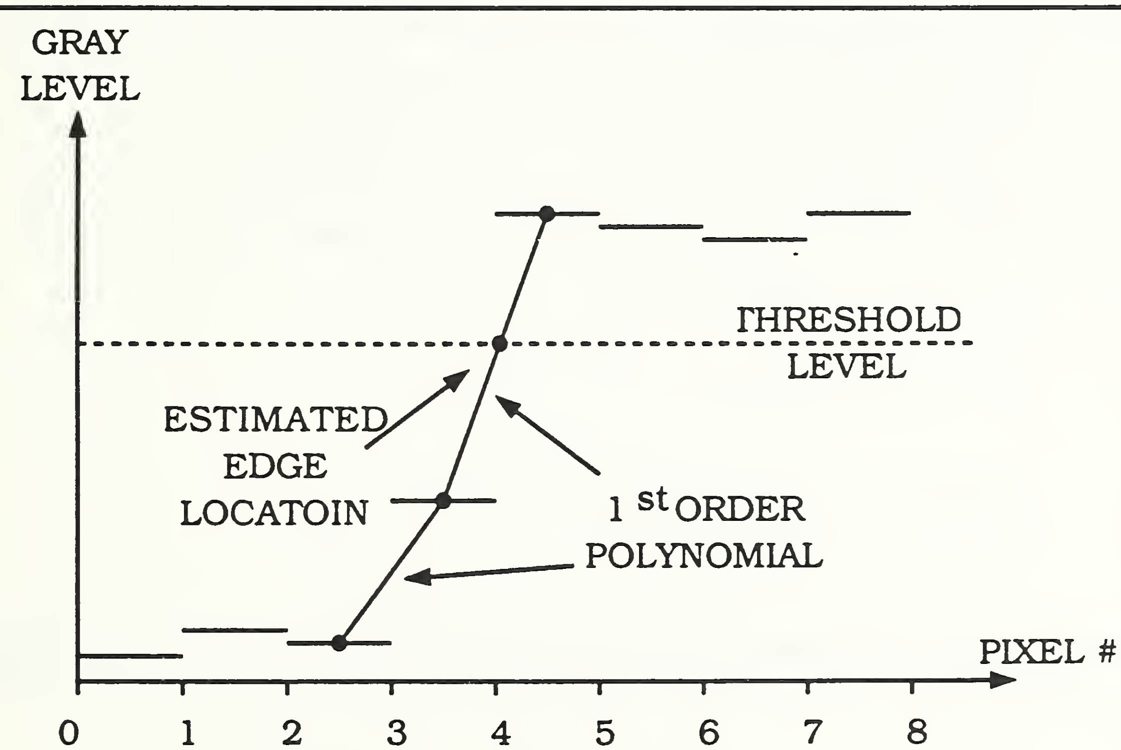

Figure 8(b). Edge estimation.

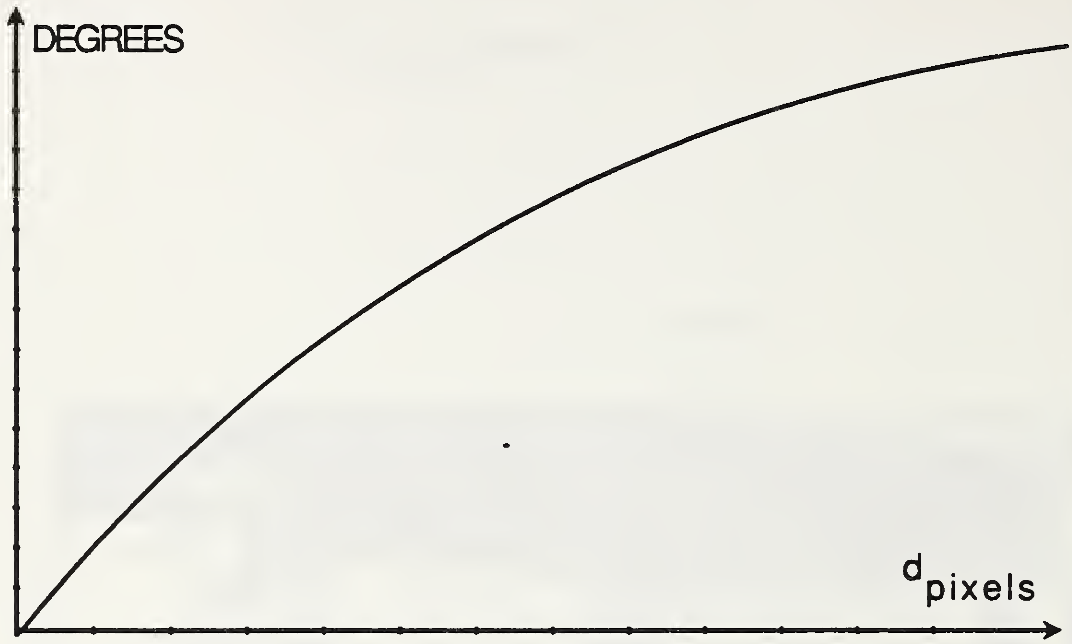

Figure 9. Conversion of pixels to angle.

NIST-114A
(REV. 3-90)

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY

BIBLIOGRAPHIC DATA SHEET

1. PUBLICATION OR REPORT NUMBER
NISTIR 4669

2. PERFORMING ORGANIZATION REPORT NUMBER

3. PUBLICATION DATE
SEPTEMBER 1991

4. TITLE AND SUBTITLE
An Integration-Based Method for Depth Estimation

5. AUTHOR(S) Daniel Raviv

6. PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)
U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
GAITHERSBURG, MD 20899

7. CONTRACT/GRANT NUMBER

8. TYPE OF REPORT AND PERIOD COVERED

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)
Robotics Center & Dept. of Electrical Engr., Florida Atlantic Univ., Boca Raton, FL 33431
and
NIST, Robot Systems Division

10. SUPPLEMENTARY NOTES

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

ABSTRACT

A closed-form, integration-based, and massively-parallel algorithm for determining depth of points in 3-D using one moving camera is presented. It is based on analyzing a sequence of images that result from a known rectilinear motion of a camera (with no rotation) in a stationary environment. A traceable point in an image sequence is reconstructed using an integration operation (no differentiation operator is involved).

The method arose from two simple observations:

(1) Stationary points in the 3-D scene appear to move away from the focus of expansion (FOE).

(2) The distance of a point in 3-D space from the camera motion-axis is the same at all instants of time.

Any visible moving point in the image can be processed independently of, and concurrently with, any other point. Laboratory results for the case where the optical axis is parallel to the motion axis show an error of less than 0.6% in absolute distance.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)
computer vision; visual motion; structure for motion; 3-D reconstruction.

13. AVAILABILITY

X	UNLIMITED
	FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).
	ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.
x	ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.

14. NUMBER OF PRINTED PAGES
23

15. PRICE
A02

ELECTRONIC FORM