Shape from Rotation using Stereo

Yizhi E. Li and M. R. M. Jenkin
Department of Computer Science
York University
North York, Ontario, Canada, M3J 1P3

Abstract

This paper examines the construction of a detailed 3D surface model of an object rotating in front of a stationary video camera. An algorithm is developed which integrates repeated stereo views of a rotating object into a 3-D model of the object. Starting with disparity estimates obtained using an existing stereo algorithm, the algorithm presented here obtains the true depth of the recovered points. As the object is rotated in front of the camera, these points are then used to construct an octree representation of the object. The resulting representation provides a full 3D representation of the objects visible exterior surfaces.

1 Introduction

Computer aided design and manufacturing are rapidly growing fields in modern industry. The flexibility and speed at which designs can be altered, examined and manufactured make these technologies very popular. One drawback of these techniques is the need to produce computer representations of existing objects in order to be able to integrate new designs with existing equipment. In this paper we present a technique for obtaining a 3D computer model of an already existing object. Utilizing a single video camera and a rotating stage, the algorithm presented here integrates repeated stereo views of the object as it rotates into a full 3D surface model. This is an initial step towards the automatic construction of full 3D surface model of objects, suitable for inclusion within a CAD/CAM model of an environment.

3D shape modeling is a fundamental research issues in computer vision. The underlying problem is the development of object representations which are sophisticated enough to model interesting objects, yet which are simple enough to permit recognition or construction from images or other sensor data. A variety of surface-based models have been proposed, including generalized cylinders[1], superquadrics[11], deformable finite-element models[14], and volumetric models such as octrees[7,10]. Once a particular representation has been chosen, there still remains the task of extracting visual information in order to build up a description of the object. For relatively small objects, one common approach to obtaining visual data from a large portion of the objects exterior is to examine the object as it rotates in front of a single stationary camera. The object can then be reconstructed by intersecting multiple silhouettes of the object as seen from different views[13,3]. This is a very limited technique as only boundary information is available, thus discarding potential information from the textured surfaces of the object. As the object rotates in front of the camera, sequential views are equivalent to left and right image pairs obtained by a converged binocular viewer. Thus an existing stereopsis algorithm (such as [4]) should be capable of obtaining dense matches between pairs of images, and these dense measurements should give rise to a more complete model of the object as it rotates.

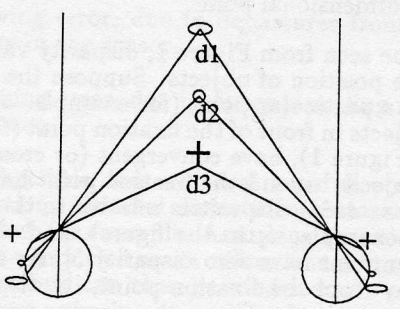

Figure 1: **Definition of Disparity** The disparity of a point refers to the angular difference in position of the image element in the two eyes. Given a fixation point, such as the circle in the figure, disparities may be crossed(d_3), uncrossed(d_1), or zero(d_2), relative to the fixation point.

A stereopsis algorithm gives a relatively dense set of image disparity measurements for each pair of rotated views of the scene without special lighting. The recovered disparity results are converted to true 3D coordinates and then represented as an octree. Representing the recovered data in an octree allows the algorithm to represent arbitrarily shaped 3D models. Experimental results show the promise of the technique.

2 Stereo

The shape from rotation algorithm relies on the use of a stereopsis algorithm to recover the disparity between successive views of feature on the surface of the

*The authors gratefully acknowledge the financial support of the Natural Sciences and Engineering Research Council.

object. The disparity and 3D position of a feature point are related, where:

- **Disparity** refers to the angular difference in position of the image element in the two eyes.
- **Distance** refers to the objective physical distance from the viewer to the object, usually measured from one of the two eyes.
- **Depth** refers to the subjective distance to the object as perceived by the viewer, usually measured relative to a fixation point or some other three-dimensional point.

As can be seen from Figure 1, disparity varies with the relative position of objects. Suppose the eyes are fixating at a particular point (for example. d_2 in Figure 1). Objects in front of the fixation point (for example, d_3 in Figure 1), have convergent (or crossed) disparities, objects beyond the fixation point have divergent (or uncrossed) disparities relative to the fixation point (for example, d_1 in the figure), and objects at the fixation point have zero disparity[5]. As an object moves away from the fixation point, its disparity increases in magnitude. Given the viewing parameters, it is straightforward to derive the relationship between distance and disparity (details for the rotational case are given later in this paper). The critical problem, however, is how should the disparity be measured? Marr[9] proposed that there are three steps involved:

1. A particular location on a surface in the scene is selected from one image.
2. The same location is found in the other image.
3. The disparity between the two corresponding image points is measured.

When two corresponding image points are identified, the actual computation of distance involves a simple geometric transformation.

In the past, the task of recovering surface height from two images of a scene has usually been expressed as either a correlation or correspondence process. Even though both methods have given some limited success, neither method has reached the performance associated with the human visual system. Correlation based processes encounter problems with perspective projection, interocular illumination differences, non-zero disparity gradients, and images that contain too much or to little structure. Correspondence based algorithms, on the other hand, reduce the computational problem to one of token matching by representing the signal as a sparse set of complex tokens. The resulting depth information is at least as sparse as the density of the monocular token used, and the problem of inferring the surface from these tokens can be formidable[8].

More recently, a number of researchers have proposed that stereo (and motion and form) processing should be considered as the task of measuring the local phase difference between bandpass filtered version

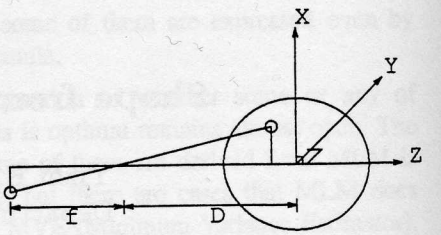

Figure 2: Camera System Calibration

of the monocular signals[4,8]. These new local phase based techniques have shown some improvements over these earlier algorithms. They provide a rich and dense set of disparity measures over a large class of image textures. The measurements provided are ideally suited to our requirements. Thus, in the work presented here we utilize a stereopsis algorithm based on local phase difference developed by Fleet, Jenkin and Jepson[4] to measure the disparity between a pair of images. For more details on the technique, the interested reader is directed to [4].

3 Framework

The shape from rotation algorithm developed in this paper converts a series of images into a 3D model of the object. In order to map disparities into an object centered coordinate system, the experimental setup must be calibrated. One approach to calibration is to image a known 3D reference model, and to use the known position of points on the object to calibrate the system. In order to automatically determine the object's rotation angle without the use of any additional sensors, a rotational stage which gives a precise measure of the rotation was used. A black background was placed behind the turntable so as to reduce the effect of background texture.

The actual shape from rotation algorithm operates in the following stages.

1. The camera system is calibrated. The transformation between the camera image coordinates and the 3D world coordinate system is computed through a known 3D model such as a cube (see Figure 2).
2. The disparity between successive image pairs is extracted using an existing stereo algorithm. The first image is considered as the left image and the second image as the right image for the first processing pair. In the next iteration the second image is the left image and the third image is the right image for the second processing pair, and so on.
3. After running the stereo algorithm on a pair of images, the disparity of the image points are obtained. As the camera system has been calibrated, it is possible to compute the true depth of each recovered point in the image (this is covered in a later section).

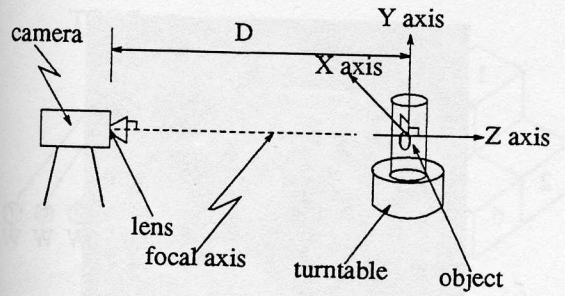

Figure 3: Experimental set up

4. The recovered points (x,y,z) of the image are represented in an octree.

5. Go back to step 2, repeat the processing for a complete 360° rotation of the object.

4 Camera Calibration (after Horn[6])

The experimental set up of the shape from rotation experiment is shown in Figure 3. The camera is positioned in front of the turntable with the camera's focal axis roughly aligned with the Z axis of the turntable. An object, such as a cup, is put on the center of the turntable. An object based co-ordinate system is centered on the top of the rotation platform, with the Y axis pointing up from the platform, and X and Z lying on the rotation plane. A calibration grid on the turntable defines the X and Z direction. There are many different techniques available for calibrating a camera system. Following Horn[6], a calibration object is used to experimentally determine the transformation from the object to image co-ordinate.

Assume a pin-hole camera model as shown in Figure 2. Let (u, v) be the image of a point (x_c, y_c, z_c) in the camera coordinates system and let f be the focal length of the camera. Furthermore, let (x_a, y_a, z_a) be the co-ordinate of the point in the rotating stage coordinate system.

Point (x_c, y_c, z_c) and point (x_a, y_a, z_a) are related as follows:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_a \\ y_a \\ z_a \\ 1 \end{bmatrix} \quad (1)$$

Where the r_{ij} elements define the transform in homogeneous co-ordinates to map one coordinate system to the other. In an ideal case (u, v) are related to (x_c, y_c, z_c) by $(u = fx_c/z_c, v = fy_c/z_c)$. In an electro-optical system, it may be very difficult to ensure the two axes of the measurement system are exactly orthogonal and the distances measured along the axes are scaled equally. In order to take these effects and others into account, a more general model of the camera imaging system is required. One approach[6] is to assume that an affine transform of the image occurs in the camera. An *affine* transformation in the plane is linear and can work for uniform scaling, translation, rotation, skewing, and shearing. This transformation can absorb all of the departures from the ideal situation as follows:

- Scaling error, due to inaccurate knowledge of focal length (f).
- Translation error, due to inaccurate knowledge of the origin.
- Rotation error, due to inaccurate knowledge of image sensor rotation.
- Skewing error, due to departures from orthogonality in the sensor.
- Shearing error, due to unequal scaling in the image axes.

Note that this does not deal with all possible image distortion, as radial distortions of the camera are not modeled, for example. Assuming that the calibration can be modeled as an affine transformation, the imaging process can then be modeled as:

$$\begin{bmatrix} u/f \\ v/f \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_c/z_c \\ y_c/z_c \\ 1 \end{bmatrix} \quad (2)$$

so that

$$\begin{aligned} u/f &= (a_{11}x_c + a_{12}y_c + a_{13}z_c)/z_c \\ v/f &= (a_{21}x_c + a_{22}y_c + a_{23}z_c)/z_c \end{aligned} \quad (3)$$

Combining (1) with (3) and simplifying, results in:

$$\begin{aligned} u/f &= (x_aA_1 + y_aB_1 + z_aC_1 + D_1)/E \\ v/f &= (x_aA_2 + y_aB_2 + z_aC_2 + D_2)/E \end{aligned} \quad (4)$$

where

$$\begin{aligned} A_1 &= a_{11}r_{11} + a_{12}r_{21} + a_{13}r_{31}, \\ B_1 &= a_{11}r_{12} + a_{12}r_{22} + a_{13}r_{32}, \\ C_1 &= a_{11}r_{13} + a_{12}r_{23} + a_{13}r_{33}, \\ D_1 &= a_{11}r_{14} + a_{12}r_{24} + a_{13}r_{34}, \\ A_2 &= a_{21}r_{11} + a_{22}r_{21} + a_{23}r_{31}, \\ B_2 &= a_{21}r_{12} + a_{22}r_{22} + a_{23}r_{32}, \\ C_2 &= a_{21}r_{13} + a_{22}r_{23} + a_{23}r_{33}, \\ D_2 &= a_{21}r_{14} + a_{22}r_{24} + a_{23}r_{34}, \\ E &= r_{31}x_a + r_{32}y_a + r_{33}z_a + r_{34}. \end{aligned} \quad (5)$$

Defining

$$s = \begin{bmatrix} s_1 & s_2 & s_3 & s_4 \\ s_5 & s_6 & s_7 & s_8 \\ s_9 & s_{10} & s_{11} & s_{12} \end{bmatrix} = \begin{bmatrix} A_1 & B_1 & C_1 & D_1 \\ A_2 & B_2 & C_2 & D_2 \\ r_{31} & r_{32} & r_{33} & r_{34} \end{bmatrix} \quad (6)$$

then (4) can be written as:

$$\frac{u}{f} = \frac{s_1x_a+s_2y_a+s_3z_a+s_4}{s_9x_a+s_{10}y_a+s_{11}z_a+s_{12}}$$
$$\frac{v}{f} = \frac{s_5x_a+s_6y_a+s_7z_a+s_8}{s_9x_a+s_{10}y_a+s_{11}z_a+s_{12}} \quad (7)$$

The problem is to determine the twelve unknowns of the transformation. One way to proceed is to use a number of calibration points whose coordinates are known in the external system. The transformation can be recovered if we measure the coordinates of the corresponding points in the image. Each measurement results in one pair of linear equations:

$$u(s_9x_a + s_{10}y_a + s_{11}z_a + s_{12}) - f(s_1x_a + s_2y_a + s_3z_a + s_4) = 0$$
$$v(s_9x_a + s_{10}y_a + s_{11}z_a + s_{12}) - f(s_5x_a + s_6y_a + s_7z_a + s_8) = 0 \quad (8)$$

Six image measurements are required to solve for the twelve unknowns. However, these equations are homogeneous. In other words the transformation is not affected by uniform scaling of all coefficients so that there are actually eleven degrees of freedom. Without any further assumptions, six image measurements will be more than enough to solve the equations.

The two constraints induced by a single measurement can be written in the form $\mathbf{u}\cdot\mathbf{c} = 0$ and $\mathbf{v}\cdot\mathbf{c} = 0$, where

$$\mathbf{u} = (-fx_a, -fy_a, -fz_a, -f, 0, 0, 0, 0, ux_a, uy_a, uz_a, u)^T,$$
$$\mathbf{v} = (0, 0, 0, 0, -fx_a, -fy_a, -fz_a, -f, vx_a, vy_a, vz_a, v)^T,$$
$$\mathbf{c} = (s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12})^T. \quad (9)$$

Given six image measurement vectors $\mathbf{u}_i$, and $\mathbf{v}_i$, it is possible to solve for the vector $\mathbf{c}$. More practically, additional measurements should be used to improve accuracy. It is then no longer possible to find a set of coefficients that will make all of the constraint equations exactly equal to zero. Instead, a least squares approach is used to solve for $\mathbf{c}$ when one of the coefficients (s_{12}) is 1.

5 Recovering Depth

Once the unknown vector $\mathbf{c}$ has been recovered, the depth of the image point and its true 3D point (x_a, y_a, z_a) can be computed as follows: Assume that an image point (x_a, y_a, z_a) is projected at time t to (u_1, v_1), and that at time $t+1$ it is projected to (u_2, v_2) (after the platform rotating by an amount $\Delta\theta$). Then, $u_2 = u_1 + disp$, where $disp$ is the horizontal component of the disparity obtained using the stereo algorithm. To solve for the unknown point (x_a, y_a, z_a), the following linear functions must be solved:

$$\frac{u_1}{f} = \frac{s_1x_a+s_2y_a+s_3z_a+s_4}{s_9x_a+s_{10}y_a+s_{11}z_a+1},$$
$$\frac{v_1}{f} = \frac{s_5x_a+s_6y_a+s_7z_a+s_8}{s_9x_a+s_{10}y_a+s_{11}z_a+1},$$
$$\frac{u_2}{f} = \frac{s_{12}x_a+s_2y_a+s_{13}z_a+s_4}{s_{14}x_a+s_{10}y_a+s_{15}z_a+1}. \quad (10)$$

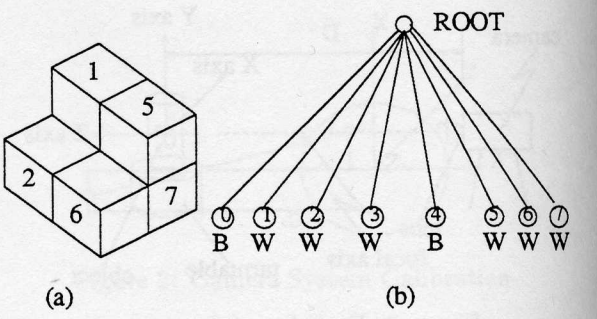

Figure 4: (a)A staircase (b)Octree representation of the staircase

where

$$s_{12} = s_1cos\theta - s_3sin\theta,$$
$$s_{13} = s_3cos\theta + s_1sin\theta,$$
$$s_{14} = s_9cos\theta - s_{11}sin\theta,$$
$$s_{15} = s_{11}cos\theta + s_9sin\theta. \quad (11)$$

The 3D point (x_a, y_a, z_a) can then be found by solving the following linear system:

$$\begin{bmatrix} u_1s_9 - s_1f & u_1s_{10} - s_2f & u_1s_{11} - s_3f \\ v_1s_9 - s_5f & v_1s_{10} - s_6f & v_1s_{11} - s_7f \\ u_2s_{14} - s_{12}f & u_2s_{10} - s_2f & u_2s_{15} - s_{13}f \end{bmatrix} \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix} = \begin{bmatrix} fs_4 - u_1 \\ fs_8 - v_1 \\ fs_4 - u_2 \end{bmatrix} \quad (12)$$

6 Representing 3D objects

An octree is used to represent the 3D points that are recovered. A rectangular octree is a regular cellular decomposition of the object space (universe) [2,12]. The universe is subdivided into eight cells of equal size. If any one of the resulting cells is homogeneous, meaning that it lies entirely inside or outside the object, the subdivision stops. On the other hand, if the cell is heterogeneous, that is, intersected by one or more of the object's bounding surfaces, the cell is subdivided further into eight subcells (see Figure 4). The subdivision process stops when all the leaf cells are homogeneous to some degree of precision. The advantage of using an octree representation is that any arbitrarily shaped object, can be represented to the precision of the smallest cell.

All the 3D points recovered from the first image pair are inserted directly into the octree. All other 3D points that are recovered from later pairs of images are rotated about y axis by $\Delta\theta$, where $\Delta\theta$ is the rotational angle of the left image, and then inserted into the octree. As mentioned before, $\Delta\theta$ can be directly read from the rotational stage.

7 Experimental results

Some preliminary results with the shape from rotation using stereo algorithm are given in the following figures. A simple calibration object is used for calibrating the camera system and rotating platform as discussed above. Each image is digitized to 128x128x8 bits. An existing stereopsis algorithm[4] is used to

Figure 5: first image in rotation sequence of a coffee cup

Figure 8: first image in rotation sequence of a toy

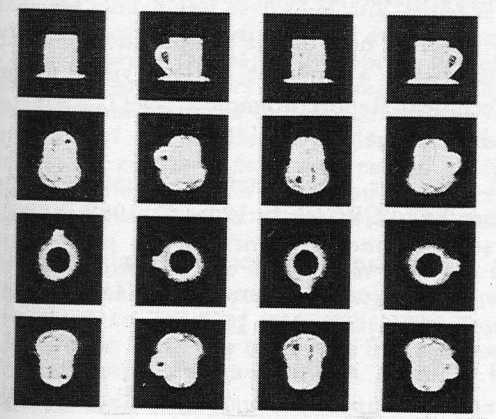

Figure 6: 16 different orientation views of the recovered cup

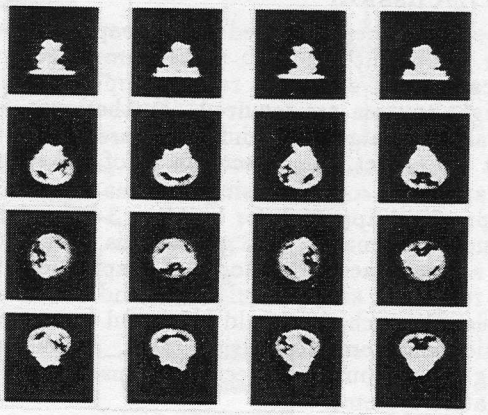

Figure 9: 16 different orientation views of the recovered toy

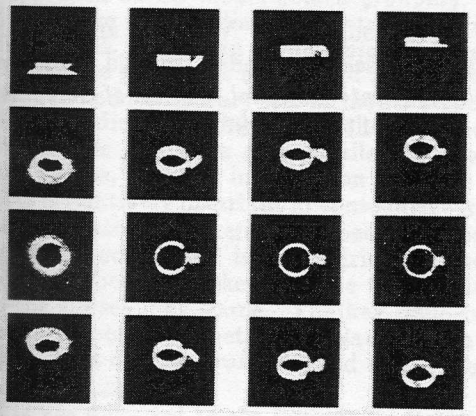

Figure 7: views of different sliced layers of the recovered cup

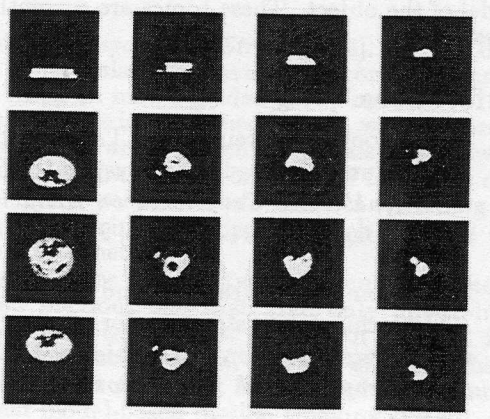

Figure 10: views of different sliced layers of the recovered toy

determine the disparities estimated from successive frames and their uncertainty, which gives quite good results. Figure 5 shows the first frame from the rotational sequence of a coffee cup. The cup was rotated 5° between measurements. The recovered 3D object viewed from 16 different orientations is given in Figure 6. Figure 7 shows the same reconstructed object sliced into layers. Note that poorly textured regions give fewer responses than highly textured regions. This results from the stereo algorithm which is unable to obtain reliable disparity estimates for regions which lacks texture. The general shape of the cup (and the rotating platform) have been recovered. The hollow nature of the cup, and the handle are clearly visible.

Figure 8 shows the first frame of the rotation sequence of a toy figure. The recovered object and slices of the recovered object are given in Figure 9 and 10. The 3D structure is clearly visible.

8 Discussion

The techniques described in this paper perform 3D shape reconstruction with rather simple equipment. The calibration system is rather simple, and no special light sources are required. Furthermore, since a general stereo algorithm and octree are used to reconstruct the object, the object can be of any shape.

Shape from rotation using a stereo algorithm is a very practical approach for building 3-D models from a sequence of images. The goal of this work is to produce a locally accurate model of shape and intensity with relatively simple equipment and non-restricted models. As such, it should be useful in a variety of robotics and computer vision tasks, as well as providing novel input for objects to be used in computer animation systems.

Some tasks remain. An analysis of the error of the recovered 3D surface remains to be done. In addition, a process is required to trim isolated spurious responses from the object description and some mechanism should be used to merge the recovered data points into a surface or more compact volumetric model of the object. These topics are currently under investigation.

References

[1] R. A. Brooks, R. Greiner, and T. O. Binford. The acronym model-based vision system. In *Sixth International Joint Conference on Artificial Intelligence*, pages 105–113, Tokyo, Japan, 1979.

[2] I. Carlbom, I. Chakravarty, and D. Vanderschel. A hierarchical data structure for representing the spatial decomposition of 3-d objects. *IEEE Computer Graphics and Applications*, pages 24–31, 1985.

[3] H. H. Chen and T. S. Huang. A survey of construction and manipulation of octrees. *CVGIP*, 43:409–431, 1988.

[4] D. J. Fleet, A. D. Jepson, and M. R. M. Jenkin. Phase-based disparity measurement. *CVGIP: Image Understanding*, 53(2):198–210, 1991.

[5] W. E. L. Grimson. *From Image to Surfaces (A Computational Study of the Human Early Visual System)*. The MIT Press, Cambridge, Massachusetts, 1981.

[6] B. K. P. Horn. *Robot Vision*. The MIT Press, Cambridge, Massachusetts, 1986.

[7] C. J. Jackins and S. L. Tanimoto. Oct-trees and their use in representing three-dimensional objects. *CVGIP*, 14:249–270, 1980.

[8] M. R. M. Jenkin, A. D. Jepson, and J. K. Tsotsos. Techniques for disparity measurement. *CVGIP: Image Understanding*, 53(1):14–30, 1991.

[9] D. Marr. A note on the computation of binocular disparity in a symbolic, lowlevel visual processor. Technical Report Memo 327, MIT Artificial Intelligence Laboratory, 1974.

[10] D. Meagher. Geometric modeling using octree encoding. *CVGIP*, 19:129–147, 1982.

[11] A. P. Pentland. Perceptual organization and the representation of natural form. *Artificial Intelligence*, 28(3):293–331, 1986.

[12] H. Samet. *Applications of Spatial Data Structures (Computer Graphics, Image Processing, and GIS)*. Addison-Wesley, Reading, Massachusetts, 1990.

[13] R. Szeliski. *Bayesian Modeling of Uncertainty in Low-Level Vision*. Kluwer Academic Publishers, Boston, Massachusetts, 1989.

[14] D. Terzopoulos, A. Witkin, and M. Kass. Symmetry-seeking models and 3d object reconstruction. *International Journal of Computer Vision*, 1(3):211–221, 1987.