One-Eyed Stereo: A General Approach to Modeling 3-D Scene Geometry

Thomas M. Strat and Martin A. Fischler
Artificial Intelligence Center
SRI International
Menlo Park, California 94025

Abstract

A single 2-D image is an ambiguom representation of the 3-I) world many different scenes could have produced the same image -yet the human visual system is extremely successful at recovering a qualitatively correct depth model from this type of representation. Workers in the field of computational vision have devised many distinct schemes that attempt to duplicate this ability of human vision; these schemes are collectively called "shape from ...." methods *(e.g.,* shape from shading, shape from texture, shape from contour). In this paper we argue that the distinct assumptions employed by each of these different schemes must be equivalent to providing a second (virtual) image of the original scene, and that all of these different approaches can be translated into a conventional stereo formalism. In particular, we show that it is frequently possible to structure the problem as that of recovering depth from a stereo pair consisting of a conventionial perspective image (the *original* image) and an orthographic image (the *virtual* image). We provide a new algorithm of the form required to accomplish this type of stereo reconstruction task.

1 Introduction

The recovery of 3-D scene geometry from one or more images, which we will call the scene modeling problem (SMP), has solutions that appear to follow one of three distinct paradigms: stereo; optic flow; and shape from shading, texture, and contour.

In the stereo paradigm, we match corresponding world/scene points in two images, and, given the relative geometry of the two cameras (eyes) that acquired the images, we can use simple trigonometry to determine the depths of the matched points [1].

In the optic flow paradigm, we use two or more images to compute the image velocity of depicted scene points. If the camera's motion and imaging parameters are known, we can again use simple trigonometry to convert velocity measurements in the image to depths in the scene [20].

In the shape from shading, texture, and contour (SSTC) paradigm, we must either know, or make some assumptions about the nature of the scene, the illumination, and the imaging geometry. Reference [2] contains an excellent collection of papers, many of which address the problem of how to recover depth from the shading, texture, and contour information visible in a single image. Two distinct computational approaches have been employed in the SSTC' paradigm: (a) integration of partial differential equations describing the relation of shading in an image to surface geometry in a scene, and (b) back-projection of planar image facets to undo the distortion in an image attribute *(e.g.,* edge orientation) induced by the imaging process on an assumed scene property *(e.g.,* uniform distribution of edge orientations).

Our purposes in this paper are to provide a unifying framework for the scene modeling problem, and to present a new computational approach for recovering scene geometry from the shading, texture, and contour information present in a single image. Our contribution is based on the following observation: regardless of the assumptions employed in the SSTC paradigm, if a 3-D scene model has been successfully derived, it will generally be possible to establish a large number of correspondences between image and scene (model) points. From these correspondences we can compute a collineation matrix [10] and extract from the matrix the imaging geometry [3] [18]. We can now construct a second image of the scene as viewed by the camera from some arbitrary location in space. It is thus obvious that any technique that is competent to solve the SMP must either be provided with at least two images, or must make assumptions that are equivalent to providing a second image. We can unify the various approaches to the SMP by converting their associated assumptions and auxiliary information into the implied second image and employ the stereo paradigm to recover depth. In the case of the SSTC paradigm, our approach amounts to "one-eyed stereo."

2 Shape from One-Eyed Stereo

Most people viewing Figure 1 get a strong impression of depth We can recover an equivalent depth model by assuming that we are viewing a projection of a uniform grid and employing the computational procedure to be described. In the remainder of this paper we will show how various simple modifications and variations of the uniform grid, as the implied second image, allow us to recover depth from shading, texture, and contour.

The one-eyed stereo paradigm can be described as a five-step process, as outlined in the paragraphs below. Differences in the scenes and the image-formation processes will require variations in the particular procedures to be used, but the general approach will remain the same,

2.1 Partition the image

As with all approaches to the SMP, the image must be segmented into regions prior to the application of a particular algorithm on any individual portion of the image. Before the one-eyed stereo computation can be employed, the image must be segmented

The work reported herein was supported by the Defense Advanced Research Projects Agency under Contract No. MDA903-83-C-0027.

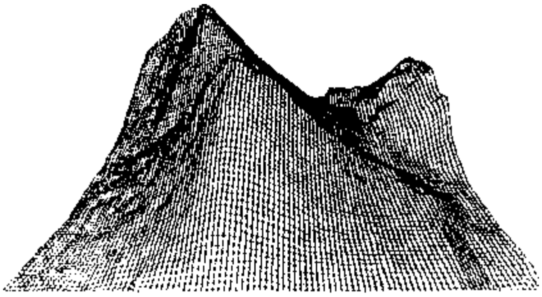

Figure 1: A synthetic image

into regions that can be described by a single underlying model. The computation can then be carried out independently in each region, and the results knitted together.

2.2 Select a model

For each region identified by the partitioning process, we must determine the underlying model that explains that portion of the image. Surface reflectance functions and texture patterns are examples of such models. Partitioning the image and selecting the appropriate models are difficult problems that are not addressed in this paper. Witkin and Kass [22] are exploring a new class of techniques that promises to provide answers to these questions. It will not be possible to recover depth where no single model can be associated with a particular image region. Similarly, inaccurate or incorrect results can be expected if the partitioning or modeling is performed incorrectly.

2.3 Generate the virtual image

The key to one-eyed stereo is using the model to fabricate a second (virtual) image of the scene. The idea is that the model often allows one to construct an image of the scene that is independent of the actual shape of the imaged surface. This allows the virtual image to be determined solely from knowledge of the model without making use of the original image. For example, the markings on the surface of Figure 1 could have arisen from a projection of a uniform grid upon the surface (Figure 2). For all images that fit this model, we can use a uniform grid as the virtual image. The orientation, position, and scale of this grid will typically be unknown, and we will show how this information can be recovered from the original image. Other models give rise to other forms of virtual images.

2.4 Determine correspondences

In order to apply stereo techniques to determine depths, we must first establish correspondences between points in the real image and the virtual image. When dealing with textures, the process is typified by counting texels in each image from a chosen starting point. With shading, the general approach is to integrate intensities. Several variations are described in the next section, and the difficulty of the procedure will depend on the nature of the model.

2.5 Compute depths using stereo

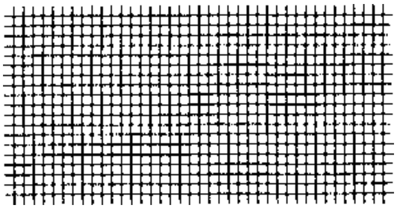

Figure 2: The virtual image of Figure 1

With two images and a number of point-to-point correspondences in hand, the techniques of binocular stereo are immediately applicable. At this point, the problem has been reduced to computing the relative camera models between the two images and using that information to compute depths by triangulation. The fact that the virtual image will normally be an orthographic projection required reformulation of existing algorithms for performing this computation. The appendix describes a new algorithm that computes the relative camera model and reconstructs the 3-D scene from eight point-correspondences between a perspective and an orthographic image.

The problem of recovering scene and imaging geometry from two or more images has been addressed by workers in both binocular stereo and monocular perception of motion (where tintwo projections are separated in time as well as space). Various approaches have been used to derive equations for the 3-D coordinates and motion parameters; these equations are generally solved by iterative techniques [4] [7] [12] [13]. Oilman [20] presents a solution for recovering 3-D shape from three orthographic projections with established correspondences among at least four points. His "polar equation" allows computation of shape when the motion of the scene is restricted to a rotation about the vertical axis and arbitrary translation. Nagel and Neumann [9] provide a compact system of three nonlinear equations for the unrestricted problem when five point-correspondences between the two perspective images are known. More recently. Huang [19] and Longuet-Higgins [8] have independently derived methods that only require the solution of a set of eight simultaneous linear equations when eight point-correspondences are known between two perspective images. In our formulation we are faced with a stereo problem involving a perspective and an orthographic image, and while the aforementioned references are related, none provides a solution to this particular problem.

The derivation described in the appendix was inspired by the formulation of Longuet-Higgins for perspective images. When either image nears orthography, Longuet-Higgins' method becomes unstable and is undefined if either image is truly orthographic Moreover, his approach requires knowledge of the focal length and principal point in each image. Our method was specifically derived for one orthographic and one perspective image whose internal imaging parameters may not be fully known.

3 Variations on the Theme

In this section we illustrate how our approach is used with several models of texture, shading, and contour. Where these models

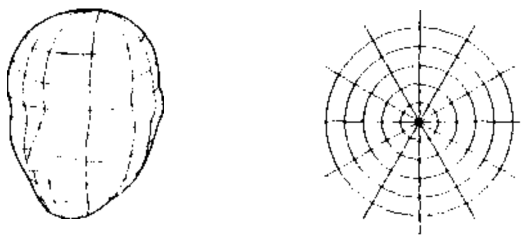

Figure 3: (a) The original image (b) The virtual image

don't match given scene characteristics, they may require additional modification. However, a qualitatively correct answer might still be obtainable by applying one of the specific models we discuss in the following subsections to what appears to be an inappropriate situation, or a situation where the validity of the assumptions cannot be established.

3.1 Shape from texture

Surface shapes are often communicated graphically to humans by drawings like Figure 1. These drawings can also be interpreted by one-eyed stereo. In this case, there is no need to partition the image; the underlying model of the entire scene is that the intersections of the lines are distributed in the form of a square grid. When viewed from directly above at an infinite distance, the surface would appear as shown in the virtual image of Figure 2 regardless of the shape of the surface. This virtual image can be construed as an orthographic projection of the object surface from an unknown viewing direction. Correspondences between the original and virtual images are easily established if there are no occlusions in the original image. Select any intersection in the original image to be the reference point and pair it with any intersection in the virtual image. A second corresponding pair can be found by moving to an adjacent intersection in both images. Additional pairs are found in the same manner, being careful to correlate the motions in each image consistently in both directions. When occlusions are present, it may still be possible to obtain correspondences for all visible junctions by following a non-occluded path around the occlusion. If no such path can be found, the shape of each isolated region can still be computed, but there will be no way to relate the distances without further information. Other techniques used to graphically represent images of 3-D shapes may require other virtual images. Figure 3a, for example, would imply a virtual image as shown in Figure 3b. Methods for recognizing which model to apply are needed, but are not discussed here.

Once correspondences have been determined, we can use the algorithm given in the appendix to recover depth. We have presumably one perspective image and one orthographic image whose scale and origin are still unknown. The depths that will be recovered will be scaled according to the scale chosen for the virtual image[1]. The choice of origin for the orthographic image is arbitrary, and will result in the same solution regardless of thf point chosen as the origin. The appendix shows how to compute the orientation of the orthographic coordinate system relative to the perspective imaging system as well as the displacement between the two, given the choice of origin for the orthographic view. 3-D coordinates of each matched point are then easily computed using back-projection. A unique solution will be obtained whenever the piercing point of the perspective image is known. A minimum of eight pairs of matched points are required to obtain a solution; depths can be computed for all matched points.

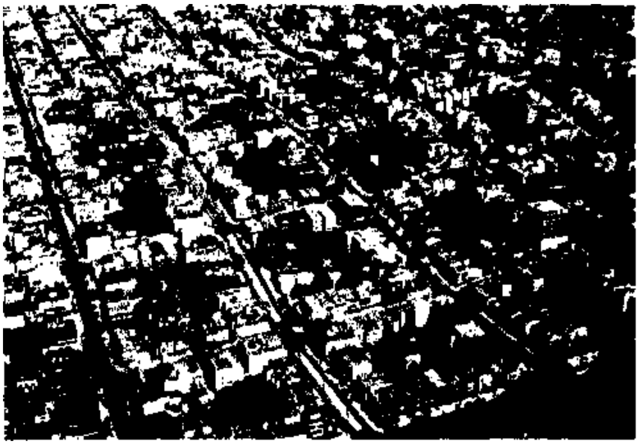

Figure 4: The streets in this scene resemble a projected texture.

There exists a growing literature on methods to recover shape from natural textures [6][II][17][21]. We will now show how the constraints imposed by one particular type of natural texture can be exploited to obtain similar results by using one-eyed stereo.

Consider the pattern of streets in Figure 4. If this city were viewed from an airplane directly overhead at high altitude, the streets would form a regular grid not unlike the one used as the virtual image in Figure 2. There are many other scene attributes that satisfy this same model. The houses in some cities would appear to be distributed in a uniform grid if viewed from directly overhead. In an apple orchard growing on a hillside, the trees would be planted in rows that are evenly spaced when measured horizontally.

Ignoring the nontrivial tasks of partitioning these images into iso-textural regions, verifying that they satisfy the model, and identifying individual texels, it can be seen how these images can be interpreted using the same techniques as in the previous section. The virtual image in each case will be a rectangular grid, and can be considered as an orthographic view from an unknown orientation. Correspondences can be determined by counting street intersections, rooftops, or apple trees. As before, one can solve for the relative camera model and compute depths of matched points. Obviously, for the situations discussed here, we must be satisfied with a qualitatively-correct interpretation due to the difficulty of locating individual texels reliably and accurately, as well as the numerical instabilities arising from the underlying nonlinear transformation.

8.2 Shape from shading

For our purposes, surface shading can be considered the limiting case of a locally uniform texture distribution, as the texels approach infinitesimal dimensions (as seen near the horizon in Figure 1). To compute correspondences, we need to appropriately integrate image intensities in place of counting lines, since

[1] Recall that, the original image does not contain the information necessary to recover the absolute size of the scene.

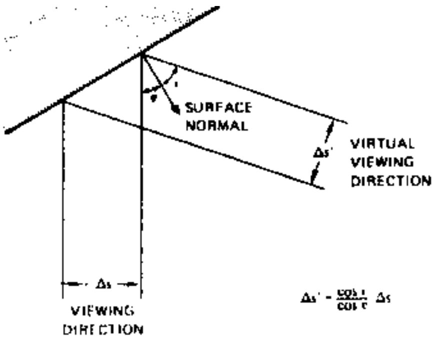

Figure 5: The geometry along a line in the direction of the light source

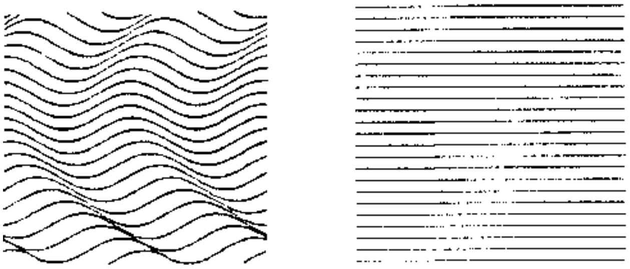

Figure 6: (a) An image of contours [16] (b) Its virtual image

the image intensities can be seen to be related to the density of lines projected on the surface. The feasibility of this procedure depends on the reflectance function of the surface.

What types of material possess the special property that allows their images to be treated like the limit of the projected texture of the previous section? It must be the case that the integral of intensity in an image region is proportional to the number of texels that would be projected in that region. This can be described in terms of i and c, where i is the angle between the local surface normal and the light source, and e is the angle between the surface normal and the viewpoint. It can be seen that the number of texels projected onto a surface patch will be proportional to cos i, the cosine of the incident angle. At the same time, the surface patch (as seen from the viewpoint) will be foreshortened by cose, the cosine of the emit Lance angle. Thus, the integral of reflected light intensity over a region will be proportional to the flux of the light striking the surface if the intensity of the reflected light at any point is proportional to cosi/cose. Horn [5] has pointed out that the material in the maria of the moon, and other rocky, dusty objects when viewed from great distances, possess a reflectance function that allows recovery of the ratio cosi/cose from the imaged intensities. This surface property has allowed unusually simple algorithms for computing shape-from-shading, so it is not surprising that it easily submits to one-eyed stereo as well.

To interpret this type of shading, we can construct a virtual image whose direction of view is the lighting direction (*i.e.,* taken from a "virtual camera* located at the light source). When the original shaded image is orthographic, we consider a family of parallel lines that lie in planes that include both the light source and the (distant) view point. When viewed from the light source, the image of the surface corresponding to these lines will also be a set of parallel lines regardless of the shape of the surface. These parallel lines constitute the virtual image. We will use the image intensities to refine these line-to-line correspondences to point-to-point correspondences. Figure 5 shows the geometry for an individual line in the family. A little trigonometry shows that

A*' =——*At* (1)

where ▼s is a distance along the line in the real image and A*' is the corresponding distance along the corresponding lino in the virtual image. Integrating this equation produces the following expression, which defines the point correspondences in the two images along the given line.

$$s' = s'_0 + \int_0^s \frac{\cos i}{\cos e} \Delta s \qquad (2)$$

To use this equation we must first compute cops/cose from the intensity value at each point along the line. This will, of course, be possible only when the reflectance function is constant for constant cosi/cose. With these point-to-point correspondences in hand, it is a simple matter of triangulation to find the 3-D coordinates of the surface points, given that we know the direction to the light source. We can explore the remainder of the surface by repeating the process for each of the successive parallel lines in the image. It still remains to tie each of the adjacent profiles together, as the scale factor of each profile has not been determined. Knowledge of the actual depth of one point along each profile provides the necessary additional information. It is important to note that our assumptions and initial conditions are those used by Horn; the fact that he was able to obtain a solution under these conditions assured the existence of a suitable virtual image for the one-eyed stereo paradigm.

3.3 Shape from contour

It is sometimes possible to extract a line drawing, such as shown in Figure 6, from scene textures. Parallel streets like those encountered in Figure 4 give rise to a virtual image consisting of parallel lines when the cross streets cannot be located; terraced hills also produce a virtual image of parallel lines. Correspondences between real and virtual image lines can be found by counting adjacent lines from an arbitrary starting point. This matches a virtual image line with each point in the real image. Point-to-line correspondences are not sufficient to employ the stereo computation of the appendix to reconstruct the surface Knowledge of the relative orientation between the two images (equivalent to knowing the orientation of the camera of the real image relative to the parallel lines in the scene) provides the necessary additional constraint; the surface can then be reconstructed uniquely through back-projection. Without knowledge of the relative orientation of the virtual image, heuristics must be employed that relate points on adjacent contours so that a regular grid can be used as the virtual image. The human visual system is normally able to interpret images like Figure 6 although just what assumptions are being made remains unclear. Further study into this phenomenon may lead to the extraction of models suitable to the employment of one-eyed stereo on this type of image without requiring prior knowledge of the virtual orientation

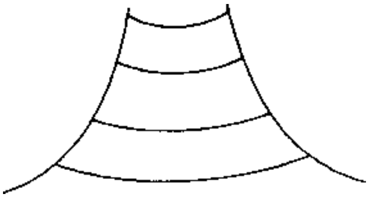

Figure 7: This simple drawing has two reasonable interpretations. It is seen as curved roller-coaster tracks if the lines are assumed to be the projection of a rectangular grid, or as a volcano when the lines are assumed to be the projection of a circular grid.

3.4 Distorted Textures and Unfriendly Shading

We have already noted that image shading can be viewed as a limiting (and, for our purposes, a degenerate) result of closely-spaced texture elements. In order to recover depth from shading, we must use integration to replace counting the texture elements that define the locations of the "grid lines" of our virtual image. The integration process depends on having a "friendly" reflectance function and an imaging geometry that allows us to convert distance along a line in the actual image to a corresponding distance along a line in the virtual image.

The recovery of lunar topography from a single shaded image (5), as discussed in Section 3.2, is one of the few instances in which "shape from shading" is known to be possible without a significant amount of additional knowledge about the scene; and even here we are required to know the actual reflectance function, the location of the (point) source of illumination, the depths along a curve on the object surface, and be dealing with a portion of the object having constant albedo. Further, the reflectance function had to have just the property that we require to replace direct counting, i.e., the reflectance function had to compensate exactly for the foreshortening" of distance due to viewing points on the object surface at an unknown tangent-plane orientation angle. Most of the commonly encountered reflectance functions, such as Lanibertian reflectance, do not have this friendly property, and it is not clear to what extent it is possible to recover depth from shading in such cases *[e.g.,* see Pentland [11] and Smith [14]). Additional assumptions will probably be needed and the qualitative nature of the recovery will be more pronounced. Just as in the case where a complex function can be evaluated by making a local linear approximation and iterating the resulting solution, it may be possible to deal with unfriendly, or even unknown, reflectance functions by assuming that they are friendly about some point, directly solving for local shape using the algorithm applicable to the friendly case, and then extending the solution to adjacent regions. We are currently investigating this approach.

The uniform rectangular grid and the polar grid that we used as virtual images to illustrate our approach to one-eyed stereo are effective in a large number of cases, because there are processes operating in the real world that produce corresponding textures (i.e., grid-like textures that appear to be orthographical!y projected onto the surfaces of the scene). However, there are also textures that produce similar-appearing images, but are due to different underlying processes. For example, a uniform grid-like texture might have been created on a flat piece of terrain, which then underwent geologic deformation in this case the virtual image needed to recover depth (or the recovery algorithm) must be different from the projective case. We have already indicated the problem of choosing the appropriate model for the virtual image, and as noted above, image appearance is probably not sufficient to make this determination some semantic knowledge about the scene is undoubtedly required. Figure 7 shows an example in which two completely different interpretations of scene structure result, both believable, depending on whether we use the rectangular grid model, or the polar grid model.

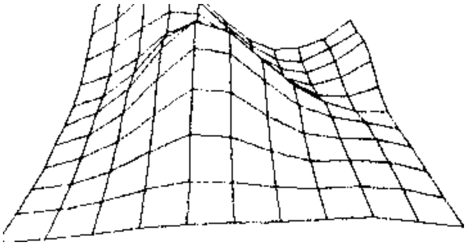

Figure 8: View of surface reconstructed from Figure I.

4 Experimental Results

The stereo reconstruction algorithm described in the appendix has been programmed and successfully tested on both real and synthetic imagery. Given a sparse set of image points and their correspondence in a virtual image, a qualitative description of the imaged surface can be obtained.

Synthetic images were created from surfaces painted with computer-generated graphic textures. Figure 1 shows a synthetic image constructed from a piece of a digital terrain model (DTM). The intersections of every 20th grid line constitute the set of 36 image points made available to the one-eyed stereo algorithm. Their correspondences were determined by selecting an arbitrary origin and counting grid lines to obtain virtual image coordinates Processing these pain by the algorithm in the appendix yields a set of 3-D coordinates in either the viewer-centered coordinate space, or the virtual image coordinate space (which, if correct, is aligned with the original DTM). Figure 8 was obtained from the 3-D coordinates in the virtual image space by fitting a surface to these points using Smith's surface interpolation algorithm [15], This gives a dense set of 3-D coordinates that can then be displayed from any viewpoint. The viewpoint that was computed by one-eyed stereo was used to render the surface as shown in Figure 8. Its similarity to the original rendering of the surface (Fig. 1) illustrates the successful reconstruction of the scene.

The same procedure was followed when working with real photographs. Using the photo of San Francisco in Figure 4, the intersections of 31 street intersections were extracted manually. Those that were occluded or indistinct were disregarded. Virtual image coordinates were obtained by counting city blocks from the lower-left intersection. The one-eyed stereo algorithm was then used to acquire 3-D coordinates of the corresponding image points in both viewer-centered and grid-centered coordinate systems. A continuous surface was fitted to both representations of these points. The location and orientation of the camera relative

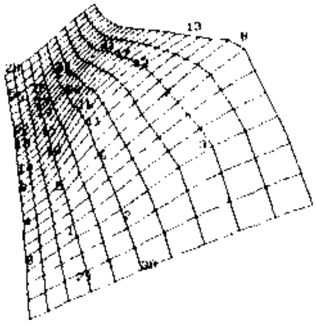

Figure 9: Perspective view of surface reconstructed from photograph of San Francisco (Figure 4)

to the grid were also computed. Figure 9 shows the reconstructed surface from the derived location of the viewpoint of the original photo The numbers superimposed are the computed locations of the original 31 points. While several of the original points were badly mislocat.ed, the general shape of the landform is apparent.

There are several reasons why the algorithm can only provide a qualitative shape description. First, the problem itself can be some what sensitive to slight perturbations in the estimates of the piercing point or focal length. This appears to be inherent to the problem of recovering shape from a single image. How humans ran determine shape monocularly without apparent knowledge of the piercing point or semantic content of the scene remains unresolved. The second factor precluding precise, quantitative description of shape is the practical difficulty of acquiring large numbers of corresponding points. While the algorithm can proceed with as few as eight points, the location of the object will only be identified at those eight points. If a more complete model is sought, then additional points will be required to constrain the subsequent surface interpolation.

The task remains to evaluate the effectiveness of the iterative technique, described in Section 3.4, for recovering (a) shape from shading in the case of scenes possessing "unfriendly" reflectance functions, and (b) shape from nonprojective and distorted textures. Our experience with the process indicates that the key to these problems lies in the ability to establish valid correspondences with the virtual image. Once these are available, reconstruction of the surface can proceed as outlined.

5 Conclusion

In this paper we have shown that, in principle, it is possible to employ the stereo paradigm in place of various approaches proposed for modeling 3-D scene geometry—including the case in which only one image is provided. We have further shown that, for the case of a single image, the approach could be implemented by:

(1) Setting up correspondences between portions of the image and variations of a uniform grid;

(2) Treating each image portion and its grid counterpart at a stereo pair, and employing a stereo technique to recover depth. (We present a new algorithm necessary to accomplish this step.)

Automatic procedures to partition the image, select the appropriate form of the virtual image, and establish the correspondences, are all difficult problems which were not addressed in this paper. Nevertheless, we have unified a number of apparently distinct problems, which, individually, would still have to contend with these same pervasive problems (i.e., partitioning, model selection, and matching).

References

|1] Barnard, S. T., and Fischler, M. A., Computational Stereo," Computing Surveys, Vol. 14, No. 4, December 1982.

|2) Brady. M., ed., Artificial Intelligence (Special Volume on Computer Vision), Volume 17, Nos. 1-3, August 1981.

[3] Ganapathy, S., "Decomposition of Transformation Matrices for Robot Vision," International Conference On Robotics, (IEEE Computer Society), March 13-16 1984, pp. 130-139.

[4] Gennery, D. B. "Stereo Camera Calibration," Proceedings of the IU Workshop, November 1979, pp. 101-107.

[5] Horn, B. K. P., "Image Intensity Understanding," MIT Artificial Intelligence Memo 336, August 1976.

|6) Kender, J. R., "Shape from Texture," PhD thesis, Carnegie Mellon University, CMU-CS-81-102, November, 1980.

[7] Lawton, D. T., "Constraint-Based Inference from Image Motion/' Proc. AAAJ-80, pp. 31-34.

(8) Longuet-Higgins, H. C, "A Computer Algorithm for Reconstructing a Scene from Two Projections," Nature, Vol. 203, September 1981, pp. 133 136.

|9] Nagel, H., and Neumann, B., "On 3-D Reconstruction from Two Perspective Views," Proc. IEEE 1981.

[10] Nitzan, D., Bolles, R.C., it et. al., "Machine Intelligence Research Applied to Industrial Automation," 12th Report SRI Project 2996, January 1983.

[11] Pentland, A. P., "Shading into Texture" Proceedings AAAI-84. AugUBt 1984, pp. 269-273.

[12| Prazdny, K., "Motion and Structure from Optical Flow," Proc IJCAI-79, pp. 704-704.

[13] Roach, J. W., and I Aggarwal, J. K., "Determining the Movement of Objects from a Sequence of Images," IEEE Trans on Pa tern Analysis and Machine Intelligence, Vol. PAMI-2, No. 6. November 1980, pp. 654 562.

[14) Smith, G. B., "The Relationship between Image Irradiance and Surface Orientation," Proc. IEEE CVPR-83.

[16| Smith, G. B., "A Fast Surface Interpolation Technique," Proceedings: DARPA Image Understanding Workshop, October 1984, pp. 211-216.

[16] Stevens, K. A., "The Line of Curvature Constraint and the Interpretation of 3-D Shape from Parallel Surface Contours," AAAI 83, pp. 1057 1061.

[17] Stevens, K. A., "The Visual Interpretation of Surface Contours," Artificial Intelligence Journal Vol. 17, No. 1, August 1981, pp. 47-73.

[18] Strat, T. M., "Recovering the Camera Parameters from a Transformation Matrix," Proceedings: DARPA Image Understanding Workshop, October 1984, pp. 264-271.

(19) Tsal, R.Y. and Huang, T.8., "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces," IEEE Trans, on Pattern Analysis and Machine Intelligence, vol. PAM1-6, No. 1, Jan 1984, pp. 13-27.

[20] Ullman, S., *The Interpretation of Visual Motion,* The MIT Press, Cambridge, Mass., 1979.

[21] Witkin, A. P., "Recovering Surface Shape and Orientation from Texture," Artificial Intelligence Journal Vol. 17, No. 1, August 1981, pp. 17-45.

[22] Witkin, A., and Kass, M., "Analysing Oriented Pattern!," in this proceedings.

Appendix

This appendix shows how 3-dimensional coordinates can be computed from point correspondences between a perspective and an orthographic projection when the relation between the imaging geometries is unknown.

We will use lower-case letters to denote image coordinates and capital letters to denote 3-D object coordinates. Unprimed coordinates will refer to the geometry of the perspective image, and primed coordinates to the orthographic image. Let X_1 and x_2 be the image coordinates of a point in the perspective image relative to an arbitrarily selected origin Let $-d_1$ and $-d_2$ be the (unknown) image coordinates of the principal point and / (> 0) he the focal length. The object coordinates associated with an image point are (X_1X_2,X_3) where the origin coincides with the center of projection and the X_3 axis is perpendicular to the image plane. The A's coordinates of any object point will necessarily be positive.

The imaging geometry is given by the following standard perspective equations:

$$x_1 + d_1 = f\frac{X_1}{X3}; \qquad x_2 + d_2 = f\frac{X_2}{X_3} \tag{3}$$

For the orthographic image, x'_1 and x'_2 are the image coordinates (relative to an arbitrary origin) and (X'_1, X'_2, X'_3) is the world coordinate system defined such that

$$x'_1 = X'_1; \qquad x'_2 = X'_2. \tag{4}$$

We use the unknown scale factor between orthographic image coordinates and the scene as our unit of measurement.

The two world coordinate systems can be related as follows.

$$X' = R(X - T) \tag{5}$$

where X is the column vector $[X_1, \ X_2, \ X_3]^T$
X' is the column vector $[X'_1, \ X'_2, \ X'_3]^T$
R is a 3x3 rotation matrix and
T is a translation vector from the center of perspective projection to the origin of the world coordinate system associated with the orthographic projection.

By substituting Equations 3 and 4 into the above, and eliminating X_3 from the two resulting equations, we get

$$\begin{aligned} 0 = \ & x'_1 x_1 R_{21} + x'_1 x_2 R_{22} + x'_1 R_2 \cdot D \\ & - x'_2 x_1 R_{11} - x'_2 x_2 R_{12} - x'_2 R_1 \cdot D \\ & + x_1 (R_{21} R_1 \cdot T - R_{11} R_2 \cdot T) + x_2 (R_{22} R_1 \cdot T - R_{12} R_2 \cdot T) \\ & + R_1 \cdot T R_2 \cdot D - R_2 \cdot T R_1 \cdot D \end{aligned} \tag{6}$$

where D is the vector $[d_1, \ d_2, \ f]$ and R_i is the i-th row of R.

The above equation relates image coordinates for corresponding points in both images. The following unknowns can be found by using eight corresponding pairs and solving the system of eight linear equations.

$$\begin{aligned} C_1 &= \tfrac{R_{21}}{R_{11}} \\ C_2 &= \tfrac{R_{22}}{R_{11}} \\ C_3 &= \tfrac{R_2 \cdot D}{R_{11}} \\ C_4 &= \tfrac{R_{12}}{R_{11}} \\ C_5 &= \tfrac{R_1 \cdot D}{R_{11}} \\ C_6 &= \tfrac{R_{21}}{R_{11}} R_1 \cdot T - R_2 \cdot T \\ C_7 &= \tfrac{R_{22}}{R_{11}} R_1 \cdot T - \tfrac{R_{12}}{R_{11}} R_2 \cdot T \\ C_8 &= R_1 \cdot T R_2 \cdot D - R_2 \cdot T R_1 \cdot D \end{aligned} \tag{7}$$

$$\begin{bmatrix} x'_1 x_1 & x'_1 x_2 & x'_1 & -x'_2 x_2 & -x'_2 & x_1 & x_2 & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \\ C_7 \\ C_8 \end{bmatrix} = \begin{bmatrix} x'_2 x_1 \\ \cdot \\ \cdot \\ \cdot \end{bmatrix} \tag{8}$$

Once we have the C_is in hand, we can solve for the components of the rotation matrix R using Equations 7 and the following properties of rotation matrices: $\| R_1 \| = 1$; $R_1 \cdot R_2 = 0$; and $R_1 \times R_2 = R_3$.

The origin of the primed coordinate system in unprimed coordinates is found to be

$$T = [C_7 \frac{R_{11}}{R_{33}}, \ -C_6 \frac{R_{11}}{R_{33}}, \ 0]. \tag{9}$$

If the location of the principal point is known but the focal length (the scale factor of the perspective image) is not, f can easily be computed from Equation 7.

$$f = \frac{C_5 R_{11} - R_{11} d_1 - R_{12} d_2}{R_{13}} \tag{10}$$

If the focal length is known, the principal point of the perspective image is found using the third and fifth expressions of Equation 7:

$$\begin{aligned} d_1 &= f\tfrac{R_{31}}{R_{33}} + \tfrac{C_5 R_{11} R_{22} - C_3 R_{11} R_{12}}{R_{33}} \\ d_2 &= f\tfrac{R_{32}}{R_{33}} + \tfrac{C_3 R_{11}^2 - C_5 R_{11} R_{21}}{R_{33}} \end{aligned} \tag{11}$$

We are now in a position to compute the world coordinates of all points for which we have correspondences. There may, of course, be many more than the 8 points used so far. The following expression is derived from Equation 5:

$$X_3 = \frac{f(x'_1 + R_1 \cdot T)}{R_{11} x_1 + R_{12} x_2 + R_1 \cdot D} \tag{12}$$

Equation 3 gives the other unprimed world coordinates:

$$X_1 = \frac{X_3}{f}(x_1 + d_1); \qquad X_2 = \frac{X_3}{f}(x_2 + d_2) \tag{13}$$

If desired, the primed coordinates are found with Equation 5.