Building, Visualizing, and Computing on Surfaces of Evolution

H. Harlyn Baker
SRI International

This article presents a new approach to giving a robot the ability to move safely through a scene using its own vision. As in man, the success of such an approach depends on the ability to operate explicitly in both space and time, and on exploiting the massive redundancy present in the hundreds of views that can be obtained when moving through a scene.

The mechanism discussed in this article for integrating these space-time factors is a 3D surface-building process called the *Weaving Wall*. In robotic navigation work, the 3D surfaces built by this process represent the space-time *evolution* of scene images, and this representation, in conjunction with geometric constraints, enables determining the 3D structure of the scene. In other domains where there is also a gradual evolution of data over a third dimension—in medical tomography, for example—the surfaces constructed by the Weaving Wall are immediately of value for their topographic structure. The designs of both the surface-building and scene-reconstructing processes make them well-suited for real-time operation given appropriate hardware.

Enabling a robot to function both safely and accurately using its own visual processing is a principal challenge in robotics. This capability is an equal necessity for robots working in a factory, exploring in the ocean depths, moving on land, or operating somewhere in deep space. These devices will have to observe in three dimensions, recognize anticipated objects either as landmarks or targets, and at the same time build, for later use, models of the various objects and terrain they encounter. Their visual sensing will have to provide for real-time navigation (including following maps and avoiding obstacles), object recognition, and object and terrain modeling (building maps and object models). Operating autonomously with minimal opportunity for manual intervention, they will have to carry out these tasks in a manner that leaves little chance for failure.

Robotic vision

Historically, the task of providing a robot with such sensing capability has been addressed with a variety of approaches, including the interpretation of single images, paired-image or stereo analysis, and the use of active ranging devices. Single-image analysis may be somewhat useful in image interpretation, where one might identify certain known forms from their appearance in images, but it is generally inadequate when the objects are three-dimensional. Moreover, it cannot provide the information necessary for constructing 3D descriptions of the scene or its components. Stereo analysis can, in many cases, recover 3D scene structure, but even the best techniques have never been demonstrated

0272-1716/88/0700-0031$01.00 © 1988 IEEE

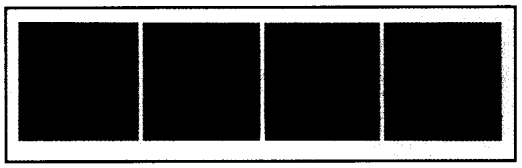


Figure 1. Four frames of a 125-frame sequence.

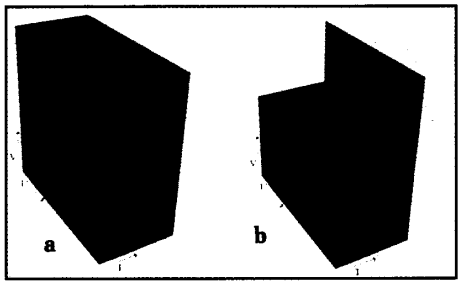


Figure 2. (a) Spatiotemporal volume, (b) spatiotemporal volume sliced horizontally.

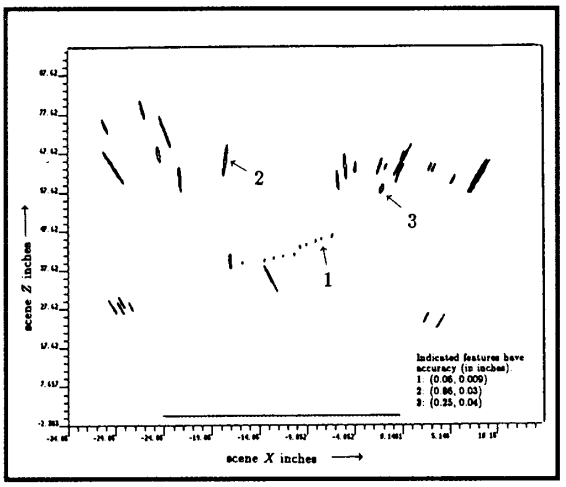


Figure 3. Estimated positions and confidence intervals from one slice.

to be broadly applicable or robust. With the development of special scanners, active sensing has become a more recent pursuit, yet even when broadcasting the device's presence does not compromise its mission, active ranging has been equally unsuccessful at achieving robust performance outside of controlled environments. Besl and Jain provide a thorough survey of these sensing and modeling requirements for a robotic system.[1]

In our research at SRI, we have taken the passive sensing of stereo, combined with it the redundancy of processing image sequences, and obtained a robust, precise, 3D vision capability ideal for many of the sensing requirements of an autonomous robot. This capability was achieved through building a process to construct 3D descriptions of the evolution of image data over time and then coupling with this an analysis procedure that exploits geometric constraints to track and estimate features in the scene. The section on sequence analysis describes our first simplified implementation of the approach and then develops the general form that more fully exploits the spatiotemporal structure of the problem. In the section on further applications we describe other uses of this space-time processor.

The visualization issue

We make extensive use of graphic aids in our work, not only for viewing and assessing the data, but also for experimenting with analysis algorithms and both displaying and evaluating the results of our computations. Indeed, the formulation we have derived for our sequence analysis work would not even have been conceived without plentiful access to 3D display. The development of this effort has depended heavily on our ability to create and examine complex sets of spatiotemporal events. This has enabled our approach to depart radically from traditional multiple-image analysis.

In viewing our data and results we rely primarily on stereoscopic display, using free fusion, a beam-splitter mirror system, or a 120-Hz Tektronix SGS420 LCD circularly polarized display. Many of the images we present here are of this stereo type, and are intended for crossed-eye viewing. This may alarm some readers, but we find such stereo viewing indispensable for work in three dimensions. To see the figures presented here in three dimensions (for example, Figure 4), view the left side with the right eye and the right side with the left eye. Special viewers can make this easier.

Often our visualization needs can be met by rendered display of three-space surfaces. Extending this, we can compose sequences of these images from various viewpoints to induce a perception of depth through motion, either when stereo fusion alone is inappropriate (for example, as shown in Figure 22, where the data has dimension greater than three), or when viewers are not comfortable with or capable of stereo fusion. Print, of course, does not allow this dynamic display. Banchoff presents a summary of these visualization issues in the context of display of point sets from higher dimension spaces.[2]

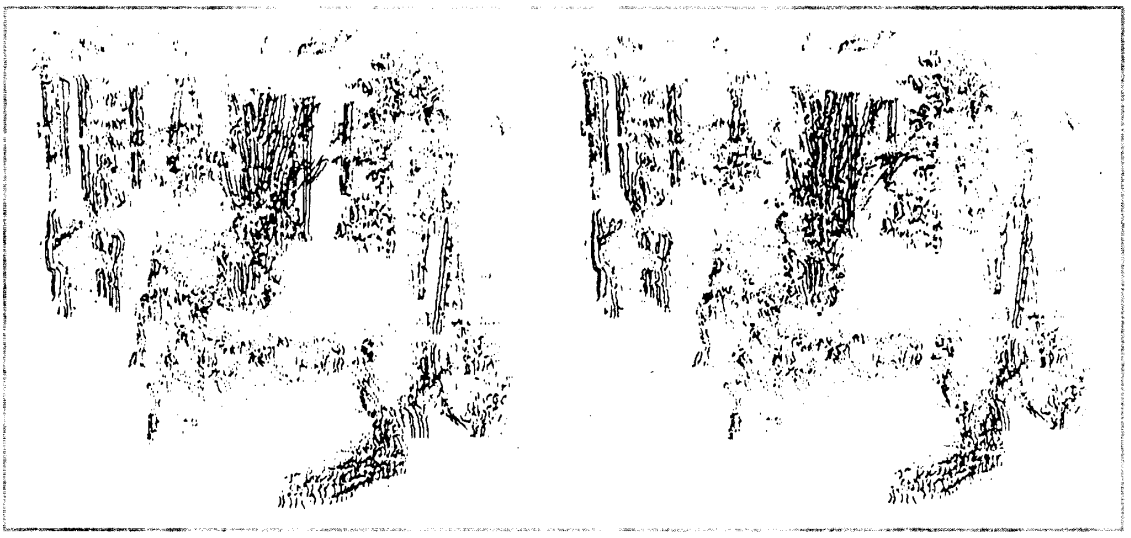


Figure 4. Resulting feature estimates for this sequence.

Sequence analysis

The principal problem in stereo vision is to put into correspondence, accurately and reliably, features that appear in two views of a scene. This correspondence and the relationship between the two imaging sites make it possible to estimate the 3D position of the features viewed. Determining the correspondence, however, is an ill-posed problem: Ambiguity, occlusion, image noise, and other influences resulting from the differing appearance of objects in the two views make feature matching difficult. In sequence analysis, where rapid image sampling produces images that change little from one to the next, matching is less problematic. In our approach we take this to the extreme, with continuous (or nearly so) sampling giving us images that vary smoothly between views. The result is a temporal continuity similar to the obvious spatial continuity in a regular image. With temporal continuity, matching of features becomes a simple matter of contour following.

Epipolar-plane image analysis

In our initial implementation of the approach, we chose a restricted camera arrangement, one whose geometry facilitated the analysis considerably. The camera moved along a straight path, acquiring images at fixed spacings, and looking at right angles to its path. Figure 1 shows several frames from a 125-image sequence obtained under these conditions. Figure 2a shows a volume formed by stacking the images together. The front, with (u,v) coordinates, is a regular image—the first in the sequence. The visible side of the volume shows the rightmost column of each image, and gives a misleading impression of also being a normal image. The top looks less imagelike, but is a crucial depiction for our processing.

Figure 2b shows the volume sliced horizontally along its middle, cut away to reveal a pattern somewhat like that at the top of the volume to its left. This pattern, showing the temporal continuity of features, is referred to as an *epipolar-plane image*. Chakravarty, Nichol, and Ono use a similar slicing mechanism for display and manipulation of seismic data.[3] The patterns in their slices reveal strata of differing acoustical properties; in our work the streaks indicate the paths of particular features over time. By following these paths (which must be straight lines for stationary objects, given our camera geometry), we can establish the position of features in the scene; their distance from the camera path depends simply on the slope of the line.

Figure 3 shows feature position estimates from one slice of the data shown in Figure 2. Each path consists of many observations of a particular feature; these overdetermine its estimate, and the statistical covariance gives us a confidence interval. These confidence intervals are depicted in the figure by ellipses. The coordinate system used here is in units of world inches, with the line at the bottom indicating the camera's path through the scene. Figure 4 is a display created for stereoscopic viewing; it depicts the feature estimates combined from all the slices through the data set shown in Figure 2. Figure 5 shows a stereo view of another scene, with colors coding the depth estimates of scene features.

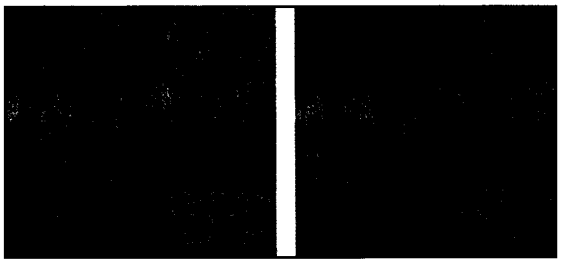


Figure 5. Color-coding of relative depth for another scene.

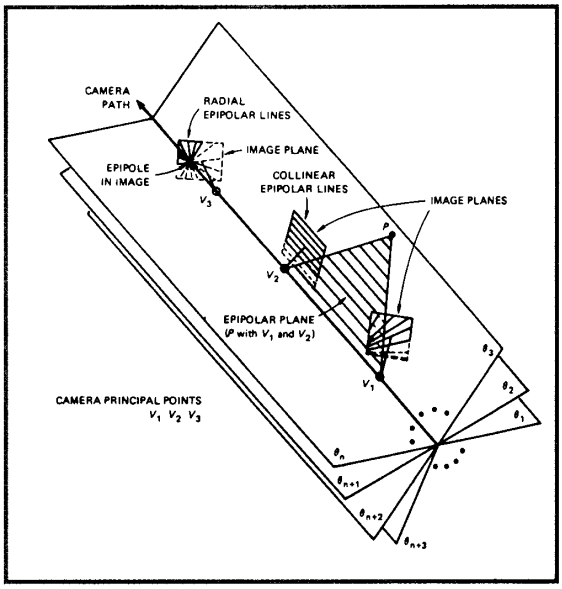


Figure 6. Camera attitudes along a linear path.

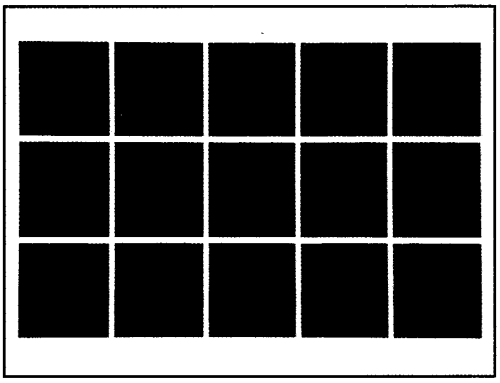


Figure 7. Fifteen frames of a synthesized image sequence.

Generalizing the camera path

The restrictions on this straight-path orthogonal-viewing arrangement limit its applicability in autonomous navigation tasks: One might need to look where one is going, or pan and tilt to track some particular feature. It was essential that we generalize the analysis to allow such flexibility. First we removed the constraints of orthogonal viewing and equally spaced sampling; we wanted the camera to be able to look anywhere and even change its direction while in motion. Our solution also provides for removal of the straight-path restriction, but implementation of this capability still lies ahead. Two other objectives for our generalization were to produce results connected in space (the results shown in Figure 4 are isolated points) and to have the analysis proceed incrementally as the vehicle moves, rather than await acquisition of all the data.

Viewing the geometry of the situation makes it easier to appreciate the problems arising from these generalizations. Consider Figure 6, which depicts three cases of a camera positioned along a straight path. The simplest case is that shown at V_2. Notice that this is the case handled in our earlier implementation—planes were formed by collecting successive scan lines, and feature paths in these planes were linear so estimation was simple. In general the planes are formed by collecting the intersections of the image planes with the pencil of planes (labeled θ_1 through θ_{n+3}) passing through the camera path. Unfortunately, in the situations demonstrated at V_1, which looks to the same side as V_2 but slightly forward, and at V_3, which looks almost straight ahead, the structure of these intersections is radial, and not scan-line based. Even worse, feature paths in the planes will not be straight lines—they will, in fact, be hyperbolic. The most complex situation occurs when the camera is allowed to change its direction while moving, varying say from V_1 to V_3 and positions between. Here the structure of the planes depends on the direction of the camera at each point along its trajectory, and feature paths are neither linear nor hyperbolic, but are arbitrary curves.

Things might have been hopeless; however, an insight from mathematical duality[4] allowed us to keep the estimation as a linear problem for all these situations. Still, for this to work we needed a way to track features through any path in the space-time volume of Figure 2. Resampling was a consideration, but the combination of singularities in the mapping, the computation required, the aliasing that would result, and the disruption of pixel-variance measures made this a very unattractive option. To better enable this tracking and to attain our other two goals of processing the images sequentially and producing spatially coherent results, we were led to develop our 3D surface constructor, the Weaving Wall. Using the Weaving Wall and constraints provided by knowledge of the camera path, we have been able to achieve a robust

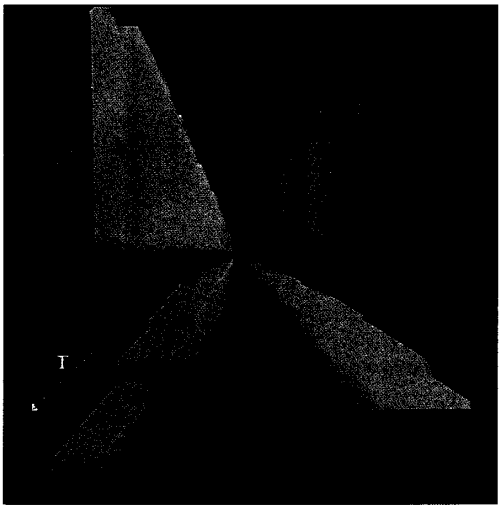


Figure 8. A rendering of surfaces.

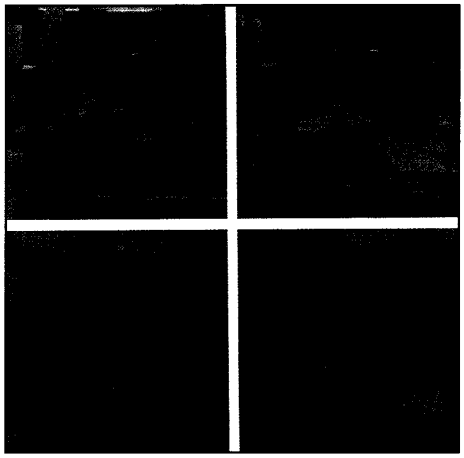


Figure 9. First and last images of zoom sequence. Top: full resolution; bottom: reduced resolution.

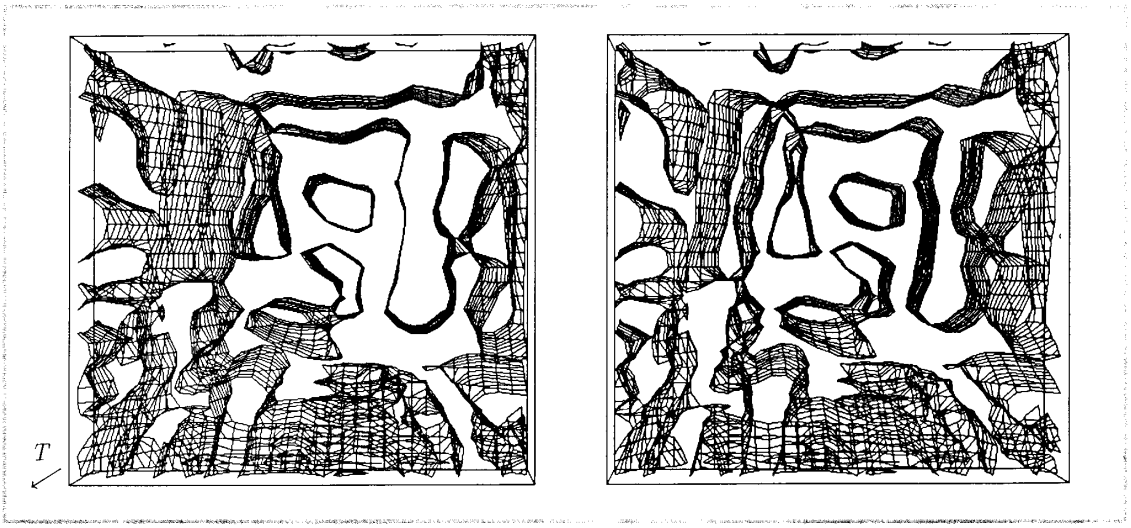


Figure 10. Spatiotemporal surfaces of first 10 frames.

estimation of scene contours for arbitrary viewing directions.

The Weaving Wall

The Weaving Wall operates as images are acquired, knitting together a connected representation of the spatial and temporal evolution of a sequence over time. In effect, it carries out a 3D counterpart of 2D edge detection; rather than detecting edges, however, it detects 3D facets. The process acts as a loom during surface construction, with a wall of accumulators weaving the surface elements together—hence its name. Figure 7 shows a synthesized image sequence: zooming in on a set of rectangles. Figure 8 shows a rendering of the spatiotemporal surfaces arising from these images. Here T_0 is at the rear, T_{14} is in the foreground, and the spatiotemporal evolution is quite apparent.

The top of Figure 9 shows the first and last images obtained with a similar forward camera motion through a scene. For simplicity in our analysis and display we use a reduced-resolution version of this sequence, the first and last images of which are shown at the bottom of Figure 9. Figure 10 is a crossed-eye display of the spatiotem-

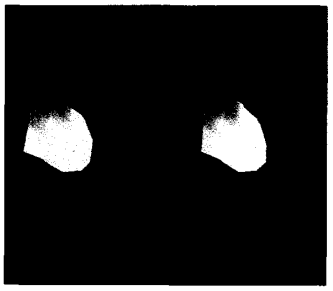


Figure 11. Single rendered surface (orange exterior, light interior).

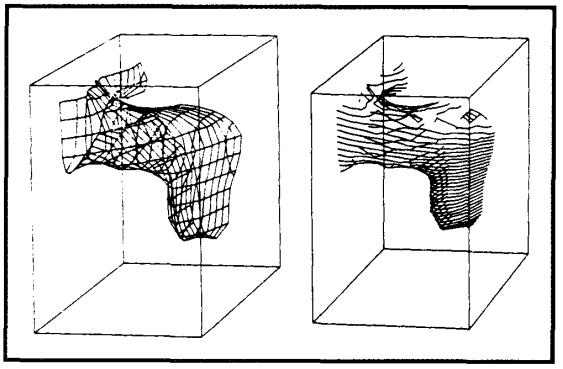


Figure 12. Wireframe of spatiotemporal surface.

Figure 13. Constraint planes intersecting spatiotemporal surface.

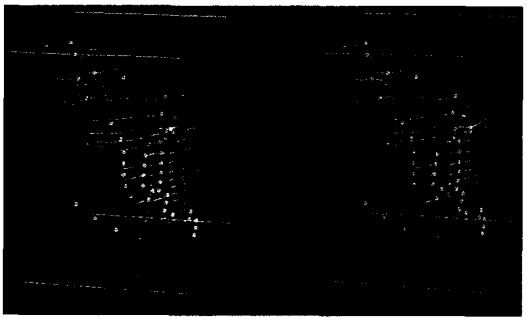


Figure 14. Tracking process in operation.

poral surfaces produced for the first 10 frames of this sequence. Surface facets are determined by convolution of the data with a 3D operator (a Laplacian of a Gaussian—the smoothing Gaussian sets the resolution, while the Laplacian locates gradient extrema). In digital images, edges generally coincide with object boundaries (although they can also be texture boundaries or noise artifacts), and the same applies to our 3D facets. Note in particular the sock-shaped surface at the upper left of Figure 10. Its right is formed by the ladder, its top by the ceiling and lights, and its left by the cabinets against the left wall. These can be seen clearly at the top left of the images in Figure 9. Figure 11 shows this surface rendered in stereo, with its exterior colored orange and its interior light.

Spatiotemporal tracking

Space-time surfaces are formed by the image-plane evolution of scene features. For tracking features on the spatiotemporal surface, we apply the constraints derived from the known camera geometry. These constraints restrict feature movement. Feature trackers on the spatiotemporal surface of Figure 12, for example, are restricted to lines as shown in Figure 13. In fact, this figure shows the intersection of the constraint planes of Figure 6 (the θ's) with the spatiotemporal surface of Figure 12. A tracker is initiated when a surface is first cut by a plane; tracking is carried out by passing the tracker along sequentially from surface element to element.[5] Figure 14 shows the tracking process operating in the vicinity of this surface. The red wire framing indicates the spatiotemporal surfaces, the yellow lines show the tracking of features, red circles encode the initiation of trackers, and magenta circles indicate a tracker termination. When sufficient observations have been made of some feature, an initial estimate of its position is made, and this is coded in the display by a yellow circle. As further images are acquired, estimates of the feature's position and confidence interval are updated using a least-squares sequential estimator (a Kalman filter[6]).

Each of the paths marked in yellow along these surfaces represents a feature in the scene whose position and confidence interval are being estimated. Figure 15 shows the sequential updating of such a feature estimate at selected frames over the nine in which it is being tracked. The feature is first observed at T_0; at T_4 sufficient observations have been made for a reasonable estimate of its position (marked by the cross) and confidence interval (the large ellipse, only part of which fits within the display frame). These estimates are refined as the feature is observed through frame T_8. The horizontal line at the bottom left of three of the frames is the camera path. These figures were made by using a 3D mouse to select and query the state of a tracker on the surface shown in Figure 14.

The positional and confidence updating depicted in Figure 15 occurs along *all* tracked paths simultaneously and, since we have the surface's 3D connectivity explicitly in the representation, we can show the evolution of *contours* in the scene. Figure 16 shows this for a set of 11 scene features adjacent on a spatiotemporal surface. The left frame shows the first estimate of the contour; the succeeding frames show how this estimate evolves as more images are acquired. Estimates of such

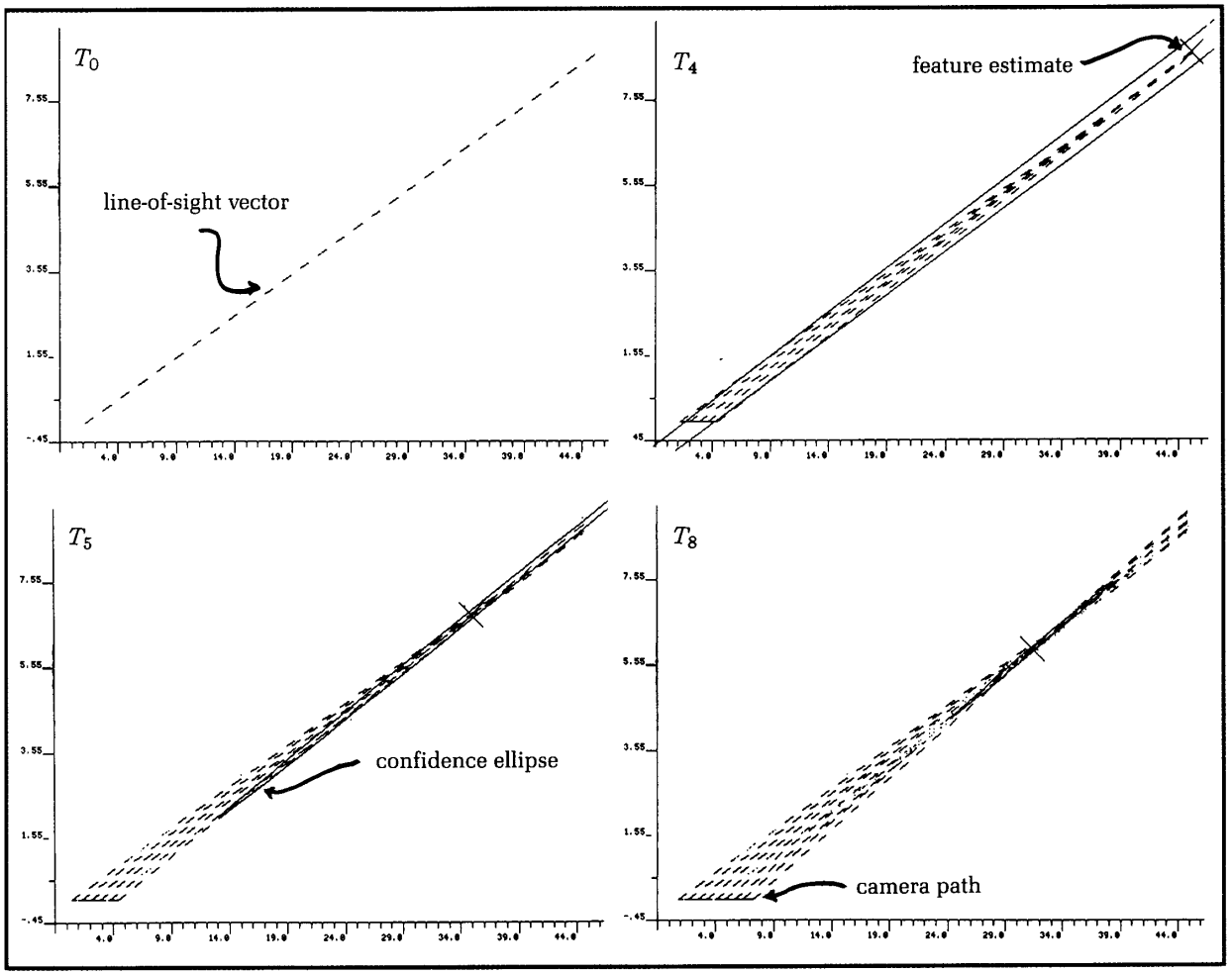
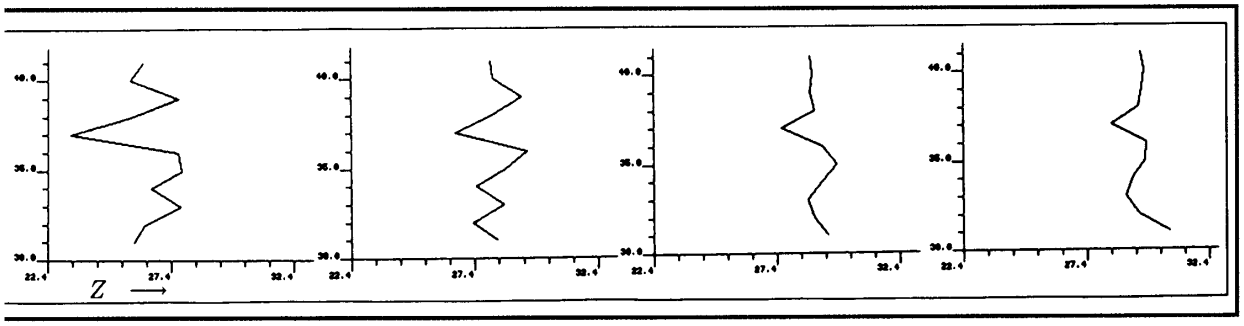


Figure 15. Updating a feature estimate over nine frames.



igure 16. Evolution of feature estimates along a connected contour.

contours are formed and refined over the entire space as the sequence progresses. Our current representation of scene structure is based on these evolving contours.

Through this analysis, we have met our processing objectives:

- There is *no restriction* on the camera's viewing direction during the scan; it can both pan and tilt, locking onto and tracking specific features of interest in the scene as it moves.
- The spatial connectivity maintained in the representation allows us to build 3D *contours*, rather than isolated point sets.
- The estimation happens *sequentially*, with all estimates and confidences updated as each new image is acquired.

Further applications

Some exciting secondary applications resulted from development of the Weaving Wall spatiotemporal surface-building process. Designed to satisfy our tracking needs where the third dimension is time, this process has characteristics that make it useful for other applications in which coherent descriptions of nearly continuous 3D data are sought, regardless of what that third dimension represents. Most obvious among these is the construction of surface models from CT (computed tomography) and other medical scanning technologies (magnetic resonance, ultrasound, tomographic electron-microscopy). Here the third dimension is spatial. In addition, we have been applying the algorithm to modeling material fractures and the visualization of higher dimension analytic functions, the latter of which we discuss in the second subsection below. The critical element for application of the algorithm is that the 2D data evolve gradually over the third dimension, allowing us to track that evolution and represent it as a set of evolving surfaces.

Medical data

The principal approach to surface reconstruction from sensed data in the medical imaging field is that of Artzy, Frieder, and Herman.[7] In that approach, surfaces are built consecutively, with each constructed by a sequential process that begins at a selected seed voxel and traverses the isocontour by means of a connected-component search. Wyvill, McPheeters, and Wyvill improved on the search efficiency by characterizing local surface structure, but maintained the technique's inherent sequential nature.[8] Our process, on the other hand, operates incrementally in the third dimension, creating *all* surfaces simultaneously. This incremental operation was necessary for our tracking work, and is the principal distinction of our process. Although currently implemented to work sequentially within the first two dimensions (as it must on a sequential machine), it could be recoded to perform its processing in parallel over the spatial images, and thereby operate in real time.

Surface element definition is made at a higher resolution than the sampling raster; facets are positioned by interpolation of 3D Laplacian values (or intensities, when these are appropriate). The computation is structured in a way that makes ancillary calculations (for example, cylindrical or line-of-sight transformations, epipolar-plane intersections, triangular tessellation for rendering, and bounding volume determinations) both simple and efficient. The approach grew out of a 2D contour-finding algorithm.[9] In three dimensions, a binary relation (*inside*) is defined over the voxels; this gives rise to 2^8 or 256 voxel combinations in a $2 \times 2 \times 2$ subvolume. These combinations enumerate the various ways the surface (or surfaces) can pass through that subvolume.[10]

We have applied the algorithm to surface reconstruction from CT slice data. Figure 17 shows the evolution of surfaces judged to be bone in a 70×30 window of a 52-image CT data set. This shows the incremental nature of the surface development. Figure 18 is a view of the three major surfaces (jaw, upper teeth, spine) of this data set, individually colored. Since these surfaces are distinct objects, they may be manipulated for many purposes—for example, simulation of kinematics and dynamics, as shown in Figure 19, and structural analysis. Figure 20 shows two stereo views of the spine alone, processed at a higher resolution. A recent surface-rendering algorithm developed at GE Laboratories, Marching Cubes,[11] shares many of these characteristics, but, while producing triangular facets for rendering, does not create connected surface descriptions and does not distinguish surfaces. A by-product of our Weaving Wall's processing is a rendering triangulation similar to that of Marching Cubes, with the advantage that ours handles saddle points correctly.[10]

Figure 21 shows surfaces from another CT data set, this one of 46 slices, each 120 pixels square. Figure 21a shows side and front views of the soft tissue. Figure 21b is a stereo pair of the skull from the side; Figure 21c is a stereo pair showing the cranial vault. Our rendering algorithm uses normals computed from 3D Gaussian derivatives of the data. Interslice spacing is five times the slice resolution, with slice sampling doubling for a section of the data set in the area about the ear. The horizontal band there suggests the patient moved slightly as the scan adjustments were made; the roughness at the jaw resulted from X-ray scattering at metal teeth fillings.

Although the surface display aspects of this technique are evidently quite worthwhile, bear in mind that the primary representation is a surface model, with all the connectivity appropriate for full model-based analysis (for example, symmetry mappings, elastic deformation operations, and computations leading to finite-element analysis). These are computational models built directly from the data.

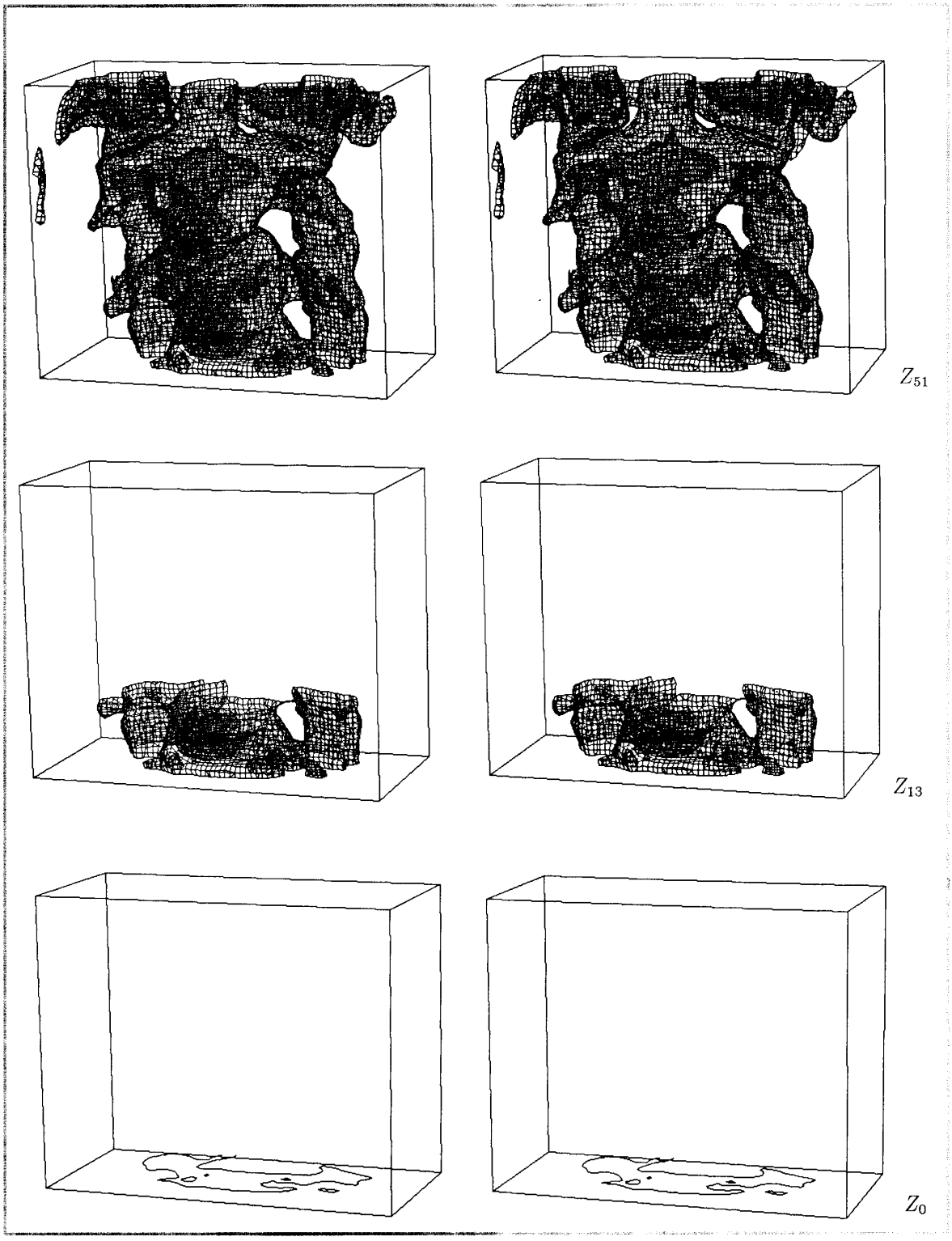


Figure 17. Evolution of surfaces judged to be bone.

Figure 18. Three major surfaces.

Figure 19. Simulation of jaw movement.

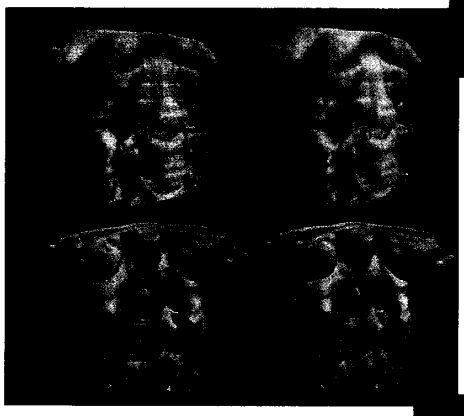


Figure 20. Two stereo views of the spine.

a

b

c

Figure 21. CT data: (a) surface skin, (b) stereo pair of skull side, (c) stereo pair of cranial vault.

Figure 22. Four 3D projections through a four-dimensional object.

Analytic functions

The Weaving Wall has proved useful in object representation studies. Our research group explores problems of representation for object modeling, and has developed a representational facility based on superquadrics.[12] Hanson[13] has developed a hyperquadric generalization permitting the description of shapes with arbitrary polyhedral bounds; superquadrics, in contrast, permit only shapes having the three orthogonal Cartesian bounds. Use of these hyperquadrics enriches the modeling by yielding a superset of the superquadric primitives. Experimenting with these higher dimension objects is complicated by the difficulty in visualizing them. To facilitate this, we display sequences of 3D projections of these n-dimensional objects ($n > 3$), using the Weaving Wall in the rendering. Figure 22 shows a selection of such projections through a four-dimensional surface. Viewing such frames as a sequence gives us insight into the structure of these objects. Banchoff discusses similar issues in assessing the structure of higher dimension functions.[2]

Further potential applications of the surface-building process employed here include representation of surfaces from other 2D sensing domains (e.g., ultrasound, geology), representation of images over scale, and the colorization of black-and-white film. In general, this process can be used in any application requiring a description of the evolution of a 2D pattern varying gradually in a third dimension—be it time, space, viewing position, resolution, or any other dimension. ■

Acknowledgments

This research has been supported by DARPA contracts MDA 903-86-C-0084 and DACA 76-85-C-0004. Bob Bolles, David Marimont, and Lynn Quam have been crucial to its development. Alex Pentland, with his Supersketch modeling and graphics system, was very helpful in tasks of generating simulated data (Figure 7) and rendering surface images. Andy Hanson provided the data and rationale for the hyperquadric analytic surface displays. The medical CT data is courtesy of C. Cuttings, MD, New York University, and CEMAX Corporation, Santa Clara, California. This article has benefited from many suggestions of the editors and reviewers.

References

1. P.J. Besl and R.C. Jain, "Three-Dimensional Object Recognition," *Computing Surveys*, Mar. 1985, pp. 75-145.
2. T.F. Banchoff, "Visualizing Two-Dimensional Phenomena in Four-Dimensional Space: A Computer Graphics Approach," in *Statistical Image Processing and Computer Graphics*, E. Wegman and D. Priest, eds., Marcel Dekker, Inc., New York, 1986, pp. 187-202.
3. I. Chakravarty, B.G. Nichol, and T. Ono, "The Integration of Computer Graphics and Image Processing Techniques for the Display and Manipulation of Geophysical Data," in *Advanced Computer Graphics*, T. Kunii, ed., Springer-Verlag, Tokyo, 1986, pp. 318-334.
4. R.C. Bolles, H.H. Baker, and D.H. Marimont, "Epipolar-Plane Image Analysis: An Approach to Determining Structure from Motion," *Int'l J. Computer Vision*, June 1987, pp. 7-55.
5. H.H. Baker and R.C. Bolles, "Generalizing Epipolar-Plane Image Analysis on the Spatiotemporal Surface." To be published in *Int'l J. Computer Vision*, 1988.
6. Analytic Sciences Corp. technical staff, *Applied Optimal Estimation*, A. Gelb, ed., MIT Press, Cambridge, Mass., 1974.
7. E. Artzy, G. Frieder, and G.T. Herman, "The Theory, Design, Implementation, and Evaluation of a Three-Dimensional Surface Detection Algorithm," *Computer Graphics and Image Processing*, Jan. 1981, pp. 1-24.
8. G. Wyvill, C. McPheeters, and B. Wyvill, "Data Structure for Soft Objects," *The Visual Computer*, Aug. 1986, pp. 227-234.
9. D.H. Marimont, "Segmentation in Acronym," *Proc. DARPA Image Understanding Workshop*, Science Applications Int'l Corp., McLean, Va., 1982, pp. 223-229.
10. H.H. Baker, "Building Surfaces of Evolution: The *Weaving Wall*." To be published in *Int'l J. Computer Vision*, 1988.
11. W.E. Lorensen and H.E. Cline, "Marching Cubes: A High Resolution 3D Surface Construction Algorithm," *Computer Graphics* (Proc. SIGGRAPH), July 1987, pp. 163-169.
12. A.P. Pentland, "Perceptual Organization and the Representation of Natural Form," *Artificial Intelligence*, May 1986, pp. 293-331.
13. A.J. Hanson, "Hyperquadrics: Smoothly Deformable Shapes with Convex Polyhedral Bounds." To be published in *Computer Vision, Graphics, and Image Processing*, 1988.

Harlyn Baker has been a computer scientist in the Artificial Intelligence Center at SRI since 1984 and has been working in computer vision for the past 15 years. From 1974 to 1976 he was a research associate at Edinburgh University, and from 1978 to 1983 he was a research assistant and then research associate at Stanford University.

Baker received his BSc in computer science from the University of Western Ontario, MPhil in Machine Intelligence from Edinburgh University, and PhD in computer science from the University of Illinois at Urbana-Champaign. He is a member of IEEE, a committee member of ISPRS (Commission II), an associate editor for *Image and Vision Computing*, and a member of the editorial board of *The Robotics Review*.

The author can be reached at the Artificial Intelligence Center, EK233, SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025. Electronic mail address: Baker@ai.sri.com.