Photogrammetrische Woche 1987
Institut für Photogrammetrie
Schriftenreihe, Heft 12, 1987

MULTIPLE–IMAGE COMPUTER VISION

H. Harlyn Baker, SRI International, Menlo Park, California.

1. INTRODUCTION

This paper discusses some new directions being taken in image processing for applications of three-dimensional image understanding. Although focussing on two pieces of work from our research laboratory, the general theme I wish to present is that recent increases in computer power, both memory and throughput, have given avenue to new approaches in computational vision, and many of these show promise of solving long-standing problems. The new approaches I will discuss can be characterized as requiring massive amounts of local computation – they are also robust. These approaches address the recovery of three-dimensional scene structure from multiple images, where multiple means more than 1 and upwards of hundreds. The computational demands mean that practical implementation of these techniques will require use of parallel processors. Indeed, before the arrival of Connection Machines[Hillis 85], Transputers[INMOS 85], etc., these approaches, if conceived, would never have been further considered. Taking the fairly radical tack where time and data volume pose no technical obstacle, some interesting approaches have arisen.

The two pieces of research I will discuss are 1) a stochastic approach to stereo analysis using simulated annealing (adapted from statistical mechanics), and 2) a technique for building depth, occlusion, and free-space maps of a scene through image sequence analysis. The first makes the valuable contribution of unifying feature and area-based stereo to yield a full, reliable disparity map of a scene; the second integrates spatial and temporal processing for scene modelling and applications in autonomous navigation. First, the stochastic stereo.

2. A STOCHASTIC APPROACH TO STEREO VISION

Traditional stereo processing divides itself into two categories: area-based correlation and feature-based matching. Neither has proven to be totally acceptable in its functioning. Area-based processing is readily confused by repeated patterns, and, by its very nature of matching only where the signal is adequately similar, loses most where one would probably want it to succeed the most – at depth discontinuities. Many efforts have been made to improve on this situation (see, for example, [Hannah 85] and [Förstner 86]). Feature-based processing is less sensitive to disruption by occlusion or steep depth gradients, but in general yields only sparse results, failing to map areas void of the 'features' being matched. Attempts to integrate the two approaches have not yet proven satisfactory either, often because of the difficulty in assuring correct feature results to use as seeds to constrain the subsequent area-based correlation, and then because of the difficulty in enforcing two-dimensional continuity in the area correlation (see [Baker 82], [Ohta 85]). A new stochastic approach, a simulation of the thermal-bath technique of physical annealing, has demonstrated considerable success at providing what these two approaches either alone or combined have been unable to deliver (see [Barnard 86] for full details).

2-a. Simulated Annealing

Simulated annealing is a computational parallel to the physical process of annealing. There, a system of molecules is raised to an elevated temperature and slowly cooled, staying as close to equilibrium as is possible. In this stochastic simulated annealing process, an image is treated as a lattice, with energy assigned locally by the disparity gradient between adjacent pixels and the local intensity difference between a pixel and the pixel judged to be its correlate. Controlled iteration drives the energy of the lattice to a minimum. At first glance this might seem inappropriate as a computational technique. The process involves upwards of hundreds of iterations over the image pairs, at each point making small probabilistic adjustments to the disparity measures. However, it exhibits very good performance, and returns a full disparity map. It is inherently massively parallel, and maps well to simple SIMD parallel architectures. This means that although inappropriate on today's sequential machines, the technique shows tremendous promise when implemented on the new breed of parallel machines.

Barnard [Barnard 86] has implemented the simulated annealing process for stereo mapping, and has recently revised the approach to use a simpler control mechanism and has significantly improved its throughput by structuring the annealing in a coarse-to-fine hierarchy. For stereo processing, an energy function, which the system is to minimize through adjusting disparity measures, is defined as:

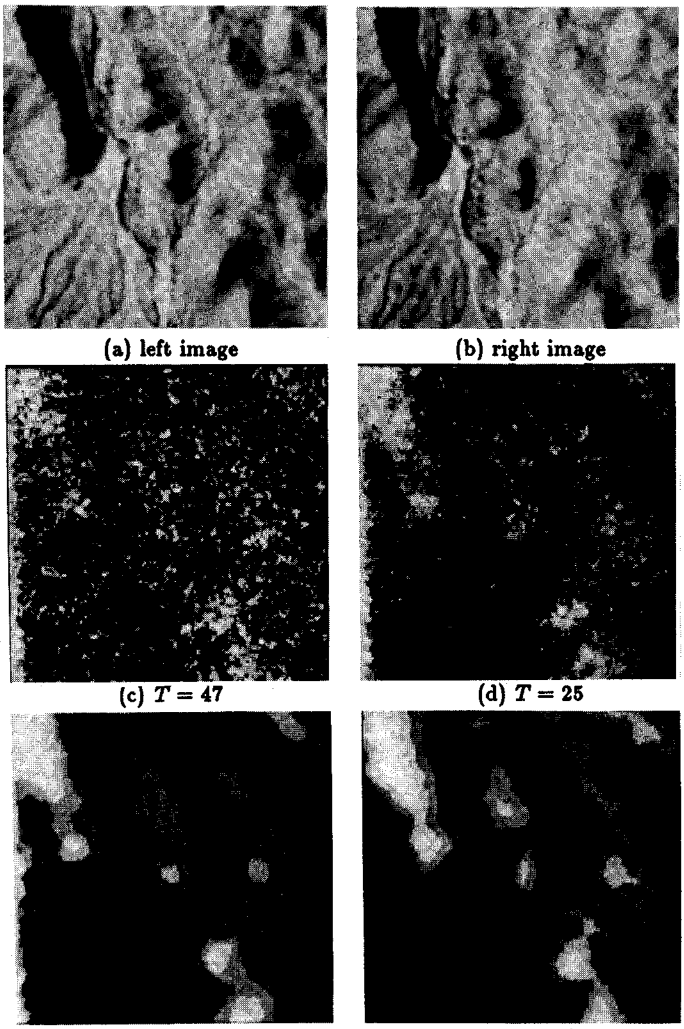
$$E_{ij} = \sum_{i,j}(|\Delta I_{ij}| + \lambda|\nabla D_{ij}|)$$

with $\Delta I_{ij} = I_L(i,j) - I_R(i,j+D_{ij})$, I_L and I_R being the left and right intensity values, and D_{ij} being the associated disparity measure.

The first term seeks to minimize the difference in image intensity between corresponding points, and the second minimizes the local surface disparity gradient. If both reach zero the images are identical (modulo a shift) and the scene is flat. The two components of the energy measure are two of the basic constraints used in stereo mapping. They indicate that matches should have similar local characteristics (here the measure is intensity), and the resulting description should reflect the (generally) smoothly varying nature of the scene. λ balances the two measures.

Starting at a very high temperature in a random state, simulated annealing uses the Metropolis ([Metropolis 53]) algorithm to bring the system to equilibrium. The temperature is then lowered slightly and the procedure repeated until a sufficiently low temperature is attained. If the temperature is lowered too rapidly the system may enter a locally optimal minimum prematurely, and be prevented from rising out of it to reach the desired globally optimal ground state. The simulated annealing process operates as follows:

```
Select a random state $S$.
Select a sufficiently high starting temperature $T$.
while $T > 0$ do
    Make a random state change $S' \leftarrow R(S)$.
    $\Delta E \leftarrow E(S') - E(S)$
    if $\Delta E \leq 0$ then $S \leftarrow S'$          ; Accept lower energy states.
    else
        $P \leftarrow \exp(-\Delta E/T)$
        $x \leftarrow$ random number in $[0,1]$
        if $x < P$ then $S \leftarrow S'$          ; Accept higher-energy states with probability $P$.
    if there has been no significant decrease in E
        for many iterations
    then lower the temperature $T$.
```



(a) left image

(b) right image

(c) $T = 47$

(d) $T = 25$

(e) $T = 0$

(f) ETL disparity map

Fig. 1. ETL results

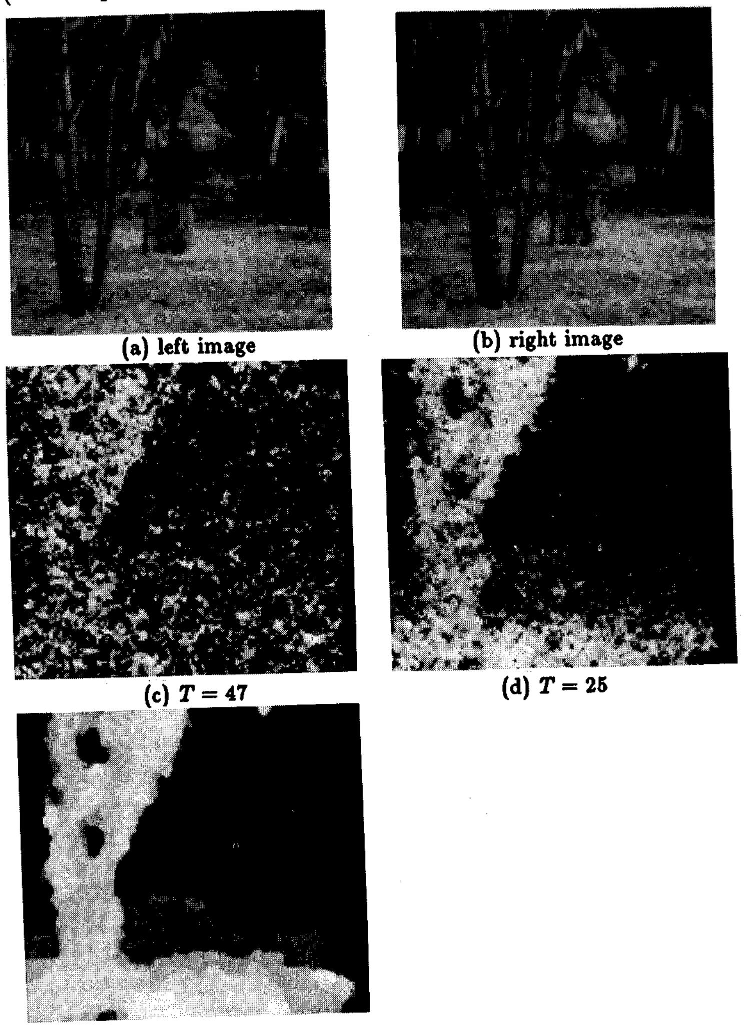
2-b. Matching Results with Simulated Annealing

Figures 1 and 2 show the results of this algorithm on a vertical aerial pair and an oblique ground-level pair. Identical parameters were used for both examples: $\lambda = 5$, $T = 0$, and T was reduced between equilibrium points by 10% until it fell below 1. Ten iterations of the Metropolis algorithm were performed at each temperature.

The first pair, shown in Figure 1, is data provided by the Engineering Topographic Laboratory (ETL). Intermediate results for $T = 47$ and $T = 25$ and the final result for $T = 0$ are shown. In addition, a disparity map supplied by ETL is included for comparison.[1] The stochastic matching algorithm produces a result that is quite similar to the ETL data, although it is somewhat smoother. Some of this difference can be attributed to the fact that the ETL result was produced from higher-resolution imagery. The error on the left border of Figure 1e arises from the images not overlapping completely.

The second stereo example is an oblique view of an outdoor scene, as seen in Figure 2. The result in Figure 2e is certainly plausible – a quantitative disparity model was not available for comparison. The matching algorithm has smoothed over the foreground trees probably more than appropriate, but the disparity at the limb and trunk boundaries is very large. The disparity energy along those lines is similarly large, and Barnard has found that this provides a good measure for scene depth segmentation.

The new stochastic stereo mapping process developed by Barnard and reported in [Barnard 87], termed "microcanonical annealing" (see [Creutz 83]), has two important features. First, the annealing is simulated using an algorithm that is more efficient and more easily controlled than the Metropolis algorithm. Secondly, it uses a hierarchical, coarse-to-fine control structure (see [Marr 77] and [Moravec 81]) employing Laplacian pyramids [Burt 83] of the stereo images. In this way, more quickly computed results at low resolutions are used to initialize the system at higher resolutions. This leads to both greater efficiency and the ability to deal with much larger ranges of disparity. Furthermore, the new approach works with integer arithmetic, making it even more atuned to a simple parallel processor, has less rigorous demand for true random numbers, does not need the exponential function of the Metropolis algorithm, and has been demonstrated to handle much larger problems (a state space larger by a factor of 10^{36000} than the earlier approach) in similar time.



(a) left image

(b) right image

(c) $T = 47$

(d) $T = 25$

(e) $T = 0$

Fig. 2. Tree results.

[1] The ETL disparity map was made with an interactive digital correlation device that depended on a human operator to detect and correct errors. The disparity map in Figure 1f has been sampled from a larger map compiled from higher-resolution imagery.

2-c. Suitability of the Computation

This thermodynamic model of optimization involves huge numbers of rather simple elements interacting over short ranges. Consequently, its main disadvantage is that it is grossly inefficient when implemented on conventional uniprocessor systems. The poor match of algorithm to processor architecture leads not only to extreme impracticality for any application implementation, but makes experimentation severely cumbersome, tending to focus work on smaller problems and limiting the insights possible from system development over large sample sets. The computation time for each of the examples above was about 12 hours on a Symbolics 3600 Lisp Machine. Since all nodes of the lattice are executing the same simple program, simulated annealing can be efficiently implemented on an SIMD machine. The Connection Machine, a fine-grained SIMD parallel processor, has an ideal architecture for this class of algorithm. A one-day programming effort on the Connection Machine gave a speedup of 100 times. A serious implementation effort could be expected to bring another order of magnitude to this figure.

But the results (and the videotape) speak for themselves, and show that a technique inconceivable by traditional standards has great utility when mapped onto a machine with the appropriate architecture.

3. IMAGE SEQUENCE ANALYSIS

Our second example of a novel, though computationally expensive, depth reconstruction process is Epipolar Plane Image Analysis (see [Baker 86] and [Bolles 87] for a description of an early version). This technique involves the processing of a very large number of images acquired by a moving camera. The analysis is based on three constraints:

1. the camera's movement is restricted to lie along a linear path;
2. the camera's position and attitude at each imaging site are known;
3. image capture is rapid enough with respect to camera movement and scene scale to ensure that the data is, in general, temporally continuous.

This approach bridges the usual dichotomy of depth sensing in that its large number of images leads to a large baseline and thus high accuracy, while the rapid image sampling gives minimal change from frame to frame, eliminating the correspondence problem. Rather than choosing quite disparate views and then working on stereo matching, with this technique one chooses to process massive amounts of similar data, but with much simpler and more robust techniques. The analysis has two characteristics which make it suitable for rapid parallel or distributed processing:

1. only local support (both spatially and temporally) is required for the tracking and depth computations;
2. the processing occurs incrementally in time, as the camera moves.

We will expand on these.

3-a. Geometry of the Temporal Sampling

Within this framework, we generalize from the traditional notion of epipolar *lines* to the idea of epipolar *planes* – a set of epipolar lines sharing the property of transitivity. We formulate a tracking process which exploits this property for determining the position of features in the scene. Critical to visualizing the approach is an understanding of the geometry of the sensing situation.

The camera is modelled as a pin-hole with image plane in front of the lens (Figure 3). For each feature P in the scene and two viewing positions V_1 and V_2, there is an *epipolar plane*, which passes through P and the line joining the two lens centers. This plane intersects[2] the two image planes along corresponding *epipolar lines*. An *epipole* is the intersection of an image plane with the line joining the lens centers. In motion analysis, an epipole is often referred to as the focus of expansion (FOE) because the epipolar lines radiate from it. In this work, the camera moves in a straight line, and the lens centers at the various viewing positions lie along this line. Here, the FOE is the camera path. This structuring divides the scene into a pencil of planes passing through the camera path. We view this as a cylindrical coordinate system with axis the camera path, angle defined by the epipolar plane, and radius the distance of the feature from the axis. Note that a scene feature is restricted to a single epipolar plane, and any scene features at the same angle (within the discretization) share that epipolar plane. This means that the analysis of a scene can be partitioned into a set of analyses, one for each EPI, and these EPIs can be processed independently. As will be seen, this structuring allows simultaneous processing of epipolar planes.

[2] Here, intersection and projection are equivalent.

Figure 4 shows a simple motion with a camera moving orthogonal to its direction of view. This corresponds to the situation depicted by V_2 of Figure 3. Here, the epipolar lines for a feature, such as P, are horizontal scanlines, and these occur at the same vertical position (scanline) in all the images. This is the camera geometry normally chosen for computer stereo vision work. Each scanline is a projected[2] observation of the features in the epipolar plane. The projection of P onto these epipolar lines moves to the right as the camera moves to the left. If one were to take a single epipolar line (scanline) from each of a series of images obtained with this camera geometry and compose a spatiotemporal image (horizontal being spatial and vertical being temporal), one would see a pattern as in Figure 5. For this type of motion feature trajectories are straight lines, as can be seen. If the camera were moving with an attitude as shown at V_3 in Figure 3, the set of epipolar lines would form a pattern as shown in Figure 6. For this type of motion feature trajectories are hyperbolas. Allowing the camera to vary its attitude along the path gives rise to spatiotemporal images (the set of corresponding epipolar lines) as shown in Figure 7. These are neither linear nor hyperbolic, and in fact are arbitrary curves.

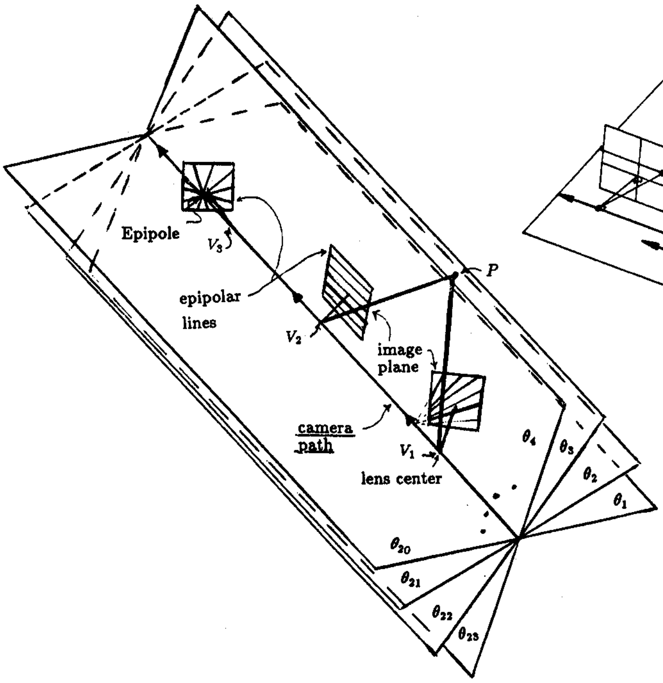


Fig. 3. General epipolar configuration.

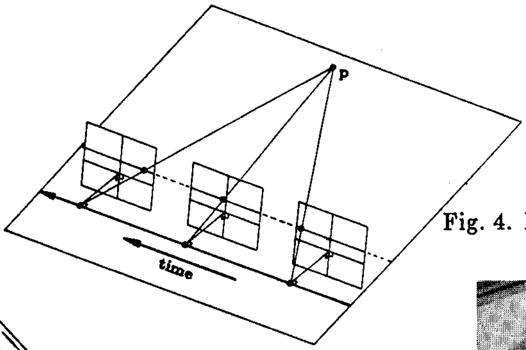


Fig. 4. Right-to-left motion.

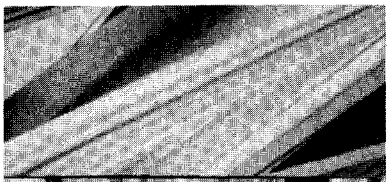


Fig. 5. Orthogonal viewing.

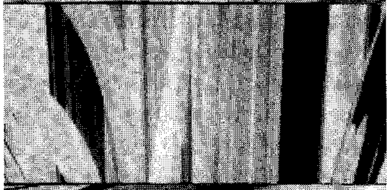


Fig. 6. Fixed, non-orthogonal viewing.

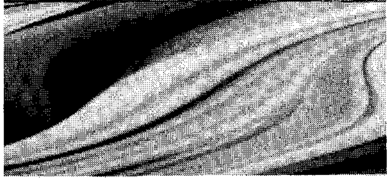


Fig. 7. Viewing direction varying.

3-b. Linear Tracking

The aim is to determine the position of observed features by tracking their appearance in these epipolar planes. Obviously in the case of orthogonal viewing (i.e., as at V_2), the tracking is linear. For general camera attitudes, including varying, it is non-linear. Computational considerations make it extremely advantageous for the tracking to be posed as a linear problem. To maintain the linearity regardless of viewing direction we find not linear feature paths in the EPIs (Figures 5 through 7), but linear paths in a *dual space*. Our insight here (see [Marimont 86a] and [Marimont 86b]) is that no matter where a camera roams about a scene, for any particular feature, the *lines of sight* from the camera principal point through that feature in space, determined by the line from the principal point through the point in the image plane where the projected feature is observed, all intersect at the feature (modulo the measurement error). The duals of these lines of sight lie along a line whose dual is the scene point (Figure 8): fitting a point to the lines of sight is a linear problem. This, then, gives us a metric for linear tracking of features: we map feature image coordinates to lines of sight, and find the point which minimizes, in a weighted-least-squares sense, the error from those various lines of sight.

We must, however, have a mechanism for extracting the observations of features from the individual images in which they occur, and grouping them by epipolar plane.[3] We could transform the images from the Cartesian space in which they are sampled to an epipolar representation (see [Baker 83] and [Jain 87]). Because of aliasing effects and non-linearities in the mapping, we prefer to avoid this. Probably the best solution would be to use a sensor which delivers the data in this form (a spherical sensor having meridian scanning would accomplish this (see also [Gibson 50])). Such a sensor not yet being available, we choose to transform the features we detect in image space to the desired epipolar space (the cylindrical coordinate system of Figure 3). The structure for implementing this brings us several other advantages, as the next section describes.

[3] Only in the simple case of viewing angle orthogonal to the motion is this grouping trivial (Figure 5).

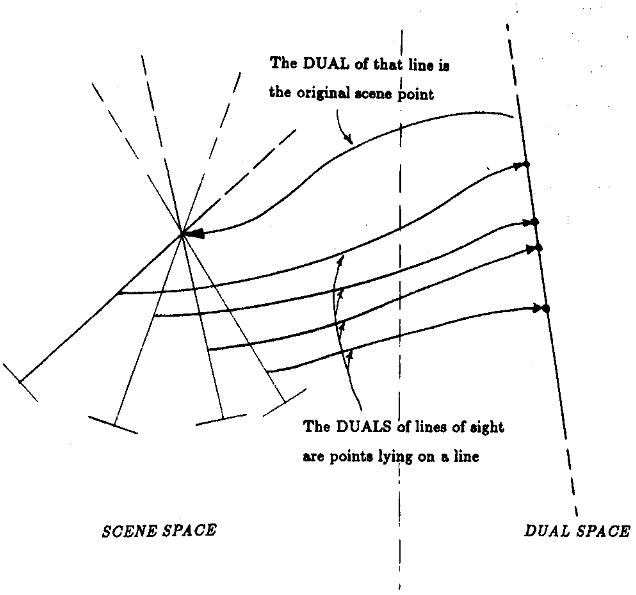


Fig. 8. Line of sight duality.

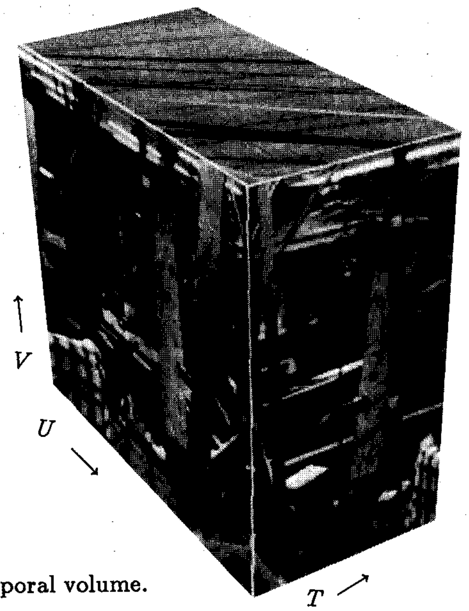


Fig. 9. Spatiotemporal volume.

3-c. Feature Tracking on the Spatiotemporal Surface

We collect the data as a sequence of images, in fact stacking them up as they are acquired into a spatiotemporal volume as shown in Figure 9. As each new image is obtained, we construct its *spatial* and *temporal* edge contours. These contours are three-dimensional zeros of the Laplacian of a chosen three-dimensional Gaussian[4] (LoG), and the construction produces a spatiotemporal *surface* enveloping the signed *volumes* (note that in two dimensions edge contours envelop signed *regions*). The *spatial* connectivity in this structure lets us explicitly maintain object coherence between features observed on separate epipolar planes; the *temporal* connectivity gives us, as before, the tracking of features over time. We can use the spatial connectivity *after* processing, to connect feature estimates.

The need for this connectivity can be demonstrated by observing our earlier results ([Baker 86]), shown in Figure 10. There, in processing the EPIs separately, we obtained separate planes of results. We used proximity of the resulting estimates on adjacent planes to filter outliers (isolated feature estimates unconnected to other estimates). These results were sparse, and overly fragmented due to the proximity measure used (overlapping error ellipses derived from the covariance matrices). But the problem lay not with the filtering, but with the loss of spatial connectivity in the first place. Our separation of the data into EPIs and then subsequent independent processing of them lost the spatial connectivity apparent in the original images. We maintained instead the temporal connectivity. For spatial connectivity in the scene reconstruction, spatial connectivity in the imagery must be preserved.

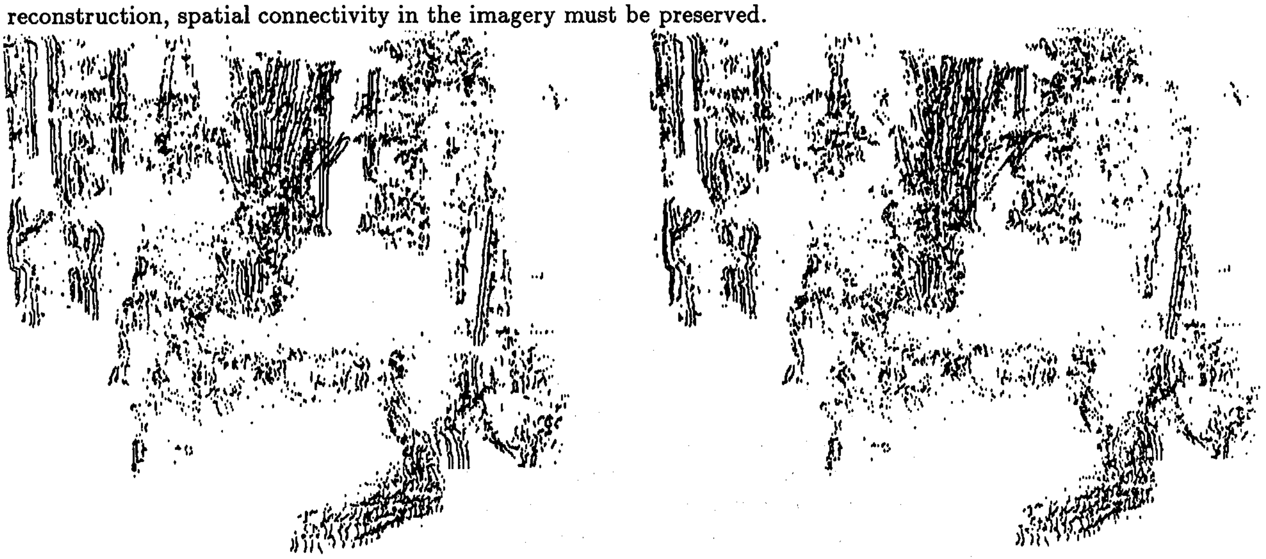


Fig. 10. Crossed-eye display of lateral-viewing results.

[4] See also [Buxton 83] and [Heeger 86] for their use of spatiotemporal convolution over an image sequence.

In this spatiotemporal surface description, feature observations bear (u, v, t) coordinates, and are spatiotemporal ***voxel facets.*** Figure 12 shows a mesh description of the facets for the spatiotemporal surfaces associated with the forward-viewing sequence whose first and last images are depicted in Figure 11a. The surface representations shown in the remaining figures are based, for clarity and development, on a reduced version of the imagery – one eighth the linear resolution of the originals. Figure 11b shows these two frames at the reduced resolution.

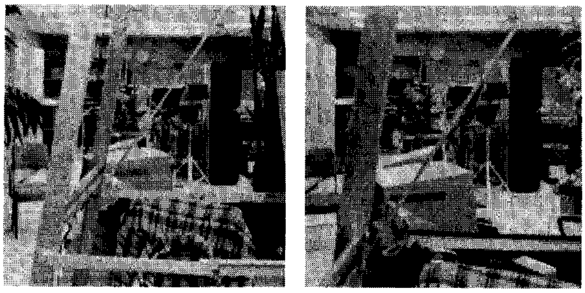


Fig. 11a. First and last images of forward viewing sequence.

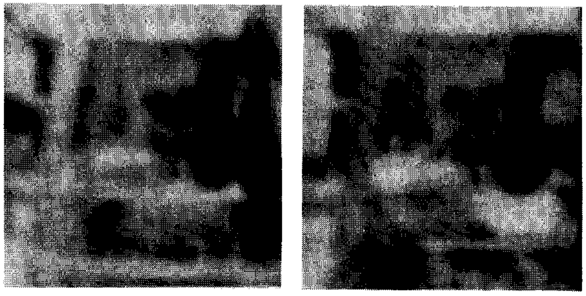


Fig. 11b. First and last images at one-eighth linear resolution.

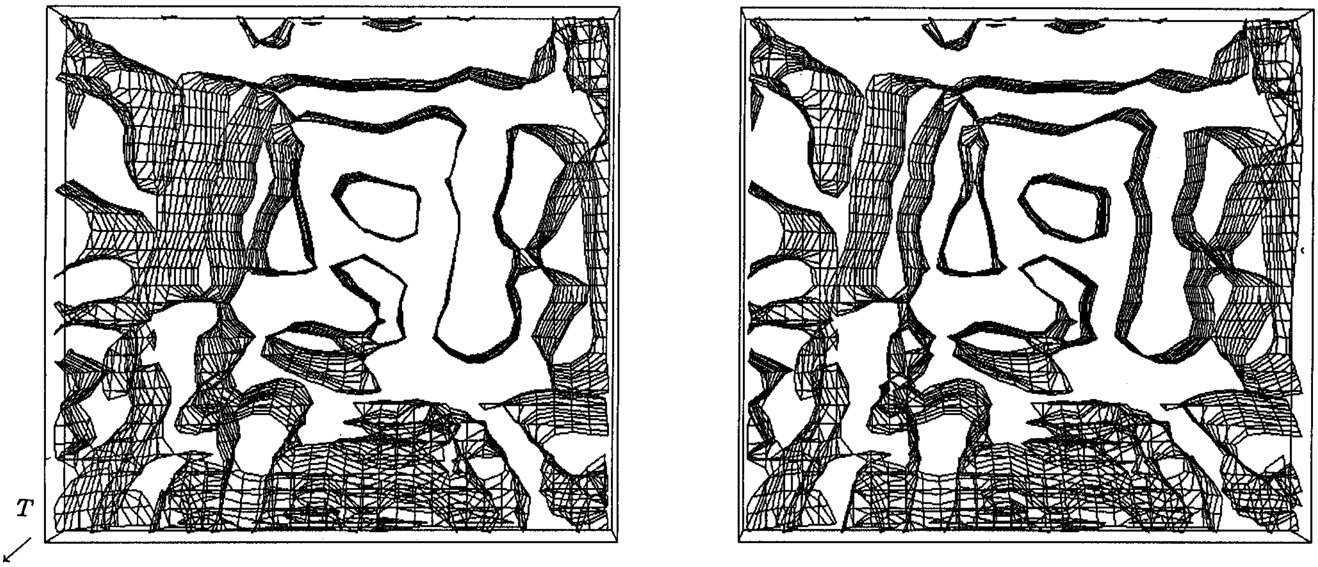


Fig. 12. Spatiotemporal surface representation for the first 10 frames (crossed-eye display).

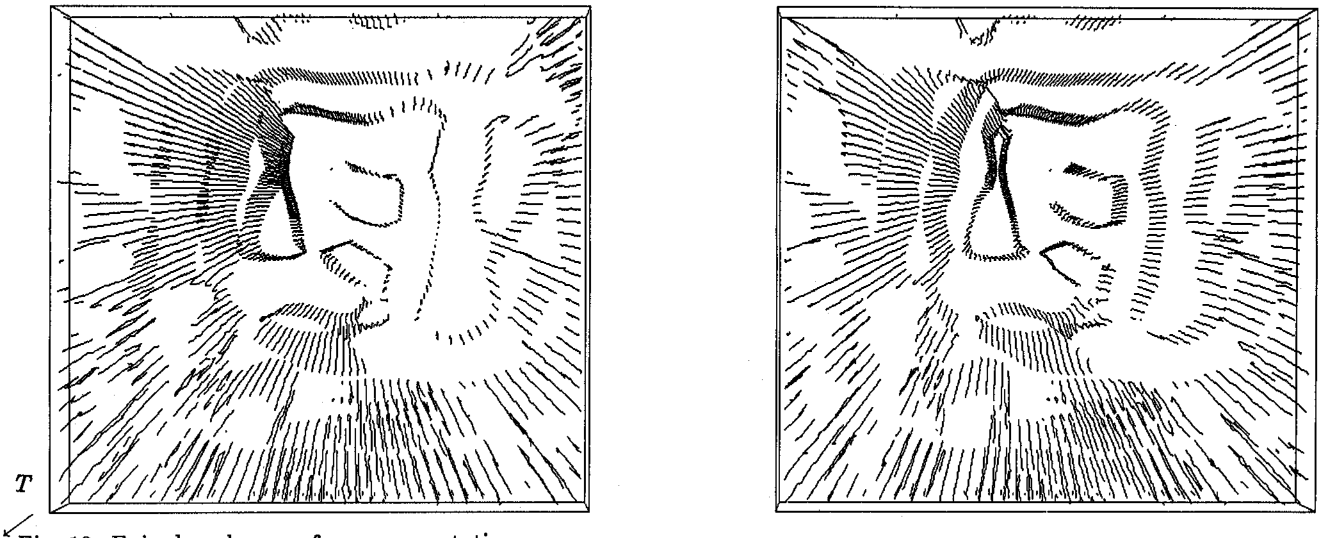


Fig. 13. Epipolar-plane surface representation.

As mentioned in the previous section, for non-orthogonal viewing directions, epipolar lines are not distinguished by the spatial v coordinate. To obtain this necessary structuring we develop within this spatiotemporal description an *embedded* representation that makes the epipolar organization explicit. Over each of the sequential images, we transform the (u, v, t) coordinates of our spatiotemporal zeros to (r, h, θ) *cylindrical coordinates* (θ indicates the epipolar-plane angle ($\theta \in [0, 2\pi]$), the quantized resolution in θ is a supplied parameter, and the transform for each image is determined by the particular camera parameters). In this new coordinate system, we build a structure similar to our earlier EPI edge contours, but dynamically organized by epipolar plane. This is done by ***intersecting*** the spatiotemporal surfaces with the pencil of appropriate epipolar planes[5] (as Figure 3). We weave the epipolar connectivity through the spatiotemporal volume, following the known camera viewing direction changes. Figure 13 shows a sampling of the spatiotemporal surfaces as they intersect the pencil of epipolar planes (every fifth plane is depicted). You will notice the obvious radial flow pattern away from the epipole (FOE). Figure 14 shows seven of these surface/plane intersections, along with the associated bounding planes (refer to Figure 3). The edge that all share is the camera path (the epipole). Figure 15 isolates a single surface from the top left of Figure 12, and shows it's spatiotemporal structure. Figure 16 shows the same surface structured by its epipolar-plane components.

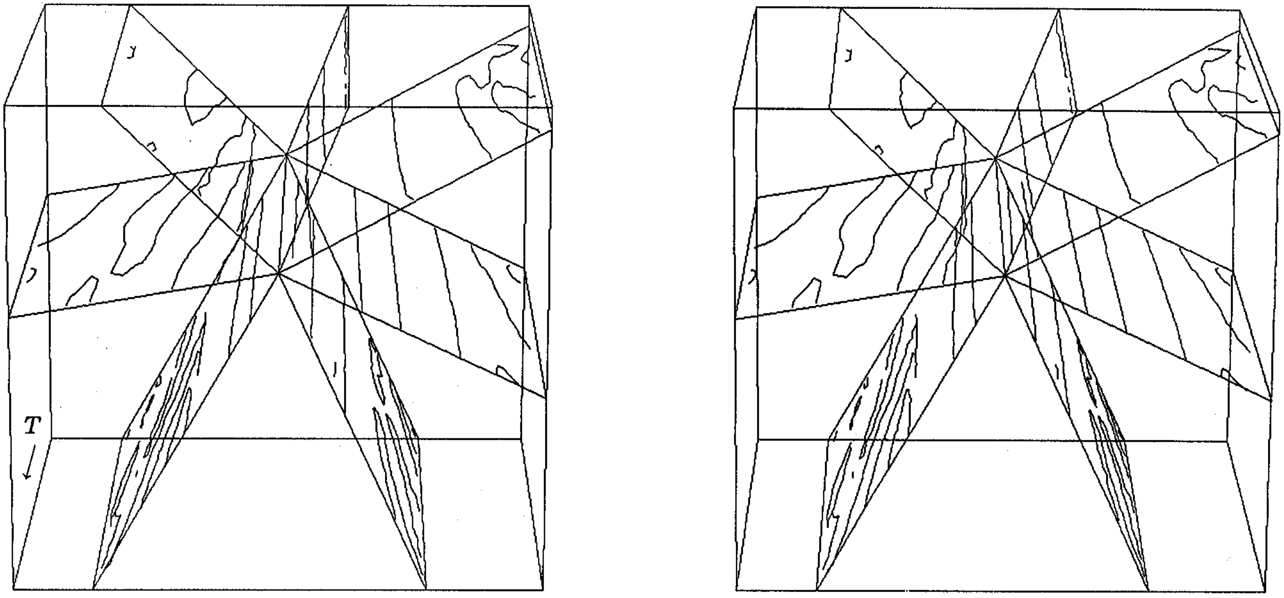


Fig. 14. Intersection of 7 epipolar planes with spatiotemporal surfaces (30 frames).

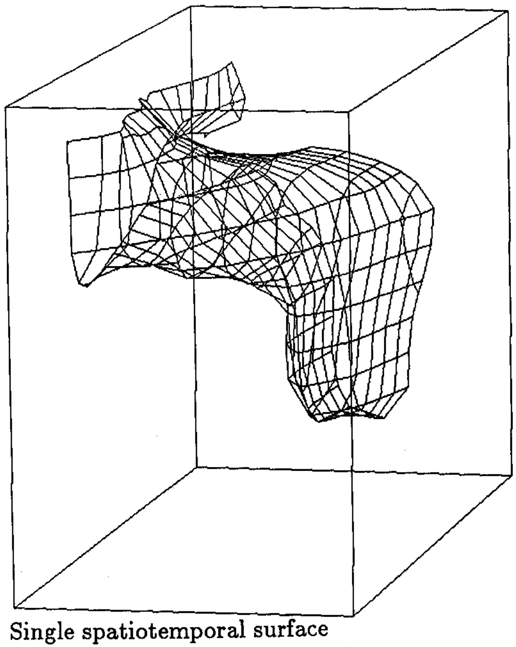


Fig. 15. Single spatiotemporal surface from top left of Figure 12.

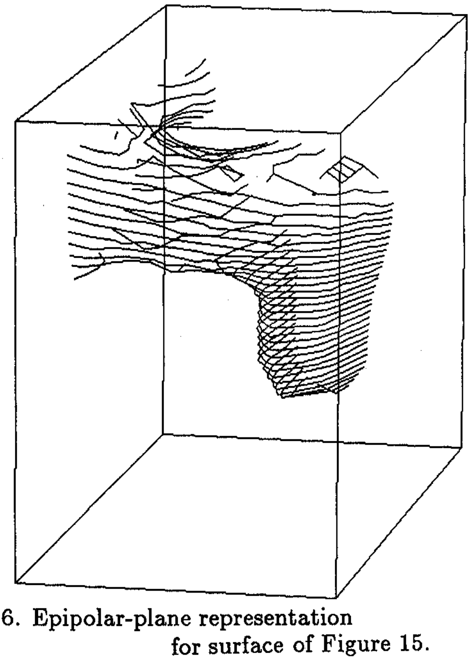


Fig. 16. Epipolar-plane representation for surface of Figure 15.

[5] Notice that, with view direction varying, the underlying epipolar plane may be far from planar in (u, v, t) space – it may undulate in a manner similar to that in which Figure 7 varies from Figure 5, and for similar reasons.

In Figure 17 we show the tracking of scene features on the spatiotemporal surfaces in the vicinity of this surface, with the final pair showing, in crossed-eye stereo form, the result of the tracking after 10 frames. The coding is as follows: initiation of a feature tracking is marked by a circle; the leading observation of a feature (active front) is shown as an x; lines join feature observations; 5 observations (an arbitrary number, greater than 2 may be sufficient) must be acquired before an estimate is made of the feature's position – at that point an initial batch estimate is made, and a Kalman filter [Gelb 74] is turned on and associated with the feature[6] – this initiation of a Kalman filter is coded by a square; where two observations merge, the tracking is stopped and the features are entered into the database – this is coded by a diamond.

Lines of sight are represented in the epipolar plane by the homogeneous line equation $ax + by - c = 0$. For the initial batch estimation, the coordinates (X) of the feature are the solution of the normal equations for the weighted least squares system: $\mathbf{X} = (\mathbf{H}^{\mathbf{T}}\mathbf{W}\mathbf{H})^{-1}\mathbf{H}^{\mathbf{T}}\mathbf{W}\mathbf{C}$, where $\mathbf{H}$ is the $m \times 2$ matrix of (a_i, b_i) observations, $\mathbf{C}$ is the vector of c_i, and $\mathbf{W}$ is the diagonal matrix of observation weights, determined by the distance from the camera to the observed feature at observation position i. We estimate $\mathbf{X}$ first without weights, then compute the weighted solution and the desired covariance matrix, $\mathbf{V}$. Given a current estimate $\mathbf{X}_{i-1}$ and covariance $\mathbf{V}_{i-1}$, the Kalman filter at observation i updates these as:

$$\begin{aligned}
\mathbf{K}_i &= \mathbf{V}_{i-1}\mathbf{H}_i^{\mathbf{T}}[\mathbf{H}_i\mathbf{V}_{i-1}\mathbf{H}_i^{\mathbf{T}} + \mathbf{w}_i]^{-1} \\
\mathbf{V}_i &= [\mathbf{I} - \mathbf{K}_i\mathbf{H}_i]\mathbf{V}_{i-1} \\
\mathbf{X}_i &= \mathbf{X}_{i-1} + \mathbf{K}_i[\mathbf{c}_i - \mathbf{H}_i\mathbf{X}_{i-1}]
\end{aligned}$$

$\mathbf{K}_i$ is the 2×2 Kalman gain matrix, $\mathbf{w}_i$ is the observation weight, a scalar, based on the distance from the camera at observation position i to $\mathbf{X}_{i-1}$.

The tracking of an individual feature is depicted in Figure 18. The camera path runs across the figure from the lower left. Lines of sight are shown from the camera path to the observation of the feature at the upper right. As the Kalman filter is begun (T_4), an estimate (marked by an x) and confidence interval (the ellipse) are produced. As further observations are acquired, the estimate and confidence interval are refined. Tracking continues until the error term begins to increase, suggesting that observations not related to the tracked feature are beginning to be included[7], or until the feature is lost. Note that although a single feature is depicted here, it is part of a spatiotemporal surface, and we have explicit knowledge of those other features to which it is spatially adjacent. We obtain estimates for all such features in the scene. Hopefully by September we will have developed a way to adequately display the ensemble of results.

3-d. Prospect of More Generality

A crucial constraint of the current epipolar-plane image analysis is that having a camera moving along a linear path enables us to divide the analysis into planes, in fact the pencil of planes of Figure 3 passing through the camera path. With this, we are assured that a feature will be viewed in just a single one of these planes, and its motion over time will be confined to that plane. Another crucial constraint is the one we generalized from the orthogonal viewing case – we know that the set of line-of-sight vectors from camera to feature over time will all intersect at that feature, and determining that feature's position is a linear problem. This latter constraint does not depend upon the linear-path constraint. In fact, the problem would remain linear even if the camera meandered in three dimensions all over the scene. This knowledge gives us a possibility of removing the restriction that the camera path be linear. All that the linear path guarantees is that the problem is divisible into epipolar planes. If we lose this constraint, then we cannot restrict our feature tracking to separate planes. The features will, however, still form linear paths in the space of line-of-sight vectors, and our spatiotemporal surface description is an appropriate representation for doing this non-planar, but still linear, tracking. The motion of features will give us *ruled* surfaces, with the rules (zeros of gaussian curvature) revealing the positions of the features in space[8]. This generality suggests that there is even broader application for the technique than we had initially thought[9]. It is also worth noticing that, when the camera attitude and position parameters are not provided, the spatiotemporal surface contains everything that is necessary for determining them; but this is another problem.

[6] [Dickmanns 87] reports on a vehicle navigation controller that similarly works sequentially, utilizing Kalman filters for estimating motion parameters of a small number of features, and is carried out on a parallel processor.

[7] or the zero-crossing is erroneous, or the feature is not stationary, or the feature is a contour rather than a single point in space, or ...

[8] Visualize pick-up-sticks jammed in a box, with the sticks being the rules.

[9] Although being computationally expensive, it would be possible here to use the pairwise epipolar constraints between images to constrain rule tracking on the spatiotemporal surface – notice that they will not have the transitivity property cited earlier.

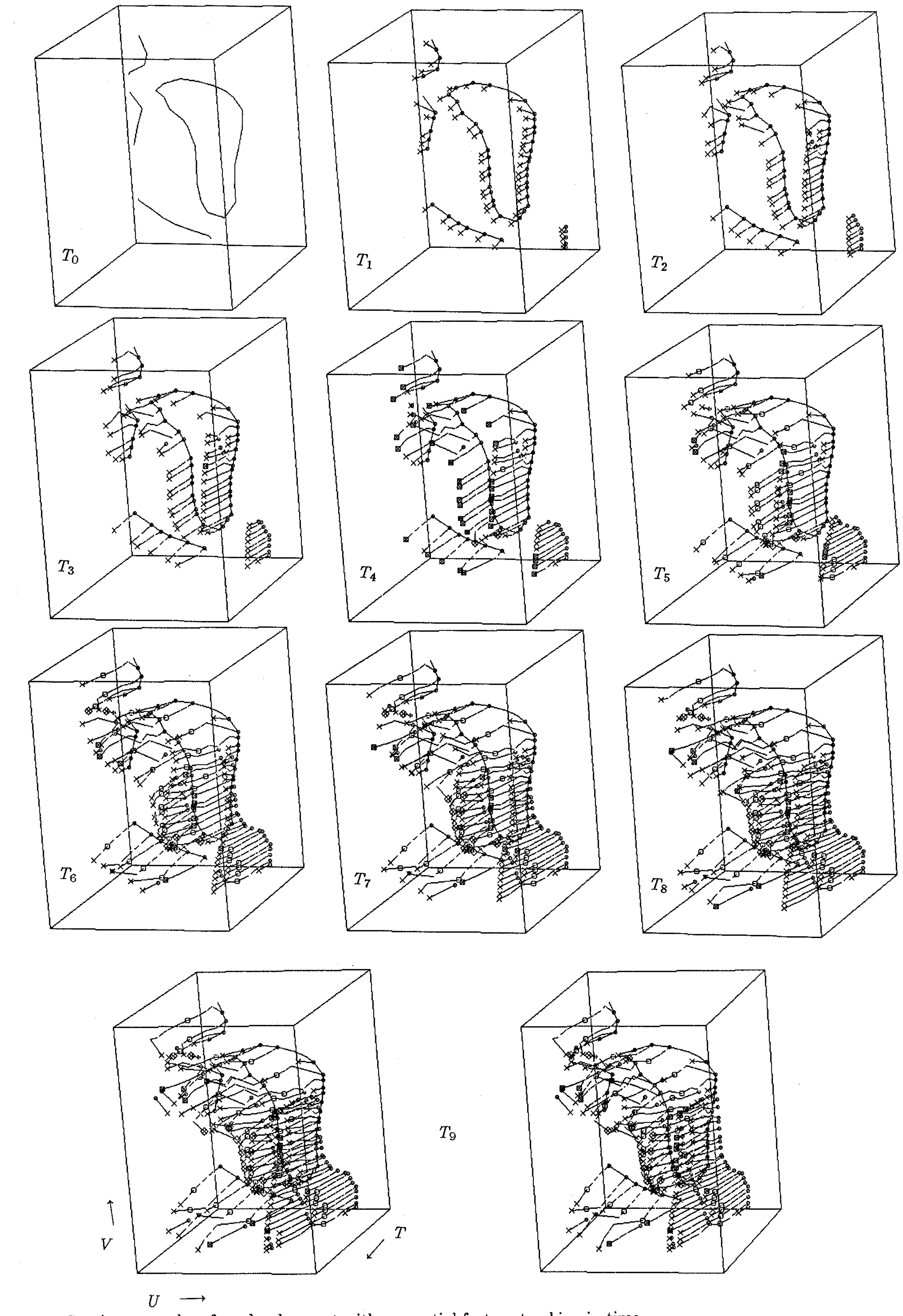


Fig. 17. Spatiotemporal surface development with sequential feature tracking in time.

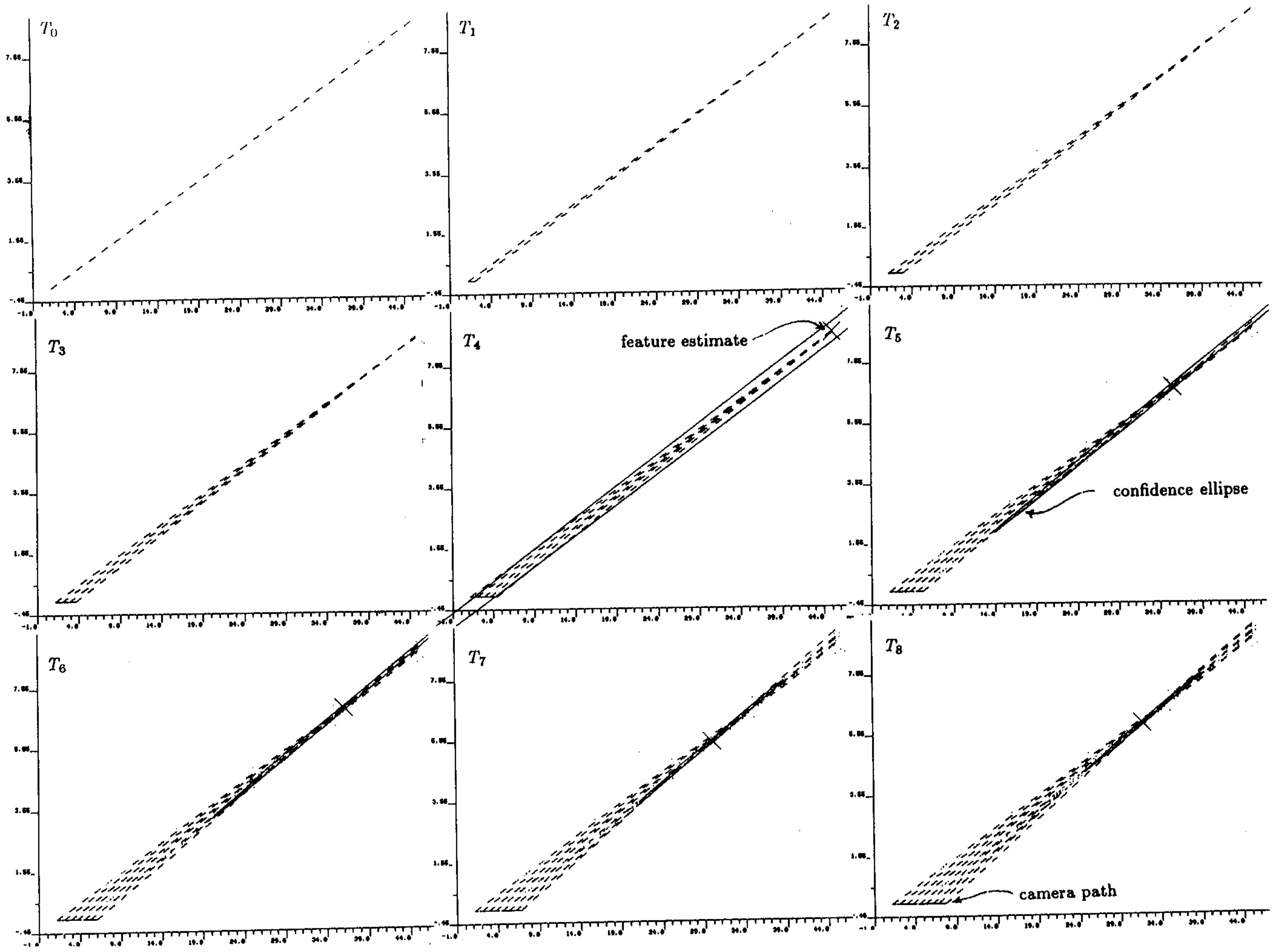


Fig. 18. Sequential estimation.

3-e. Conclusions

We showed, in our earlier work, the feasibility of extracting scene depth information through ***Epipolar-Plane Image Analysis***. Our theory applies for any motion where the camera (lens center) moves in a straight line, with the earlier implementation covering the special case of viewing direction orthogonal to the camera path. The generalizations obtained through spatiotemporal surface analysis bring us the advantages of:

- incremental analysis;
- unrestricted viewing direction (including direction varying along the path);
- spatial coherence in our results, providing connected surface information for scene objects, rather than point estimates structured by epipolar plane;
- the possibility of removing the restriction that fixes us to a linear path.

Bear in mind that the tracking of Figure 18 is occurring simultaneously for all features in the scene. Figure 17 showed a $6 \times 6 \times 10$ subvolume of this data. Visualize the $256 \times 240 \times 128$ data set that comprises the full sequence at full resolution, and you will appreciate both the quality of the processing, and the enormity of its computational load. The crucial understanding when judging the technique's potential is that everything being computed uses only local information in the volume, and is performed sequentially as images are acquired.

4. MULTIPLE-IMAGE COMPUTER VISION: Summary

I have described two novel approaches to depth determination in computer vision – both offering a great deal of promise for three-dimensional mapping. They are distinguished from earlier efforts in that they trade complex processing involving search for simple processing involving massive amounts of local computation. The simpler processing brings the advantages of design simplicity and operational robustness, while necessitating a departure from the traditional computing paradigm of uniprocessor machines. Others are involved in similar transformation of computing approach, and, given the demands for realtime visual processing, this change from sequential sophistication to parallel simplicity will probably be essential.

The stochastic stereo has obvious applicability to the mapping problems of the photogrammetric community; it is not so apparent how the multi-image sequence analysis will impact here. But this work has been developed primarily with autonomous navigation in mind, and navigation requires maps. A process which can deliver robust and accurate spatial estimates for a freely flying vehicle will in fact be building a map as it is progressing – and it could equally be building one for later navigation or assessment. This suggests a fairly radical departure from traditional two-view stereo photogrammetry. The difficult problem of matching is gone. For this, we pay the price in computing cycles, but obtain the benefits of precise accuracy estimates, insensitivity to occlusions, immediate freespace provision (basically gratis), and coherent connected descriptions of the scene's components. Future developments may even add to this list. We solve a difficult mapping problem through a massively redundant analysis (when compared with two-view stereo), and the massively redundant parallel architectures being developed today are the most appropriate for folding the computation back for realtime performance.

REFERENCES

[Baker 82] H. Harlyn Baker and Thomas O. Binford, "A System for Automated Stereo Mapping," *Proceedings ISPRS Commission II Symposium on Advances in Instrumentation for Processing and Analysis of Photogrammetric and Remotely Sensed Data*, Ottawa, Canada, August 1982, 156–171.

[Baker 83] H. Harlyn Baker, Thomas O. Binford, Jitendra Malik, and Jean-Fredric Meller, "Progress in Stereo Mapping," *Proceedings DARPA Image Understanding Workshop,* Arlington, Virginia, June 1983, 327–335.

[Baker 86] H. Harlyn Baker, Robert C. Bolles, and David H. Marimont, "A New Technique for Obtaining Depth Information from a Moving Sensor," *Proceedings of the ISPRS Commission II Symposium on Photogrammetric and Remote Sensing Systems for Data Processing and Analysis*, Baltimore, Maryland, May 1986, 120–129.

[Barnard 86] S. Barnard, "A Stochastic Approach to Stereo Vision," *Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86)*, Philadelphia, Pennsylvania, August 1986, 676–680.

[Barnard 87] Stephen T. Barnard, "Stereo Matching by Hierarchical Microcanonical Annealing," Technical Note No. 414, SRI International, Menlo Park, CA 94025, February 1987.

[Bolles 87] Robert C. Bolles, H. Harlyn Baker, and David H. Marimont, "Epipolar-Plane Image Analysis: An Approach to Determining Structure from Motion," *International Journal of Computer Vision,* Kluwer Academic Publishers, Vol.1, No.1, June 1987, 7–55.

[Burt 83] P. Burt, "The Laplacian Pyramid as a Compact Image Code," *IEEE Transactions on Communications*, Vol. COM-31, No. 4, April 1983, 532–540.

[Buxton 83] B.F. Buxton and Hilary Buxton, "Monocular Depth Perception from Optical Flow by Space Time Signal Processing," *Proceedings of the Royal Society of London, Series B,* 218 (1983), 27–47.

[Creutz 83] M. Creutz, "Microcanonical Monte Carlo Simulation," *Physical Review Letters*, Vol. 50, No. 19, May 9, 1983, 1411–1414.

[Dickmanns 87] E. D. Dickmanns and A. Zapp, "Autonomous High Speed Road Vehicle Guidance by Computer Vision," *Tenth IFAC Congress,* München, West Germany, July 1987.

[Förstner 86] W. Förstner, "A Feature Based Correspondence Algorithm for Image Matching," *Int. Arch. of Photogrammetry,* Vol. 26-III, Rovaniemi, Finland, 1986.

[Gelb 74] **Applied Optimal Estimation,** Arthur Gelb, editor, written by the Technical Staff, The Analytic Sciences Corporation, MIT Press, Cambridge, Massachusetts, 1974.

[Gibson 50] James. J. Gibson, **The Perception of The Visual World**, Houghton Mifflin, Boston, 1950.

[Hannah 85] Marsha Jo Hannah, "SRI's Baseline Stereo System," *Proceedings DARPA Image Understanding Workshop,* Miami Beach, Florida, December 1985, 149–155.

[Heeger 86] David J. Heeger, "Depth and Flow from Motion Energy," *Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86),* Philadelphia, Pennsylvania, August 1986, 657–663.

[Hillis 85] W. Daniel Hillis, *The Connection Machine.* Cambridge, MA., MIT Press, 1985.

[INMOS 85] *Transputer Reference Manual,* INMOS Limited, Box 424, Bristol BS99 7DD, England, 1985.

[Jain 87] R. Jain, S. L. Bartlett, and N. O'Brien, "Motion Stereo Using Ego-Motion Complex Logarthmic Mapping," *IEEE Transactons on Pattern Analysis and Machine Intelligence,* Vol. PAMI-9, No. 3., May 1987, 356–369.

[Marimont 86a] David H. Marimont, "Inferring Spatial Structure from Feature Correspondences," Ph.D. dissertation, Electrical Engineering Department, Stanford University, 1986.

[Marimont 86b] David H. Marimont, "Projective Duality and the Analysis of Image Sequences," *Proceedings of the Workshop on Motion: Representation and Analysis,* IEEE Computer Society, Kiawah Island, South Carolina, May 1986, 7–14.

[Marr 77] D. Marr and T. Poggio, "A Theory of Human Stereo Vision," Memo 451, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts, November 1977.

[Metropolis 53] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller, "Equations of State Calculations by Fast Computing Machines," *J. Chem. Phys.,* Vol. 21, No. 6, June 1953, 1087–1092.

[Moravec 81] H. Moravec, "Rover Visual Obstacle Avoidance," *Proceedings of the 7th International Joint Conference on Artificial Intelligence,* Vancouver, Canada, August 1981, 785–790.

[Ohta 85] Y. Ohta and T. Kanade, "Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. PAMI-7, No. 2, March 1985, 139–154.

ABSTRACT

This paper discusses some new directions being taken in image processing for applications of three-dimensional image understanding. It's focus is on a stochastic stereo process and a sequence analysis process, and uses these to demonstrate the general thesis that new computer power, particularly parallelism, has brought new approaches to three-dimensional image processing. Both of these processes, inconceivable as production systems on conventional architectures, show great promise in parallel implementation. Substituting simple but massive local computation for sophisticated approaches involving search, they produce robust, reliable results in three-dimensional depth estimation.

ACKNOWLEDGEMENTS

This research was supported by DARPA Contracts MDA 903-86-C-0084 and DACA 76-85-C-0004. Bob Bolles has been critical in the inception of the EPI work, and he and David Marimont have been instrumental throughout its continued development. Special thanks to Stephen Barnard for his help in summarizing and reporting on the stochastic stereo work.

Dr. H. Harlyn Baker
Artificial Intelligence Center, SRI International,
333 Ravenswood Avenue, Menlo Park, California, USA 94025.