3D WIRE-FRAME INTEGRATION FROM IMAGE SEQUENCES

Mike Stephens & Chris Harris

Plessey Research Roke Manor, United Kingdom

When integrating visual features into 3D for a Structure From Motion algorithm, the connectivity and relationships of features are an important adjunct to any quantitative 3D geometry. This paper describes a vision system which aims to perceive and refine this topology, in conjunction with geometry, using edges and vertices extracted from a sequence of monocular images of an unconstrained scene. Rules which tackle the practical difficulties of imperfect image processing and of obscuration features are defined. Results are shown for a rotating view of a polyhedral object, and for an outdoor scene viewed from a moving vehicle.

INTRODUCTION

A useful aim for a general purpose computer vision system is to provide a representation which promotes understanding of the diverse 3D world. Stereopsis and Structure From Motion both form a basis for 3D perception from more than one image, without high-level constraints on the scene.

This paper describes Alvey supported research into 3D vision based on monocular Structure From Motion. A static scene is observed and coherently integrated into a 3D representation via image feature extraction and matching over an arbitrary number of images from a moving camera. Two important image feature types are edges and vertices. Both are expected to correspond mostly to well-defined 3D events; edges to surface folds and markings, vertices to edge discontinuities and junctions. However, a practical system must admit that a 2D image often contains misleading features; for example a T-junction where one edge obscures another at greater depth. Perhaps more importantly, a practical system must cope with imperfect feature perception; edges may bend or break, vertices may wobble or vanish.

The chosen representation is a connected list structure; the region-edge-vertex (REV) graph (regions are not extracted from images, but may be represented later). Such a structure readily allows refinement and extension for an image sequence of any length[1]. A 2D REV-graph represents features extracted from each image. A 3D REV-graph represents the integrated 3D structure. The information they contain can be divided broadly into *geometry* (where features are) and *topology* (how features connect to each other). The geometric part has been implemented separately, and is reviewed below. The topological part is the main subject of this paper.

REVIEW OF GEOMETRIC SYSTEM

The geometric part of the REV-graph (excluding the need for edges) has been implemented as a vision system in its own right, during the first part of the project. The procedure for integrating 3D positional information for isolated feature-points, extracted from a sequence of images, has been described in a previous paper[2]. The following is a brief outline of the method:

- Each image is processed by a corner operator to extract a list of 2D feature-points.
- Features are matched from image to image using prior estimates of camera motion, and image-plane attributes of feature-points.
- The visual matches are used to solve for camera motion via an iterative ego-motion algorithm[3].
- The first two images are used to instantiate the 3D representation - this is the *Boot* mode, analogous to a stereo problem.
- Subsequent images are used to refine the 3D representation, in terms of completeness and accuracy. This is the *Run* mode.
- The 3D positions of features are refined as the features are tracked in the Run mode. This is performed by an independent Kalman Filter for each 3D feature.
- Positional uncertainties are modelled by 3D normal distributions, which reduce as features are tracked and their positions refined.

This system, now known as DROID, has proved to be versatile and robust. It has been used to recognise a polyhedral object and guide a robot arm to pick it up. It is also the core of a system which provides 3D navigational information from a long sequence of outdoor images taken from a moving vehicle[4].

However, the level of vision provided by this system is crude in the sense that it only represents a set of isolated 3D "corners". Any connectivity must be inferred rather than perceived. Important connectivity clues are provided by edges in images. Hence the aim of recent work has been to build a 3D representation based upon feature points connected by edges - a 3D Wire-Frame view of the world. This is achieved by adding the topological part of the REV-graph.

EDGES IN 3D VISION

Edges in images are likely to be interesting features for a 3D vision system, either in stereo or motion, because most edges are expected to correspond to real 3D structure in the scene; for instance surface folds or discontinuities in surface brightness. In this sense, they are good image tokens like feature points, and they also provide connectivity information.

However, unlike feature points, edges are spatially extended features and this causes several difficult problems. The first is that edges extracted from real images are subject to inconsistencies, such as dropout, from image to image. The second is that edges are often not viewable as complete entities - one or both endpoints may be obscured by other structure or be outside the field of view. A third

AVC 1988 doi:10.5244/C.2.25

problem is that 3D positional integration is not directly achievable for general points on edges.

One approach to these problems is to extract a set of edge fragments, which are typically unconnected, parametrise these (for instance as straight-edge segments or circular arcs), and integrate these features into 3D. Connectivity can be inferred later. This sort of technique has been used successfully in stereo algorithms, for example[5,6].

Our approach is to try to extract connected edge segments from images, and to retain and refine connectivity during 3D integration. This requires that each image be decomposed into a connected list of edges and vertices. The edges link vertices, which are typically edge junctions, sharp bends or endpoints. Several advantages are offered; the connectivity of the resulting 3D wire-frame is perceived rather than inferred; some curved edges may be easier to deal with, merely as topological connectors; and the 3D geometry can be largely retained at the vertices. The latter allows 3D positions to be processed in the manner established for the DROID system.

A potential disadvantage of the connectivity approach is that it imposes severe requirements on the edge-vertex operator, in order to provide reasonable consistency of visual information from image to image[7]. However, consistency is important to all feature matching approaches. In this context, a promising edge-vertex operator[8] has been developed from the corner operator used by the independent Geometric system.

INTEGRATION OF EDGES AND VERTICES

Each edge-vertex decomposition produces a list of vertices, connected by edge segments. Example edge-vertex images are shown in figure 1b, for samples (figure 1a) from a sequence of views around a wooden polyhedral object, known as a widget. These images exhibit typical problems caused by self obscuration of the widget, and by imperfect feature extraction. In an edge-vertex list, taken from each image, the vertices may be classified by their *degree*; the number of attached edges. Hence the major classifications of vertices are:

degree 0 - isolated interest points (not strictly vertices)
degree 1 - edge endpoints
degree 2 - sharp bends, or breakpoints in straight-lines
degree 3 - edge junctions, eg from 3D corners

Degree 0 points are a special case which is discussed later. Degree 1 vertices indicate incomplete edges, either by dropout or at the edge of the field of view. Degree 3 vertices correspond to feature points with well-defined positions, except that a T-junction suggests a possible obscuration between two edges. The situation is muddied by imperfect connectivity, which may cause junctions to be perceived as degree 2 vertices, and so on.

All vertices (degree > 1) are connected to other vertices by at least one linking edge. In the simplest case, all edges are taken to be straight and the 3D geometry is completely described by the 3D positions held and refined at tracked vertices. Hence the edges may be regarded as topological features only, describing the connectivity of the wire-frame. The geometric information from each image is integrated into 3D by updating the positional estimates of tracked 3D vertices. The topological information in the 3D representation should also be "integrated" or refined. This refinement process must cope with the two main problems caused by the spatial extent of edges:

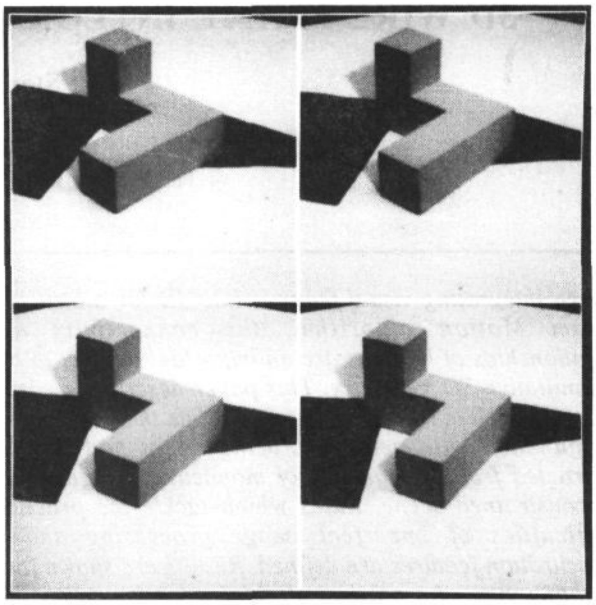

Figure 1a. Four images from a tour around a widget

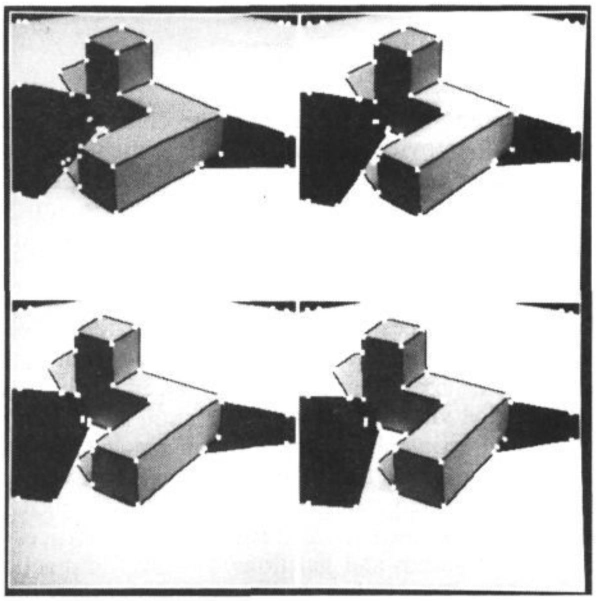

Figure 1b. Edge-vertex features for the images of figure 1a

1) edges and their connectivity will not be extracted with perfect consistency, from image to image,
2) obscurations and limited field of view cause false junctions and incomplete edges.

Hence, it is desirable to maintain multiple descriptions of the connectivity in the 3D REV-graph, together with a note of how often and when each element is seen - its track *history*. This provides several possible interpretations of a scene, together with a record of the strength of visual evidence for each. A "best" wire-frame model may then be extracted for use at any time. Such a system integrates topology, rather than attempting to modify it at each frame to fit new data[1].

The multiple representation is facilitated by including a third, notional element (after edges and vertices) in the wire-frame representation - *direction* nodes. A vertex of degree N has N attached directions, to which edges are linked. Now more than one edge may be attached to a direction in the 3D REV-graph, indicating more than one perceived connectivity. The direction nodes can also hold geometric information on the directions-of-leaving of edges, potentially even for curved edges.

DROID WITH WIRE-FRAME PERCEPTION

An extended version of DROID has now been written, and a simplified flowchart is shown in figure 2. The aim is to integrate a connected 3D wire-frame from a sequence of images, and provide information which allows a best representation of the scene to be extracted at any stage. The required tasks and algorithms divide almost entirely into two systems which can be regarded as operating in tandem. The Geometric system retains and refines the 3D metrical information, usually at the vertices. The Topological system retains and refines the connectivity information, using edges as links (via direction nodes).

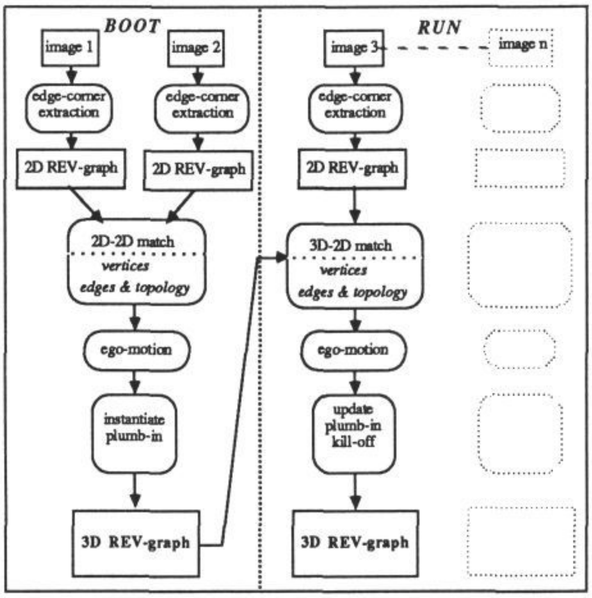

Figure 2. Edge-vertex processing by DROID.

The Geometric system is in fact very similar to the established DROID, because (at least for straight edges) all 3D geometric information is held at vertices. Hence, the algorithms for geometric matching, ego-motion estimation and positional refinement by Kalman filtering are still valid.

In edge-DROID, each edge-vertex decomposition of a new image results in a 2D REV-graph, where vertices are simply connected and any geometry relates to the 2D image plane. The current integrated representation is held in the 3D REV-graph, whose connectivity can be more complex, and where geometry is held for vertices which have been tracked and cast into 3D. The REV-graphs themselves are essentially dimensionless, since they describe topology. This means that components of the 2D and 3D REV-graphs can be compared and combined.

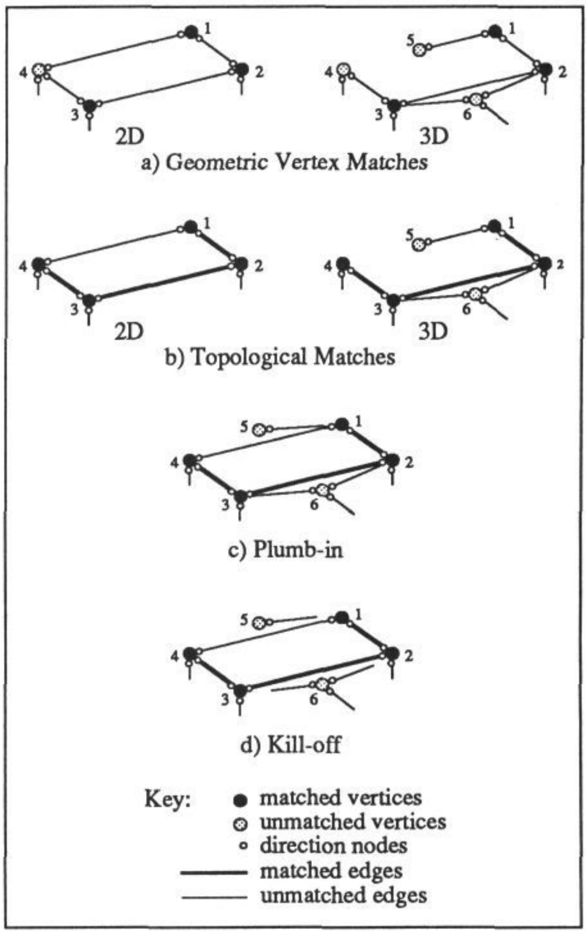

Figure 3. Stages in REV-graph update

Run Mode Processing

To explain the integration process, assume that the system has already booted-up and is in the Run mode. Figure 3 shows a simplified view of a fragment of the 2D (image) REV-graph being matched to the 3D graph, and the updating of the latter. The current 3D structure initially shows (figure 3a) one broken edge (between vertices 1 and 5) and a multiply-represented edge (between vertices 2 and 3, one possibility including vertex 6 as a T-junction). The major steps in processing a new image are:

Edge-Vertex Decomposition
This produces a connected edge-vertex list which is read by the system as the next 2D REV-graph, replacing the last in memory. A fragment of the 2D graph is shown on the left of figure 3a.

Geometric Vertex Matching
This is derived from the established DROID system, and treats the vertices as feature-points with image plane attributes[7]. The exit directions (angles) of edges are important attributes in the new system. An estimate of camera motion is used to predict candidate match regions for 3D points to be tracked, by projection of the positions of 3D vertices. In figure 3a, the three vertices labelled 1,2,3 are matched. (Vertex 4 is assumed unmatched because of poor geometry and/or attributes.)

Topological Matching
This is an additional phase of matching which travels along linking edges in the 3D and 2D REV-graphs, undoing topologically inconsistent vertex matches (*prune*), and creating topologically implied vertex and edge matches

(*grow*)[7]. In figure 3b, vertex 4 is matched by topological growth, because it is attached by an edge of geometrically consistent length to the matched vertex, 3. Additionally, three edges are matched (shown bold) because they are each bounded by matched vertices. Note that only whole (in terms of the graph) edges are matched - edge pixels are not considered.

Ego-Motion and 3D Point Update
This is exactly as for the established DROID system. Ego-motion re-estimates camera motion from the vertex matches[3]. The 3D positions of tracked vertices are then updated by Kalman Filtering, and the uncertainty models reduced. In figure 3b, the tracked vertices are 1,2,3 and 4. Where straight edges terminate in 3D vertices, they are implicitly updated geometrically at this stage.

Plumb-In
Unmatched topology is retained, and connected to the 3D REV-graph. In the example of figure 3c, a new complete edge is established between vertices 1 and 4, because it was seen in the 2D graph. Note that vertex 4 has become a degree 3 vertex (and a new direction node is established). Although not shown here, unmatched 2D vertices are also retained and plumbed-in. This is necessary because, for example a tracked 3D vertex may be linked by an edge to an unmatched vertex in the current image. The unmatched vertex is referred to as a *Limbo* point, and is held awaiting possible matching and casting into 3D.

Kill-Off
The system is capable of deciding to retire or kill-off vertices, edges and directions at the end of each frame. This can be driven by geometry or topology. The main reasons are:

- The geometric matcher marks a 3D vertex as well outside the current field of view (eg behind the camera) during projection. The vertex is scheduled for retirement (and possible archiving).
- A Limbo vertex has failed to match for several frames, and is considered too decrepit to be matchable. It is marked for destruction.
- The topological system marks a weak (in terms of track history) vertex, direction or edge for deletion, in order to limit the complexity of a multiply-perceived connectivity.

Because the vertices, directions and edges are linked by topology, any request for removal can propagate to other structure. For instance, if a vertex is to be deleted, its directions and edges must also be discarded to maintain consistency of the graph (all edges must terminate at a direction attached to a vertex). Hence, the kill-off phase consists of marking all requests for deletion, propagating this through the topology, and then freeing up the information slots in the Geometry and Topology. In figure 3d, the T-junction of vertex 6, and the broken edge terminating in vertex 5 are shown retiring (assuming a weak track history).

Boot and Limbo Processing

For clarity, the above discussion has not included the instantiation of 3D structure, either at Boot or as new structure is matched in the Run mode. The first two images constitute a Boot phase (see figure 2), which matches two 2D REV-graphs, in order to instantiate the 3D REV-graph. This consists of geometric matching via epipolar candidate regions (analogous to the stereo problem) followed by a topological prune/grow phase. Each matched vertex becomes a 3D feature, and each unmatched vertex becomes a Limbo point. The 3D and Limbo topology are linked to initiate the 3D REV-graph. During the Run mode, the matcher includes an additional phase which tries to match Limbo points (via epipolar candidate regions) to those vertices in the 2D REV-graph which failed to match to any 3D vertex. This Limbo matching allows the 3D REV-graph to be extended to include new scene detail, or to repair weak detail.

Isolated Interest Points

In general, DROID retains a best wire-frame representation of the scene which improves with each new viewpoint, in terms both of completeness and of positional accuracy. However, there may be no sensible edge-vertex description extractable from some parts of a scene. Here we are thinking particularly of natural objects such as trees and bushes, and of unresolved scene detail. With conventional edge-vertex extraction, DROID is likely to be confused by spurious and inconsistent detail. However, we know from experience that a corner operator *can* provide consistent features on objects such as trees, and that the representation is rich[4]. Consequently, an edge-vertex operator based upon the corner operator is being developed, with the aim of providing isolated interest points (corners) on densely textured regions[8]. With such an operator as a front end, DROID can very simply allow isolated (degree 0) "vertices", with appropriate attributes. Their 3D positions can be integrated as normal but they will not take part in topological update (unless they are later perceived as plumbed-in).

SOME RESULTS

Widget sequence

The widget sequence (see also figure 1) consists of 20 images (each 256x256 pixels) of the object rotating on a turntable, moving about 6 degrees between images. From DROID's perception, this is equivalent to the camera moving in a circular arc, focused on the widget. However, the shadows cast by the widget cause a moderate non-static element to be injected into the sequence. Although the shadows are not 'understood', the resulting inconsistent features tend to fade out and the overall representation is not seriously damaged. Figure 4 shows the projected 3D REV-graph, integrated from the extracted features, at several stages as it evolves. The shaded ellipsoidal shapes are projections of (16 sigma) surfaces of the 3D normal distributions, one for each vertex position. These are maintained by the Kalman filters in the Geometric system. The bright lines are the "best" linking edges between vertices, maintained by the Topological system.

In figure 4a, two images have been processed, and the 3D REV-graph has been instantiated. In figure 4b, the REV-graph has been refined, as four frames have been integrated. Note that the ellipsoids corresponding to tracked vertices are reduced and compacted, indicating improved 3D location from multiple viewpoints. The connecting topology is more complete, as broken edges are repaired from later views. Figure 4c is the situation at frame 20. Although the wire-frame connectivity improves little after frame 4, the positional geometry is much more precise.

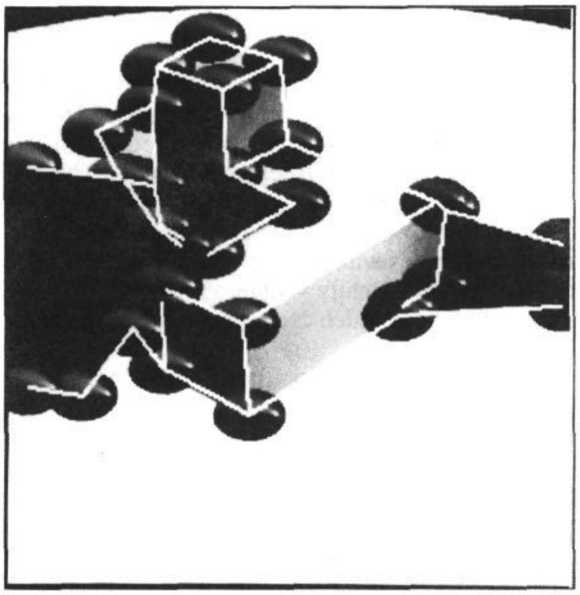

Figure 4a. Wire-frame and 3D vertex ellipsoids, image 2

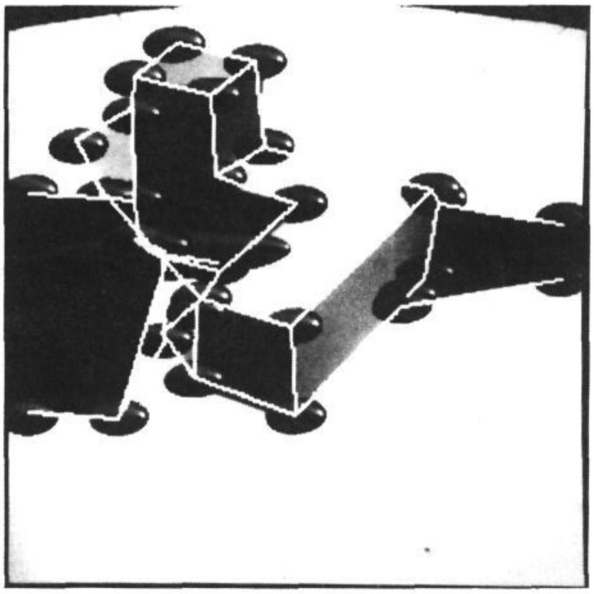

Figure 4b. Wire-frame and 3D vertex ellipsoids, image 4

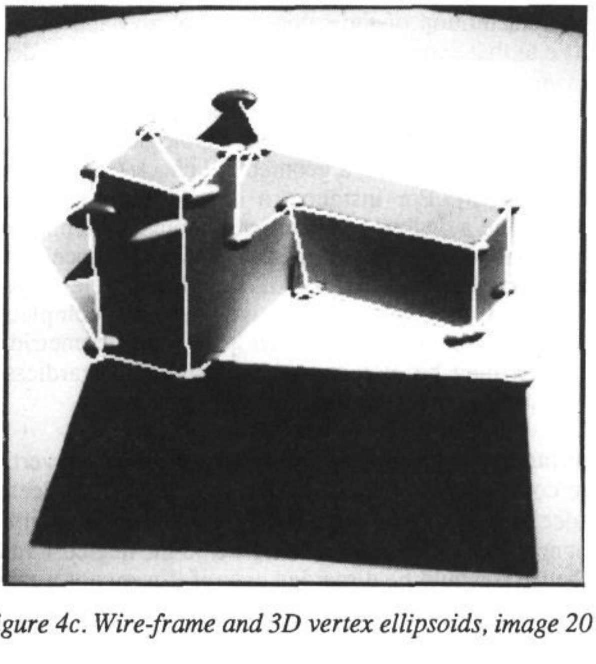

Figure 4c. Wire-frame and 3D vertex ellipsoids, image 20

Outdoor Sequence

Figure 5 shows an example image from the outdoor sequence. The scene includes straight-edge structure (the building, the road markings), curved road edges and natural vegetation. The camera moves about 30 cm forwards between frames. Figure 6 shows the projected structure of the 3D REV-graph as it evolves up to the fourth image. In this sequence, many isolated corners are extracted, on the vegetation, as well as edges and vertices around the building and road.

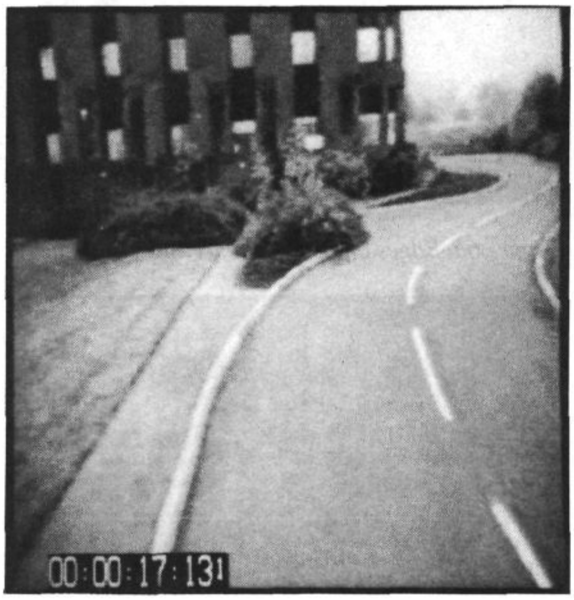

Figure 5. Example raw image from the outdoor sequence

Figure 6a shows the edges and corners which have matched and initiated the 3D REV-graph at image 2 (after Boot). Note that, instead of confusing edge detail in the vegetation, the system is perceiving and integrating reliable corner features. Conversely, instead of merely isolated corners on the building and road edges, the Topological system makes connected structure available. In figure 6b the bright corners and edges are those in the 3D REV-graph which have matched and been updated at image 3. The dark corners and edges are those parts of the 3D REV-graph not matched, but retained from Boot. Where the connectivity has changed in image 3, that at image 2 is remembered in the multiple representation. Unmatched corners are also retained as Limbo points. Hence, at image 4 (figure 6c), an attempt is made to match all remembered structure; those elements which match are updated geometrically and/or reinforced topologically.

The plotting for figure 6 assumes that all edges are straight, but this is *not* assumed in the topological representation itself. Some weakly curved edges, for example at the base of the bush in the near-field, are successfully tracked as topological links. Although the 3D position of any point along such curved edges is not strictly known, a straight-edge interpolation from the positions of its endpoint vertices may be sufficient. Alternatively, the endpoint positions and the exit directions (angles) of the edge are known. For the truly straight edges, for example the windows on the building, full 3D geometry is implicitly known.

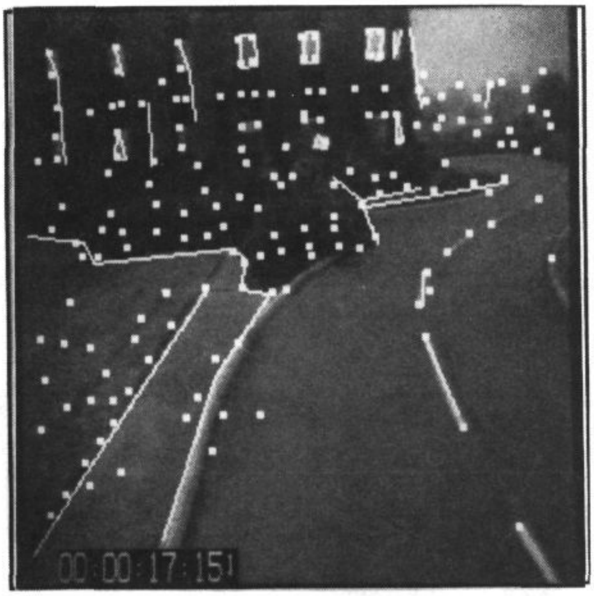

Figure 6a. Edges and corners cast into 3D at image 2

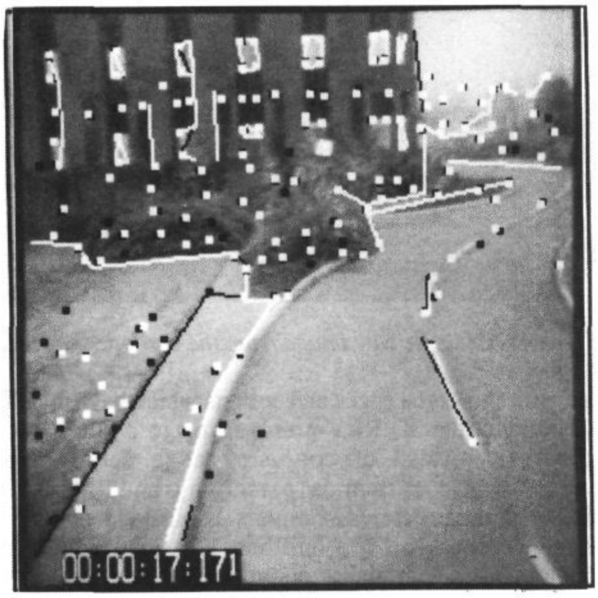

Figure 6b. 3D structure at image 3
(bright edges and corners are matched, dark are unmatched)

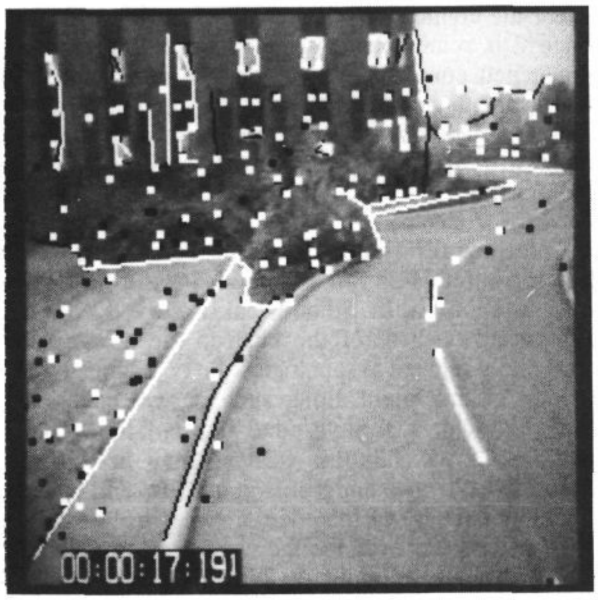

Figure 6c. 3D structure at image 4

CONSISTENCY PROBLEMS REVISITED

An important problem for a system such as edge-DROID is that some image vertices will not correspond to real 3D structure, but will only be perceived from some viewpoints; they are *subjective* features. The major culprits are:

- degree 1 vertices which imply an edge whose endpoint is weak or not currently visible,
- degree 2 vertices which correspond to weak or spurious bend points,
- degree 3 vertices (in particular, T-junctions) which are actually obscuration junctions between two edges at different depths.

The use of a multiply-connected 3D REV-graph allows decisions on these recalcitrant vertices to be postponed until more visual information is accumulated. The simple procedures described for the kill-off phase then cater in some measure for subjective features. For instance, an obscuration T-junction will be discarded as a weak vertex if the unobscured structure is viewed over enough frames.

However, positive rules are required to deal with these problems fully. The approach used is to allow vertices in the 3D REV-graph to be labelled at any stage as *non-geometric* vertices which are likely to correspond to subjective 3D features. The non-geometric candidates are:

- All degree 1 vertices - their apparent positions are expected to wander; as more or less of an incomplete edge is visible, or as an edge break behaves inconsistently.
- Bend points or T-junctions, which have failed to match geometrically over several frames - they are suspected to correspond to subjective features.

In the 3D graph of figure 3a, vertices 5 (endpoint of a broken edge) and 6 (T-junction, assuming no track history) are examples of vertices which would be labelled as non-geometric. Such vertices are treated differently during matching and during topological update.

The geometric matcher ignores non-geometric vertices, because a match would be spurious; occurring only if the apparent motion of the subjective feature happens to be close to that expected for a real feature at particular depth. However, topological matching can match these vertices, where they are connected to matched geometric vertices. This is similar to the concept of topological growth, but deliberately excluding a geometrical check (for consistent edge length). For instance, a degree 1 vertex can be matched to a degree 1 vertex, even though it terminates a line whose length has changed. The significance of this match is that the history of the incomplete edge is reinforced, and the edge survives to await completion. Similarly, the edges attached to a non-geometric T-junction may be matched and reinforced, regardless of inconsistent apparent motion of the junction.

During topological update, some non-geometric vertices are considered to be superseded by "proper" vertices, in order to bias the 3D wire-frame towards more complete connectivity. For instance, in figure 3d the degree 1 vertex labelled 5 would be discarded even if it were not weak on track history, because the broken edge is superseded by the complete edge between vertices 1 and 4.

POSSIBLE EXTENSIONS

This section briefly considers several areas of possible future work.

Curves

The current edge-vertex extraction does not admit curved edges in whole, but breaks them into (approximately) straight-edge segments. Consequently, no additional geometry for curves is attempted. For weak or short curves, extracted as single edges, some geometry is still retained; at terminating vertices, and in terms of edge exit angles. Because the REV-graph does not require that edge links be straight, the parametrisation and geometric integration of larger curves is possible if the edge-vertex operator can extract and label them. However, curved surfaces present a difficult consistency question - the treatment of profile edges, which are spatially extended subjective features. But, in outdoor scenes (for example, figures 5,6), the more immediate problems of non-regular structure are successfully tackled by DROID's multiply-connected topological edges, and its geometric treatment of isolated corners.

Regions

Regions are readily represented in the REV-graph, by the linking of bounding edges. They are of interest in dealing with surfaces of objects. However, extraction and integration of regions from images are likely to be very difficult, since regions are grossly extended (and hence fragile) features on the image plane. One solution is the inference of regions, after edge-vertex integration, but some support from image information is desirable.

Stereo plus motion

One advantage of the feature representation, and the matching/update methods employed by DROID is that the system can readily be extended to incorporate moving stereo sequences. Stereo offers the advantages of improved accuracy at close range and regularisation of camera motion estimates. Conversely, the integration of multiple viewpoints provides useful 3D geometry at longer range, and more complete scene representation from viewpoints which see different structure. Hence, the Geometric system has already been extended to allow N-camera stereo, and a small study of its performance is in progress. Stereo matching is performed in the same way as for Boot matching (between camera pairs). Motion matching is performed for each camera as for Run matching, and 3D geometry is updated from all matching viewpoints. The topological system, which operates in tandem with geometry in monocular edge-DROID, is similarly extendable to multiple cameras.

ACKNOWLEDGEMENTS

This work was carried out under Alvey project MMI/149. Ed Sparks, Mary Pike and Debra Charnley have helped develop the DROID vision algorithms and software.

The image sequence for figures 5 and 6 was digitised from a video tape supplied by RSRE;

REFERENCES

1 Herman, M & T Kanade "Incremental Reconstruction of 3D Scenes" *Artificial Intelligence*, December 1986

2 Harris, C G & J M Pike "3D Positional Integration from Image Sequences" *Proceedings third AVC*, 1987

3 Harris, C G "Determination of Ego-Motion from Matched Points" *Proceedings third AVC*, 1987

4 Charnley, D & R J Blissett "Surface Reconstruction from Outdoor Image Sequences" *AVC 88*, 1988

5 Ayache, N & F Lustman "Fast and Reliable Passive Trinocular Stereovision" *Proceedings first ICCV*, 1987

6 Pridmore, A P, J Porrill & J E W Mayhew "Segmentation and Description of Binocularly Viewed Contours" *Image & Vision Computing*, Vol 5 No 2, May 1987

7 Stephens, M J "Matching Features from Edge-Processed Image Sequences" *Proceedings third AVC*, 1987

8 Harris, C G & M J Stephens "A Combined Corner and Edge Detector" *AVC 88*, 1988