Algebraic polyhedral constraints and 3D structure from motion

D W Murray
University of Oxford
Department of Engineering Science
Parks Road, Oxford, OX1 3PJ

We describe the application of algebraic polyhedral constraints to the computation of the 3D structure and motion of polyhedral objects. The method, which works when complete 2D linedrawing information is available, guarantees the recovery of planar faces. The normals to these faces are used for matching to models. Several examples are given to illustrate the scope of the method.

In [1] Murray *et al.* describe a motion processing system, ISOR, which is able to recover the 3D motion and structure of polyhedral objects from an image sequence and goes on, where possible, to recognize the object as one from a database of object models. The system performs a 'bottom-up' pass through a vision processing hierarchy in the four stages: (i) **Low level** – Compute visual motion at intensity edgels in a sequence of time-varying imagery; (ii) **Segmentation** – Segment the edgels (and thereby visual motion) into groups lying on the same straight edges in the image; (iii) **Structure-from-motion (SFM)** – Compute the 3D structure and motion of the partial wireframe of which the linked straight edges in the image are the perspective projection; and (iv) **Recognition** — Match the 3D partial wireframe to a complete wireframe stored in a database of object models.

Here we make two modifications. First, the computation of 3D structure from 2D visual motion is made under algebraic constraints which force the reconstructed partial wireframe in 3D to be *strictly* polyhedral. The motivation is to make surfaces explicit at an earlier stage of the processing, in particular, before model matching. One way of achieving this would be to fit planes through the 3D edges and vertices computed by the existing, unconstrained, algorithm. If sets of edges were ajudged coplanar, it would then be possible to re-execute the algorithm with these additional constraints. However, such a method neglects the structural information contained in a single image of a polyhedral scene. This work utilizes the information from linedrawing analysis to provide *a priori* constraints to the SFM computation, using the techniques described by Sugihara [2,3]. The second modification is that we match using the recovered planar surfaces as primitives. using the search method of Grimson and Lozano-Pérez [4] and the geometrical constraints described by Murray [5,13].

1 THE UNMODIFIED SFM METHOD

For our present purpose, only details of the third stage of the system, that where 3D scene structure is computed from 2D visual motion, are of direct concern. Prior to a résumé of that stage, we give the briefest details of the earlier stages to clarify what information is explicit (see [1] for a fuller discussion).

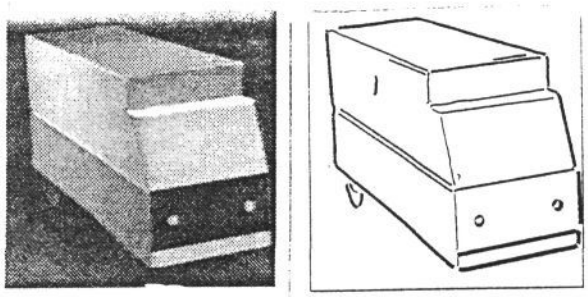

Figure 1: (a) The "current" image of a toy truck from a sequence where the camera translates towards the camera and (b) the visual motion components

1.1 Computing visual motion

The images in a sequence are processed in groups of three. The second of the group is regarded as the "current" frame and those captured one time interval earlier and one interval later defined as the backward and forward frames, respectively. Intensity edgels are computed in all three frames using the Canny detector [6] which provides the position $\underline{r}_e = (x_e, y_e)^T$ of each edgel e in the image to sub-pixel precision, along with its orientation and strength (change in grey value). A thresholding operation filters out weak, isolated edgels. Visual motion is computed at each edgel in the central frame by analysing matching strength distributions between edgels in consecutive frames [1]. Sitting at an edgel e in the current frame, a search is made around the position $\underline{r}_e$ in the forward frame for edgels f to which to match and initial matching strengths between e and f at $\underline{r}_f$ are defined initially using similarity measure which favours matching between edgels of similar strength and orientation. These initial strengths are improved using neighbourhood support within an iterative relaxation scheme. A similar search is made in the backward frame, and probability distributions combined simply by time-reversing the backwards displacements. The resulting distribution around position $\underline{r}_e$ is analysed using a principal axis decomposition [7], yielding two orthogonal vector components of visual motion and associated confidences. Figure 1a shows the "current" image of a sequence of a toy truck approaching the camera and the higher confidence components of visual motion are shown in Figure 1b. Note that along the extended edges the aperture problem prevails, and the major components are mostly normal to the edge direction. In this situation the lower confidence minor tangential components are of such little statistical worth that they may be discarded.

1.2 Segmentation

This stage segments the visual motion into groups lying

AVC 1989 doi:10.5244/C.3.37

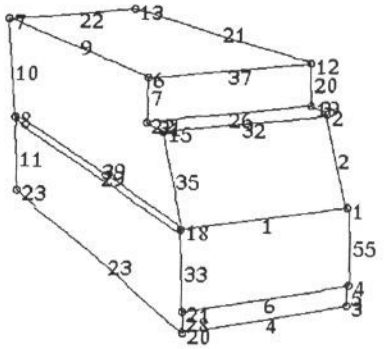

Figure 2: The segmentation for the truck

along extended straight edges. Because the visual motion is at edges, it suffices (in simple worlds!) to segment the edgel *positions*, with no reference to the visual motion per se. The segmentation proceeds through edgel linking into extended strings, breaking the strings into sections at points of high curvature, and by determining which sections comprise straight edges. Attempts are then made to link up straight edges which appear to converge to a single vertex. Figure 2 shows the final result for the truck, where the circles indicate vertices. As the visual motion is computed at the edgels, it is a trivial matter to import the visual motion into the segmentation graph.

1.3 The unmodified SFM algorithm

The unmodified SFM algorithm [1] is founded on the assumptions that, first, each subgraph is the projection of a rigidly moving object in the scene and, secondly, that straight edges and vertices in the image map to straight edges and vertices in the scene.

The scene can therefore be (over-)described by $n+6$ parameters $\{\zeta_1, \ldots, \zeta_n, \underline{V}, \underline{\Omega}\}$ where ζ_i is the inverse or reciprocal depth of the scene vertex which projects to image endpoint i, $\underline{V}$ is the translational velocity relative to the camera, and $\underline{\Omega}$ is the instantaneous angular velocity relative to the camera. These scene parameters are varied so as to minimize

$$D = \sum_e w_e(|\underline{v}_e| - |\underline{v}_e{}^{pred}|)^2 \tag{1}$$

where $\underline{v}_e$ is the *measured* (major) component of visual motion at edgel e, $\underline{v}_e{}^{pred}$ is the *predicted* component and w_e is the confidence associated with the measurement. The remainder of this section explains how $\underline{v}_e{}^{pred}$ is derived in terms of the unknown scene parameters and known image quantities.

The overdetermination arises because, without external knowledge, it is impossible to derive more than $n+5$ of the parameters because of the inevitable depth/speed scaling ambiguity in monocular motion processing. There are two obvious ways of reducing the dimensionality of the parameterization: (i) by fixing one of the reciprocal depth values or (ii) by fixing the magnitude of the translational velocity.

Figure 3 sketches the scene and camera geometries under consideration. Consider the image endpoint i at $\underline{r}_i$. It is related to the corresponding scene point $\underline{R}_i$ by $\underline{r}_i = -l\zeta_i\underline{R}_i$ where l is the focal length of the camera, and $\zeta_i = 1/(\underline{R}_i \cdot \hat{\underline{z}})$. The full projected motion at $\underline{r}_i$ is the time differential

$$\dot{\underline{r}}_i = -l\zeta_i\left(\dot{\underline{R}}_i - \underline{R}_i\zeta_i\dot{\underline{R}}_i \cdot \hat{\underline{z}}\right). \tag{2}$$

The motion of the scene point can always be expressed as

$$\dot{\underline{R}}_i = \underline{V} + \underline{\Omega} \times \underline{R}_i \tag{3}$$

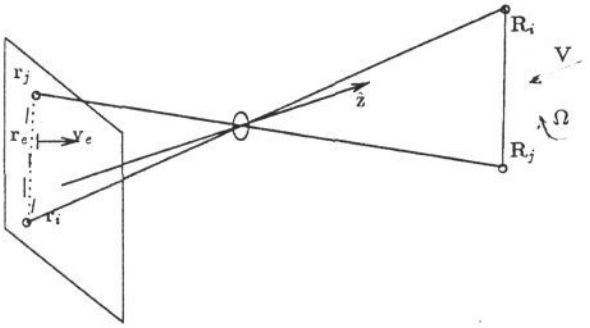

Figure 3: The camera and scene geometries

so that after substitution:

$$\dot{\underline{r}}_i = -l\zeta_i\underline{V} + \underline{\Omega} \times \underline{r}_i - \zeta_i(\underline{V} \cdot \hat{\underline{z}})\underline{r}_i + \frac{(\underline{\Omega} \times \underline{r}_i) \cdot \hat{\underline{z}}}{l}\underline{r}_i. \tag{4}$$

Now consider a point $\underline{r}$ on the straight edge between endpoints $\underline{r}_i$ and $\underline{r}_j$:

$$\underline{r} = \lambda\underline{r}_j + (1-\lambda)\underline{r}_i : 0 \le \lambda \le 1. \tag{5}$$

The visual motion at this point must be just

$$\dot{\underline{r}} = \lambda\dot{\underline{r}}_j + (1-\lambda)\dot{\underline{r}}_i. \tag{6}$$

This is *almost* the data we measure and wish to predict, but there are two details which must be taken care of. First, the edgel position $\underline{r}_e$ will probably not lie directly on the line between i and j: edgels will meander either side of the true line. To overcome this, we estimate λ to be that describing the nearest point on the straight line. In other words, given an edgel at $\underline{r}_e$ between endpoints i and j

$$\lambda_e = (\underline{r}_e - \underline{r}_i) \cdot (\underline{r}_j - \underline{r}_i)/|\underline{r}_j - \underline{r}_i|^2 \tag{7}$$

and hence the predicted full visual motion at the edgel is

$$\dot{\underline{r}}_e^{pred} = \lambda_e\dot{\underline{r}}_j + (1-\lambda_e)\dot{\underline{r}}_i. \tag{8}$$

Secondly, we wish to derive a *component* of $\dot{\underline{r}}_e^{pred}$. This is found simply by vector projection onto the measured component $\underline{v}_e$. That is, our predicted value of the component is

$$\underline{v}_e{}^{pred} = \underline{v}_e(\dot{\underline{r}}_e^{pred} \cdot \underline{v}_e)/|\underline{v}_e|^2. \tag{9}$$

After some routine working, the magnitude of the predicted component is given by [1]:

$$\begin{aligned} |\underline{v}_e{}^{pred}| \;=\; & \underline{V} \cdot \hat{\underline{x}}[\zeta_i(\lambda_e - 1) - \zeta_j\lambda_e]l\cos\theta + \\ & \underline{V} \cdot \hat{\underline{y}}[\zeta_i(\lambda_e - 1) - \zeta_j\lambda_e]l\sin\theta + \\ & \underline{V} \cdot \hat{\underline{z}}[\zeta_i(\lambda_e - 1)f_i - \zeta_j\lambda_e f_j] + \\ & \underline{\Omega} \cdot \hat{\underline{x}}[(1-\lambda_e)f_i y_i + \lambda_e f_j y_j + l^2\sin\theta]/l + \\ & \underline{\Omega} \cdot \hat{\underline{y}}[(\lambda_e - 1)f_i x_i - \lambda_e f_j x_j - l^2\cos\theta]/l + \\ & \underline{\Omega} \cdot \hat{\underline{z}}[(\lambda_e - 1)g_i - \lambda_e g_j], \end{aligned} \tag{10}$$

where $\cos\theta = (\hat{\underline{x}} \cdot \underline{v}_e)/|\underline{v}_e|$, $\sin\theta = (\hat{\underline{y}} \cdot \underline{v}_e)/|\underline{v}_e|$, and where $f_i = x_i\cos\theta + y_i\sin\theta$, $g_i = y_i\cos\theta - x_i\sin\theta$ and similarly for f_j, g_j.

2 POLYHEDRAL CONSTRAINTS

Although there is an implicit polyhedral assumption in the existing SFM algorithm, in that we consider a 3D scene to be made up of straight edges linked by vertices, nowhere do we exploit the fact that the straight edges lying around a face should be coplanar. To impose this though, obviously requires that we discover which edges comprise the border of a

face. This can be achieved by analysing the 2D line drawing, at least providing it is *complete*, a process which also provides other clues about relative depth. The two major methods of reconstructing polyhedra from 2D line drawings are due to Kanade [8], who recovered shape from line drawings using a gradient space approach and Sugihara [2,3] who developed linear algebraic constraints imposed in real space. Sugihara's technique has advantages over that of Kanade. First, the former's constraints impose necessary and sufficient conditions that the object is a polyhedron, where the latter's apply only a necessary condition. Secondly, gradient space techniques appear more sensitive to errors in 2D vertex positions than the algebraic constraints. Here we utilize Sugihara's method but, unlike previous published experimental work, we apply the constraints under perspective projection.

2.1 Sugihara's algebraic constraints

Using the information in the 2D graph derived for segmentation, we first create a labelled 2D line drawing [9,10,11] with lines corresponding to convex edges labelled '+', those corresponding to concave edges labelled '−' and those corresponding to occluding edges labelled '>', where the arrow points such that the area to the right of the arrow is the occluding face.

Following [2,3], let $\mathcal{V}$ be the set of visible vertices, so that $|\mathcal{V}| = n$, and let $\mathcal{F}$ be the set of (partially or wholly) visible faces, with $|\mathcal{F}| = m$. Now define the set $\mathcal{R}$ as $\mathcal{R} \subseteq \mathcal{V} \times \mathcal{F}$: $(v, f) \in \mathcal{R}$ iff $v \in \mathcal{V}$ lies on $f \in \mathcal{F}$. Each pair $(v, f) \in \mathcal{R}$ is called an *incidence pair* and the triple $S = (\mathcal{V}, \mathcal{F}, \mathcal{R})$ is an *incidence structure*. This is easily computed from the labelled line drawing.

Define scene points $\underline{R}$ lying on the face f_j by

$$\underline{R} \cdot \underline{N}_j = -1 \tag{11}$$

where $\underline{N}_j$ is normal to the face and sticks out of the surface into free space. Using the perspective projection (equation (3)) and writing $\underline{N}_j = (a_j b_j c_j)^T$, each $(v_i, f_j) \in \mathcal{R}$ gives rise to an equation

$$-a_j x_i/l - b_j y_i/l + c_j + \zeta_i = 0. \tag{12}$$

Collecting these together for every incidence pair in $\mathcal{R}$ results in the system:

$$\mathbf{A}\underline{s} = \underline{0} \tag{13}$$

where

$$\underline{s} = (a_1 b_1 c_1 \ldots a_m b_m c_m \zeta_1 \ldots \zeta_n)^T \tag{14}$$

is an unknown column vector of length $(3m + n)$ and $\mathbf{A}$ is a known $|\mathcal{R}| \times (3m + n)$ matrix.

Any face f_j divides space in two. If a point $\underline{R}'$ is such that $\underline{R}' \cdot \underline{N}_j + 1 > 0$ then the point lies in front of the plane of the face and if $\underline{R}' \cdot \underline{N}_j + 1 < 0$ it lies behind the plane. Now consider two faces f_j and f_k sharing a concave edge, as illustrated in Figure 4a. Consider the vertex v_i such that $(v_i, f_k) \in \mathcal{R}$ but $(v_i, f_j) \notin \mathcal{R}$. Clearly,

$$-a_j x_i/l - b_j y_i/l + c_j + \zeta_i > 0. \tag{15}$$

But suppose these faces share a convex edge (Figure 4b). Then

$$+a_j x_i/l + b_j y_i/l - c_j - \zeta_i > 0. \tag{16}$$

In fact the situation is a little more complicated. The analysis above is only straightforwardly applicable when the joining edge is not a re-entrant edge on a non-convex face. In practice a test is made (in 2D) whether all the vertices of at least one face lie to one side of the line created by extending

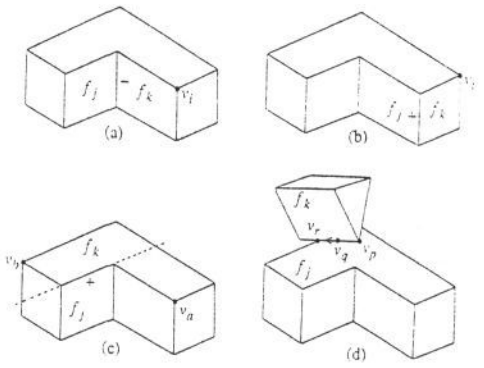

Figure 4: Linedrawings of polyhedra

the shared edge. If they do, then the correct inequality can be chosen. For example in Figure 4c, all the vertices of face f_j lie on one side, so we can easily decide that vertices v_a and v_b lie on in front of, and behind, the plane of f_j, respectively.

Then consider Figure 4d where f_k occludes f_j. Let v_p, v_q, v_r be the initial, end and mid point of the occluding edge, following along the label direction. (Note that v_r is not an obvious member of $\mathcal{V}$. Sugihara explains that such pseudo-vertices are added to $\mathcal{V}$ and the pseudo-incidence pair (v_r, f_k) to $\mathcal{R}$ during creation of the incidence structure. They are then treated just like any other members.)

Either none or one of v_q or v_p could touch f_j, but not both. Thus three constraints become available [3]:

$$-a_j x_p/l - b_j y_p/l + c_j + \zeta_p \geq 0, \tag{17}$$

$$-a_j x_q/l - b_j y_q/l + c_j + \zeta_q \geq 0, \tag{18}$$

and

$$-a_j x_r/l - b_j y_r/l + c_j + \zeta_r > 0. \tag{19}$$

Unfortunately, we cannot apply these constraints within a single SFM computation, because of the possibility that the occluding and occluded objects *move* differently, although they can of course be used to constrain depths between separate applications of the SFM algorithm.

Constraints of the type (17) - (18) (and indeed the occlusion constraints, if used) can be expressed as

$$\mathbf{B}\underline{s} > \underline{0} \tag{20}$$

(where, for occlusions, the inequality sometimes permits equality).

Hence, given that we wish to recover a polyhedral object, one might pose the structure from motion calculation as:

Minimize

$$D = \sum_e w_e (|\underline{v}_e| - |\underline{v}_e{}^{pred}|)^2 \tag{21}$$

subject to the conditions

$$\mathbf{A}\underline{s} = \underline{0} \tag{22}$$

$$\mathbf{B}\underline{s} > \underline{0}. \tag{23}$$

However, Sugihara highlights several difficulties with applying the constraints naïvely. The principal one is that not all the equations in the equality constraint (Equation 24) are linearly independent. We outline here the steps used by Sugihara [2,3] to eliminate this problem.

2.2 Eliminating dependent constraints

Because only a few of the vertex positions in a polyhedron are independent, some of the constraints expressed by the set

$\mathcal{R}$ depend on others. It is necessary both to eliminate these dependent constraints and to elicit the set of independent vertices [2,3].

First, it is necessary for the image vertices $(x_1 y_1 \ldots x_n y_n)$ to be in *general position*, that is, they must be algebraically independant over the rational field so that there are no special relationships between their positions. (Eg., three vertices must not always be collinear, nor three edges concurrent.) Given this condition, we seek a *position-free* incidence structure S, one where the constraint system has a non-trivial solution when the vertices are in general position. Sugihara proves the following:

Theorem 1 *If $S = (\mathcal{V}, \mathcal{F}, \mathcal{R})$ is a incidence structure in which no three faces sharing a vertex have a common line of intersection then S is position-free if and only if for all $\mathcal{X} \subseteq \mathcal{F} : |\mathcal{X}| \geq 2$,*

$$|\mathcal{V}(\mathcal{X})| + 3|\mathcal{X}| \geq |\mathcal{R}(\mathcal{X})| + 4;$$

where $\mathcal{V}(\mathcal{X})$ is the set of vertices that are on some faces in $\mathcal{X}$ and $\mathcal{R}(\mathcal{X})$ is the set of incidence pairs involving elements of $\mathcal{X}$.

Theorem 2 *If S as described in Theorem 1 is position-free and the vertices are in general position, then the system $\mathbf{A}\underline{s} = \underline{0}$ is linearly independent.*

Given some set of incidence pairs $\mathcal{R}$, we can use Theorem 1 to test whether it is position free. If it is not, we search for a maximal set $\mathcal{R}^* \subset \mathcal{R}$ for which the reduced incidence structure S^* is position-free by testing that for all $\mathcal{X} \subseteq \mathcal{F} : |\mathcal{X}| \geq 2$,

$$|\mathcal{V}^*(\mathcal{X})| + 3|\mathcal{X}| \geq |\mathcal{R}^*(\mathcal{X})| + 4; \tag{24}$$

where $\mathcal{R}^*(\mathcal{X})$ is the subset of $\mathcal{R}^*$ involving elements of $\mathcal{X}$ and $\mathcal{V}^*(\mathcal{X}) = \{v | v \in \mathcal{V}, (\{v\} \times \mathcal{X}) \cup \mathcal{R}^* \neq \phi\}$.

Let the reduced matrix associated with the constraints in $\mathcal{R}^*$ be $\mathbf{A}^*$. Theorem 2 indicates that it must be possible to transform $\mathbf{A}^*$ by appropriate column permutation into $\mathbf{A}'$, which may be partitioned so that

$$\mathbf{A}'\underline{s}' = (\mathbf{A}_1 | \mathbf{A}_2)\,\underline{s}' = \underline{0} \tag{25}$$

where $\mathbf{A}_1$ is a non-singular $|\mathcal{R}^*| \times |\mathcal{R}^*|$ matrix whose inverse therefore exists. The vector $\underline{s}'$ has the same members as $\underline{s}$ but certain of the ζ values will have been permuted. Splitting $\underline{s}'$ into two vectors $\underline{s}' = (\underline{\eta}, \underline{\xi})^T$, it is possible to write

$$\underline{\eta} = -{\mathbf{A}_1}^{-1}\mathbf{A}_2\underline{\xi}. \tag{26}$$

It is clear that we may associate the vector $\underline{\xi}$ with the reciprocal depths of the independent vertices, and $\underline{\eta}$ with the other, dependent, reciprocal depths and plane parameters. The number of independent parameters is $|\underline{\xi}| = 3m + n - rank(\mathbf{A}_1)$.

2.3 Finding the independent set of vertices

Section 2.2 shows that a set of independent vertices must exist. Here we briefly indicate the method proposed by Sugihara to find such a set, and thus how to find the permutation of columns that transforms $\mathbf{A}^*$ to $(\mathbf{A}_1|\mathbf{A}_2)$, $\underline{s}$ to $\underline{s}'$, and $\mathbf{B}$ to $\mathbf{B}'$ (used later).

It is possible to define the degree of freedom $\sigma_D(\mathcal{Y})$ of a set of vertices $\mathcal{Y} \subseteq \mathcal{V}$ such that the pair $(\mathcal{V}, \sigma_D)$ is a *matroid*. The subset of vertices we require is that which is the maximal independent subset of $\mathcal{V}$, that is a *base* of the matroid. Sugihara proves the following:

Theorem 3 *If $S^* = (\mathcal{V}, \mathcal{F}, \mathcal{R}^*)$ is a position free incidence structure then $\mathcal{Y} \subseteq \mathcal{V} - \mathcal{V}(\mathcal{R} - \mathcal{R}^*)$ is an independent set of the matroid $(\mathcal{V}, \sigma_D)$ if and only if for all $\mathcal{X} \subseteq \mathcal{F}$*

$$|\mathcal{V}^*(\mathcal{X})| + 3|\mathcal{X}| \geq |\mathcal{R}^*(\mathcal{X})| + |\mathcal{V}^*(\mathcal{X}) \cap \mathcal{Y}|.$$

Using this, and the fact for any $\mathcal{Y} \subseteq \mathcal{V}$:

$$\sigma_D(\mathcal{Y}) = max\{|\mathcal{Y}'|\}$$

such that $\mathcal{Y}' \subseteq \mathcal{Y}$ and $\mathcal{Y}'$ is an independent set of matroid $(\mathcal{V}, \sigma_D)$, we can build an independent subset $\mathcal{Y}$ by choosing vertices $\{v\}$ one by one from $\mathcal{V} - \mathcal{V}(\mathcal{R} - \mathcal{R}^*)$. Starting with $\mathcal{Y} = \phi$ we test whether $\{v\} \cup \mathcal{Y}$ is independent using Theorem 1. If it is, $\mathcal{Y} \rightarrow \{v\} \cup \mathcal{Y}$, otherwise $\{v\}$ is discarded. As soon as $|\mathcal{Y}| = \sigma_D(\mathcal{V}) = |\underline{\xi}|$, $\mathcal{Y}$ must be the required base [2] .

3 A NEW SFM ALGORITHM

Under the constraints, the structure-and-motion of the 3D wireframe is fully described by the depths or reciprocal depths of the vertices in the base $\mathcal{Y}$, that is, by $\underline{\xi}$, and by the six motion parameters $\underline{V}$ and $\underline{\Omega}$. However, the constraints have done nothing to resolve the depth/speed scaling ambiguity, and so we must still reduce the number of the parameters by one to $|\underline{\xi}| + 5$. Here, we fix the reciprocal depth of one of these independent vertices, say ξ_1, to unity.

The SFM problem becomes one of minimizing

$$D(\underline{p}) = \sum_e w_e(|\underline{v}_e| - |\underline{v}_e{}^{pred}|)^2 \tag{27}$$

subject now only to the inequality conditions

$$\mathbf{B}'\underline{s}' = \mathbf{B}'\mathbf{H}\underline{\xi} > \underline{0}. \tag{28}$$

Here, the parameter vector is $\underline{p} = (\xi_2 \ldots \xi_{|\underline{\xi}|}, \underline{V}, \underline{\Omega})^T$, $\mathbf{B}'$ is $\mathbf{B}$ after column permutation, and $\mathbf{H}$ is a linear transformation.

The complete procedure to obtain structure from motion is then:

1. Label the line drawing or 2D vertex-edge graph.
2. Find the maximal position-free incidence structure S^* using Theorem 1.
3. Find the maximal independent set of vertices and thereby which vertices are associated with $\underline{\xi}$.
4. Set $\xi_1 = 1$ and guess initial values for the parameters $(\xi_2 \ldots \xi_{|\underline{\xi}|})$ that satisify $\mathbf{B}'\mathbf{H}\underline{\xi} > \underline{0}$ and guess initial values for the six motion parameters $\underline{V}$ and $\underline{\Omega}$.
5. Starting with these initial values, minimize D with respect to the parameters. If D_{min} is below some threshold, and the $\underline{\xi}$ at minimum satisfies the inequalities, goto Step 6. Otherwise go to Step 4.
6. If $\mathcal{R}^* = \mathcal{R}$, end. Otherwise if $\mathcal{R}^* \neq \mathcal{R}$ the scene positions might not satisfy the constraints in $\mathcal{R} - \mathcal{R}^*$ because these have been removed. Correct the positions of the vertices involved with elements in $\mathcal{R} - \mathcal{R}^*$ by finding the intersections of the surfaces already computed. Then end.

4 SOME EXPERIMENTS

4.1 Toy truck

Figure 5 shows the line labelling derived from the segmentation of the truck. The entire incidence structure proves to be position-free in this case and the base set contains five independent vertices which are used in the SFM optimization.

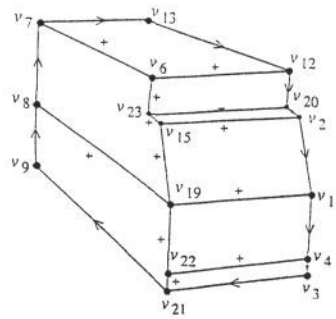

Figure 5: The line labelling for the truck

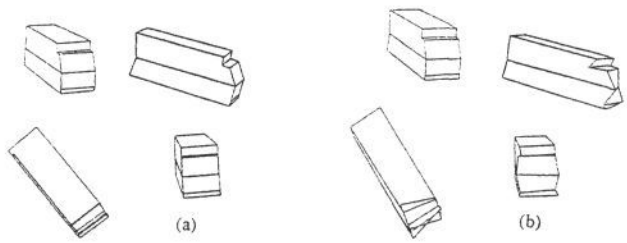

Figure 6: Structure recovered (a) with and (b) without the constraints

Thus the dimensionality of the optimization is reduced from 21 to 10. The reconstruction with constraints is shown in Figure 6a and that without in Figure 6b. There is a substantial improvement in the recovered structure, particularly marked for this case of translation towards the camera because there is very little depth information around the focus of expansion, here at the image centre.

As well as reducing the dimensionality of the problem, we have also recovered explicitly the planar faces of the object. Recall that the first $3m$ components of $\underline{\eta}$ contain the surface normals of the planar faces of the reconstructed object. Murray [5] has described a method of matching surface normal and relative position data to CAD-type models. The method is based on that of of Grimson and Lozano-Pérez [4], but develops geometrical matcing constraints appropriate when the overall scale of the 3D data is unknown. This is the case here, because the structure data still suffer the depth/speed scaling ambiguity.

A data to model match is grown by considering the compatibility of the following metrics between pairs of data patches (a and b) and pairs of model faces (i and j):

$$\begin{array}{lcccc} \textbf{Data} & \hat{\underline{N}}_a \cdot \hat{\underline{N}}_b & \hat{\underline{N}}_a \cdot \hat{\underline{D}}_{ab} & \hat{\underline{N}}_b \cdot \hat{\underline{D}}_{ab} & \hat{\underline{N}}_{ab} \cdot \hat{\underline{D}}_{ab} \\ & \updownarrow & \updownarrow & \updownarrow & \updownarrow \\ \textbf{Model} & \hat{\underline{n}}_i \cdot \hat{\underline{n}}_j & \hat{\underline{n}}_i \cdot \hat{\underline{d}}_{ij} & \hat{\underline{n}}_j \cdot \hat{\underline{d}}_{ij} & \hat{\underline{n}}_{ij} \cdot \hat{\underline{d}}_{ij}. \end{array}$$

The vector $\hat{\underline{N}}_a$ is the unit normal to data patch a, $\hat{\underline{D}}_{ab}$ is the unit vector in the direction between patches a and b, and $\hat{\underline{N}}_{ab} = \hat{\underline{N}}_a \times \hat{\underline{N}}_b$, and similarly for the model metrics. The various vectors are illustrated in Figure 7. Because the data normals have sensing errors, and because the model faces have finite extent, both sets of metrics exhibit *ranges* of validity, which much overlap for consistency.

The surface normals from $\underline{\eta}$ are normalized and placed at the centre of each reconstructed face, as shown in Figure 8a. Figure 8b shows the labelling of faces of the surface model. Because of symmetry, there are two matches which are feasible under the constraints, shown in Table 1.

A match which is feasible under the local *pairwise* constraints does not necessarily possess a valid *global* transformation $(\mathbf{R}, \underline{t}, F)$ relating model and sensor spaces, $\underline{\mu}$ and $\underline{\sigma}$,

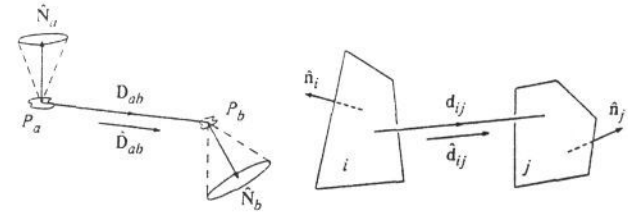

Figure 7: Vectors on the data and model

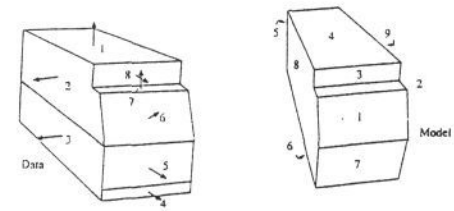

Figure 8: Vector labels in the case of the truck

as $\underline{\sigma} = F\mathbf{R}\underline{\mu} + \underline{t}$ where $\mathbf{R}$ is a rotation matrix, $\underline{t}$ is a translation and F is a scaling factor. Using each feasible match we derive first the rotation $\mathbf{R}$ (using the quaternion technique of Faugeras and Hébert [12]) and then the translation and scaling that best relate model and data spaces. We then assess whether this represents a good global transformation by determining the overall deviation of the sensed patch positions from their respective matched faces after transformation.

In the case of the toy truck, this process enables us to reject the second feasible interpretation as globally invalid. The scale factor derived for the first feasible and globally valid interpretation finally resolves the depth/speed scaling ambiguity [5], enabling the recovery of *absolute* depths and translation velocity. For example, the veridical width of the toy truck was 76mm and that recovered was 71.1mm; the veridical translational velocity was $\underline{V} = (0, 0, -20)$mm per frame and that computed was $(0.3, 0.1, -17.9)$mm per frame.

4.2 A chipped block

We include a second example with an indicidence structure which is not position-free. Figure 9 shows the visual motion, the labelled line drawing and the reconstruction, where the vertex and edge indices are those given by the segmentation stage. The full incidence structure S comprises

$$\begin{aligned} \mathcal{V} = &\quad \{v_1\, v_2\, v_5\, v_7\, v_{11}\, v_{12}\, v_{13}\, v_{14}\, v_{16}\}, \\ \mathcal{F} = &\quad \{f_1\, f_2\, f_3\, f_4\}, \\ \mathcal{R} = &\quad \{(v_2, f_1)\,(v_{13}, f_1)\,(v_{14}, f_1)\,(v_1, f_2)\,(v_{11}, f_2)\,(v_{14}, f_2) \\ &\quad (v_2, f_2)\,(v_{12}, f_2)\,(v_{16}, f_3)\,(v_5, f_3)\,(v_{11}, f_3)\,(v_{14}, f_3) \\ &\quad (v_{13}, f_3)\,(v_7, f_4)\,(v_{12}, f_4)\,(v_2, f_4)\,(v_{13}, f_4)\,(v_{16}, f_4)\}. \end{aligned}$$

This is not position free, but the removal of, for example, the pair (v_{13}, f_3) from $\mathcal{R}$ (that is, setting $\mathcal{R}^* = \mathcal{R} - \{(v_{13}, f_3)\}$) makes S^* so. Using theorem 3, a base set of independent vertices is found as

$$\mathcal{Y} = \{v_2\, v_1\, v_{14}\, v_{16}\},$$

thus reducing the size of the parameter space from fourteen to nine.

Data patches	→	1	2	3	4	5	6	7	8
Model faces	Match 1 →	4	8	8	7	7	1	2	3
	Match 2 →	4	9	9	7	7	1	2	3

Table 1: The two interpretations feasible under the pairwise constraints. The second is found globally invalid by transformation.

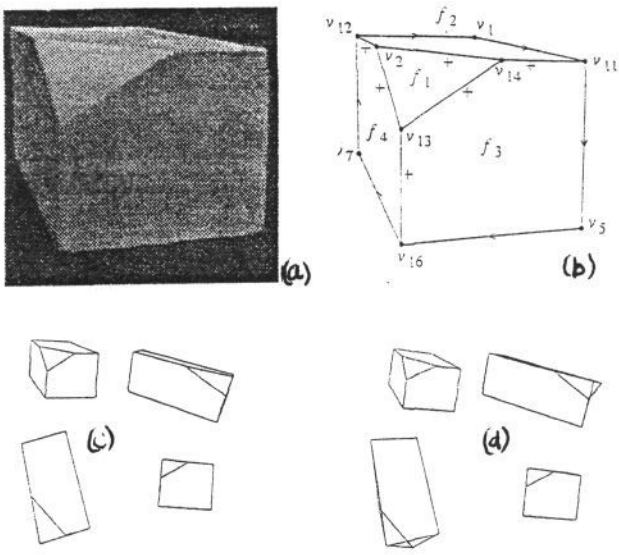

Figure 9: Image (a), line drawing (b) and reconstruction with (c) and without (d) constraints

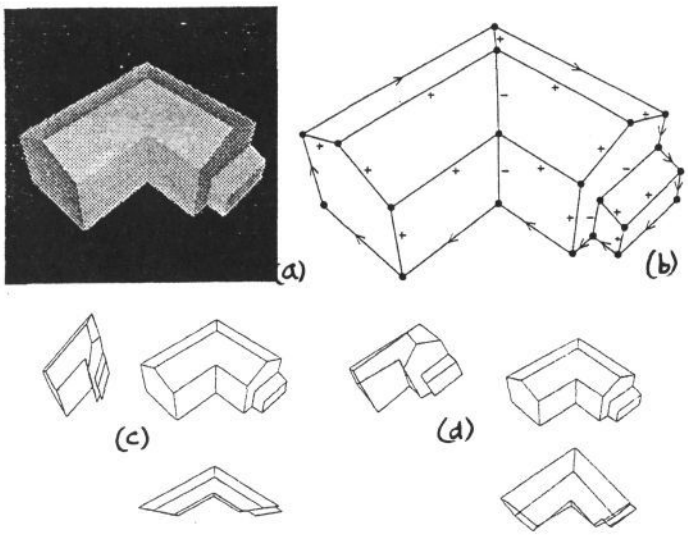

Figure 10: Image (a) and line drawing (b) of the moving CSG House. Reconstruction with (c) is worse than that without (d) constraints

There is a clear improvement in the quality of the recovered structure when compared with the reconstruction without the constraints [1].

5 COMMENTS

We have demonstrated that the geometrical reasoning method of Sugihara can be used successfully within the framework of a SFM algorithm and that the surface information recovered is of sufficient quality to match to simple CAD models, enabling absolute depths and motion to be recovered.

Matching to surfaces rather than edges has the advantage that the search space for matching is considerably reduced, because faces are always fewer in number than edges. By way of empirical illustration, to obtain the feasible matches using 8 data surface patches and 9 model faces took around 2 cpu-seconds on a Sun 3/160. Even if we restrict the edge matching problem, using only 8 of the 22 data edges, matching to the model of 26 edges took 54 cpu-seconds. (The code per attempted compatibility test is of similar complexity in the two cases.)

In most cases explored, the recovered structure was improved over that obtained by the unconstrained algorithm. However, it is clear that by adding extra constraints or expectations about the scene, there is a risk of failure because those expectations are not met, perhaps through noise or, more fundamentally, if the quite strict requirements of Sugihara's method are not met. Figure 10a,b shows an image and line labelling of a CSG model house. If we attempt to find an independent set and supply noise-free values of their 3D positions only half the object is successfully described. If we attempt to recover SFM with the constraints, the reconstruction is a failure, as shown in Figure 10c,d.

Perhaps the most alarming requirement is that the 2D line-drawing be complete. Given that most existing edge detectors are designed to preserve geometry rather than topology, this is almost impossible to guarantee. There are other drawbacks which temper enthusiasm for the method. Firstly, the computational cost of the method is quite high, requiring in the case of finding the base set of vertices multiple passes through the power set of the faces. Secondly, apart from checking the initial and final reciprocal depths for consitency, the inequality constraints, which yield clues about relative depth, do not guide the SFM minimization. Although standard techniques exist for active use of inequality constraints in the minimization of linear and quadratic functions, the present minimization function can not be set into such forms.

Acknowledgements: Much of this work was performed while the author was with GEC Research, Wembley.

References

[1] D W Murray, D A Castelow and B F Buxton. From image sequences to recognized moving polyhedral moving objects. Accepted for publication, Int J. Computer Vision.

[2] K Sugihara. Mathematical structures of line drawings of polyhedrons – toward man-machine communications by means of line drawings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-4**(5), 1982, pp458–469.

[3] K Sugihara. An algebraic approach to shape-from-image problems. *Artificial Intelligence*, **23**(1), 1984, pp59–95.

[4] W E L Grimson and T Lozano-Pérez. Model-based recognition and localization from sparse range or tactile data. *International Journal of Robotics Research*, **3**, 1984, pp3–35.

[5] D W Murray. Model-based recognition using 3d shape alone. *Computer Vision, Graphics and Image Processing*, **40**, 1987, pp250–266.

[6] J F Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-8**(6), 1986, pp679–698.

[7] G L Scott. *Local and Global Interpretation of Moving Images.* Pitman, London, 1988.

[8] T Kanade. Recovery of the three-dimensional shape of an object from a single view. *Artificial Intelligence*, **17**, 1981, pp409–460.

[9] D Huffmann. Impossible objects as nonsense sentences. In B Meltzer and D Michie, editors, *Machine Intelligence 6*. Edinburgh University Press, 1971.

[10] D Waltz. Understanding line drawings of scenes with shadows. In P H Winston, editor, *The Psychology of Computer Vision*. McGraw-Hill, New York, 1975.

[11] M Clowes. On seeing things. *Artificial Intelligence*, **2**(1), 1971.

[12] O D Faugeras and M Hébert. A 3d recognition and positioning algorithm using geometric matching between primitive surfaces. In *Proc. Int. Joint Conf. on Artificial Intelligence IJCAI-83*, pp996–1002.

[13] D W Murray. Recognition from structure from motion Proceedings of the 2nd Alvey Vision Conference, Bristol, 1986.