Using Automatically Constructed View-Independent Relational Model in 3D Object Recognition

S. Zhang, G. D. Sullivan and K. D. Baker
Intelligent Systems Group
University of Reading, RG6 2AY, UK

Abstract This paper describes and demonstrates a view-independent relational model (VIRM) in a vision system designed for recognising a known 3D object from single monochromatic images. The aim is to derive a model of an object able to effect recognition without invoking pose information. The system inspects a CAD model of the object from a number of different viewpoints to identify relatively view-independent relationships among component parts of the object. These relations are represented in the form of a hypergraph. The VIRM can be searched using a best-first technique to obtain hypotheses of vehicle poses which match image features.

1. Introduction

The recognition of 3D objects from single 2D monochromatic images of unknown scenes is a major problem for computer vision. In this paper we discuss the construction of a view-independent relational model (VIRM), derived automatically from a CAD wireframe model, and its use in recognising 3D objects. The VIRM of the object is a weighted hypergraph, associated with procedural constraints. Weights attached to the hyperedges, representing the probability of co-visibility of component parts of the object, are used to control the search for object hypotheses in recognition, and procedural constraints associated with the hyperedges prune the interpretation tree during the search. An early report of this model scheme has been given in [12]. We report here several major improvements brought about by adding an automatic model feature selection process, and by using the hypergraph to represent relations among clusters of more than two model features. We also report the use of the VIRM for recognition of multiple objects.

In this paper we are only concerned to establish plausible initial hypotheses of objects and their poses. The system is used in conjunction with a hypothesis refinement process to recover accurate pose and object classes (see Sullivan [9], Worrall *et al* [10]). The VIRM is used to identify extended groups of 2D image features compatible with a hypothesis of the class and pose of the object. The VIRM encodes the object by means of five relations: co-visibility, parallelism, colinearity, side relation, and relative size. The model scheme is viewer-centred, so that the hypothesis generation process does not depend on an initial estimate of pose. However, unlike other viewer-centred models, such as those based on aspect graphs ([3,6,7,8]), the present model is comprised entirely of view-independent attributes and relations. No search over alternative viewpoints is needed during the object recognition process, and the storage required for the model is small. The model uses a hypergraph representation associated with procedural constraints, expressing relations among two or more features, which can be used to control the search in the labelling (or hypothesis generation) process.

The model is generated automatically from a CAD wireframe model of the object. We illustrate the system by using a model of a car, but the approach can be applied to any geometrically defined object. The output of the model building phrase is a triple M={V, G, C}, which consists of:

1. A set of extended model features, V, including their shapes and the types of extended 2D image features, that they might match.
2. Extended adjacency matrices, G, representing the co-visibility of the model features.

3. A set of procedural constraints, C, each associated with a pair of co-visible model features, representing selected view-independent relations based on: parallelism, colinearity, side relation and relative size.

The first two elements V and G form the hypergraph representation of the VIRM. A hypergraph [1] is defined as an ordered pair H={X, E} where $X = \{x_1, x_2, \ldots, x_n\}$ is a set of vertices and $E = \{e_1, e_2, \ldots, e_m\}$ is a set of hyperedges such that $e_i \subseteq X$, $e_i \neq \emptyset$, $i = 1, 2, \ldots, m$ and $\bigcup_i^n e_i = X$. If $|e_i| = k$, then e_i is called an *order-k hyperedge* of the hypergraph. In particular, e_i is called an *edge* if $|e_i| = 2$. In this work we consider only edges and order-3 hyperedges.

2. Construction of the VIRM from a CAD Model

2.1. The Wireframe Model and Its Projection

Wireframe models of three different types of vehicles are shown in Fig.1, with the features of a hatchback car labelled symbolically (the hatchback car is used in this paper to illustrate the approach). The primitives in the wireframe model are line segments. Each is labelled uniquely, e.g. nfw_4 denotes the bottom line of the nearside front window, rws_1 denotes the bottom line of the rear windscreen, etc. (Note that the vehicles are British, the right side is the offside.) Throughout this research we assume that both the object and the image are approximately upright and that the angle between the axis of the camera and the ground is between 0 and θ_{max} (=60°). These assumptions cover all likely views of a car in normal conditions.

We start the construction of the VIRM of an object by projecting the CAD wireframe model from a number of different viewpoints. Let

$$O = \{o_1, o_2, \ldots, o_m\}$$

be the set of line segments in the wireframe model of the object. The model of the object is placed at the centre of the Gaussian viewsphere. The area ($0<\varphi<2\pi$, $0<\theta<\theta_{max}$) on the surface of the Gaussian viewsphere is sampled randomly giving n (=500) viewpoints equally distributed over the area. Each sampled image gives rise to a set of projected model line segments:

$$S_i = \{s_1^i, s_2^i, \ldots, s_m^i\}, \; i = 1, 2, \ldots, n$$

in which s_k^i represents the projection of model feature o_k in sample i. Each s_k^i is either a line segment in the 2D image plane represented by its coordinates $[[x_1\ y_1]\ [x_2\ y_2]]$ or an empty set, if o_k is occluded from the given viewpoint.

2.2. Building Nodes and Node Attributes of the VIRM

The model primitives are 1-D line segments which provide only very poor constraints for object recognition. These 1-D line segments are therefore grouped into *2-D feature complexes* which form invariant patterns, for example, a planar quadrilateral in the 3D world reliably projects to a 2-D quadrilateral in the image. The 2-D complexes can therefore be used as "focus features" (Bolles & Horaud [2]) providing a starting point for searching for consistent cliques.

A subset of model features is grouped into a 2D complex if they satisfy the following:

1. For a given viewpoint, if any one feature is visible, then all are visible,
2. The features form connected sets on the 3D model and hence the image,
3. They conform to a known class of shapes (quadrilateral or U-shape curve).

For example, the windows of the car satisfy the above conditions, but the bonnet of the car does not because the line at the bottom of the windscreen may be occluded in views from the rear. For the hatchback car, this process groups all 6 windows and the 4 wheel arches into 2D complexes, represented as quadrilaterals and U-shape curves respectively. Complexes of model features are used in the same way as single model features, and in the following discussion, we use O_{2d} to express the set of 2-D complexes and O_{1d} to represent the remaining (1-D) model features.

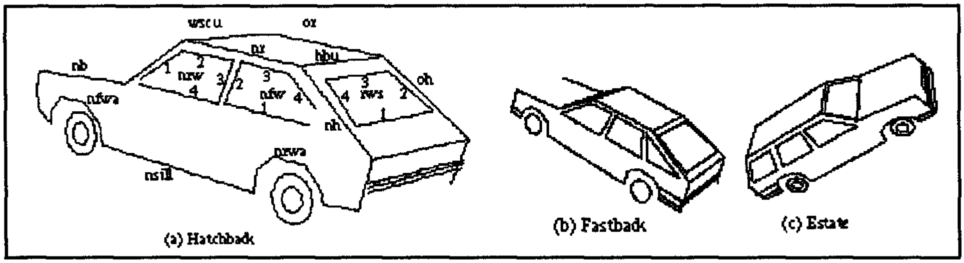


Fig.1. Wireframe models of vehicles

2.3. Building the Co-visibility Hypergraph of VIRM

The VIRM consists of nodes corresponding to model features, and hyperedges indicating co-visibility of features. Co-visibility of model features is generally view-dependent. However, we accept features to be co-visible if the probability of their co-occurring in images is high. This relaxation of the co-visibility constraints introduces errors, but these errors can be eliminated by finding combinations of mutually consistent matches. Currently we only use the co-visibility of pairs and triples of features.

2.3.1. Pairwise Co-visibility - Edges of the hypergraph

The pairwise co-visibility of two features is quantified by the conditional probability of observing one feature given the presence of another, represented by

$$A_2(i,j) = p\{o_j | o_i\}$$

Fig. 2 shows a part of co-visibility hypergraph of the VIRM of the hatchback car (a sub-hypergraph of G), where only the windows of the car are shown. (NB: $p\{o_i | o_j\}$ is shown to the right side of the arc from i to j).

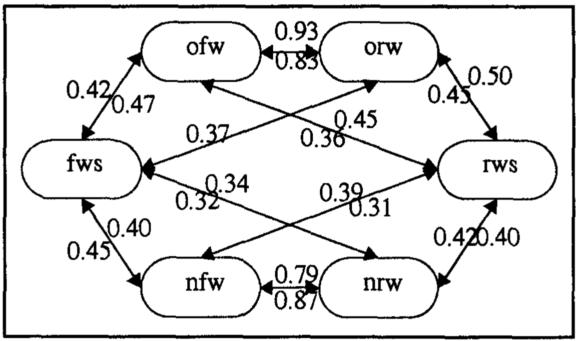


Fig.2. Part of the pairwise co-visibility hypergraph of the VIRM of a car

2.3.2. Co-visibility of Feature Clusters - Order-3 Hyperedges of the Hypergraph

Pairwise co-visibility provides only weak constraints on component parts of the object. It can be extended to include triples of features containing at least two 2-D complexes. For a cluster of model features $\{o_i, o_j, o_k\}$, we consider all the n projections of the model features and estimate the probability of their co-occurrence in the sampled images. The quantified co-visibilities of feature clusters are represented by means of the adjacency matrix of the hypergraph

$$A_2(i,j,k) = p\{o_k | (o_i, o_j)\}$$

in which $o_i, o_j \in O_{2d}$. This gives an order-3 hyperedge connecting nodes $\{o_i, o_j, o_k\}$ with weight $p\{o_k | (o_i, o_j)\}$ to represent their co-occurrence.

2.4. Geometrical Constraints- Procedures Associated with Edges of the VIRM

Although simple geometrical relations are inherently view-dependent, some are at least partially insensitive to view and provide weak constraints on pose. We consider, in particular, four pairwise relationships: parallelism, colinearity, side relation and relative size. Other view-independent relations, such as connectivity, symmetry, etc., are possible but have proved to be more difficult to compute and contribute little to car recognition.

The constraints on feature pairs are examined by quantifying the above relations as scalars. There are no generally accepted ways to quantify parallelism, colinearity, side relation, and we have adopted measures which we call parallel ratio, co-line ratio, and side ratio (see below). Each is based on overlaps, lengths, and angles among the line segments concerned.We use the Monte Carlo methods to obtain statistical evidence of the four relations between each pair of model features. If a specified relation between a pair of model features is reasonably stable over all views, then it defines a constraint associated with the edge connecting the corresponding nodes in the co-visibility hypergraph of the VIRM.

2.4.1. Parallel Ratio

Parallel model features appear approximately parallel in the image except when viewed from extreme angles. The *parallel ratio* between a pair of line segments is defined as follows. If the two line segments do not overlap (as shown in Fig.3(a)), or if the crossing point of the two lines is within both segments (Fig.3(b)), the parallel ratio is defined as zero. Otherwise, (e.g. Fig.3(c)), the parallel ratio is defined as

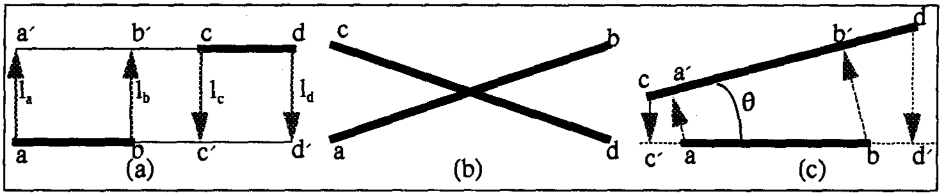
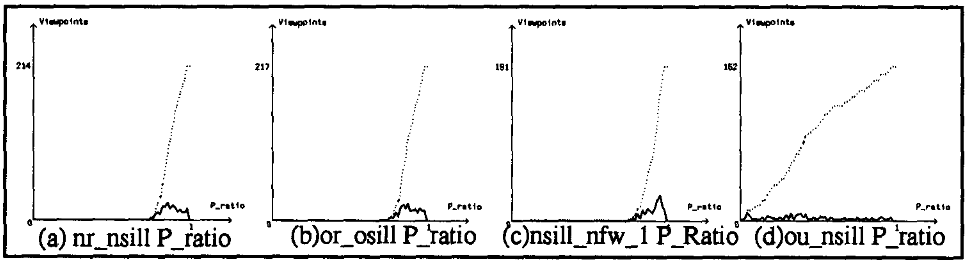


Fig.3. Definition of parallel ratio

$$p\,(ab, cd) = \frac{min\,\{l_a, l_b, l_c, l_d\}}{max\,\{l_a, l_b, l_c, l_d\}}\cos\theta$$

in which l_a is the distance between *a* and its orthogonal projection onto *cd*, etc., and θ is the angle between the two line segments. The parallel ratio between two line segments lies in [0,1], with 1 indicates absolutely parallel. If the mean value of p is high (>0.75) and the standard deviation is small (<0.25) then the parallel relation is accepted.



(a) nr_nsill P_ratio (b)or_osill P_ratio (c)nsill_nfw_1 P_Ratio (d)ou_nsill P_ratio

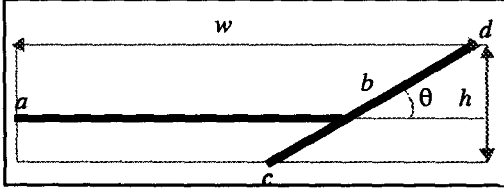
Fig.4. Parallel ratio between feature pairs

If one of the features is two dimensional (e.g. a quadrilateral or U-shape curve) and the other is a line segment, we record the parallel ratio between the line segment and each of the lines of the 2-D feature. If both are two dimensional, we record the parallel ratio of all pairs.

The *pdfs* of the parallel ratios (δ(p)) for 4 pairs of features are shown in Fig. 4: (a) nearside roof and nearside sill, (b) offside roof and offside sill, (c) nearside sill and the bottom line of the nearside front window, (d) offside upright (windscreen pillar) and nearside sill. The solid curves show the distribution function δ(p) of the variable. The dotted curves show the cumulative distributions. The first three show good parallel ratio of the pair of model features, with most values in the region of [0.8, 1].Therefore the constraints that these features pairs are parallel are accepted and are coded into the model. In Fig. 4(d), the parallel ratio is evenly distributed, and no parallel constraint exists between the pair.

2.4.2. *Colinear Ratio*

Colinearity between object features is preserved by the perspective transformation. Given two line segments *ab* and *cd* (assuming that *ab* is longer), as shown in Fig.5, we construct a minimal rectangle whose long axis is parallel to *ab* and encloses *ab* and *cd*. Let *w* be the length of the side of the rectangle parallel to *ab, h* be the length of the perpendicular side of the rectangle, and θ be the angle between the two line segments. Coline ratio is defined as in Fig.5.



$$c(ab, cd) = \begin{cases} 0 & h>w \\ \left|(1-\frac{h}{w})\cos\theta\right| & h<=w \end{cases}$$

Fig.5. Definition of coline ratio

The expected value for true colinearity, and the acceptance criterion is similar to that of parallelism.

2.4.3. *Side Relation*

Side relation, meaning one feature is to the left (or right) of another in the image, is view-independent if the object is solid and if the roll of the camera is limited (i.e. if the object and the image are both "upright"), though this breaks down if the object has significant concavities. The ***side relation*** of a point ***P*** to a directed line segment $\overline{ab}$ is used in the following way: Let ***V*** be the vector from ***a*** to ***b***. ***P*** is said to the left (right) of line segment $\overline{ab}$ if ***P*** is to the left (right) of ***V***. To define the side relation between two features, all the line segments involved are labelled with directions so that they can be used as vectors. Given line segment ***ab*** and a feature ***f*** and assuming that there are n_1 points in ***f*** which are on the left side of ***ab*** and n_2 points which are on the right side, the side ratio of feature ***f*** with respect to line segments ***ab*** is defined as:

$$s(ab,f) = \frac{n_1 - n_2}{n_1 + n_2}$$

The definition of side relation involving compound features is similar to that of parallelism. If one feature is always at the right (left) side of another the side relation constraint is accepted.

2.4.4. *Relative Size*

Relative size depends on the position and orientation of the camera, the focal length and the view angle. But provided that the focal length to distance ratio is not very small, and the object is not very large, the size changes of the different component parts of the object remain similar. The main effect on relative size is then due to the view angle. In the case of a two dimensional feature (e.g. a quadrilateral), at least one of its two dimensions is relatively stable and thus can provide bounds on the relative size of features. As an example, Fig. 6 shows a quadrilateral (Q) together with several projected shapes (Q_1, Q_2, Q_3) when viewed from different angles. At least one of its two dimensions remains at least 70% ($1/\sqrt{2}$) of its original scale, and can be used to provide bounds for the relative sizes of component parts of the object.

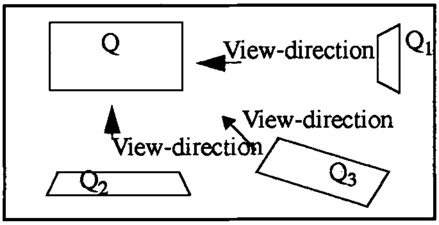


Fig.6. Variation of a quadrilateral viewed from different directions

Relative size is only used in feature pairs including at least one 2-D complex. Under perspective transformations its longest observed edge (the *pseudo-height* of the feature) in the image is expected to remain stable to some degree. The relative size of two features (r) is defined as the ratio of the distance between the features in the image to the pseudo-height of the 2D feature (if both features are 2D, the larger pseudo-height is used). The distance between two features is defined as the shortest distance from the vertices in one feature to the vertices in the other. If both features are 2D, their pseudo-heights are also compared and stored as a separate constraint. These ratios vary considerably, and therefore only provide weak bounds for the distributions. The acceptance criterion is similar to that of parallelism.

2.4.5. Procedural Constraints

As a result of the statistical analysis, each edge is associated with a set of constraints

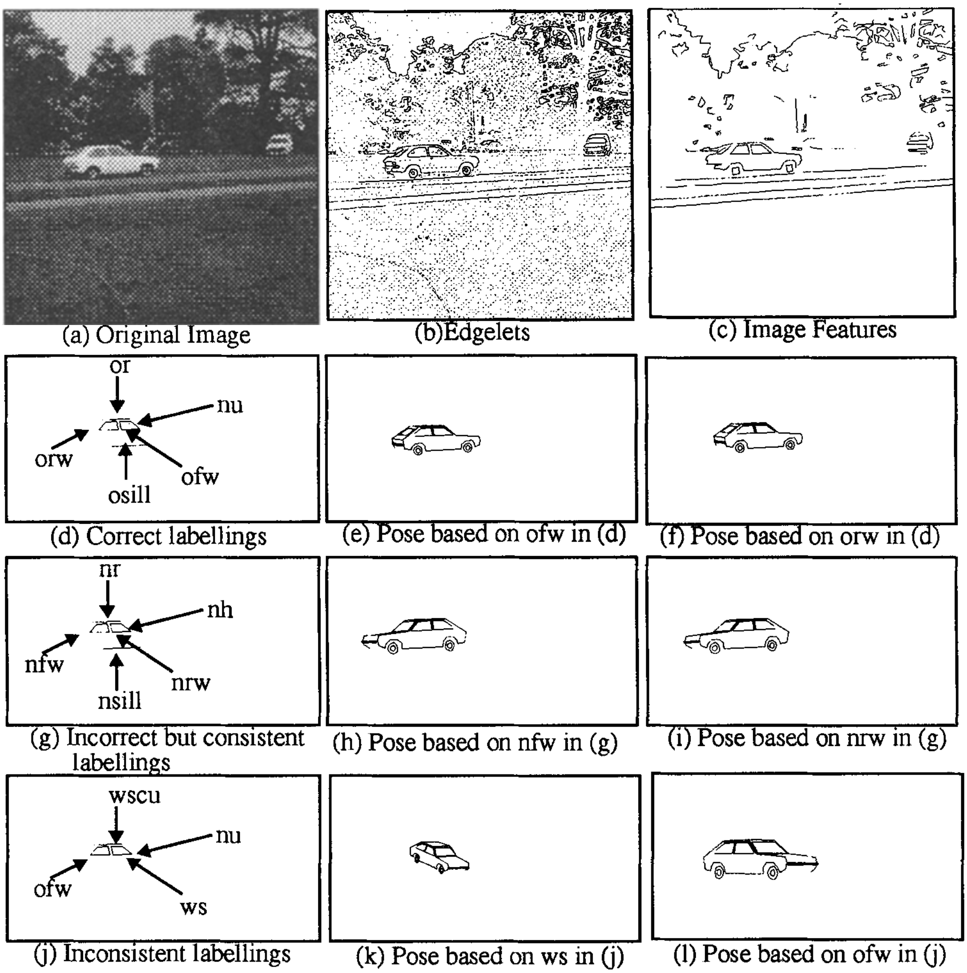
$$c_{ij}=\{p_{ij}, c_{ij}, l_{ij}, r_{ij}\}$$

representing relations between the corresponding pair of features $\{o_i, o_j\}$. If a relation is not applicable to this pair, the entry is open. We combine the hypergraph and the constraints to obtain the VIRM, with the selected constraints being compiled as procedures associated with edges of the hypergraph. The compiled constraints are Boolean-valued procedures with two inputs (the polyline representations of the two image features concerned). All of the constraints are defined in terms of the individual line segments constituting the corresponding features. Vertices in both 2D model features and 2D image features are labelled in an counter-clockwise order beginning with the bottom left vertex.

3. Application of the VIRM in the Generation of Pose Hypotheses

The VIRM is used to generate hypotheses of the class and pose of the object of interest. Fig.7 illustrates the different stages of the hypothesis generation process. The Canny [5] edge detector is first applied to the original image to get edgelets (Fig. 7(b)). These edgelets are grouped into 2D features (Fig. 7(c)). Each image feature is associated with all permissible model features, for example, quadrilateral image features are associated with any of the windows of the car. The image features are then matched with the VIRM by a depth-first search, restricted to only those cases with high co-visibility as recorded in the hypergraph. Typically around 10 consistent hypotheses are generated which reflect the inherent symmetries of the car (examples are shown in Fig. 7(d), (g) & (j)).

The hypotheses are used to estimate the pose using a quantitative method described by Du [12] in which two labelled non-parallel, non-colinear, co-planar lines are used to estimate the position and orientation of the camera, by means of pre-compiled look-up tables. Each of the 10 extended features groups identified by the VIRM gives rise to a number of pose hypotheses, based on the labelled 2-D features in the extended group. Fig. 7 shows three labelled feature groups ((d), (g) & (j)), each containing two 2-D features, so that each gives two pose hypotheses. Where these pose hypotheses are not consistent with each other, the labelling is rejected (Fig. 7



(a) Original Image (b)Edgelets (c) Image Features

(d) Correct labellings (e) Pose based on ofw in (d) (f) Pose based on orw in (d)

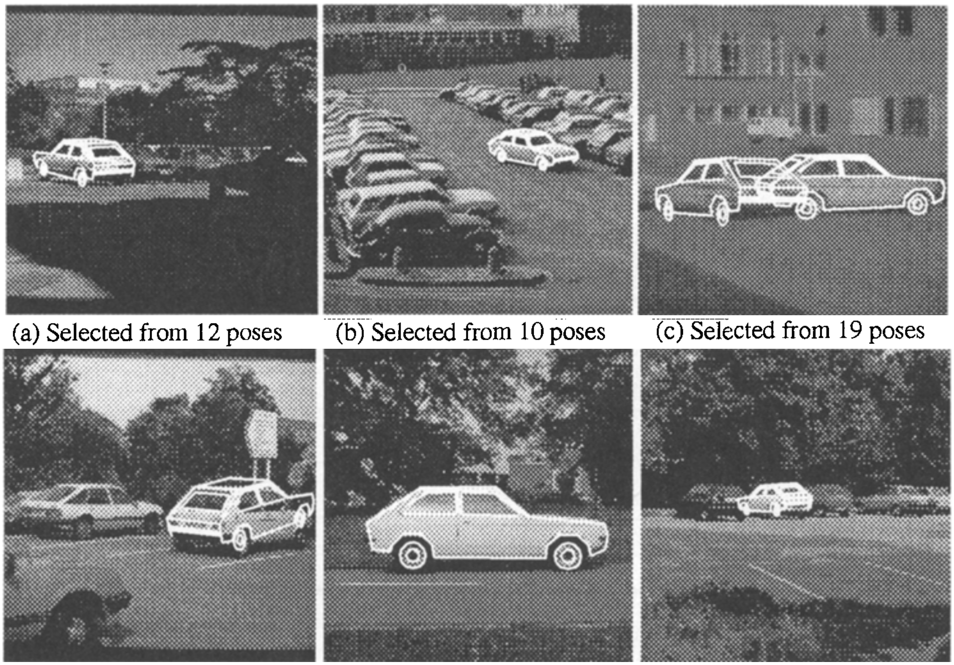
(g) Incorrect but consistent labellings (h) Pose based on nfw in (g) (i) Pose based on nrw in (g)

(j) Inconsistent labellings (k) Pose based on ws in (j) (l) Pose based on ofw in (j)

Fig.7. Hypothesis generation process

(k), (l)). In the case here only two of the 10 hypotheses are retained by this requirement (Fig. 7 (d) & (g)), giving two pairs of very similar possible poses. These accepted hypotheses must be subjected to further evaluation using view specific methods which are not discussed here. Details of the pose verification process can be found in Brisdon [4], and Worrall [10].

Fig.8 shows further examples of hypotheses superimposed on the images, which in theses examples have been selected manually from the few candidates. It should be noted that, as expected, the pose recovered is only approximate. The model can be used in the recognition of occluded objects (Fig.8(f)), as well as scenes containing multiple objects (Fig.8(c)). Table 1 shows the number of hypotheses generated against the size of the combinatorial search space. To make the comparison realistic, the search space quoted represents the number of possible triples of feature labellings, containing at least one 2-D feature - these could form a comparable basis for viewpoint inversion and subsequent view-dependent reasoning. It can be seen that the VIRM is very effective in identifying the very few labellings in the interpretation tree which are mutually consistent with a single view of the vehicle model.



(a) Selected from 12 poses (b) Selected from 10 poses (c) Selected from 19 poses

(d) Selected from 14 poses (e) Selected from 7 poses (f) Selected from 7 poses

Fig.8. Correct instances superimposed on a representative set of original images

Table 1: Number of hypotheses against the search space

Image	Number of Hypotheses	Number of Quadrilaterals	Number of U-shape Curves	Number of Line Segments	Size of Search Space
Fig. 8(a)	12	9	7	112	8.3×10^7
Fig. 8(b)	10	8	4	128	9.4×10^7
Fig. 8(c)	19	17	6	98	1.3×10^8
Fig. 8(d)	14	7	6	57	1.7×10^7
Fig.8(e)	7	2	2	112	1.7×10^7
Fig. 8(f)	7	6	8	121	6.2×10^7

4. Results and Discussion

We have built VIRMs for the three vehicles shown in Fig. 1.Table 2 summarises the result of the model building process. The results are obtained from 500 samples of the view with the camera upright and its position limited to within 0 and 60° of the model's ground plane. In representing the fastback car, an extra shape feature, a triangle, was introduced. The data change slightly each time the model is built because viewpoints are selected randomly, but we have found that this change is small and has no appreciable influence in the later object recognition process. The number of constraints generated in the model building process depends on the thresholds selected for the acceptance criteria. Such thresholds are inherent in any recognition problem, and must be determined by experience. However, in our experiments the effects of the thresholds on final recognition performance appear not to be dramatic, mainly affect the time used in recognition.

The time used to construct the relational model is high, since all the relations among the component parts of the object need to be assessed. At the present state of development the code runs in pop11 and we have made no attempt to make the code efficient. Model generation takes about one and a half hours on a Sun Sparc 2 with 24 MB memory. However, storage of the eventual VIRM is very efficient. We need an m by m matrix to represent the pairwise co-visibility of the object and a similar m_{2d} by m_{2d} by m matrix (m_{2d} is the number of 2-D complexes) to store co-visibility of feature clusters, and a set of procedures (typically 100 because a procedure may include more than one geometrical constraint) to represent geometrical relations among the component parts of the object.

Table 2: VIRMs for different types of vehicles

Number of	Hatchback	Fastback	Estate
1-D Features	22	26	28
2-D Features	10	12	12
Pairwise Co-visibility	107	134	152
Triple Co-visibility	300	424	468
Parallel Constraints	43	58	67
Colinear Constraints	21	25	31
Side Relation Constraints	72	85	85
Relative Size Constraints	35	42	49

5. Summary

A method has been described for creating a view-independent relational model of an object used in object recognition to aggregate features related to a pose hypothesis. A match is accepted as a hypothesis, and therefore will be further evaluated, only when its relational support passes a certain threshold. The model is created off-line and its use in object recognition requires no non-linear calculations.

Reference

1. Berge, C., *Graph and Hypergrapgh*, New York: North-Holland, 1973.
2. Bolles, R. C., Horaud, P., "3DPO: A Three Dimensional Part Orientation System", The International Journal of Robotics Research, Vol.5, No, 3, Fall 1986.
3. Bray, A. J., "Recognising and Tracking Polyhedral Objects", Ph. D Dissertation, University of Sussex, Oct., 1990.
4. Brisdon, K., Sullivan, G. D., Baker, K. D., "Feature Aggregation in Iconic Model Evaluation", Proc. of AVC-88, Manchester, Sept., 1988.
5. Canny, J. F., "A Computation Approach to Edge Detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8, No. 6, pp. 179-698, Nov., 1986.
6. Gigus, Z., Malik, J., "Computing the Aspect Graph for Line Drawings of Polyhedral Objects", IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-12, No. 2, pp. 113-122, Feb., 1990.
7. Goad, C., "Special Purpose Automatic Programming for 3D Model-Based Vision", Proceedings Image Understanding Workshop, Virginia, USA, pp.94-104,1983.
8. Koenderink, J., J., Von Door, A., J., "The Internal Representation of Solid Shape with Respect to Vision", Biological Cybernetics, Vol. 32, pp. 211-216, 1979.
9. Sullivan, G., D, "Alvey MMI-007 Vehicle Exemplar: Performance & LImitations", Proc. AVC-87, Cambridge, England, Sep., 1987.
10. Worrall, A. D., Baker, K. D. and Sullivan, G. D., "Model Based Perspective Inversion", Proc. AVC-88, Manchester, Aug., 1988.
11. Zhang, S., Du, L., Sullivan, G. D. and Baker, K. D., "Model-Based 3D Grouping by Using 2D Cues", Proc. BMVC90, Oxford, Sept., 1990.
12. Zhang, S., Sullivan, G.D., and Baker, K. D., "Relational Model Construction and 3D Object Recognition", Proc. BMVC91, Glasgow, Sept., 1991.