# Performance Assessment of Model-based Tracking

K. D. Baker and G. D. Sullivan

Department of Computer Science,
University of Reading, RG6 2AY, UK.

## Abstract

*Model-based vision techniques, originally developed for the recognition and pose recovery of vehicles in a single image, are used here to track vehicles through a sequence of images. Knowledge of the position of the camera with respect to the ground plane is used to reduce the search space of possible vehicle positions from six dimensions to three. The expected dynamics of vehicles are expressed in a Kalman filter, which predicts the likely poses in successive frames and provides a smoothed description of the vehicles' motion.*

*The notion of equivalence classes defined by a search of the parameter space is developed as an indicator of the performance of the pose-refinement sub-system. The system is illustrated and assessed by using the size of the correct class as a performance measure.*

## 1: Introduction

The work reported in this paper is part of a European vision research programme, Esprit P2152: VIEWS "Visual Interpretation and Evaluation of Wide-area Scenes", which seeks to combine the recognition and situation assessment of traffic in natural daylight scenes. In its current state the research has successfully demonstrated methods for model-based tracking of moving vehicles in long sequences of monochrome video images obtained from a single camera viewpoint. The project has also demonstrated in principle the recovery of high level behavioural descriptions of the vehicle movements and interactions as part of a situation assessment of the observed scene. A longer term objective of this research is the integration of these two aspects of scene perception into a real-time vision system for surveillance applications.

Two application domains have been addressed: road traffic and ground traffic at airports. In the former, the task is to recover the pose and tracks of vehicles for the purposes of traffic census, incident analysis and detection of traffic violations. In airports, the main purpose is to identify and track aircraft and ancillary service vehicles, and to monitor their conformance with servicing schedules and taxi-ing instructions from the ground traffic controller. The experimental image sequences obtained in these two domains contain a level of visual complexity to be expected in a wide range of similar applications.

Our work takes a strongly knowledge-based approach to the solution of specific vision problems. By doing so we are able to avoid many of the problems of general vision and yet still provide very useful vision systems. Practical applications of traffic monitoring systems will normally have access to a great deal of contextual information about the perceptual task in hand, such as the lay-out of the scene, the position of the camera(s) with respect to the scene, the expected routes of vehicles in the scene, and the expected types of vehicles. Such *a priori* knowledge may be highly specific, as in the airport monitoring task, or relatively uncertain, as in the census of road junctions. In either case, the proper use of expectation can greatly simplify the visual problem, and we show below that we can achieve very useful performance with quite modest computing facilities.

In recent years a variety of model-based methods for object tracking have been reported in the literature [1, 2, 3, 4, 5]. In all these cases the images used were of indoor laboratory scenes, and dealt with the tracking of relatively simple objects. In our work, model-based methods are used to classify and track natural objects (cars and other moving vehicles) in uncontrolled, cluttered outdoor scenes. The complexity of the methods we have had to employ has created a need for quantitative performance criteria to guide the design of the system.

In this paper we briefly review the model-based object recognition methods, and then report a method for analysing the effectiveness of iterative 3D pose-recovery algorithms. This allows the many design alternatives to be investigated and compared, and provides measures of performance which are essential prerequisites of any system if it is to form part of well-engineered products.

## 2: Overview of methods

The geometrical reasoning used in our tracking algorithms is based on extended wire-frame models, and is essentially 3D. This overcomes in a natural way many of the classical problems in vision, such as size and shape



0-8186-2840-5/92 $03.00 © 1992 IEEE

invariance, and partial occlusion. The system is initiated by means of an approximate pose, derived from analysis of regions of coherent movement in the image. All subsequent image analysis is then performed "top-down" under the control of the current pose hypothesis.

Each position and orientation of an object can be instantiated into the scene and a goodness-of-fit score evaluated. The initial pose estimate is refined by searching locally for a maximum of the evaluator function. Once a maximum score is found, which meets specific acceptance criteria, it is taken as an estimate of the true position and orientation of the object. The refined estimate of pose is then used as an input measurement for a Kalman filter, which imposes dynamic constraints on the visual interpretation. The updated filter state is used to predict a provisional position and orientation for the same object in the next frame of the sequence, and the process is repeated. Thus, given an initial frame in the sequence, an object to track and an initial estimate of its position we continue to track the object in subsequent frames while at all times maintaining an understanding of its location and direction of travel in the three dimensional scene.

In this paper we do not discuss the problem of how the model is initially selected or how the initial estimate of the position and orientation of the object is obtained. In the VIEWS project this information is derived from the analysis of coherently moving image features. Their position and shape attributes in the image are used to "cue" the model-based processes [6]. Other methods may also be used, such as seeking view-independent object-specific cues in edge-descriptions of the images[7].

### 2.1: Iconic evaluation

The object model comprises a set of line features specified in a three dimensional object-centred coordinate reference frame. On instantiation, the model is translated and rotated to the appropriate position in the scene model and finally each line which is visible from the given camera position is projected onto the image. The evidence in the image for each visible line feature is then assessed, using a process which we have called "iconic evaluation". The scores from the evaluation of individual lines are then aggregated to give an overall score for the model in the given position.

The iconic evaluator has previously been described in [8, 9]. Its essential features are as follows. The visible lines of the instantiated wire-frame vehicle model determine a set of lines in the image plane. For each line, we derive an average cross-section of the image, by integrating parallel to the line for a series of points on the normal, close to the line. Object features are associated either with edge-like or bar-like properties. In the former case, the absolute maximum of the derivative along the integrated normal is found; in the latter case the absolute difference between the positive and negative maxima of the derivative is found. These provide measures of the match of an individual predicted edge or bar with the image.

The evidence from all the visible lines is pooled, by expressing the scores of individual features as probabilities, which take account of the differences in length and type of the features. Probability tables are established empirically, by placing lines and bars of several different lengths randomly in a calibration image (ideally the image under investigation, but in practice a previously computed temporal median image), and determining the score. These tables are interpolated to give probability estimates for features of any length. Any given score can thus be associated with a probability that a score at least as high would have been obtained by chance from a randomly placed feature of equivalent type and length.

The individual model features are treated as independent samples from the underlying probability distributions†, and their probabilities are pooled using a $\chi^2$ test. The result is a single scalar in $\chi^2$ units, having expectation 0, and typically ranging from -3 (in areas of the image with far less than average detail) to 4 or 5 when the projected model fits the image very well. Scores over 2 are treated as significant indications of the presence of a vehicle.

The evaluation score is reasonably independent of the position and pose of the object, since it measures (to a first approximation) how likely the score was to have been obtained by chance, taking into account the number of visible features, and their lengths in the image.

### 2.2: Pose refinement

The evaluation score defines a scalar function of six variables - in world coordinates these are most simply defined as the three cartesian coordinates of the object's position and the three angles needed to specify its orientation. In general, we expect that peaks in this function will indicate likely matches between the model and the image. The problem is to locate the most prominent peak, and thereby to determine the pose of the vehicle.

A considerable computational simplification can be made by introducing a constraint limiting the object's position to the ground plane thus only permitting two dimensions of translation and one of rotation about the vertical axis. Using these simple but (normally) realistic physical assumptions, only three independent dimensions remain unspecified. Even so, an exhaustive search over

† This assumption is obviously false (since the presence of one image feature due to a car will be strongly correlated with the presence of others), however it is very difficult to take conditional probabilities into account, and we have found the method to be robust in practice.

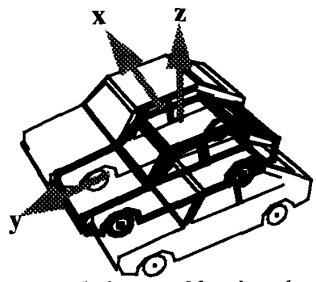



*Figure 1(A):For each degree of freedom the model is displaced, instantiated and its fit evaluated*

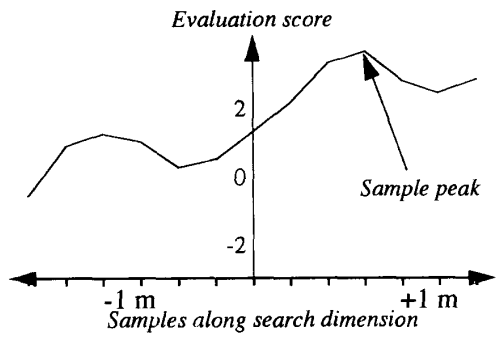



*Figure 1(B): Evaluation scores obtained by displacing the model along a single dimension*

three degrees of freedom is computationally too expensive. We have therefore developed an iterative method which successively decomposes the problem into three separate one dimensional searches, each based on the current estimate of the object's coordinate frame.

## 2.3: Separated ascent

The separated gradient ascent algorithm is applied iteratively to find peaks in the evaluation function. Each iteration consists of three separate one dimensional searches, each search taking a number of samples either side of the initial position. This is illustrated for the x-coordinate (the left-to-right axis of the car) in Figure 1(A). For each sample the model is displaced by an appropriate amount (in the object coordinate frame), instantiated into the image, and its fit to the image is evaluated as described in section 2.1. The results obtained for a typical search along a single dimension are shown in Figure 1(B), where the abscissa represents the displacement and the ordinate represents the evaluation score obtained (the higher the score the better the fit). Note the subsidiary peak, due to accidental alignment of one side of the car model with the wrong side in the image. The best score, and its position, are noted and the process is repeated for the other two variables - here, y and rotation about z (i.e. $\theta$) - each time

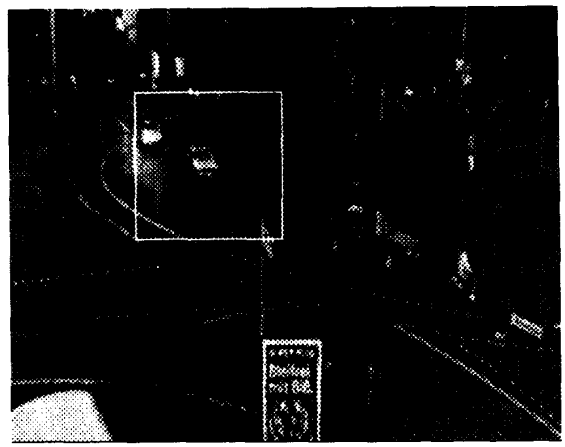
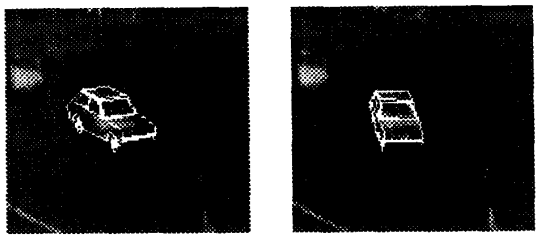


*Figure 2: Roundabout scene, frame 00200*

starting from the same initial pose. When all three dimensions have been searched the pose having the highest score is adopted as the initial position for the next iteration, and the search coordinate frame is changed accordingly. If no higher score is found then the three ranges of the search are reduced, to be equal to the previous sampling interval. The process is repeated until all the sampling intervals are below criterion values (see below)

The initial search range, and the terminating conditions are object- and pose-dependant. To establish these, the object model is approximated as a sphere of diameter equal to the greatest diameter of the model. Simple geometry uses the "seed" pose (in the world-coordinated frame) to compute the displacements in x, y and $\theta$ (in the object-coordinate frame) which would cause a 1 pixel change in the image for a worst-case object feature. These determine the terminating condition, which can be weighted to give the accuracy required by the application. We typically use initial search areas corresponding to +/-0.5 times the dimensions of the vehicle and +/-10 deg, and use a1.0 pixel terminating condition. At extreme distances, or in poses in which one of the object coordinate axes is directed towards the camera, the initial ranges may already imply a sampling interval below the terminating conditions - in this case only one iteration is performed for that axis.

The model instantiation process comprises two main parts: perspective projection of the 3D wire-frame model, and hidden line removal. The latter is far more expensive to compute. Very considerable savings can be made by exploiting the fact that (except near view catastrophes) the visibility of lines is only weakly affected by small perturbations of pose. We therefore update the visibility calculations on a per seed, per iteration, or per evaluation basis according to circumstances, in order to trade off accuracy against computational speed.



A: frame 200 B: frame 200

*Figure 3: (A) Before and (B) after gradient ascent.*

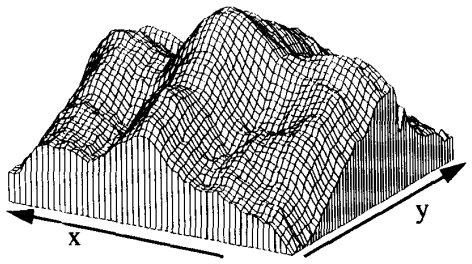



*Figure 4: Evaluation score plotted against x and y (see Figure 1).*

## 3: Assessment of pose recovery

A number of experiments have been carried out to explore the sensitivity and selectivity of the iconic evaluation process. Figure 2 shows a typical image from a test site at a roundabout in Germany. These images are 512*512 pixels, each resolved to 8 bits of intensity. To illustrate the methods, we concentrate on the car in the area of interest shown boxed.

### 3.1: Evaluator performance

Figure 3(A) shows a model that was very approximately instantiated near the vehicle by hand. This is fairly typical of an initial estimate of the position obtained from simple image-based cuing processes. The result of the separated ascent algorithm is shown in Figure 3(B). Informally, by eye, it seems that the fit is very good.

Figure 4 shows the evaluation scores for a two dimensional slice through the three dimensional search space. In this case the slice is in the plane of the two translations, x and y in the object coordinate frame, and the ranges of translation of the model are ±3.2 metres about the peak. It should be noted that the evaluation function is strongly ridged in the y direction (along the front-to-back axis of the vehicle). From the viewing position used, displacements in y are approximately in the line of sight, and cause relatively little change in the image. The exact shape of the evaluation function is of course strongly pose- and view-dependent.

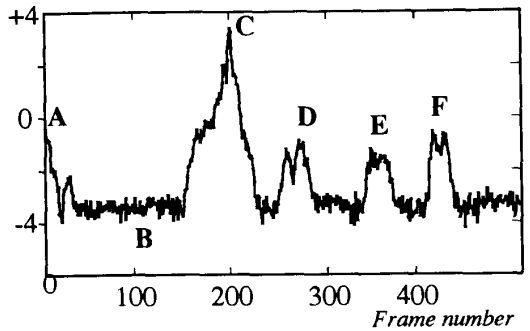



*Figure 4: Evaluation function for a fixed position on different images (A-F refer to Figure 6).*

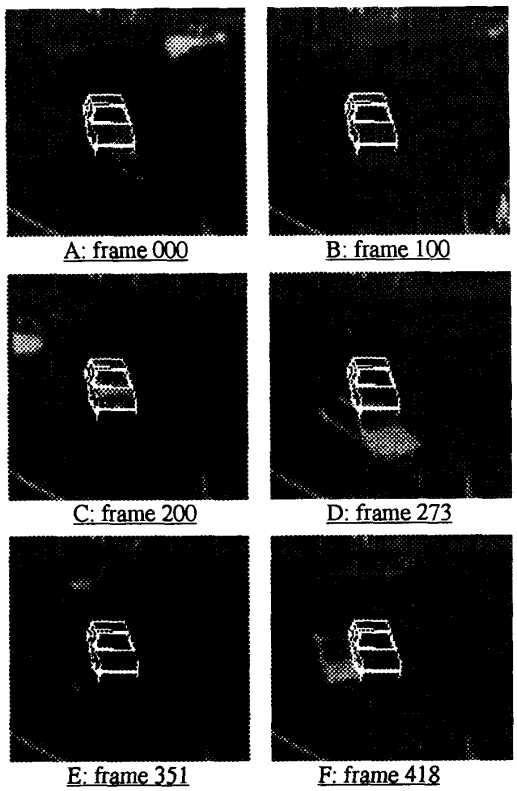



*Figure 6: Fixed model in different frames (A-F refer to Figure 5).*

To estimate the overall signal-to-noise ratio of the method, we have taken the best fitting position and orientation for the model in this image (as illustrated in Figure 3(B)), and then evaluated this particular model instance in a sequence of 500 images taken at 25 Hz. The results are shown in Figure 5. We see one conspicuous peak, with several subsidiary peaks. The images

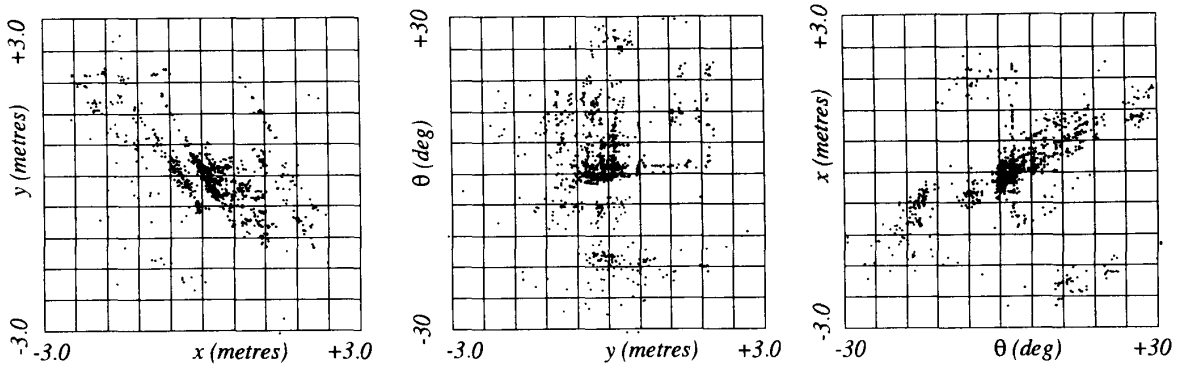



*Figure 7: Orthographic projections of sampled results*

corresponding to the points on the function labelled A to F are shown in Figure 6(A-F).

The response of the evaluator is typically about -3.5 in the absence of vehicles. The minor peaks correspond to accidental alignments between the model and a "wrong" vehicle, but these scores never exceed 0, and the peaks are fairly flat and ragged. The score for the "correct" vehicle reaches 3.7, and is well-localised. In this simple case, where there is little distracting background image detail, the evaluator provides an intuitively acceptable measure, and has good signal to noise ratio.

## 4: Assessment of search methods

A continuing problem in object recognition research is the need to compare different algorithms, both to compare methods which differ fundamentally, and also to assess the impact of the many design options within a given approach. This section describes methods developed within the VIEWS project to quantify the performance of the overall pose-recovery sub-system.

The performance of an iterative maximisation algorithm is mainly determined by (i) the smoothness of the function, (ii) the prominence of the maximum amongst (false) local maxima, (iii) the search routine used, and (iv) the initial (seed) pose. The first three factors are, in turn, affected by many free parameters of the algorithms, which must be established in the course of the implementation. To do this efficiently and effectively we need to define a summary measure of performance.

One such measure can be obtained by studying the range of seed positions which converge to the correct answer. In principal, a search algorithm defines an equivalence relationship on the domain of seed positions, in which all seeds giving rise to the same result form an equivalence class. In practice, we must tolerate a small error in the result, to take into account inaccuracy in the resulting poses, due to fine-grain noise in the evaluation function and (more importantly) the terminating conditions adopted by the algorithm. One such equivalence class can be identified (by eye) as correct. The size, convexity and compactness of the correct class provides important measures of the quality of the search algorithm.

Figure 7 shows the results of trials in which the search space (x, y, θ) was sampled to provide seed positions within a range close to the correct result. The search algorithm was run and the recovered pose was identified. The resulting points in (x, y, θ) are shown in 3 orthogonal projections in Figure 7, as scatter diagrams. We see a strong cluster of points around the origin (the correct pose), with significant clusters at false maxima.

These data can be used to estimate the size of the correct equivalence class, by computing the probability of success as a function of the initial displacement of the seed. A simple measure of performance is then given by the distance at which the probability falls to a criterion value (say 50%). This approach depends on a distance metric ($r$) which normalises the three axes. At present we scale the axes so that the scatter of points in diagrams such as Figure 7 seem fairly isotropic, but a more formal statistical approach to normalising the axes could be adopted.

The data for Figure 7 were derived from the image shown in Figure 8(A). Figure 8(B) shows the data plotted as probabilities of success as a function of $r$, for three different search algorithms: (i) the separated ascent (section 2.3), (ii) the simplex algorithm [10], and (iii) steepest ascent. Figure 8(C) shows the average computational cost as a function of $r$; as usual with such algorithms, this is dominated by the number of computations of the evaluation function.

The results shows that the steepest ascent algorithm performs very badly, and only rarely recovers from seed positions beyond 0.5m or 5deg. The Simplex algorithm achieves far better performance, mostly at a lower cost. The separated ascent algorithm has somewhat better convergence properties, but at considerably greater cost. For this image and object model, we obtained the criterion

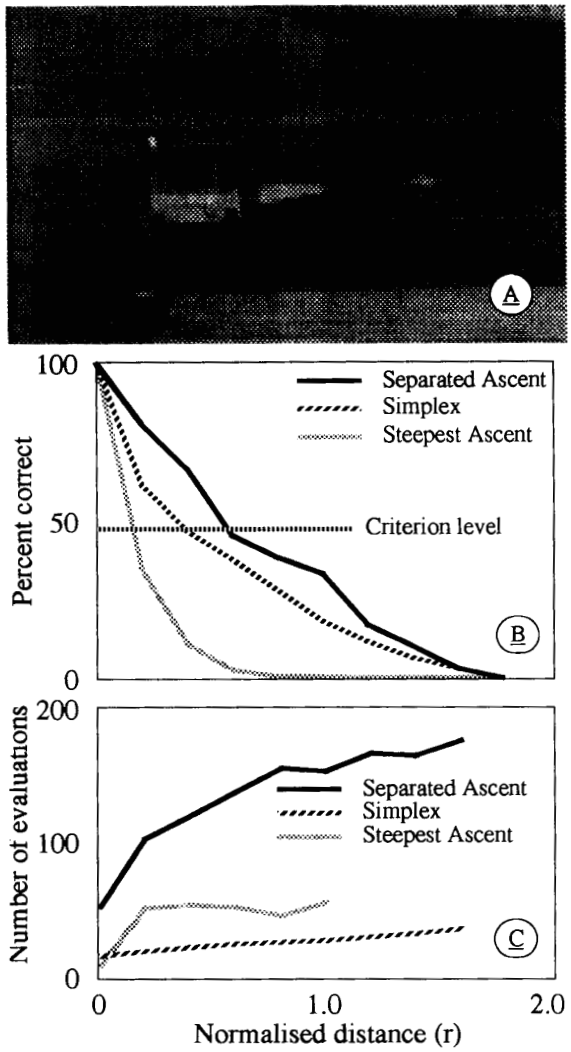



*Figure 8:* *(A) Segment of uncluttered image*
*(B) Probability of correct convergence*
*(C) Average number of evaluations.*
*In both graphs 1.0 on the abscissa represents 2.0 metres or 20 deg.*

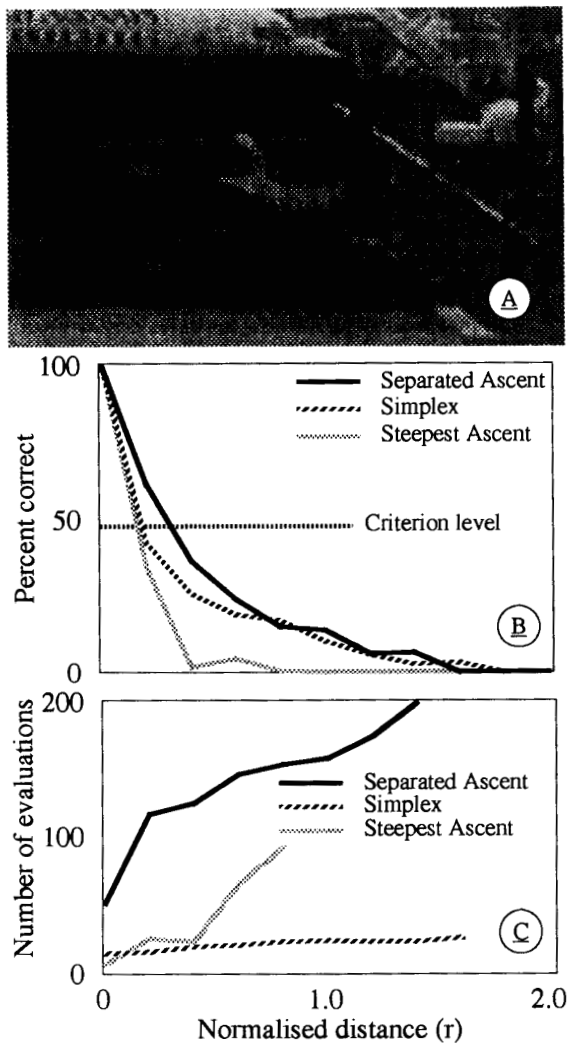



*Figure 9:* *Cluttered image - as Figure 8.*

50% success rate for seed errors equivalent to 0.3m or 3deg, 0.7m or 7 deg, and 1.1m or 11 deg, for the three search algorithms respectively.

Results such as these are strongly dependent on the object and image under consideration. The vehicle in Figure 8(A) is well isolated from distracting image detail. Figure 9(A) shows results obtained when the same model is fitted to a second object in a part of the image where there is considerably more irrelevant detail. The convergence properties of all three algorithms is significantly poorer, but the same preference order of performance occurs.

This method of comparing search algorithms has also been used to assess other minor changes to the pose-recovery method. Systematic analyses are currently being made of alternative methods for feature evaluation and the method used for updating the visibility calculations (Section 2.1), as well as the significance of individual features of the model, the feature aggregation method, and the sensitivity of the system to small geometrical errors between the model and the object. It also provides us with a formal measurement of the performance of the system as a function of object types, viewing distance, and occlusion.

These data are fundamental to any principled design of a surveillance system for practical use

## 5: Model-based tracking

The method for pose-refinement starting from an approximate seed pose has been used to track vehicles in video sequences. For each new image, the pose recovered from the iconic search is used as an input measurement to a Kalman filter, which imposes simple dynamic constraints appropriate to vehicles.

Instantaneously, a car has only two degrees of freedom - its forward (or backward) velocity, $v$, and its turning angle. We have implemented a Kalman filter with freedom in v and $\dot{v}$ (the forward velocity and its derivative with respect to time), and $\alpha$ and $\dot{\alpha}$ (the rotational angle about the model's z axis, and its derivative). The three parameters v, $\dot{v}$ and $\dot{\alpha}$ are constrained to be within plausible bounds - since the filter is expressed in the object-centred coordinate frame, these constraints may be estimated by using common knowledge of car dynamics. Note that this characterisation of the vehicle's dynamics prohibits it from sliding sideways (but doesn't prevent spinning).The measurement error assumed in the Kalman filter is also pose-dependent, and is estimated on the same basis as the terminating conditions for the pose refinement process (cf. Section 1.4). We typically assume that the error of a recovered pose is gaussian distributed with a standard error equivalent to a 1 pixel worst-case displacement.

Tracking proceeds as follows. The initial "cued" pose hypothesis is refined (as in Section 2.3). Starting with plausible default values, the Kalman filtered estimates of $v$ and $\alpha$, together with the newly measured position, provide a prediction for the object's position in the next frame to be examined. This becomes the seed pose for the next search, and the process continues.

When the filter parameters become stable, the forward prediction improves in accuracy, and fewer iterations of the maximisation algorithm are required. In these conditions we have found that the less costly simplex algorithm is sufficient. It also becomes possible to predict several frames in advance and thus reduce the computational burden of the process (though there is a trade-off with the requisite size of the initial search space). We have found good performance when tracking in images of traffic taken at 5 Hz.

## 6: Experimental results

The Kalman-filtered iconic search has been used to track cars in the mid- and fore-ground of the scene shown in Figure 2. The initial positions of the models were fitted very approximately by eye as each vehicle first came into view as in Figure 3(A), but usually at a considerably further distance from the camera. Each car was then tracked through the image sequence (at 25 Hz).

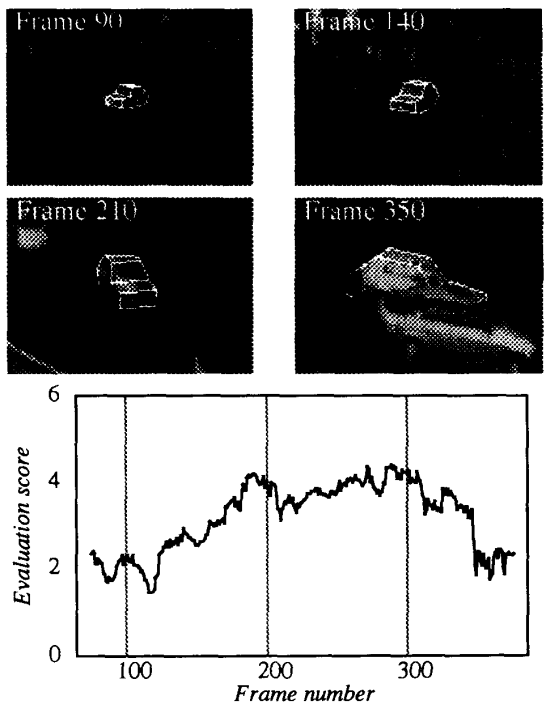



*Figure 10: Evaluator score*

The results are illustrated for the vehicle discussed in Section 2 in Figure 10, which shows the score of the evaluator obtained for each of the frames in the sequence. It demonstrates that our feature pooling method shows good shape and size invariance: the evaluation score compensates well for radical changes in the appearance of the images as the car rotates and approaches the camera. Occlusion by stationary objects such as lamp posts, as well as other tracked vehicles, was taken into account in computing feature visibility. It should be noted that all these scores are well outside the typical subsidiary peaks shown in Figure 5. [The drop in the scores obtained beyond frame 350 are due to the fact that in this case our implementation did not take proper account of the edges of the field of view of the camera.]

Vehicles from many different traffic sequences have been tracked in this way, and their positions have been recovered with respect to a plan view of the scene model. The method has been applied to traffic scenes containing cars, lorries, aircraft and airport ancillary vehicles. Figure 11 shows a still from a sequence of airport ground traffic, together with the recovered positions on the airport model.

## 7: Conclusion

Model-based vision techniques, originally developed for the recognition and recovery of the pose of vehicles in

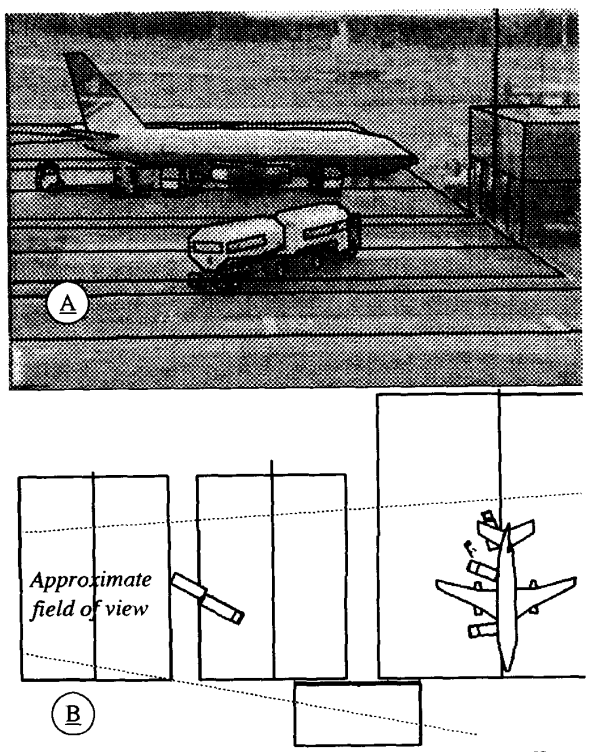



*Figure 11: Example of airport ground-traffic application: the aircraft, two baggage-handlers, a fork-lift truck, a fuel tanker and trailer and several ancillary vehicles were tracked through a 20 min. video sequence taken at 5 Hz.*
*(A) recognised objects instantiated onto the image*
*(B) recovered positions on the ground-plane*

single images, have been applied to the problem of tracking vehicles through sequences of images. Knowledge of the camera calibration and of the position of the ground plane is used to reduce the search space of possible vehicle positions from six dimensions to three. The Kalman filter mechanism provides dynamic stability in the recovered pose over time and imposed physically plausible behaviour to obtain the seed poses for the iconic search.

There are several particular merits of the approach we have adopted.

- We make effective use of contextual knowledge of the scene and camera position. Such information can be expected to be available in realistic applications of machine vision for traffic analysis.
- Pose refinement by means of iconic evaluation obviates the need for "knowledge-free" image processing and subsequent feature analysis. In particular, all feature grouping is done top-down, and the combinatorial problems of feature labelling common to other model-based methods are avoided.
- All geometrical reasoning is in 3D, so that occlusions due to known objects can easily be taken into account. Furthermore, the poses are recovered with respect to the scene model, i.e. in the known 3D world.
- Constraints on vehicle dynamics are modelled explicitly, allowing the introduction of high-level expectations of "car-ness".

The current implementation is very close to working in real-time. Using per-frame visibility calculations and the separated ascent algorithm (Section 2.3) single vehicles can currently be tracked in image sequences taken at 5 Hz on a single SUN Sparc2, at 1-2 Hz, depending on the complexity of the model. [The simplex algorithm gives slightly less stable performance at 2-5Hz.] The control code is written in pop11, and most low-level code is written in C. During the search around 80 iconic evaluations are typically performed and this represents approximately 85% of the computational cost. Work is currently planned to develop hardware support for the evaluator, and the system is then expected comfortably to exceed the requirements for 25 Hz operation.

A major problem in developing complex vision systems such as that of the VIEWS project has been to establish methods for assessing the effects of minor changes in parameters and component algorithms. The analysis of pose refinement performance described in Section 4 has provided a valuable tool for comparing alternative algorithms. It could also be applied to other work on similar problems, to help establish quantitative criteria with which to assess vision systems.

## 8: References

1 Bray A.J. Tracking Objects Using Image Disparity, Image and Vision Computing, Vol. 8, No. 1, Feb. 1990, pp4-9.

2 Stevens R. S. Real-time 3D object tracking, Image and Vision Computing, Vol. 8, No. 1, Feb. 1990. pp91-96.

3 Harris C. & Stennett C. Rapid - A Video Rate Object Tracker, Proc. BMVC-90, Oxford, 1990 pp73-77.

4 Lowe D. Fitting Parametrized 3-D Models to Images, IEEE-TPAMI, Vol. 13, No. 5, 1991 pp 441-450.

5 Koller D, Daniilidis K, Thorhallson T and Nagel H-H. Model-based object tracking in traffic scenes. Proc ECCV92, pp437-452 (Springer-Verlag)

6 Hussain Z., Godden R., Sullivan G. D., Worrall A. D. & Marslin R. F. D102: Knowledge-based Image Processing, ESPRIT P2152 report PM-03-CEC.D102-01. Dec. 1990.

7 Zhang S, Sullivan G D and Baker K D Automatic Construction of a View-independent Relational Model for 3D Object Recognition T-PAMI (to appear)

8 Brisdon K., Sullivan G. D. & Baker K. D. Feature Aggregation in Iconic Model Matching, Proc Alvey Vision Conference, AVC-88, Manchester, 1988, pp19-24.

9 Brisdon K., *Hypothesis Verification using Iconic Matching* Doctoral Thesis, University of Reading, November 1990.

10 Press W H, et al *Numerical Recipes*; Cambridge University Press, 1986.