Pose Determination and Recognition of Vehicles in Traffic Scenes

T N Tan, G D Sullivan and K D Baker

Department of Computer Science, University of Reading,
Reading, Berkshire RG6 2AY, England

Abstract. This paper concerns the pose determination and recognition of vehicles in traffic scenes, which under normal conditions stand on the ground-plane. Novel linear and closed-form algorithms are described for pose determination from an arbitrary number of known line matches. A form of the generalised Hough transform is used in conjuction with explicit probability-based voting models to find consistent matches. The algorithms are fast and robust. They cope well with complex outdoor scenes.

1 Introduction

In many practical applications of computer vision, the objects to be recognised are constrained to be in contact with a known plane. In this paper we are concerned with the localization and recognition of vehicles in traffic scenes from monochromatic image sequences recorded by one or more stationary calibrated cameras, where under normal conditions vehicles stand on the ground-plane (GP). Although our primary interest is in traffic scene analysis, other similar applications such as the recognition of objects on a table, or parts on a conveyor belt, are commonplace. The ground-plane constraint (GPC) reduces the number of degrees of freedom of a rigid object from 6 to 3; these are most simply parameterised as the position (X, Y) on the GP and the orientation (θ) about the vertical axis.

We show in this paper that the GPC significantly simplifies pose determination based on 2D image to 3D model line matches (called, here 2D-3D line matches). The simplicity of the pose constraints and their solutions makes a form of Hough transform an appropriate choice for establishing correspondences between image and model line segments.

2 Non-iterative Pose Estimation from Line Matches

We first discuss pose determination using a set of known 2D-3D line matches. Assume an image line S and a model line M form a match. Then from the known equation of the image line, one can easily derive the equation of the so-called interpretation plane Π in which the model line M must lie. The necessary and sufficient conditions for M to lie in Π are [3-4]

- The direction vector of M is perpendicular to the normal vector of Π; and
- A known point on M lies in Π.

The above two conditions, in conjunction with the GPC, lead to the following

Lecture Notes in Computer Science, Vol. 800
Jan-Olof Eklundh (Ed.)
Computer Vision – ECCV '94

two constraints on the three pose parameters X, Y and θ (see [3] for details):

$$\begin{aligned} F\cos\theta + G\sin\theta &= H \\ A\cos\theta + B\sin\theta + CX + DY &= E \end{aligned} \tag{1}$$

where A, B, C, D, E, F, G and H are terms computable from known variables [3]. Therefore, for a set of N 2D-3D line matches, the three pose parameters are constrained by a total of $2N$ equations:

$$\begin{aligned} F_i\cos\theta + G_i\sin\theta &= H_i, \quad i = 1, 2, \ldots, N \\ A_i\cos\theta + B_i\sin\theta + C_iX + D_iY &= E_i, \quad i = 1, 2, \ldots, N \end{aligned} \tag{2}$$

The number of independent equations in (2) depends on the line configurations, and lies between N and $2N$ [5].

The system of $2N$ equations in (2) may be solved in a number of ways. Here we outline two non-iterative closed-form solutions.

2.1 Linear Least Squares Solution (LLS)

The linear solution is made possible by treating $\cos\theta$ and $\sin\theta$ as two independent unknowns. The equations in (2) can be written in matrix form as

$$\boldsymbol{P}q = \boldsymbol{Q} \tag{3}$$

where $q = (\cos\theta \;\; \sin\theta \;\; X \;\; Y)^T$ is the unknown vector, and $\boldsymbol{P}$ and $\boldsymbol{Q}$ are the appropriate coefficient matrices. Equation (3) can easily be solved using pseudo-inversion to obtain the LLS solution.

2.2 Non-Linear Closed-Form Least Squares Solution (NLS)

A non-linear least squares technique is outlined in the following which does not neglect the trigonometric constraint and involves no iterations. We rewrite (3) as

$$\boldsymbol{P}_1q_1 + \boldsymbol{P}_2q_2 = \boldsymbol{Q} \tag{4}$$

where $q_1 = (\cos\theta \;\; \sin\theta)^T$, $q_2 = (X \;\; Y)^T$, and $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ are appropriate sub-matrices of $\boldsymbol{P}$. The least squares solution of (4) is then given by

$$q_1, q_2 = \arg\{ \min_{q_1, q_2} \|\boldsymbol{P}_1q_1 + \boldsymbol{P}_2q_2 - \boldsymbol{Q}\|^2 \}; \text{ subject to } \|q_1\|^2 = 1 \tag{5}$$

The above minimisation problem is solved by using a Lagrange multiplier λ. After some tedious manipulations, the minimisation is mapped into the solution of a fourth-order polynomial equation on λ whose roots can be solved in closed-form. Once the Lagrange multiplier is obtained, the optimal q_1 and q_2 can easily be computed.

The performance of the techniques outlined in this section has been studied by numerous Monte Carlo simulations, and tested by pose recovery in real outdoor traffic scenes. The experimental results have confirmed the robustness of the techniques (detailed discussion on performance is given in [6]). It should be pointed out that while in the general case of 6 dof there are no closed-form pose solutions from an arbitrary number of line matches [1], we have shown in this section that the GPC allows simple and robust closed-form pose recovery from line matches.

3 Matching and Recognition

The pose determination algorithms described in the preceding section assume known correspondences between image and model lines. We now discuss how to establish such correspondences. Although a number of matching schemes (e.g. the IT scheme [8]) may be tailored to make use of the GPC, there are good reasons for choosing the generalised Hough transform (GHT) [6].

3.1 Pose from a Single Noisy Line Match

The first equation of (1) does not involve the translation parameters, and can easily be solved to obtain the orientation parameter. When data is perfect, this gives two values for the rotation angle. When data is noisy, the two discrete values are replaced by two orientation intervals where the PDFs of the rotation angle can be computed from explicit noise models [2, 6]. In many cases, one of the solutions can be eliminated because the model line would be occluded in the derived pose.

Once the orientation is known, the object is confined to slide along a *confusion line* on the GP defined by the second constraint of (1). The confusion line is bounded by requiring overlap between the image line and the projected model line.

3.2 Pose Clustering and Feature Matching

Pose clustering is decomposed into orientation histogram followed by GP location clustering. Image line segments are first extracted from the input image (see Fig.1(a)-(b)). The orientations recovered from all possible matches between image and model lines are then histogrammed according to the respective PDFs and the visibilities of the model lines [2]. Fig.1(c) shows the results obtained for Fig.1(b). Each peak in the histogram identifies the orientation (θ) of the model on the GP where multiple line matches agree.

For each orientation peak, the confusion lines of the line matches which are consistent with the peak orientation are then tallied in an (X, Y) accumulator. Local maxima in the accumulator are identified to recover the most likely locations of the object on the GP. In the case of an isolated vehicle such as Fig.1(a), we usually obtain one conspicuous peak as seen in Fig.1(d) (where darker points indicate higher accumulator values). Fig.1(e) shows the object instantiated in the image in this pose; the fit is very close. In our work, the goodness-of-fit between the model projection and the original image may also be measured by an evaluation score (see [7] for the computation of such scores). A high score indicates a good fit between the model projection and the image, and thus also signifies a good pose.

Once an initial pose is obtained, a clique of consistent line matches is identified by retaining those matches for which the distance between the confusion line segments and the GP location is less than a threshold. Fig.1(f) shows the image lines (thick lines) of the identified clique for Fig.1(b). Note, almost all visible lines of the vehicle which are modelled have been correctly matched. The clique of consistent matches is then used by the pose determination algorithms described in Section 2 to compute a more accurate object pose. Fig.1(g) shows the model instantiated at the pose returned by the NLS algorithm.

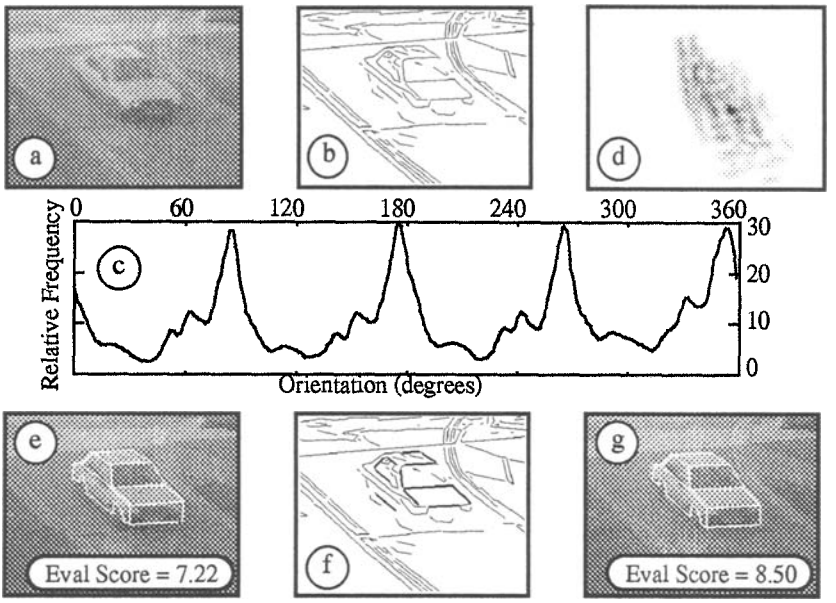

Fig. 1. (a) original image (350x270 pixels); (b) extracted line segments; (c) orientation histogram; (d) XY accumulator at global orientation peak (=175°); (e) initial peak pose from pose clustering; (f) matched line segments; (g) pose returned by the NLS algorithm.

3.3 Pose and Model Discrimination

The global peak of the orientation histogram and that of the (X, Y) accumulator do not always identify the correct pose. Usually a number of peaks in the histogram and the accumulator need to be located to identify a set of possible poses. The correct pose is then taken as the one which produces the highest evaluation score.

Similarly, to identify the correct model for the object, the evaluation scores obtained under different models are compared. The model which achieves the best score is taken as the correct model for the object. Experiments with outdoor traffic scenes have confirmed the discrimination capability of the algorithm [2].

3.4 Multiple Objects

The algorithm described in this section has also been applied to image regions which contain multiple occluding objects. An example is shown in Fig.2. Note, the

Fig. 2. (a) an image region containing multiple occluding objects; (b) extracted line segments; (c) and (d) recovered poses for the two occluding vehicles.

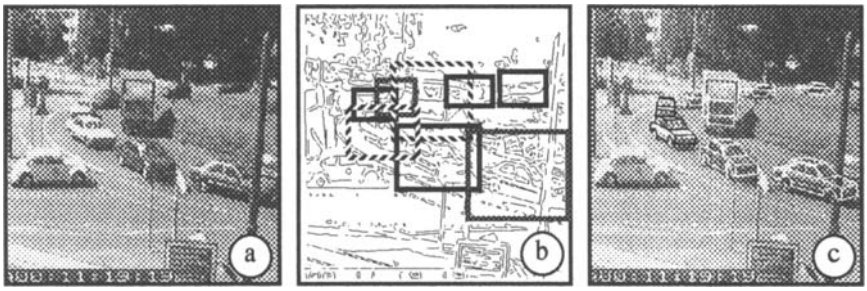

Fig. 3. (a) a wide area image (512x512 pixels); (b) extracted line segments and 8 regions of interest; (c) objects instantiated at the recovered poses and classes.

poses recovered for both vehicles appear to be very accurate. A more difficult case is given in Fig.3 where all objects have been correctly classified and located (the VW "Beetle" on the left of the image was not considered because of the lack of an appropriate 3D model).

4 Use of Multiple Cameras

In the examples discussed so far, only one static camera has been involved. In many practical machine vision applications (e.g., in wide area traffic monitoring and surveillance), the use of multiple (stationary) cameras is often advantageous or even essential [6]. Since all reasoning is in a fixed world coordinate system, the matching and recognition scheme outlined in the preceding section allows simple fusion of data from multiple cameras. For each camera, the orientation histogram is first computed. The individual histograms from all cameras are then aggregated to produce the overall orientation histogram. The overall (X, Y) accumulator is similarly obtained. The cliques of consistent matches from different cameras yield independent pose constraints which are collectively solved by the NLS algorithm.

Fig.4 shows an example of using multiple (two) cameras in a toy but fairly realistic traffic scene. Fig.4(a) and (b) depict the views (768x576 pixels) of the scene from two cameras (called here CD50 and F10 for reference). Each camera has "blind spots" due to occlusion by the buildings. The toy car would be extremely difficult (if possible) to locate were only F10 (Fig.4(b)) used. However, when data from both cameras are used, the location of the toy car proves to be straightforward as illustrated in Fig.4(c)-(f). The pose shown in Fig.4(e)-(f) corresponds to the global orientation peak of Fig.4(c) and the global accumulator peak of Fig.4(d).

5 Conclusions

Novel algorithms have been described for pose determination and recognition of vehicles in traffic scenes under the ground-plane constraint. Linear and closed-form algorithms for pose determination from an arbitrary number of line matches have been presented. The ground-plane constraint significantly reduces the pose redundancy of a line match, and a form of the generalised Hough transform has been used to find groups of consistent line matches. The algorithm is conceptually simple, fast, robust and inherently parallel, and has been shown to work well in

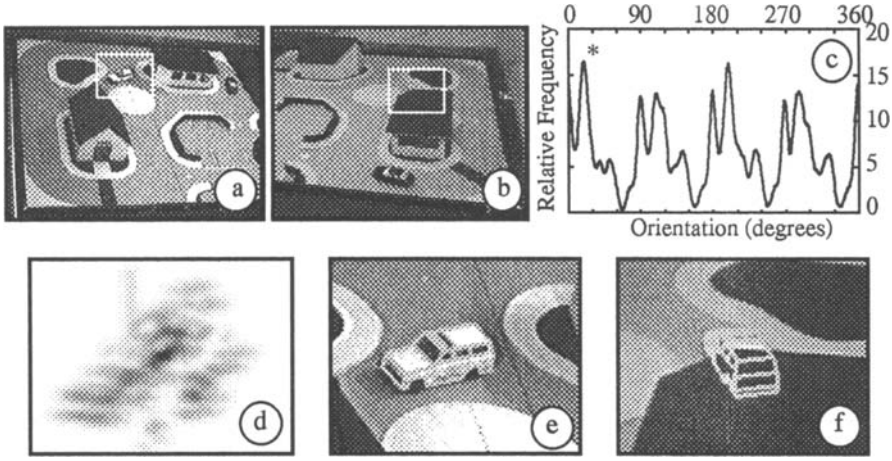

Fig. 4. Use of multiple cameras in object recognition. (a) toy scene in camera CD50; (b) toy scene in camera F10; (c) overall orientation histogram from both cameras (* marks the global peak); (d) overall (X,Y) accumulator at global peak orientation (=19°) of (c); (e) recovered pose in camera CD50; (f) recovered pose in camera F10.

routine images of out-door traffic scenes. Multiple calibrated cameras have been used to overcome the limitations of a single static camera, and the algorithms developed for a single camera can be extended to fuse data from multiple cameras in a simple and straightforward way.

References

1. O. D. Faugeras and M. Hebert, The representation, recognition, and locating of 3-D objects, Int. J. Robotics Res., vol.5, 1986, pp.27-52.
2. T. N. Tan, G. D. Sullivan and K. D. Baker, Recognising Objects on the Ground-plane, Proc. of 4th BMVC, 1993, pp.85-94.
3. T. N. Tan, G. D. Sullivan and K. D. Baker, Linear Algorithms for Object Pose Estimation, Proc.of 3rd BMVC, 1992, pp.600-609.
4. M. Dhome et. al., Determination of the Pose of an Articulated Object From a Single Perspective View, Proc. of 4th BMVC, 1993, pp.95-104.
5. T. N. Tan, G. D. Sullivan and K. D. Baker, Line-Based Object Scale and Pose Estimation, Proc. of ACCV, Osaka, Japan, November 1993, pp.347-350.
6. T. N. Tan, G. D. Sullivan and K. D. Baker, Pose Determination and Recognition of Vehicles in Traffic Scenes, Research Report, RUCS/RS/ISG/9317, University of Reading, October 1993.
7. G. D. Sullivan, Visual Interpretation of Known Objects in Constrained Scenes, Phil. Trans. R. Soc. Lond. B, vol.337, 1992, pp.361-370.
8. W. E. L. Grimson and T. Lozano-Perez, Localizing Overlapping Parts by Searching the Interpretation Tree, IEEE Trans. PAMI, vol.9, 1987, pp.469-482.