

Viewpoint Independent Detection of Vehicle Trajectories and Lane Geometry from Uncalibrated Traffic Surveillance Cameras

José Melo^{1,2}, Andrew Naftel¹, Alexandre Bernardino², José Santos-Victor²

¹University of Manchester Institute of Science and Technology, PO Box 88, Sackville Street, Manchester M60 1QD, UK
j.melo@postgrad.umist.ac.uk, andrew.naftel@co.umist.ac.uk

²Instituto Superior Técnico, Av. Rovisco Pais 1049 - 001 Lisboa, Portugal
{jpqm.alex.jasv}@isr.ist.utl.pt

Abstract. In this paper, we present a low-level object tracking system that produces accurate vehicle trajectories and estimates the lane geometry using uncalibrated traffic surveillance cameras. A novel algorithm known as Predictive Trajectory Merge-and-Split (PTMS) has been developed to detect partial or complete occlusions during object motion and hence update the number of objects in each tracked blob. This hybrid algorithm is based on the Kalman filter and a set of simple heuristics for temporal analysis. Some preliminary results are presented on the estimation of lane geometry through aggregation and K-means clustering of many individual vehicle trajectories modelled by polynomials of varying degree. We show how this process can be made insensitive to the presence of vehicle lane changes inherent in the data. An advantage of this approach is that estimation of lane geometry can be performed with non-stationary uncalibrated cameras.

1 Introduction

Intelligent traffic surveillance systems are assuming an increasingly important role in highway monitoring and city road management systems. Their purpose, amongst other things, is to provide statistical data on traffic activity such as monitoring vehicle density and to signal potentially abnormal situations.

This paper addresses the problem of vehicle segmentation and tracking, screening of partial and complete occlusions and generation of accurate vehicle trajectories when using non-stationary uncalibrated cameras such as operator controlled pan-tilt-zoom (PTZ) cameras. We demonstrate that by building a self-consistent aggregation of many individual trajectories and by taking into account vehicle lane changes, lane geometry can be estimated from uncalibrated but stable video sequences.

In our work, rather than performing object tracking under partial or total occlusion, we describe an occlusion reasoning approach that detects and counts the number of overlapped objects present in a segmented blob. Trajectory points are then classified according to whether they are generated by a single or overlapped object. This paper

describes the Predictive Trajectory Merge-and-Split (PTMS) algorithm for performing the aforementioned task. It uses a Kalman filter (KF) and a set of simple heuristic rules to enforce temporal consistency on merging and splitting overlapping objects within detected blobs. The method is independent of the camera viewpoint and requires no *a priori* calibration of the image sequences.

2 Review of Previous Work

The starting point for much work in analysing surveillance images is the segmentation of moving objects based on background subtraction methods [1-2]. Typically, each pixel is modelled using a Gaussian distribution built up over a sequence of individual frames and segmentation is then performed using an image differencing strategy. Shadow detection and elimination strategies have been commonly employed to remove extraneous segmented features [4-7].

It is also important to handle partial and complete occlusions in the video data stream [7-10]. Occlusion detection can be performed using an extended Kalman filter that predicts position and size of object bounding regions. Any discrepancy between the predicted and measured areas can be used to classify the type and extent of an occlusion [9], [10].

Higher level traffic analysis systems have also been developed specifically for accident detection at road intersections [9], [11] and estimating traffic speed [12], [13]. More general techniques for object path detection, classification and indexing have also been proposed [10], [14-17].

Our work is most closely related to [10], [12], [13]. In [12] an algorithm to estimate mean traffic speed using uncalibrated cameras is presented. It employs geometric constraints in the image, inter-frame vehicle motion and distribution of vehicle lengths. Traffic flow histograms and the image vanishing point are used in [13] to measure mean speed but it has similar limitations to the previous approach.

The work in this paper shows that accurate vehicle trajectories can be built from uncalibrated image sequences and can be aggregated to model lane geometry and ultimately determine traffic speed and classify normal and anomalous situations.

3 Predictive Trajectory Merge-and-Split (PTMS) Algorithm

The proposed system uses a multi-stage approach to determining the vehicle motion trajectories and eventually the lane geometry. Firstly, we build a background model to segment foreground objects. A detected foreground blob comprises a connected region having more than a certain pre-defined minimum number of pixels (K_{min}) in its area. A constant acceleration Kalman Filter (KF) is used to track the blobs through image coordinate space. The PTMS algorithm is then used to perform a time-consistent analysis of those detected blobs allowing for merging and splitting due to partial and complete occlusions. An overview of the system is shown in Fig. 1.

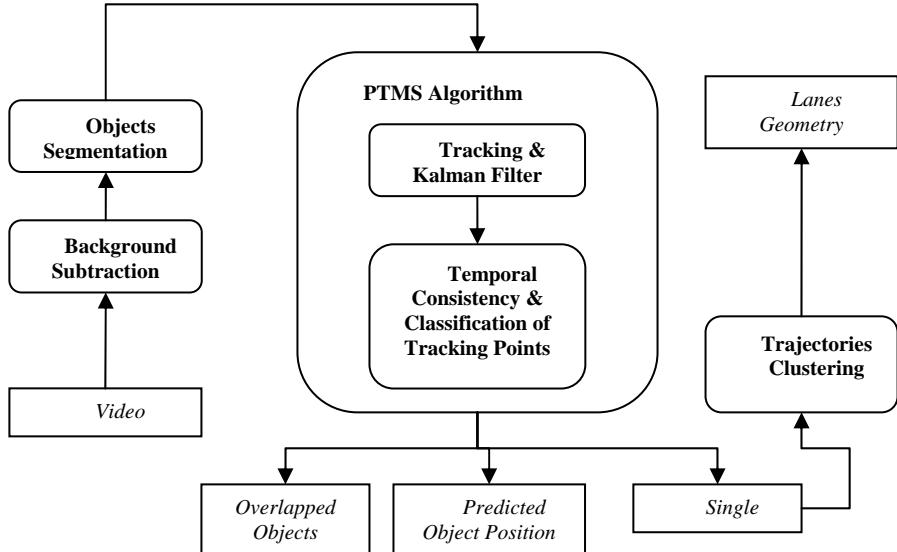


Fig. 1. Block diagram of the proposed system

3.1 Background Initialization

We use a Gaussian distribution in the Adaptive Smoothness Method [1] to build a background model. Detected blobs having an area smaller than K_{min} are deemed to be noise and disregarded. Erode and dilate operations are used to eliminate small holes within blobs. Shadow removal is not incorporated, but during the background update stage, a double thresholding operation is performed to eliminate self-shadowing.

3.2 Steady State Kalman Filter

If we wish to build complete motion histories for each tracked object, i.e. to determine the position of an object at each time step, it is necessary to implement KF[19] to resolve tracking instabilities caused by near and partial occlusions, shadows and image noise. If the case of multiple simultaneous object tracking, if we lose track of one vehicle and another vehicle is suddenly detected nearby, there is an obvious danger of mistaken vehicle identification.

Even assuming that vehicles drive at constant velocity, due to camera perspective effects their velocity in the image plane is time varying. Therefore, we approximate vehicle position in the image with a constant acceleration Kalman Filter. In the equations that follow, we work in image coordinates and assume that tuning parameters are the same for objects moving towards and away from the camera. At this stage we are not modelling the noise in vehicle position, thus we use a constant coefficient KF, whose coefficients are manually tuned for good performance

We use a steady state version of the KF, often referred to alfa-beta-gamma filter [19]. Let measurement vector $X = (x, y)$ represent the centroid of the detected blob, and the state vector $S = (x, y, x', y', x'', y'')$ where prime and double prime denote first and second derivatives with respect to time, i.e. velocity and acceleration in the x , y directions. In the initial state the velocity and acceleration are set to zero.

Let $X(k|k)$, $V(k|k)$ and $A(k|k)$ be, respectively, the estimated position, velocity and acceleration at time step k , and $X(k+1|k)$, $V(k+1|k)$ and $A(k+1|k)$ their predicted values. If $X(k)$ is the blob centroid position and T the sampling period, then the filter equations are the following:

Update equations:

$$A(k|k) = (1-\gamma)A(k|k-1) + \gamma/T^2 (Y(k) - X(k|k-1)) \quad (1)$$

$$V(k|k) = (1-\beta)V(k|k-1) + \beta/T (Y(k) - X(k|k-1)) \quad (2)$$

$$X(k|k) = (1-\alpha)X(k|k-1) + \alpha(Y(k) - X(k|k-1)) \quad (3)$$

Prediction equations:

$$A(k+1|k) = A(k|k) \quad (4)$$

$$V(k+1|k) = V(k|k) + T A(k|k) \quad (5)$$

$$X(k+1|k) = X(k|k) + TV(k|k) + 0.5 T^2 A(k|k) \quad (6)$$

A value of $\alpha = \beta = \gamma = 0.5$ is chosen for the parameters. When the PTMS detects an occlusion, the KF is not updated with the new value of X .

3.3 Heuristic Merge-and-Split Rules

The presence of shadows or ‘near’ occlusions caused by traffic congestion can seriously degrade accuracy of blob detection. Typically, several vehicles may be misdetected as one single vehicle with consequent problems for generating an object trajectory. Approaches based on spatial reasoning use more complex object representations such as templates or trained shape models. However, this is dependent on image resolution and only works under partial occlusion. A better approach is to use a temporal smoothness constraint in checking vehicle positions under different types of occlusion. Here, we propose a set of temporal rules that can easily complement a spatial approach.

The algorithm works as follows: First, we define a blob as a connected region resulting from the background subtraction process. Then use KF to predict for each blob the most likely position in the next frame that the blob will appear. Each blob is considered to have a number of children, i.e. number of different objects a blob is composed of. At the beginning, every blob is initialized as having one child. For each frame and for every blob:

1. Determine whether there is a 1-1 correspondence by checking size and position of blobs in consecutive frames and comparing positions and sizes.
2. For every blob that does not match the previous condition; determine whether the size has decreased by more than Ω expressed as a percentage. If so, decrease the number of its occluded objects by 1.
3. If any blob has decreased its size by less than Ω , store that information.
4. Determine whether any new blob has appeared in the vicinity of a blob whose size decreased and had a number of children greater than 1. If so, decrease the number of occluded objects in the old blob - the old blob was occluding the new blob.
5. Check if there are any new blobs in the new frame.
6. If there are any new blobs in the same position of several old blobs, it means that the new blob is composed of the old blobs, and the number of its children is increased by the number of the old blobs minus 1.

The algorithm works fairly well for most of the time, the principal drawback is when the initial blob is composed of several objects. In this case, it will be mis-detected as one single object. To tackle this problem, a spatial algorithm could be applied to the initial blobs to determine whether they are composed of one or more objects. The results of applying PTMS algorithm are presented in section 5.

4 Estimating Lane Geometry from Object Trajectories

In highly constrained environments such as highways, it is tempting to use vehicle motion trajectories rather than conducting image analysis of static scenes when determining lane geometry. The former approach has a number of advantages:

- Allows the use of controlled pan-tilt-zoom cameras rather than static cameras.
- Object trajectories are independent of scale and viewpoint considerations.
- Motion is more robust than spatial data with respect to light variation and noise.

The method assumes that the average lane width in image coordinates is known in advance. However, it does not require *a priori* knowledge of the number of lanes or road geometry, i.e. whether it is a straight or curved section of highway.

First, we apply a pre-filtering stage to remove obviously invalid trajectories that are produced by poor background initialization. Excluded trajectories are those that have consecutive inter-point differences greater than some threshold, or the total length less than some pre-defined threshold.

To calculate the approximate centre of each lane, first we fit a least squares polynomial of degree M for each trajectory. The average residual error of fit can be used to ascertain the optimal value of M . Next, we apply a robust K-means clustering algorithm that works in the coefficient space of the polynomials. To reduce the time complexity, we use a heuristic to limit the number of candidate trajectories to those with greater likelihood of belonging to a lane. Finally, the RANSAC [18] algorithm is used on the clustered trajectories to determine a least squares polynomial fit to the

lane centres. RANSAC is robust to outlier trajectories produced by frequent vehicle lane changes, undetected overlapped vehicles and noise in the video sequence. Further details of this method are presented in a companion paper.

5 Results

The results of applying PTMS algorithm are now presented. The video sequences were recorded in grey scale at a rate of 15 frames/sec with a 176x144 pixel resolution. In Fig. 2 we show the result of background subtraction. Segmented objects whose areas $< K_{min}$ are denoted in red whereas detected vehicles are coloured purple. We use a different colour to signify the bounding box of a tracked vehicle. When tracking of one vehicle is lost, we place a cross to highlight the position predicted by KF. In this sequence, the influence of KF prediction was not very significant.



Fig. 2. Tracked vehicles



Fig. 3. Tracking and occlusion handling

Fig. 3 shows the result of occlusion handling applied to the previous figure. Observe that the two cars in the left of the image are detected as a single blob, and through the use of PTMS algorithm, we can determine that it corresponds to two cars in the previous frame. The detected blob is displayed with its bounding box in red with a cross drawn in the middle.

In Fig. 4 we display the trajectories generated by use of KF and PTMS algorithm applied to the same sequence from which Figs. 2 and 3 were drawn. Trajectories in green correspond to single vehicles successfully tracked, whereas those in purple correspond to vehicles previously detected but whose tracking was subsequently lost. The points are predicted by output of KF. The red points correspond to trajectories of averaged position of two or more overlapped vehicles detected through use of PTMS.

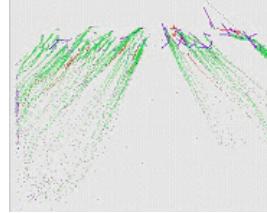


Fig. 4. Vehicle trajectories generated through hybrid tracking and PTMS algorithm

Since the approach adopted is low-level and independent of camera viewpoint and type of object motion, we tested the hybrid tracking and PTMS approach with a

different data set recorded at a road intersection. A typical frame taken from the sequence is shown in Fig. 7a.

In Fig. 5 we can observe that there are no object occlusions, and all the vehicles are detected as single objects. In Fig. 6 the PTMS algorithm detects a blob comprised of two vehicles and a second blob with four occluding vehicles. An unidentified moving object is mis-detected as comprising two occluding vehicles.

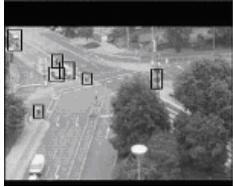


Fig. 5. Tracked vehicles



Fig. 6. Tracking and occlusion handling

In Fig. 7b we display the set of trajectories calculated from the sequence¹ 7a, with the colours employing the same semantics as in Fig. 4.

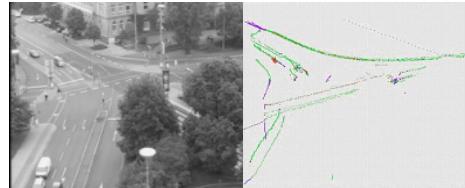


Fig. 7 (a) Typical scene at a road intersection. (b) Trajectories

We now show some preliminary results of applying the clustering approach to the computed trajectories described in section 4. The computed point trajectories of single vehicles (Fig. 8a) are used to estimate the lane centres (Fig. 8b) on a curved segment of highway. From a total of 175 partial trajectories in the image sequence, the K-means clustering algorithm uses 20 trajectories per lane to estimate the centres. It should be noted that although the original trajectory data contains vehicle lane changes, the RANSAC fitting method can be made insensitive to these by careful parameter tuning. The clustering is carried out as a post-processing operation.



Fig. 8. (a) Original trajectories of single tracked vehicles containing outliers. (b) Estimated lane centres. (c) Processing time for applying clustering algorithm

¹ Image sequence downloaded from http://i21www.ira.uka.de/image_sequences

Next figure illustrates similar results for a straight highway segment using uncalibrated PTZ cameras. Here we start from an initial total of 200 partial trajectories and again use 20 trajectories per lane to estimate the centres.

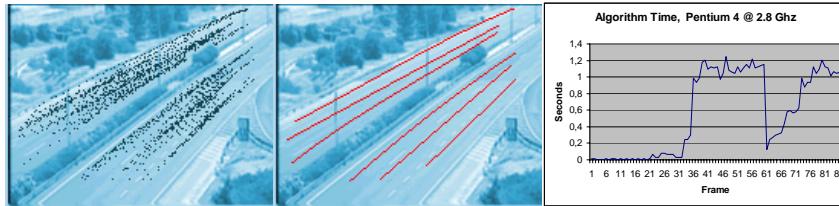


Fig. 9. (a) Original trajectories of single tracked vehicles containing outliers. (b) Estimated lane centres. (c) Processing time for applying clustering algorithm

The processing times for each frame in the respective sequences are shown in Fig. 8c and Fig. 9c. In each case, the algorithm starts with zero clusters and adds 2 new trajectories per frame. More results can be found at the author webpage².

6 Discussion and Conclusions

This paper proposes an algorithm for vehicle tracking with the following characteristics; temporal integration with a Kalman Filter, time-consistent merging-and-splitting of overlapped detected blobs, aggregation of trajectory data to estimate lane centres and removal of the need for calibrated cameras.

The preliminary results demonstrate the feasibility of using ordinary uncalibrated stationary or PTZ cameras to analyse traffic behaviour in real-time. The algorithm is viewpoint independent and does not make any *a priori* assumption regarding lane geometry. The results can be used as input to higher level traffic monitoring systems for estimating traffic speed, frequency of lane changes, accident detection and classification of anomalous driver behaviour. We use some limited assumptions regarding camera zoom and image scale.

One drawback of the clustering approach is that due to occlusions, vehicle trajectories are sometimes miss detected and hence partitioned into erroneous cluster sets. It is often difficult to distinguish these from genuine lane changes at the post-processing stage. In future work, we intend to tackle this limitation.

Acknowledgements:

This work is partially funded by the Portuguese project ADI-INTELTRAF. The authors would like to thanks to ISR and Observit for the video sequences.

² <http://omni.isr.ist.utl.pt/~jpqm/inteltraf.htm>

References

1. Gutchess D., Trajkovics M., Cohen-Solal E., Lyons D., Jain A.K, A Background Model Initialization Algorithm for Video Surveillance, in proc. IEEE ICCV 2001, Pt.1 (2001) 744-740.
2. I. Haritaoglu, D. Harwood, and L.S. Davis. W4: Real-Time Surveillance of People and Their Activities. IEEE Trans. Patt. Anal. Mach. Intell. 22 (2000) 809-830.
3. Prati, I. Mikié, C. Grana, M. Trivedi , Shadow Detection Algorithms for Traffic Flow Analysis: a Comparative Study, IEEE Trans. Intell. Transport. Syst. (2001) 340-345.
4. A. Elgammal, R. Duraiswami, Background and Foreground Modeling Using Nonparametric Kernel Density Estimation for Visual Surveillance, Proc. IEEE, 90 (2002).
5. R. Cucchiara, C. Grana, M. Piccardi, A. Prati, Detecting Objects, Shadows and Ghosts in Video Streams By Exploiting Colour and Motion Information Proceedings of 11th International Conference on Image Analysis and Processing ICIAP (2001)
6. S. Kamijo, Y. Matsushita, K. Ikeuchi, M. Sakauchi: Traffic Monitoring and Accident Detection at Intersections, IEEE Trans. Intell. Transport. Syst. 1 (2000).
7. Koller,D.; Weber,J.; Malik.J.: Robust Multiple Car Tracking with Occlusion Reasoning, Proc.Third European Conference on Computer Vision, LNCS 800, Springer-Verlag (1994).
8. H. Veeraraghavan, O. Masoud, N. Papanikolopoulos: Computer Vision Algorithms for Intersection Monitoring, IEEE Trans. Intell. Transport. Syst. 4 (2003) 78-89.
9. Y.Jung, K. Lee, Y. Ho, Content-Based event retrieval using semantic Scene interpretation for automated traffic surveillance, IEEE Trans. Intell. Transport. Syst. 2 (2001) 151-163.
10. S. Kamijo, Y. Matsushita, K. Ikeuchi, M. Sakauchi Traffic Monitoring and Accident Detection at Intersections, IEEE Trans. Intell. Transport. Syst. 1 (2000).
11. Daniel J. Dailey, F. W. Cathey, S. Pumrin: An Algorithm to Estimate Mean Traffic Speed Using Uncalibrated Cameras, IEEE Trans. Intell. Transport. Syst. 1 (2000)
12. Todd N. Schoepflin, Daniel J. Dailey, Dynamic Camera Calibration of Roadside Traffic Management Cameras for Vehicle Speed Estimation, IEEE Trans. Intell. Transport. Syst. 4 (2003) 90-98.
13. D. Makris, T. Ellis, Path Detection in Video Surveillance, Image and Vision Computing, 20 (2002) 895-903.
14. C. Stauffer, W. Grimson, Learning Patterns of activity using real-time tracking, *in* IEEE Trans. Patt. Anal. Mach. Intell. 22 (2000) 747-757.
15. N. Johnson, D. Hogg, Learning the distribution of Object Trajectories for Event Recognition, Image and Vision Computing, 14 (1996) 609-15.
16. A. Elgammal, R. Duraiswami, Background and Foreground Modeling Using Nonparametric Kernel Density Estimation for Visual Surveillance, Proc. IEEE 90.
17. M. A. Fischler, R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM, 24 (1981) 381-395.
18. "Tracking and Kalman Filtering Made Easy", Eli Brookner, John Wiley & Sons, (1998)