Czech Pattern Recognition Workshop 2000, Tomáš Svoboda (Ed.)
Peršlák, Czech Republic, February 2–4, 2000
Czech Pattern Recognition Society

Car tracking in tunnels

Roman Pflugfelder and Horst Bischof
Pattern Recog. and Image Proc. Group
Vienna University of Technology
Favoritenstr. 9-1832
A-1040 Vienna, Austria

Abstract *Tracking methods are fundamental operations in traffic scene analysis. In this work we report on a tracking algorithm with a Kalman filter for traffic surveillance in tunnels. The difficulties of solving the problem are the illumination conditions and the image quality. We demonstrate our work on short sequences of tunnel scenes.*

1 Introduction

Vehicle tracking is a key feature of traffic surveillance systems. It is realized in two stages:

1. Detection of a moving vehicle (Initialization stage of the tracker)
2. Tracking the vehicle (Tracking loop)

The need for such systems increases rapidly with the growing number of cars that are foreseen by many experts and reported in the last Delphi study of the Austrian Government. Especially traffic surveillance in tunnels can be a decisive contribution to increase safety. The requirements for the car tracking system reported in this paper are as follows:

- The usage of the existing infrastructure (cameras, video network).
- The system must work in real-time.
- Robust detection and tracking of different kinds of vehicles
- Robustness against different illumination conditions (e.g. reflections, lights).
- The System must provide an interface for future data processing. The goal is not track each individual car, but to have estimates on the average numbers of cars, and the average driving speed. Moreover, the system should facilitate incident detection.

The images of the cameras mounted in tunnels are very noisy (see Fig. 1). Sometimes even complete frames are missing or replaced by frames from other cameras due to crosstalk problems in the video network.

Many motion detection and tracking algorithms have been investigated in the last years [1][3][6][7]. The simplest and by researchers and practitioners mostly used detection algorithms are based on background differencing [3]. In the simplest case these methods subtract the actual image frame at time t, I(t), from a background reference image. The difference frame is then segmented in areas with and without motion. A shortcoming of all these methods is the robustness against illumination conditions and noise. The problems associated with illumination, reflections and noise do not allow the usage of background differencing methods in an application like car detection in tunnels.

The next step after motion detection is tracking the car. There are two important approaches, calculation of optical flow [1][5] and feature tracking methods [3][4]. The former ascertains an approximation of the dense motion field where the latter tracks merely a feature from one to the next frame and therefore calculates a sparse motion field. The choice of the method is also determined by the application. Optical Flow estimation has large computational costs, because calculation is done for every pixel of the frame. Besides optical flow estimation is difficult if objects with large homogeneous areas (e.g. vehicles in tunnels) are in motion.

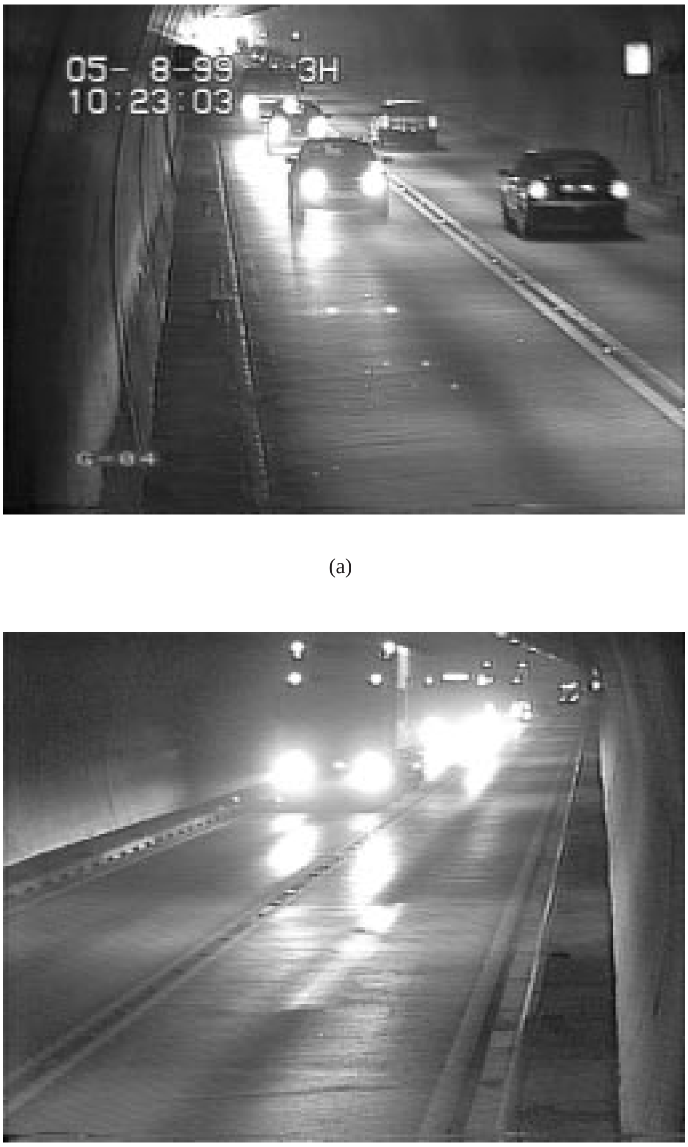
For this reason a Kalman filter based tracker operating on features seems more suitable for the present application. The robustness of such an algorithm depends mainly on reliable feature detection

In the next section we explain the requirements for a robust detection algorithm and that detection and tracking are highly coupled. In section 3, we explain the tracker. In section 4 we show some illustrative experiments which demonstrate the feasibility of our approach.

2 Detection of vehicles

Many car tracking applications use background differencing to detect areas (blobs) of motion, because the assumption of a static background is allowable. These blobs are the cars in motion. This method is shown to be efficient and reliable in outdoor applications with constant illumination conditions and in daylight [4]. But it is inappropriate to track cars in tunnels. The problems are as follows:

1. Static background and cars show similar intensity values,



(a)

(b)

Figure 1: Two representative scenes

therefore automatic segmentation of the whole vehicle by thresholding is impossible.

2. Many distortions like reflections of car-light on road, wall and other cars are present in the images. Therefore motion detection yields many false alarms.

3. Noisy images due to noisy signals of cameras and noise inference in leaky cables.

For tracking we need reliable features. As an initial analysis has shown the most reliable features are the lights of the vehicles. Moreover, we can exploit a lot of background knowledge in the detection stage, e.g., driving direction, where a car is to be expected in the image, etc. This knowledge can be used to increase the speed and the reliability of the detection stage.

The lights of vehicles are distinct from the background, and in addition the always (expect for motor-cycles and cars with broken lights) they appear in pairs with a strong geometrical relation.

Exploiting the knowledge of the scene we restrict the detection of lights to a small detection window, which is positioned in the image such that the lights appear as distinct blobs. Finding initial hypotheses for the lights can be performed by a simple threshold. The parameter estimation of the threshold is not a problem, because lights are always pixel areas of very high intensity while background is relatively dark. Since threshold results in many false positive detections, further criteria are necessary. Following properties have been proved to yield reliable detection results:

MSE of the difference of blob's convex hull and fitted ellipse: For a car-light, the convex hull is as similar to the fitted ellipse. As a measure we can calculate the ratio between the area under the fitted ellipse and the convex hull.

Center of mass (COM) of light blobs: Since lights appear in pairs and are mounted at the same hight, we can use the Y-disparity (see Fig. 2) of the center of mass as a reliable criterion for finding lights.

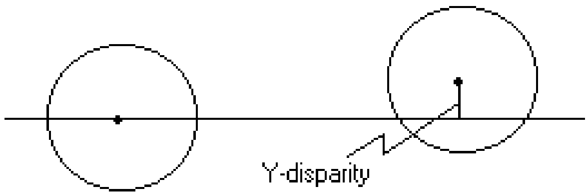


Figure 2: Illustration of the Y-disparity.

Both criteria properties rank our hypotheses of car-lights. A car-light is found if the minimality of MSE of the two blobs is fulfilled regarding to all blobs in the frame and the disparity in Y-direction is maximally two pixels. To increase reliability further we require that a light has to be found in three consecutive frames.

3 Tracking vehicles

If we have found the car's lights we can start the tracking process. Tracking is the problem of matching features from frame to frame in long video sequences. We can assume that motion is nearly linear from frame to frame. The linear motion assumption holds, because the vehicle's motions are limited by the road and random shifts in direction can be neglected. We can predict the feature location in image $I(t+1)$ with knowledge of feature location in image $I(t)$).

We approach tracking in the general framework of optimal estimation theory. Our solution is the Kalman filter [1][2]. For our purposes a Kalman filter is a recursive algorithm which estimates the position and uncertainty of a moving feature in the next frame. It gives a predicted search area where it is sure to find the feature with a certain confidence.

Let us formalize the tracking problem for the car tracking application. In our approach we consider only one feature point p_t at time t. It is the center of mass of the right car light. We describe the motion by a state vector $x_t = [p_t\ v_t]^\top$ where v_t is the velocity in p_t of the feature point. Assuming linearity in the motion parameters assumption we write the system's model of the Kalman filter as

$$\begin{aligned} p_t &= p_{t-1} + v_{t-1} + \xi_{t-1} \\ v_t &= v_{t-1} + \eta_{t-1} \end{aligned} \tag{1}$$

where ξ_{t-1} and η_{t-1} are zero-mean white Gaussian random processes modeling system noise.

In terms of the state vector x_t rewrites

$$x_t = \Phi_{t-1} x_{t-1} + w_{t-1} \tag{2}$$

with the time-invariant state matrix

$$\Phi_{t-1} = \begin{bmatrix} 1010 \\ 0101 \\ 0010 \\ 0001 \end{bmatrix}$$

and noise

$$w_{t-1} = \begin{bmatrix} \xi_{t-1} \\ \eta_{t-1} \end{bmatrix}$$

The measurement model of our car light detector which estimates z_t the position of the COM of car light at every frame of the sequence becomes

$$z_t = \begin{bmatrix} 1000 \\ 0100 \end{bmatrix} \begin{bmatrix} p_t \\ v_t \end{bmatrix} + \mu_t \tag{3}$$

μ_t is like in the system model above a white Gaussian random process modeling the measurement noise. Further information about the theory can be found in [2].

The Kalman filter is summarized in the following equations [1]. These equations are executed recursively while Φ is the time-invariant state matrix, H the time-invariant measurement matrix, Q and R are the corresponding constant covariance matrices for state and measurement. z_t is the measurement at time t, $\hat{x}_t$ the prediction of position and velocity at time t and P_t is the corresponding covariance matrix with $\hat{x}_t$ uncertainties given by the diagonal elements. P_0 is set to an initial value.

$$\begin{aligned} &while\ car\ is\ trackable \\ &\quad P'_t = \Phi P_{t-1} \Phi_{t-1}^\top + Q \\ &\quad K_t = P'_t H_t^\top (H_t P'_t H_t^\top + R)^{-1} \\ &\quad \hat{x}_t = \phi_{t-1}\hat{x}_{t-1} + K_t(z_t - H_t \Phi_{t-1} \hat{x}_{t-1}) \\ &\quad P_t = (I - K_t H_t) P'_t (I - K_t H_t) + K_t R K_t^\top \end{aligned} \tag{4}$$

To improve measurement evaluation and to make the tracking algorithm more efficient feature detection is only done within the uncertainty ellipse described by the covariance matrix P_t.

$$(z_t - \hat{x}_t)(P_t)^{-1}(z_t - \hat{x}_t)^\top \le c^2 \tag{5}$$

The ellipse contains the measurement z_t with given probability p which is set to 0.95 in our application. Equation 5 has a χ-square distribution so that the measurement lies with probability p in the uncertainty ellipse. c^2 is the p-th percentile of a χ-square distribution with two degrees of freedom (depends on the dimension of the state vector). Here c^2=5.991. Two problems arise in our implementation:

Missing information: We need initial assumptions and parameters to start the tracking.

- Which system model we should choose and how are the values of the corresponding covariance matrix Q?
- Which measurement model we should choose and how are the values of the corresponding covariance matrix R?
- How are the initial values for the state vector $\hat{x}_0$ and the corresponding covariance matrix P_0?

As described the system model is linear. The measurement model is available, as we assume that feature positions are computed at each frame. As experiments have shown entries of $R \approx \frac{1}{4}Q$ have produced satisfactory results. The tracker relies more on the measurements.

Data Association: If more than one candidate exists in the uncertainty ellipse, we have to choose which we want to track. This is a nontrivial problem. In our application neither high clutter in the detection window exists nor the blobs interfere. Therefore nearest neighbor data association is the most effective and is used.

4 Experiments

We have tested our algorithm on different tunnel sequences. In the following we will discuss such a typical tunnel image sequence.

Consider figure 4. In this sequence three vehicles are correctly tracked. This example shows that our tracking algorithm can reliably track cars in tunnel sequences. In every frame the car-light detection is performed within the detection window. The founded blobs and their properties are depicted in figure 4. We see, that one criteria alone (e.g. Y-disparity) is not reliable enough to detect car lights with a high probability. See the blobs in area 1 and 3. Both are reflections of car-light on the road. The blobs labeled with 2, 4 and 5 are the car-lights of the three cars in the sequence. However, by combining these two criteria we can achieve a high reliability in the detection result.

Every criteria of classification is determined by several parameters. The optimal estimation is a key problem in this approach. Unfortunately, the experiments have shown that problems arise if light blobs significantly vary (i.e. in the case of heavy goods vehicles) and cars are occluded by others (i.e. heavy traffic). However, for the present application (approximate number of cars, approximate velocity. etc.) The tracking results are good enough, though a detailed analysis has still to be done.

Finally, the proposed method to detect robustly car lights is not sufficient enough to find every vehicle reliable in the video sequence. In this application it was not a requirement, because on top of the tracker a traffic density and census system should be build where average data is sufficient.

5 Conclusion

We have presented Kalman filter tracking for the important application of car tracking in tunnels. Noise and poor contrast, reflections of lights do not allow us to use background differencing methods for motion detection. This problem is solved by detection of features of the vehicles - i.e. their lights. As we can only detect pairs of lights, bikes and vehicles with broken lights can not be tracked, but for our purposes this fact does not matter. Future work will concentrate on more powerful detection algorithms to track the whole vehicles. Detection itself are three stages beginning with filtering, then segmentation and as a last step classification. We have seen that the latter is the hardest task to realize. In order to be independent of the position of the detection window within the frame we base our classification on relative criteria. After a pair of car-lights is detected, the tracking process is initiated. We have got satisfactory results if we rely more on the measurements than on the state model. Data Association is implemented as a nearest neighbor variant, because no interfering blobs and high clutter exist.

6 REFERENCES

References

[1] Trucco E., Verri A. *Introductory Techniques for 3-D Computer Vision* Prentice Hall. 1998.

[2] Maybeck W. S. *Stochastic Models, Estimation and Control Vol. I* Academic Press NY. 1979.

[3] Ullman S. *The Interpretation of Visual Motion* MIT Press. Cambridge (MA). 1979.

[4] Koller D., Weber J., Malik J. *Robust Multiple Car Tracking with Occlusion Reasoning* UCB.

[5] Krueger S., Calway A. *Motion Estimation and Tracking Using Multiresolution Affine Models* Univ. of Bristol. Dep. of CS.

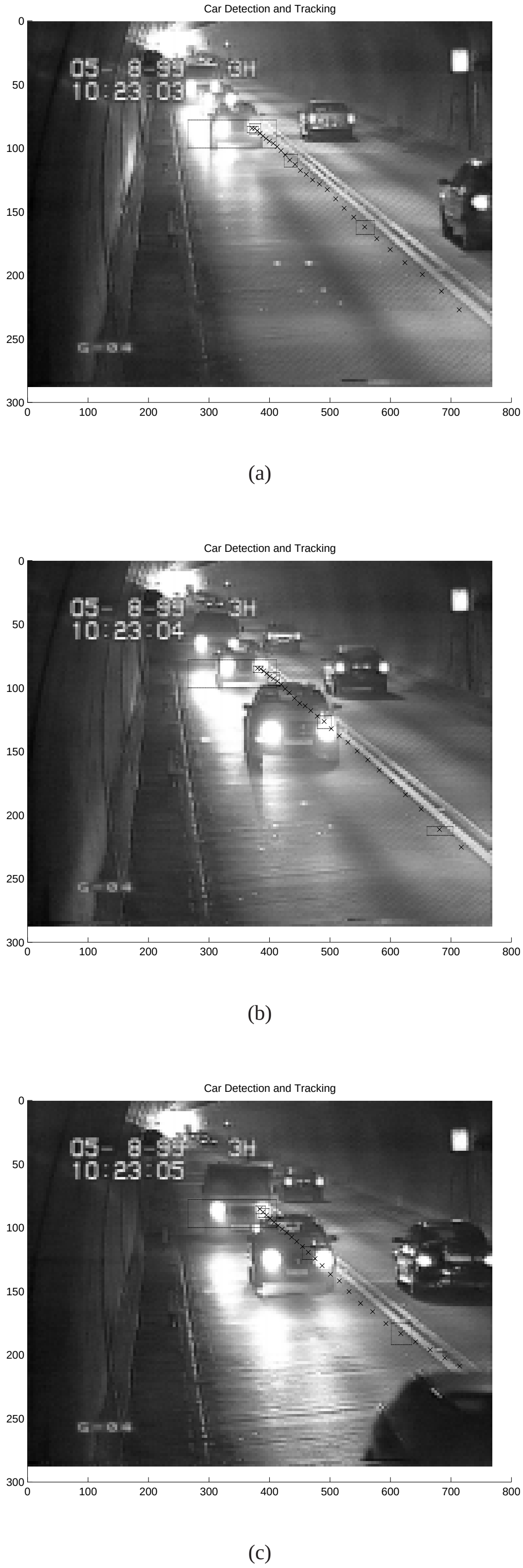
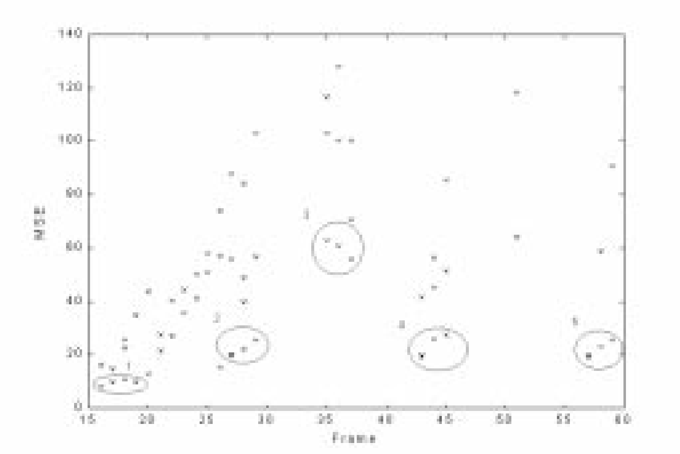


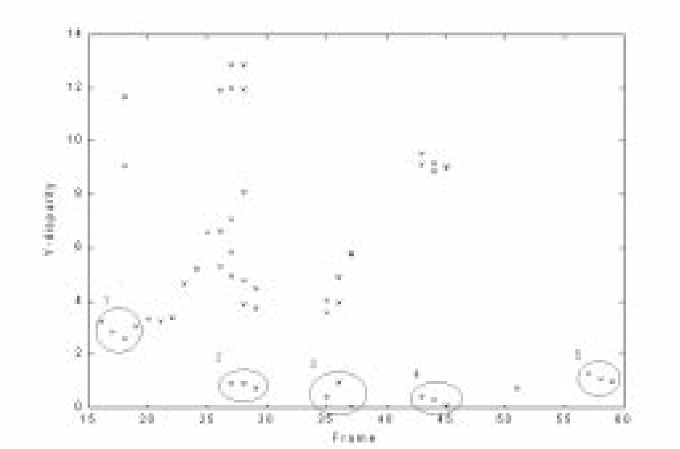
Figure 3: Tracking three cars in a video sequence. The initial frame is shown. Crosses mark the tracked light in every frame.

[6] Shi J., Malik J. *Motion Segmentation and Tracking using Normalized Cuts* ICCV 1998.

[7] Staufer C., Grimson W.E.L. *Adaptive background mixture models for real-time tracking* AI Lab. MIT.



(a)



(b)

Figure 4: Classification of blobs found in each frame of a tunnel sequence. Three vehicles cross the detection window in the frames 27-29, 43-35, 57-59 (a) relation of axis of the approximated ellipse (expansion) (b) disparity of COM in y direction