Model-Based Object Tracking in Traffic Scenes

D. Koller[1], *K. Daniilidis*[1], *T. Thórhallson*[1] *and H.-H. Nagel*[1,2]

[1] Institut für Algorithmen und Kognitive Systeme
Fakultät für Informatik, Universität Karlsruhe (TH),
Postfach 6980, D-7500 Karlsruhe 1, Germany; E-mail: koller@ira.uka.de
[2] Fraunhofer-Institut für Informations- und Datenverarbeitung (IITB), Karlsruhe

Abstract. This contribution addresses the problem of detection and tracking of moving vehicles in image sequences from traffic scenes recorded by a stationary camera. In order to exploit the a priori knowledge about the shape and the physical motion of vehicles in traffic scenes, a parameterized vehicle model is used for an intraframe matching process and a recursive estimator based on a motion model is used for motion estimation. The initial guess about the position and orientation for the models are computed with the help of a clustering approach of moving image features. Shadow edges of the models are taken into account in the matching process. This enables tracking of vehicles under complex illumination conditions and within a small effective field of view. Results on real world traffic scenes are presented and open problems are outlined.

1 Introduction

The higher the level of abstraction of descriptions in image sequence evaluation, the more a priori knowledge is necessary to reduce the number of possible interpretations as, for example, in the case of automatic association of trajectory segments of moving vehicles to motion verbs as described in [Koller *et al.* 91]. In order to obtain more robust results, we take more a priori knowledge into account about the physical inertia and dynamic behaviour of the vehicle motion.

For this purpose we establish a motion model which describes the dynamic vehicle motion in the absence of knowledge about the intention of the driver. The result is a simple circular motion with constant magnitude of velocity and constant angular velocity around the normal of a plane on which the motion is assumed to take place. The unknown intention of the driver in maneuvering the car is captured by the introduction of process noise. The motion model is described in Section 2.2.

The motion parameters for this motion model are estimated using a recursive maximum a posteriori estimator (MAP), which is described in Section 4.

Initial states for the first frames are provided by a step which consists of a motion segmentation and clustering approach for moving image features as described in [Koller *et al.* 91]. Such a group of coherently moving image features gives us a rough estimate for moving regions in the image. The assumption of a planar motion yields then a rough estimate for the position of the object hypothesis in the scene by backprojecting the center of the group of the moving image features into the scene, based on a calibration of the camera.

To update the state description, straight line segments extracted from the image (we call them data segments) are matched to the 2D edge segments — a view sketch — obtained by projecting a 3D model of the vehicle into the image plane using a hidden-line algorithm to determine their visibility.

The 3D vehicle model for the objects is parameterized by 12 length parameters. This enables the instantiation of different vehicles, e.g. limousine, hatchback, bus, or van from the same generic vehicle model. The estimation of model shape parameters is possible by including them into the state estimation process. Modeling of the objects is described in Section 2.1.

The matching of data and model segments is based on the Mahalanobis distance of attributes of the line segments as described in [Deriche & Faugeras 90]. The midpoint representation of line segments is suitable for using different uncertainties parallel and perpendicular to the line segments.

In order to track moving objects in long image sequences which are recorded by a stationary camera, we are forced to use a wide field of view. This is the reason for a small image of an individual moving object. In bad cases, there are few and/or only poor line segments associated with the image of a moving object. In order to track even objects mapped onto very small areas in the image, we decided to include the shadow edges in the matching process if possible. In a very first implementation of the matching process it was necessary to take the shadow edges into account to track some small objects. In the current implementation the shadow edges appear not to be necessary for tracking these objects but yield more robust results. The improvement of the very first implementation compared to the current implementation was only possible by testing the algorithms in various real world traffic scenes. The results of the last experiments are illustrated in Section 5.

2 Models for the Vehicles and their Motion

2.1 The parameterized Vehicle Model

We use a parameterized 3D generic model to represent the various types of vehicles moving in traffic scenes. Different types of vehicles are generated from this representation by varying 12 length parameters of our model. Figure 1 shows an example of five different specific vehicle models derived from the same generic model.

In the current implementation we use a fixed set of shape parameters for each vehicle in the scene. These fixed sets of shape parameters are provided interactively.

In initial experiments on video sequences from real world traffic scenes, the algorithm had problems in robustly tracking small vehicular objects in the images. These objects span only a region of about 20×40 pixels in the image (see for example Figure 2). In bad cases, we had not enough and/or only poor edge segments for the matching process associated with the image of the moving vehicle, which caused the matching process to match lines of road markings to some model edge segments. Such wrong matches resulted in wrong motion parameters and, therefore, in bad predictions for the vehicle position in subsequent frames.

Since vehicle images in these sequences exhibit salient shadow edges, we decided to include the shadow edges of the vehicle into the matching process. These shadow edges are generated from the visible contour of the object on the road, as seen by the sun. The inclusion of shadow edges is only possible in image sequences with a well defined illumination direction, i. e. on days with a clear sky (see Figures 7 and 8). The illumination direction can be either set interactively off-line or it can be incorporated as an unknown parameter in the matching process.

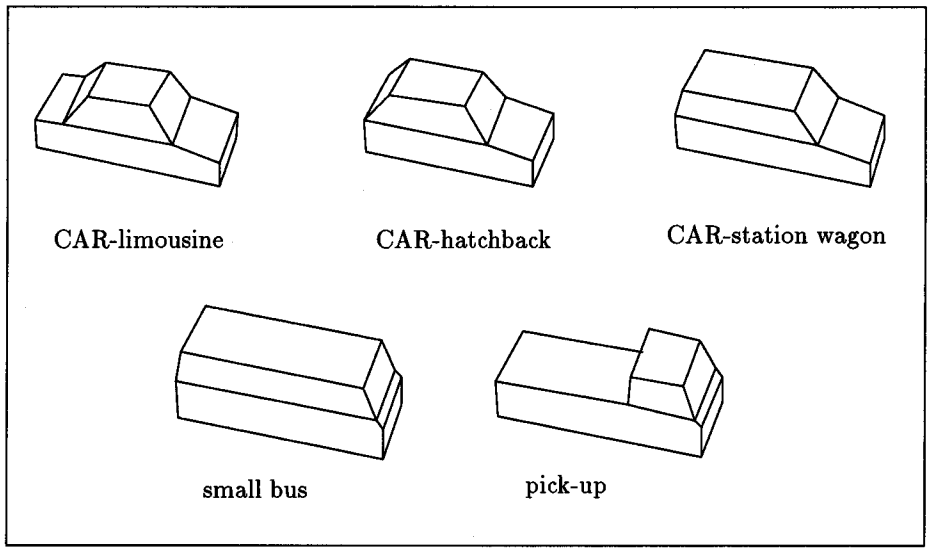

Fig. 1. Example of five different vehicle models derived from the same generic model.

2.2 The Motion Model

We use a motion model which describes the dynamic behaviour of a road vehicle without knowledge about the intention of the driver. This assumption leads to a simple vehicle motion on a circle with a constant magnitude of the velocity $v = |\boldsymbol{v}|$ and a constant angular velocity ω. The deviation of this idealized motion from the real motion is captured by process noise due to v and ω. In order to recognize the pure translational motion in the noisy data, we evaluate the angle difference $\omega\tau$ ($\tau = t_{k+1} - t_k$ is the time interval). In case $\omega\tau$ is less then a threshold we use a simple translation with the estimated (constant) angle ϕ and $\omega = 0$.

Since we assume the motion to take place on a plane, we have only one angle ϕ and one angular velocity $\omega = \dot{\phi}$. The angle ϕ describes the orientation of the model around the normal (the z-axis) of the plane on which the motion takes place. This motion model is described by the following differential equation:

$$\begin{array}{lll} \dot{t}_x = v\cos\phi, & \dot{v} = 0, & \dot{\phi} = \omega, \\ \dot{t}_y = v\sin\phi, & & \dot{\omega} = 0. \end{array} \tag{1}$$

3 The Matching Process

The matching between the predicted model data and the image data is performed on edge segments. The model edge segments are the edges of the model, which are backprojected from the 3D scene into the 2D image. The invisible model edge segments are removed by a hidden-line algorithm. The position $\boldsymbol{t}$ and orientation ϕ of the model are given by the output of the recursive motion estimation described in Section 4. This recursive motion estimation also yields values for the determination of a window in the image in which edge segments are extracted. The straight line segments are extracted and approximated using the method of [Korn 88].

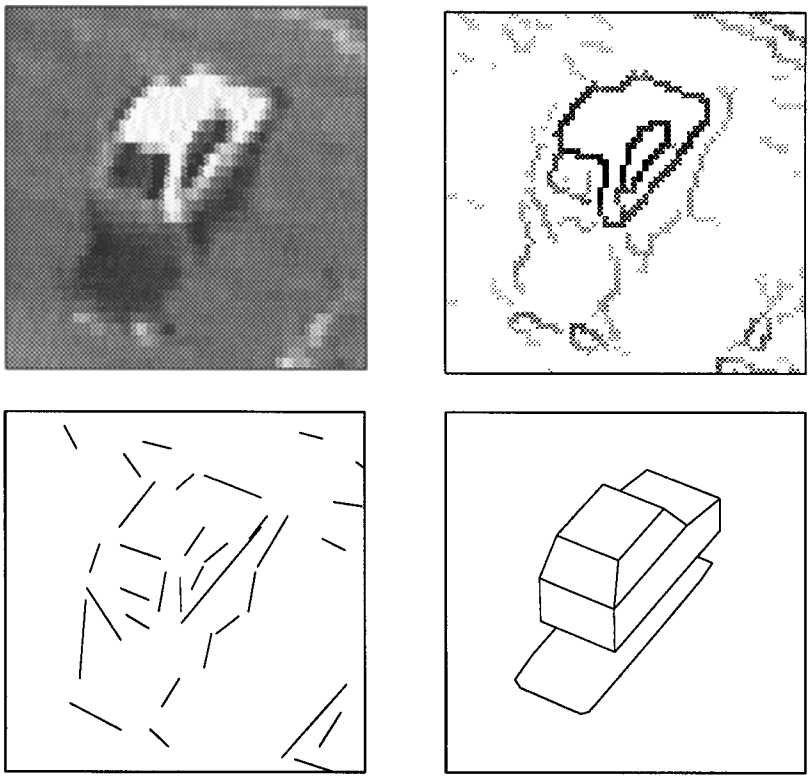

Fig. 2. To illustrate the complexity of the task to detect and track small moving objects, the following four images are given: the upper left image shows a small enlarged image section, the upper right figure shows the greycoded maxima gradient magnitude in the direction of the gradient of the image function, the lower left figure shows the straight line segments extracted from these data, and the lower right figure shows the matched model.

The Matching Algorithm

Like the method of [Lowe 85; Lowe 87] we use an iterative approach to find the set with the best correspondence between 3D model edge segments and 2D image edge segments. The iteration is necessary to take into account the visibility of edge segments depending on the viewing direction and the estimated state of position and orientation, respectively. At the end of each iteration a new correspondence is determined according to the estimated state of position and orientation. The iteration is terminated if a certain number of iterations has been achieved or the new correspondence found has already been investigated previously. Out of the set of correspondences investigated in the iteration, the correspondence which leads to the smallest residual is then used as a state update. The algorithm is sketched in Figure 3. We use the average residual per matched edge segment, multiplied by a factor which accounts for long edge segments, as a criterion for the selection of the smallest residual.

$i \leftarrow 0$
$\mathcal{C}_i \leftarrow$ get_correspondences($\boldsymbol{x}^-$)
DO
  $\boldsymbol{x}_i^+ \leftarrow$ update_state($\mathcal{C}_i$)
  $r_i \leftarrow$ residual($\mathcal{C}_i$)
  $\mathcal{C}_{i+1} \leftarrow$ get_correspondences($\boldsymbol{x}_i^+$)
  $i \leftarrow i + 1$
WHILE$((\mathcal{C}_{i+1} \neq \mathcal{C}_j \; ; j = 0, 1 \ldots, i) \wedge i <$ IMAX)
$i_{min} \leftarrow \{i | r_i = \min(r_j) \; ; j = 0, 1 \ldots,$IMAX$\}$
$\boldsymbol{x}^+ \leftarrow \boldsymbol{x}_{i_{min}}^+$

Fig. 3. Algorithm for the iterative matching process. $\mathcal{C}_i$ is the set of correspondences between p data segments $\mathcal{D} = \{D_j\}_{j=1\ldots p}$ and n model segments $\mathcal{M} = \{M_j\}_{j=1\ldots n}$ for the model interpretation i: $\mathcal{C}_i = \{(M_j, D_{ij})\}_{j=1\ldots n}$.

Finding Correspondences

Correspondences between model and data segments are established using the Mahalanobis distance between attributes of the line segments as described in [Deriche & Faugeras 90]. We use the representation $\boldsymbol{X} = (x_m, y_m, \theta, l)$ of a line segment, defined as:

$$\begin{aligned} x_m &= \tfrac{x_1+x_2}{2}, & \theta &= \arctan\left(\tfrac{y_2-y_1}{x_2-x_1}\right), \\ y_m &= \tfrac{y_1+y_2}{2}, & l &= \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}. \end{aligned} \tag{2}$$

where $(x_1, y_1)^T$ and $(x_2, y_2)^T$ are the endpoints of a line segment.

Denoting by $\sigma_\parallel$ the uncertainty in the position of the endpoints along an edge chain and by $\sigma_\perp$ the positional uncertainty perpendicular to the linear edge chain approximation, a covariance matrix Λ is computed, depending on $\sigma_\parallel, \sigma_\perp, \theta$ and l. Given the attribute vector $\boldsymbol{X}_m$ of a model segment and the attribute vector $\boldsymbol{X}_d$ of a data segment, the Mahalanobis distance between $\boldsymbol{X}_m$ and $\boldsymbol{X}_d$ is defined as

$$d = (\boldsymbol{X}_m - \boldsymbol{X}_d)^T (\Lambda_m + \Lambda_d)^{-1} (\boldsymbol{X}_m - \boldsymbol{X}_d). \tag{3}$$

The data segment with the smallest Mahalanobis distance to the model segment is used for correspondence, provided the Mahalanobis distance is less than a given threshold. Due to the structure of vehicles this is not always the best match. The known vehicles and their models consist of two essential sets of parallel line segments. One set along the orientation of the modeled vehicle and one set perpendicular to this direction. But evidence from our experiments so far supports our hypothesis that in most cases the initialisation for the model instantiation is good enough to obviate the necessity for a combinatorial search, such as, e.g., in [Grimson 90b].

The search window for corresponding line segments in the image is a rectangle around the projected model segments. The dimensions of this rectangle are intentionally set by us to a higher value than the values obtained from the estimated uncertainties in order to overcome the optimism of the IEKF as explained in Section 4.

4 Recursive Motion Estimation

In this section we elaborate the recursive estimation of the vehicle motion parameters. As we have already described in Section 2.2, the assumed model is the uniform motion of a known vehicle model along a circular arc.

The state vector $\boldsymbol{x}_k$ at time point t_k is a five-dimensional vector consisting of the position $(t_{x,k}, t_{y,k})$ and orientation ϕ_k of the model as well as the magnitudes v_k and ω_k of the translational and angular velocities, respectively:

$$\boldsymbol{x}_k = \left(t_{x,k} \;\; t_{y,k} \;\; \phi_k \;\; v_k \;\; \omega_k \right)^T. \tag{4}$$

By integrating the differential equations (1) we obtain the following discrete plant model describing the state transition from time point t_k to time point t_{k+1}:

$$\begin{aligned} t_{x,k+1} &= t_{x,k} + v_k\tau \cdot \tfrac{\sin(\phi_k+\omega_k\tau)-\sin\phi_k}{\omega_k\tau}, \\ t_{y,k+1} &= t_{y,k} - v_k\tau \cdot \tfrac{\cos(\phi_k+\omega_k\tau)-\cos\phi_k}{\omega_k\tau}, \end{aligned} \qquad \begin{aligned} \phi_{k+1} &= \phi_k + \omega_k\tau, \\ v_{k+1} &= v_k, \\ \omega_{k+1} &= \omega_k. \end{aligned} \tag{5}$$

We introduce the usual dynamical systems notation (see, e.g., [Gelb 74]). The symbols $(\hat{\boldsymbol{x}}_k^-, P_k^-)$ and $(\hat{\boldsymbol{x}}_k^+, P_k^+)$ are used, respectively, for the estimated states and their covariances before and after updating based on the measurements at time t_k.

By denoting the transition function of (5) by $\boldsymbol{f}(\cdot)$ and assuming white Gaussian process noise $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, Q_k)$, the prediction equations read as follows

$$\hat{\boldsymbol{x}}_{k+1}^- = \boldsymbol{f}(\hat{\boldsymbol{x}}_k^+), \qquad P_{k+1}^- = F_k P_k^+ F_k^T + Q_k, \tag{6}$$

where F_k is the Jacobian $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}$ at $\boldsymbol{x} = \hat{\boldsymbol{x}}_k^+$.

The four dimensional parameter vectors $\{\boldsymbol{X}\}_{i=1..m}$ from m matched line segments in the image plane build a $(4m)$-dimensional measurement vector $\boldsymbol{z}_k$ assumed to be equal to the measurement function $\boldsymbol{h}_k(\boldsymbol{x}_k)$ plus white Gaussian measurement noise $\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, R_k)$. The measurement noise covariance matrix R_k is block-diagonal. Its blocks are 4×4 covariance matrices as they are defined in equation 12 in [Deriche & Faugeras 90]. As already formulated in Section 3, the line segment parameters are functions of the endpoints of a line segment. We will briefly explain how these endpoints are related to the state (4). A point (x_i, y_i) in the image plane at time instant t_k is the projection of a point $\boldsymbol{x}_{w_{i,k}}$ described in the world coordinate system (see Figure 4). The parameters of this transformation have been obtained off-line based on the calibration procedure of [Tsai 87], using dimensional data extracted from a construction map of the depicted roads. In this way we constrain the motion problem even more because we do not only know that the vehicle is moving on the road plane, but the normal of this plane is known as well. The point $\boldsymbol{x}_{w_{i,k}}$ is obtained by the following rigid transformation from the model coordinate system

$$\boldsymbol{x}_{w_{i,k}} = \begin{pmatrix} \cos\phi_k & -\sin\phi_k & 0 \\ \sin\phi_k & \cos\phi_k & 0 \\ 0 & 0 & 1 \end{pmatrix} \boldsymbol{x}_{m,i} + \begin{pmatrix} t_{x,k} \\ t_{y,k} \\ 0 \end{pmatrix}, \tag{7}$$

where $(t_{x,k}, t_{y,k}, \phi_k)$ are the state parameters and $\boldsymbol{x}_{m,i}$ are the known positions of the vehicle vertices in the model coordinate system.

As already mentioned, we have included the projection of the shadow contour in the measurements in order to obtain more predicted edges segments for matching and to avoid false matches to data edge segments arising from shadows that lie in the neighborhood of predicted model edges. The measurement function of projected shadow edge segments differs from the measurement function of the projections of model vertices in one step. Instead of only one point in the world coordinate system, we get two. One point $\boldsymbol{x}_s$ as vertex of the shadow on the street and a second point $\boldsymbol{x}_w = (x_w, y_w, z_w)$ as vertex on the object which is projected onto the shadow point $\boldsymbol{x}_s$. We assume a parallel projection in

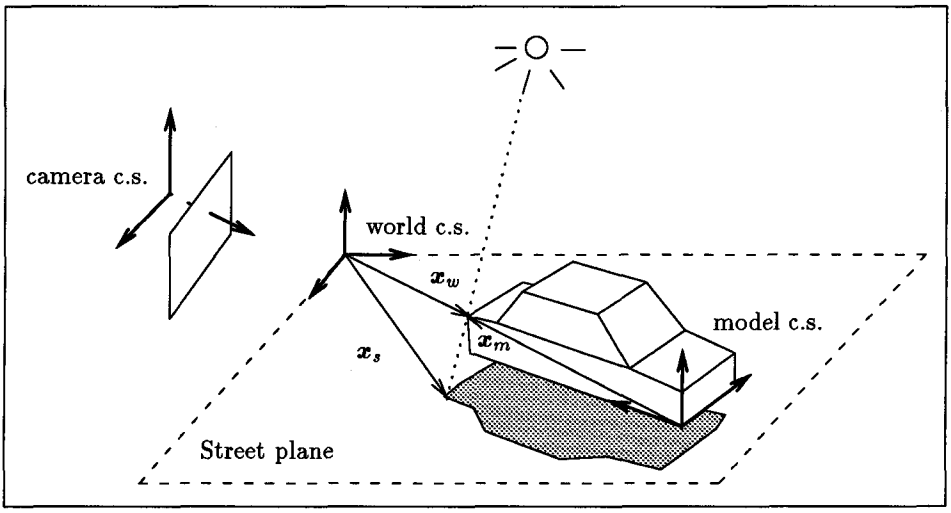

Fig. 4. Description of coordinate systems (c.s.)

shadow generation. Let the light source direction be $(\cos\alpha \sin\beta, \sin\alpha \sin\beta, \cos\beta)^T$ where α and β — set interactively off-line — are the azimuth and polar angle, respectively, described in the world coordinate system. The following expression for the shadow point in the xy-plane (the road plane) of the world coordinate system can be easily derived:

$$\boldsymbol{x}_s = \begin{pmatrix} x_w - z_w \cos\alpha \tan\beta \\ y_w - z_w \sin\alpha \tan\beta \\ 0 \end{pmatrix}. \tag{8}$$

The point $\boldsymbol{x}_w$ can then be expressed as a function of the state using (7). A problem arises with endpoints of line segments in the image which are not projections of model vertices but intersections of occluding line segments. Due to the small length of the possibly occluded edges (for example, the side edges of the hood and of the trunk of the vehicle) we cover this case by the already included uncertainty $\sigma_{\|}$ of the endpoints in the edge direction. A formal solution uses a closed form for the endpoint position in the image as a function of the coordinates of the model vertices belonging to the occluded and occluding edge segments. Such a closed form solution has not yet been implemented in our system.

The measurement function $\boldsymbol{h}_k$ is nonlinear in the state $\boldsymbol{x}_k$. Therefore, we have tested three possibilities for the updating step of our recursive estimation. In all three approaches we assume that the state after the measurement $\boldsymbol{z}_k$ is normally distributed around the estimate $\hat{\boldsymbol{x}}_{k-1}^+$ with covariance P_{k-1}^+ which is only an approximation to the actual a posteriori probability density function (PDF) after an update step based on a nonlinear measurement. An additional approximation is the assumption that the PDF after the nonlinear prediction step remains Gaussian. Thus we state the problem as the search for the maximum of the following a posteriori PDF after measurement $\boldsymbol{z}_k$:

$$p(\boldsymbol{x}_k|\boldsymbol{z}_k) = \tfrac{1}{c} \exp\left\{-\frac{1}{2}(\boldsymbol{z}_k - \boldsymbol{h}_k(\boldsymbol{x}_k))^T R_k^{-1} (\boldsymbol{z}_k - \boldsymbol{h}_k(\boldsymbol{x}_k))\right\} \cdot \exp\left\{-\frac{1}{2}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)^T P_k^{-\,-1} (\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)\right\}, \tag{9}$$

where c is a normalizing constant.

This is a MAP estimation and can be stated as the minimization of the objective function

$$(\boldsymbol{z}_k - \boldsymbol{h}_k(\boldsymbol{x}_k))^T R_k^{-1}(\boldsymbol{z}_k - \boldsymbol{h}_k(\boldsymbol{x}_k)) + (\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)^T P_k^{-1}(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-) \longrightarrow \min_{\boldsymbol{x}_k} \qquad (10)$$

resulting in the updated estimate $\hat{\boldsymbol{x}}_k^+$. In this context the well known Iterated Extended Kalman Filter (IEKF) [Jazwinski 70; Bar-Shalom & Fortmann 88] is actually the Gauss-Newton iterative method [Scales 85] applied to the above objective function whereas the Extended Kalman Filter (EKF) is only one iteration step of this method. We have found such a clarification [Jazwinski 70] of the meaning of EKF and IEKF to be important towards understanding the performance of each method.

A third possibility we have considered is the Levenberg-Marquardt iterative minimization method applied on (10) which we will call Modified IEKF. The Levenberg-Marquardt strategy is a usual method for least squares minimization guaranteeing a steepest descent direction far from minimum and a Gauss-Newton direction near the minimum, thus increasing the convergence rate. If the initial values are in the close vicinity of the minimum, then IEKF and Modified IEKF yield almost the same result.

Due to the mentioned approximations, all three methods are suboptimal and the computed covariances are optimistic [Jazwinski 70]. This fact practically affects the matching process by narrowing the search region and making the matcher believe that the current estimate is much more reliable than it actually is. Practical compensation methods include an addition of artificial process noise or a multiplication with an amplification matrix. We did not apply such methods in our experiments in order to avoid a severe violation of the smoothness of the trajectories. We have just added process noise to the velocity magnitude v and ω (about 10% of the actual value) in order to compensate the inadequacy of the motion model with respect to the real motion of a vehicle.

We have tested all three methods [Thórhallson 91] and it turned out that the IEKF and Modified IEKF are superior to the EKF regarding convergence as well as retainment of a high number of matches. As [Maybank 90] suggested, these suboptimal filters are the closer to the optimal filter in a Minimum Mean Square Error sense the nearer the initial value lies to the optimal estimate. This criterion is actually satisfied by the initial position and orientation values in our approach obtained by backprojecting image features clustered into objects onto a plane parallel to the street. In addition to the starting values for position and orientation, we computed initial values for the velocity magnitudes v and ω during a bootstrap process. During the first n_{boot} (= 2, usually) time frames, position and orientation are statically computed. Then initial values for the velocities are taken from the discrete time derivatives of these positions and orientations.

Concluding the estimation section, we should mention that the above process requires only a slight modification for the inclusion of the shape parameters of the model as unknowns in the state vector. Since shape parameters remain constant, the prediction step is the same and the measurement function must be modified by substituting the model points $\boldsymbol{x}_{m,i}$ with the respective functions of the shape parameters instead of considering them to have constant coordinates in the model coordinate system.

5 Experiments and Results

Parking Area

As a first experiment we used an image sequence of about 80 frames in which one car is moving from the left to the right leaving a parking area (see the three upper images of

Figure 5). The image of the moving car covers about 60 × 100 pixels of a frame. In this example it was not necessary, and due to the illumination conditions not even possible, to use shadow edges in the matching process. The matched models for the three upper frames are illustrated in the middle row of Figure 5, with more details given in the lower three figures. In the lower figures we see the extracted straight lines, the backprojected model segments (dashed lines) and the matched data segments, emphasized by thick lines.

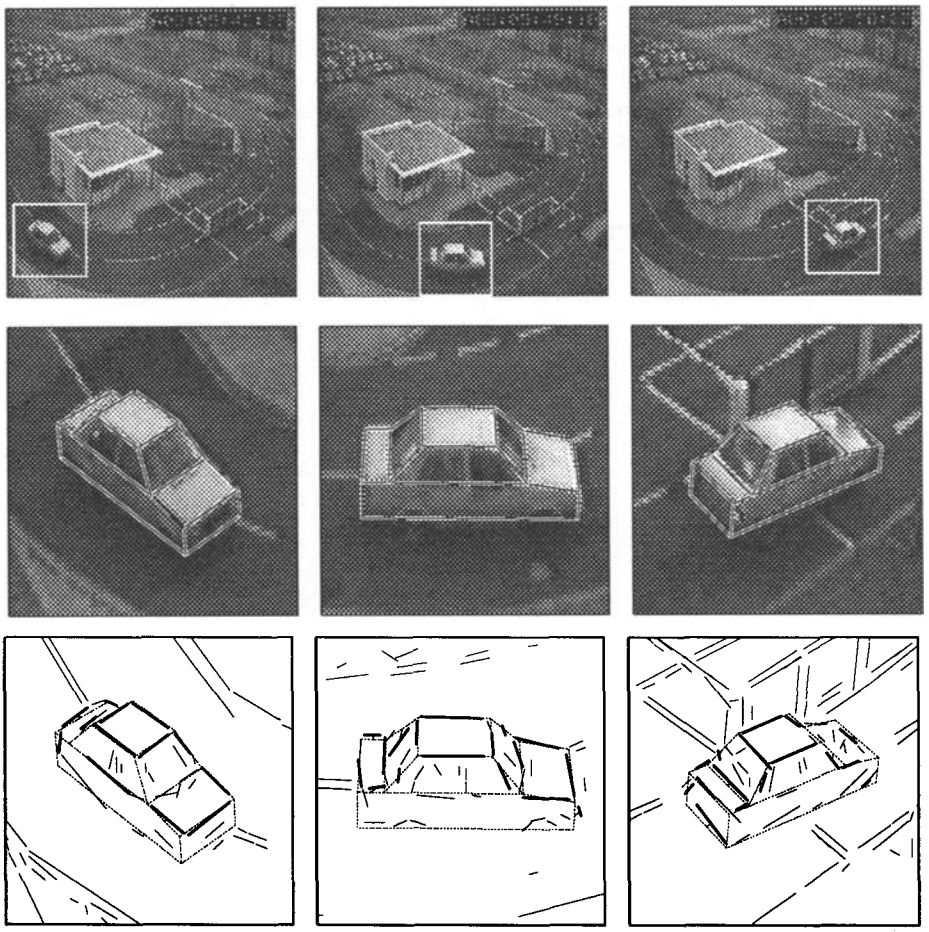

Fig. 5. The first row shows the 4^{th}, 41^{st} and 79^{st} frame of an image sequence. The three images in the middle row give an enlarged section of the model matched to the car moving in the image sequence. The lower three figures exhibit the correspondences (thick lines) between image line segments and model segments (dashed lines) in the same enlarged section as in the middle row.

The resultant object trajectories will be used as inputs to a process of associating motion verbs to trajectory segments. Since such subsequent analysis steps are very sensitive to noise we attempt to obtain smoother object trajectories. In order to obtain such smooth motion we use a small process noise for the magnitude of the velocity v and the angular velocity ω. In this and the subsequent experiments, we therefore use a

process noise of $\sigma_v = 10^{-3}\,\frac{\mathrm{m}}{\mathrm{s}}$ and $\sigma_\omega = 10^{-4}\,\frac{\mathrm{rad}}{\mathrm{s}}$. Given this σ_v and σ_ω, the majority of the translational and angular accelerations are assumed to be $\dot{v} < \sigma_v/\tau = .625\,\frac{\mathrm{m}}{\mathrm{s}^2}$ and $\dot{\omega} < \sigma_\omega/\tau = 2.5 \cdot 10^{-3}\,\frac{\mathrm{rad}}{\mathrm{s}^2}$, respectively, with $\tau = t_{k+1} - t_k = 40\,\mathrm{ms}$.

The bootstrap phase is performed using the first two frames in order to obtain initial estimates for the magnitudes of the velocities v and ω. Since the initially detected moving region does not always correctly span the image of the moving object, we used values equal to approximately half of the average model length, i.e. $\sigma_{t_{x_0}} = \sigma_{t_{y_0}} = 3\,\mathrm{m}$. An initial value for the covariance in the orientation ϕ is roughly estimated by considering the differences in the orientation between the clustered displacement vectors, i.e $\sigma_{\phi_0} = .35\,\mathrm{rad}$.

The car has been tracked during the entire sequence of 80 frames with an average number of about 16 line segment correspondences per frame. The computed trajectory for this moving car is given in Figure 6.

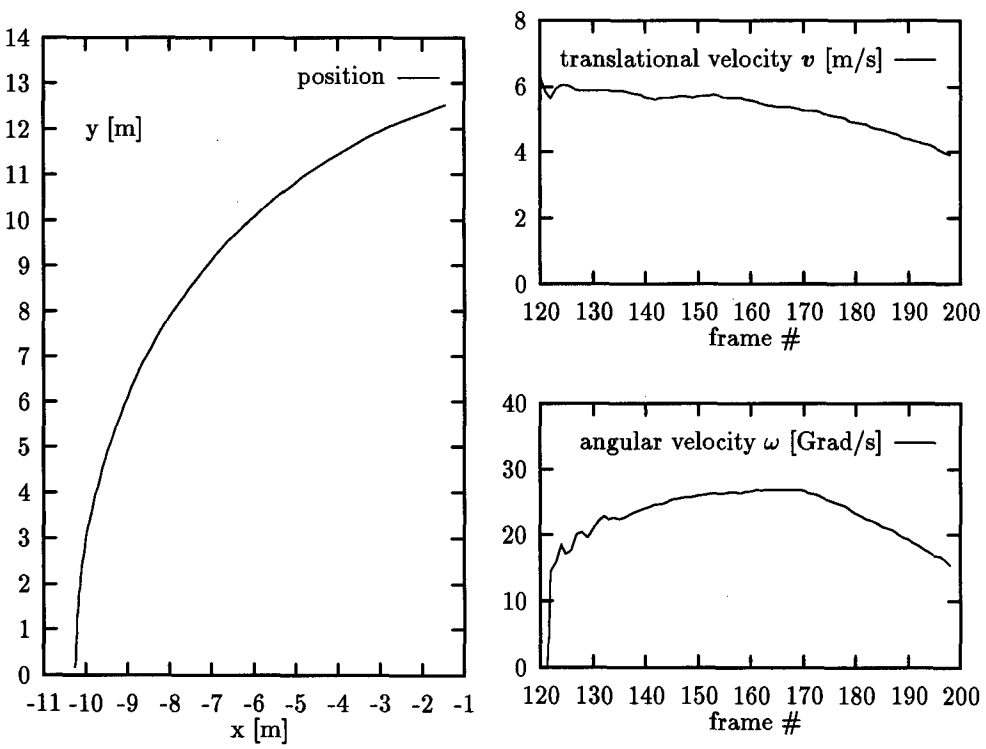

Fig. 6. The estimated position as well as the translational and angular velocity of the moving car of Figure 5.

Multilane Street Intersection

The next group of experiments involved an image subsequence of about 50 frames of a much frequented multilane street intersection. In this sequence there are several moving vehicles with different shapes and dimensions, all vehicles turning to the left (Figure 7).

The size of the images of the moving vehicles varies in this sequence from 30×60 to 20×40 pixels in a frame. Figure 2 shows some intermediate steps in extracting the line

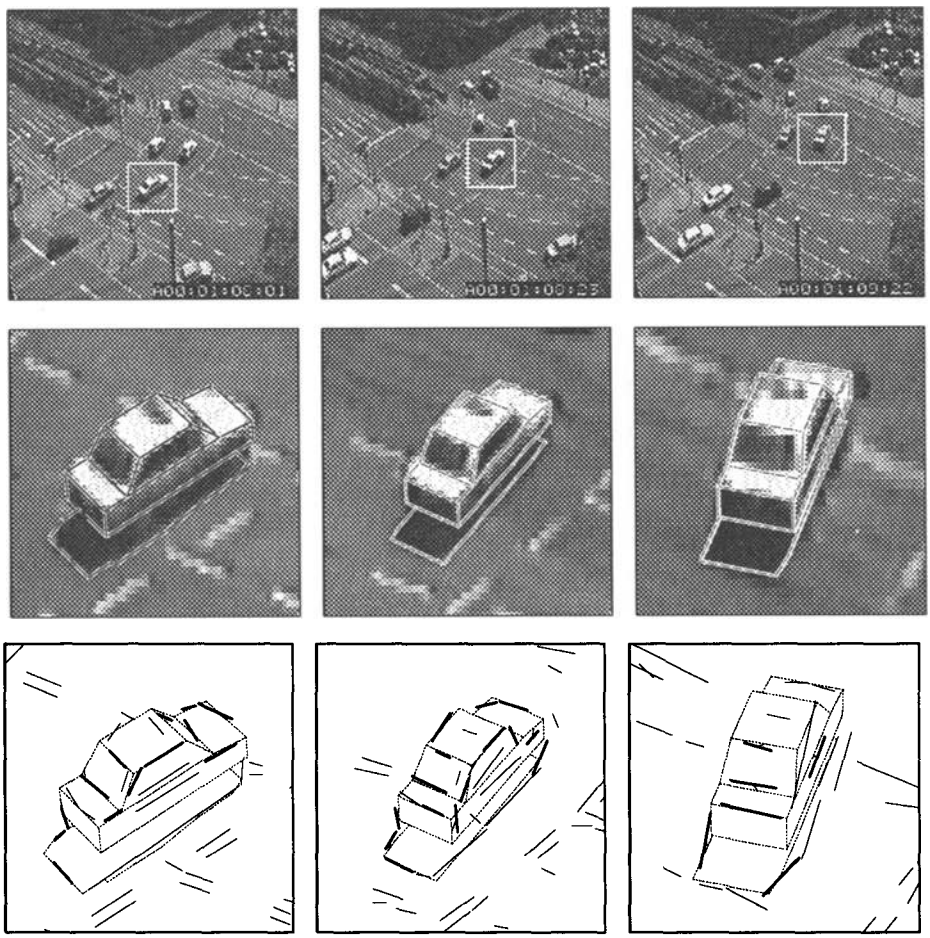

Fig. 7. The first row shows the 3^{rd}, 25^{th} and 49^{th} frame of an image sequence recorded at a much frequented multilane street intersection. The middle row shows an enlarged section of the model matched to the taxi (object #6) moving in the center of the frame. The lower three figures exhibit the correspondences between image line segments and model segments in the same enlarged section as in the middle row.

segments. We explicitly present this Figure in order to give an idea of the complexity of the task to detect and track a moving vehicle spanning such a small area in the image. We used the same values for the process noise and the initial covariances as in the previous experiment. As in the previous example we used the first two frames for initial estimation of v and ω. In this experiment we used the shadow edges as additional line segments in the matching process as described in Section 4.

Five of the vehicles appearing in the first frame have been tracked throughout the entire sequence. The reason for the failure in tracking the other vehicles has been the inability of the initialization step to provide the system with appropriate initial values. To handle this inability an interpretation search tree is under investigation.

In the upper part of Figure 7 we see three frames out of this image sequence. In the middle part of Figure 7, the matched model of a taxi is given as an enlarged section

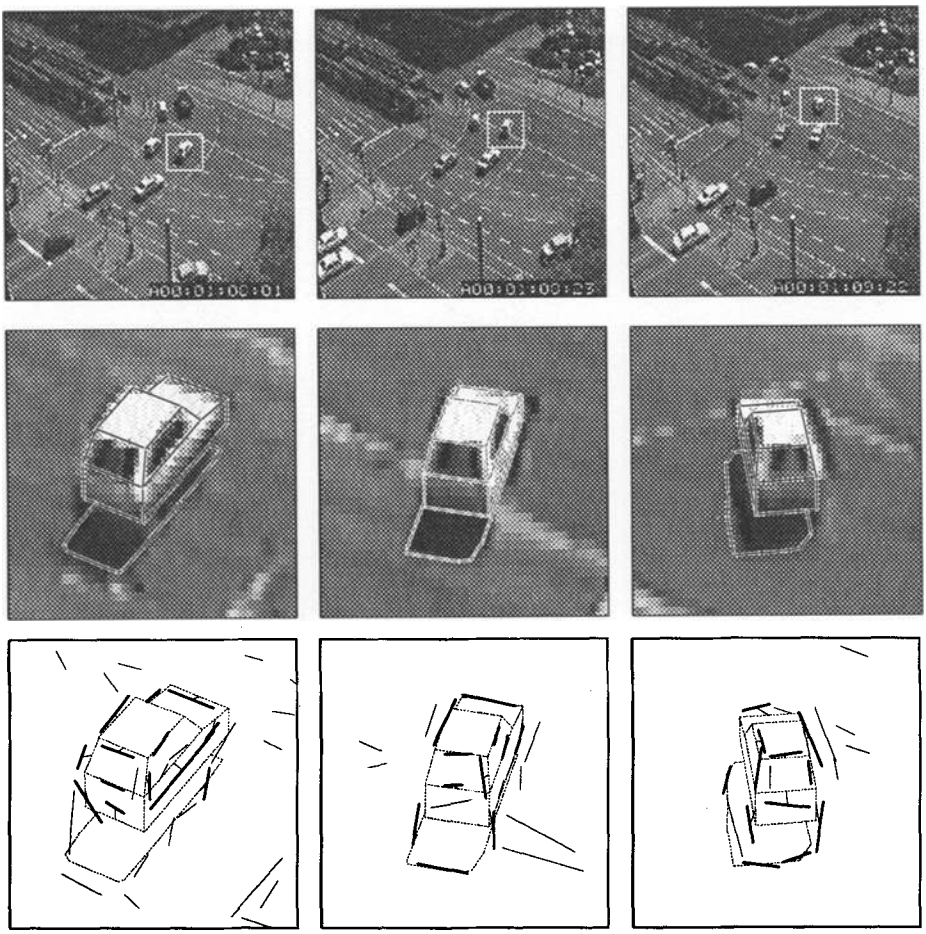

Fig. 8. The first row shows the 3^{rd}, 25^{th} and 49^{th} frame of an image sequence recorded at a much frequented multilane street intersection. The middle row shows an enlarged section of the model matched to the small car (object #5) moving left of the center of the frame. The lower three figures exhibit the correspondences between image line segments and model segments in the same enlarged section as in the middle row.

for the three upper images. In the lower three figures the correspondences of image line segments and the model line segments are given. Figure 9 shows the resultant object trajectory. Figure 8 shows another car of the same image sequence with the resultant trajectory also displayed in Figure 9.

6 Related Works

In this section we discuss related investigations about tracking and recognizing object models from image sequences. The reader is referred to the excellent book by [Grimson 90a] for a complete description of research on object recognition from a single image.

[Gennery 82] has proposed the first approach for tracking 3D-objects of known structure. A constant velocity six degrees of freedom (DOF) model is used for prediction and

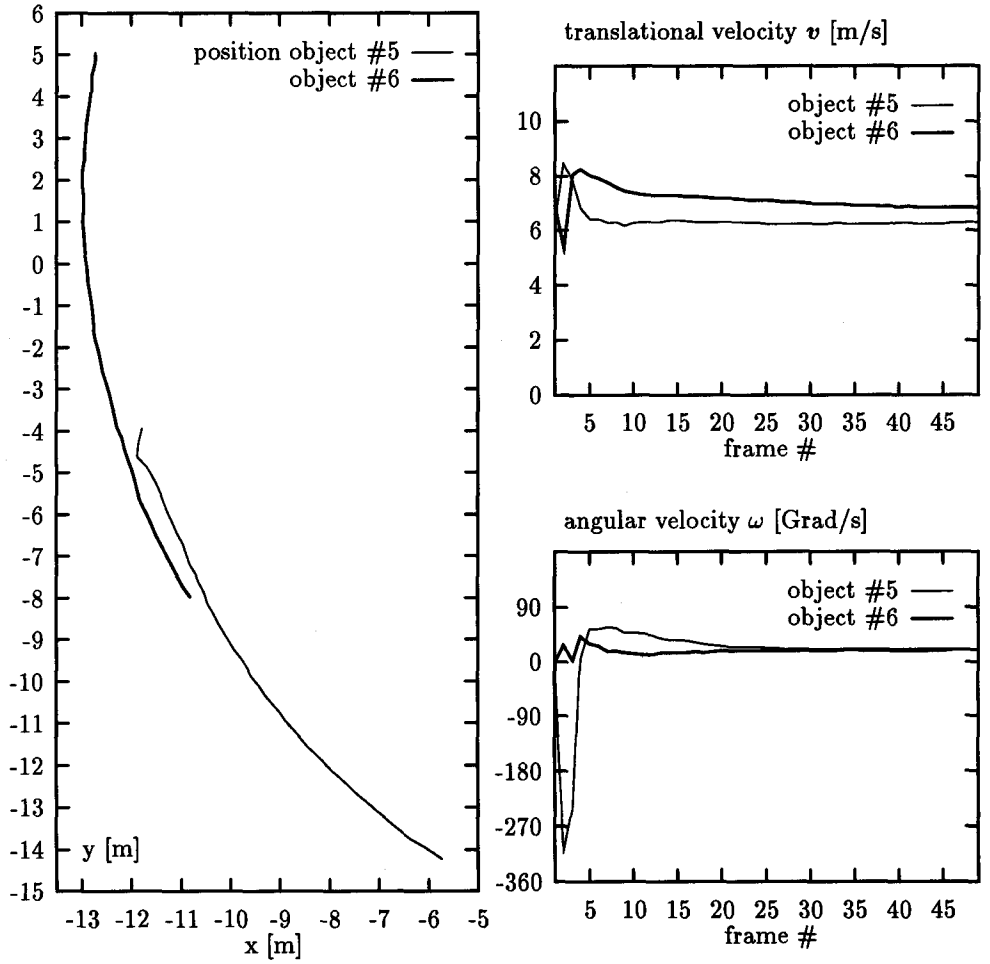

Fig. 9. The estimated positions as well as the translational and angular velocities of the moving cars in Figure 8 (object # 5) and Figure 7 (object # 6).

an update step similar to the Kalman filter – without addressing the nonlinearity – is applied. Edge elements closest to the predicted model line segments are associated as corresponding measurements.

[Thompson & Mundy 87] emphasize the object recognition aspect of tracking by applying a pose clustering technique. Candidate matches between image and model vertex pairs define points in the space of all transformations. Dense clusters of such points indicate a correct match. Object motion can be represented by a trajectory in the transformation space. Temporal coherence then means that this trajectory should be smooth. Predicted clusters from the last time instant establish hypotheses for the new time instants which are verified as matches if they lie close to the newly obtained clusters. The images we have been working on did not contain the necessary vertex pairs in order to test this novel algorithm. Furthermore, we have not been able to show that the approach of [Thompson & Mundy 87] is extensible to handling of parameterized objects.

[Verghese *et al.* 90] have implemented in real-time two approaches for tracking 3D-known objects. Their first method is similar to the approach of [Thompson & Mundy 87] (see the preceding discussion). Their second method is based on the optical flow of line segments. Using line segment correspondences, of which initial (correct) correspondences are provided interactively at the beginning, a prediction of the model is validated and spurious matches are rejected.

[Lowe 90, 91] has built the system that has been the main inspiration for our matching strategy. He does not enforce temporal coherence however, since he does not imply a motion model. Pose updating is carried out by minimization of a sum of weighted least squares including a priori constraints for stabilization. Line segments are used for matching but distances of selected edge points from infinitely extending model lines are used in the minimization. [Lowe 90] uses a probabilistic criterion to guide the search for correct correspondences and a match iteration cycle similar to ours.

A gradient-ascent algorithm is used by [Worrall *et al.* 91] in order to estimate the pose of a known object in a car sequence. Initial values for this iteration are provided interactively at the beginning. Since no motion model is used the previous estimate is used at every time instant to initialize the iteration. [Marslin *et al.* 91] have enhanced the approach by incorporating a motion model of constant translational acceleration and angular velocity. Their filter optimality, however, is affected by use of the speed estimates as measurements instead of the image locations of features.

[Schick & Dickmanns 91] use a generic parameterized model for the object types. They solve the more general problem of estimating both the motion and the shape parameters. The motion model of a car moving on a clothoid trajectory is applied including translational as well as angular acceleration. The estimation machinery of the simple EKF is used and, so far, the system is tested on synthetic line images only.

The following approaches do not consider the correspondence search problem but concentrate only on the motion estimation. A constant velocity model with six DOF is assumed by [Wu *et al.* 88] and [Harris & Stennet 90; Evans 90], whereas [Young & Chellappa 90] use a precessional motion model.

A quite different paradigm is followed by [Murray *et al.* 89]. They first try to solve the structure from motion problem from two monocular views. In order to accomplish this, they establish temporal correspondence of image edge elements and use these correspondences to solve for the infinitesimal motion between the two time instants and the depths of the image points. Based on this reconstruction [Murray *et al.* 89] carry out a 3D-3D correspondence search. Their approach has been tested with camera motion in a laboratory set-up.

7 Conclusion and future work

Our task has been to build a system that will be able to compute smooth trajectories of vehicles in traffic scenes and will be extensible to incorporate a solution to the problem of classifying the vehicles according to computed shape parameters. We have considered the task to be difficult because of the complex illumination conditions and the cluttered environment of real world traffic scenes and the small effective field of view that is spanned by the projection of each vehicle given a stationary camera. In all experiments mentioned in the cited approaches in the last section, the projected area of the objects covers a quite high portion of the field of view. Furthermore, only one of them [Evans 90] is tested under outdoor illumination conditions (landing of an aircraft).

In order to accomplish the above mentioned tasks we have applied the following constraints. We restricted the degrees of freedom of the transformation between model and camera from six to three by assuming that a vehicle is moving on a plane known a priori by calibration. We considered only a simple time coherent motion model because of the high sampling rate (25 frames pro second) and the knowledge that vehicles do not maneuver abruptly.

The second critical point we have been concerned about is the establishment of good initial matches and pose estimates. Most tracking approaches do not emphasize the severity of this problem of establishing a number of correct correspondences in the starting phase and feeding the recursive estimator with quite reasonable initial values. Again we have used the a priori knowledge of the street plane position and the results of clustering picture domain descriptors into object hypotheses of a previous step. Thus we have been able to start the tracking process with a simple matching scheme and feed the recursive estimator with values of low error covariance.

The third essential point we have addressed is the additional consideration of shadows. Data line segments arising from shadows are not treated any more as disturbing data like markings on the road, but they contribute to the stabilization of the matching process.

Our work will be continued by the following steps. First, the matching process should be enhanced by introducing a search tree. In spite of the good initial pose estimates, we are still confronted occasionally with totally false matching combinations due to the highly ambiguous structure of our current vehicle model. Second, the generic vehicle model enables a simple adaptation to the image data by varying the shape parameters. These shape parameters should be added as unknowns and estimated along time.

Acknowledgements

The financial support of the first author by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) and of the second as well as the third author by the Deutscher Akademischer Austauschdienst (DAAD, German Academic Exchange Service) are gratefully acknowledged.

References

[Bar-Shalom & Fortmann 88] Y. Bar-Shalom, T.E. Fortmann, *Tracking and Data Association*, Academic Press, New York, NY, 1988.

[Deriche & Faugeras 90] R. Deriche, O. Faugeras, Tracking line segments, *Image and Vision Computing* **8** (1990) 261–270.

[Evans 90] R. Evans, Kalman Filtering of pose estimates in applications of the RAPID video rate tracker, in *Proc. British Machine Vision Conference*, Oxford, UK, Sept. 24-27, 1990, pp. 79–84.

[Gelb 74] A. Gelb (ed.), *Applied Optimal Estimation*, The MIT Press, Cambridge, MA and London, UK, 1974.

[Gennery 82] D.B. Gennery, Tracking known three-dimensional objects, in *Proc. Conf. American Association of Artificial Intelligence*, Pittsburgh, PA, Aug. 18-20, 1982, pp. 13–17.

[Grimson 90a] W.E.L. Grimson, *Object recognition by computer: The role of geometric constraints*, The MIT Press, Cambridge, MA, 1990.

[Grimson 90b] W. E. L. Grimson, The combinatorics of object recognition in cluttered environments using constrained search, *Artificial Intelligence* **44** (1990) 121–165.

[Harris & Stennet 90] C. Harris, C. Stennet, RAPID - A video rate object tracker, in *Proc. British Machine Vision Conference*, Oxford, UK, Sept. 24-27, 1990, pp. 73–77.

[Jazwinski 70] A.H. Jazwinski, *Stochastic Processes and Filtering Theory*, Academic Press, New York, NY and London, UK, 1970.

[Koller *et al.* 91] D. Koller, N. Heinze, H.-H. Nagel, Algorithmic Characterization of Vehicle Trajectories from Image Sequences by Motion Verbs, in *IEEE Conf. Computer Vision and Pattern Recognition*, Lahaina, Maui, Hawaii, June 3-6, 1991, pp. 90–95.

[Korn 88] A. F. Korn, Towards a Symbolic Representation of Intensity Changes in Images, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-10** (1988) 610–625.

[Lowe 85] D. G. Lowe, *Perceptual Organization and Visual Recognition*, Kluwer Academic Publishers, Boston MA, 1985.

[Lowe 87] D. G. Lowe, Three-Dimensional Object Recognition from Single Two-Dimensional Images, *Artificial Intelligence* **31** (1987) 355–395.

[Lowe 90] D. G. Lowe, Integrated Treatment of Matching and Measurement Errors for Robust Model-Based Motion Tracking, in *Proc. Int. Conf. on Computer Vision*, Osaka, Japan, Dec. 4-7, 1990, pp. 436–440.

[Lowe 91] D.G. Lowe, Fitting parameterized three-dimensional models to images, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **13** (1991) 441–450.

[Marslin *et al.* 91] R.F. Marslin, G.D. Sullivan, K.D. Baker, Kalman Filters in Constrained Model-Based Tracking, in *Proc. British Machine Vision Conference*, Glasgow, UK, Sept. 24-26, 1991, pp. 371–374.

[Maybank 90] S. Maybank, Filter based estimates of depth, in *Proc. British Machine Vision Conference*, Oxford, UK, Sept. 24-27, 1990, pp. 349–354.

[Murray *et al.* 89] D.W. Murray, D.A. Castelow, B.F. Buxton, From image sequences to recognized moving polyhedral objects, *International Journal of Computer Vision* **3** (1989) 181–208.

[Scales 85] L. E. Scales, *Introduction to Non-Linear Optimization*, Macmillan, London, UK, 1985.

[Schick & Dickmanns 91] J. Schick, E. D. Dickmanns, Simultaneous estimation of 3D shape and motion of objects by computer vision, in *Proc. IEEE Workshop on Visual Motion*, Princeton, NJ, Oct. 7-9, 1991, pp. 256–261.

[Thompson & Mundy 87] D.W. Thompson, J.L. Mundy, Model-based motion analysis - motion from motion, in *The Fourth International Symposium on Robotics Research, R. Bolles and B. Roth (ed.), MIT Press, Cambridge, MA*, 1987, pp. 299–309.

[Thórhallson 91] T. Thórhallson, *Untersuchung zur dynamischen Modellanpassung in monokularen Bildfolgen*, Diplomarbeit, Fakultät für Elektrotechnik der Universität Karlsruhe (TH), durchgeführt am Institut für Algorithmen und Kognitive Systeme, Fakultät für Informatik der Universität Karlsruhe (TH), Karlsruhe, August 1991.

[Tsai 87] R. Tsai, A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses, *IEEE Trans. Robotics and Automation* **3** (1987) 323–344.

[Verghese *et al.* 90] G. Verghese, K.L. Gale, C.R. Dyer, Real-time, parallel motion tracking of three dimensional objects from spatiotemporal images, in V. Kumar, P.S. Gopalakrishnan, L.N. Kanal (ed.), *Parallel Algorithms for Machine Intelligence and Vision*, Springer-Verlag, Berlin, Heidelberg, New York, 1990, pp. 340–359.

[Worrall *et al.* 91] A.D. Worrall, R.F. Marslin, G.D. Sullivan, K.D. Baker, Model-Based Tracking, in *Proc. British Machine Vision Conference*, Glasgow, UK, Sept. 24-26, 1991, pp. 310–318.

[Wu *et al.* 88] J.J. Wu, R.E. Rink, T.M. Caelli, V.G. Gourishankar, Recovery of the 3-D location and motion of a rigid object through camera image (an Extended Kalman Filter approach), *International Journal of Computer Vision* **3** (1988) 373–394.

[Young & Chellappa 90] G. Young, R. Chellappa, 3-D Motion estimation using a sequence of noisy stereo images: models, estimation and uniqueness results, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-12** (1990) 735–759.

This article was processed using the LaTeX macro package with ECCV92 style