Beginning a Transition from a Local to a More Global Point of View in Model–Based Vehicle Tracking

Michael Haag[1] and Hans–Hellmut Nagel[1,2]

[1] Institut für Algorithmen und Kognitive Systeme, Fakultät für Informatik der Universität Karlsruhe (TH), Postfach 6980, D–76128 Karlsruhe, Germany
[2] Fraunhofer–Institut für Informations– und Datenverarbeitung (IITB), Fraunhoferstr. 1, D–76131 Karlsruhe, Germany

Abstract. This contribution attempts to move beyond the status where single moving objects in video image sequences are tracked separately in the scene domain, based on individually adapted approaches and parameters. Instead, we investigate which performance can be achieved by a combination of approaches based on edge element orientation and on optical flow, applied to a variety of image sequences and vehicles. Five different image sequences of traffic scenes recorded under different conditions have been evaluated. Quantitative statements are provided about the success rates of the approach after evaluating over 5.500 full video–frames, i. e. more than $3\frac{1}{2}$ minutes of real–world video, using *one single approach and a single parameter set.* Remaining tracking failures are analyzed and classified.

1 Introduction and Related Work

Surveillance of traffic scenes by means of video cameras potentially can have a variety of applications, including extraction of traffic statistics or the realization of intelligent control of traffic lights. Such applications require a robust system which is able to track vehicles and to characterize their maneuvers even under difficult conditions. Image sequences of road traffic recorded by a stationary standard S–VHS video camera are characterized by small vehicle images, shadows, changing illumination conditions, low contrast object images, uncertain camera calibration, and occluded scene objects. Since it is sometimes even impossible for a human observer to identify the low contrast image of a vehicle in a single grey value image, we combine purely spatial image features with motion analysis. Consequent exploitation of knowledge about the 3D–structure of the scene helps us to deal with significant occlusions between objects and with changing illumination conditions (for example objects moving from a bright region into a shadow).

This contribution does not address the problem of optimizing a tracking approach and its parameters for a particular moving object. Instead, we want to apply one and the *same approach* with the *same parameter* set in order to

track as many objects as possible in different image sequences, well knowing that we may be able to track single objects better if we adapt the approach or the parameters to the specific circumstances. A summary after evaluation of over 5.500 frames and more than 50 tracking experiments should help to identify problems which have to be solved in order to meet the requirements of traffic surveillance systems as outlined above. With this contribution, we are leaving the status where basic research addressed tracking single vehicles under particular circumstances, for example low contrast images or occlusion. Instead, we shift the discussion from specific model–based techniques to an engineering–like process by systematically identifying and subsequently solving the most pressing among the remaining problems.

Surveys of the literature up to 1995 concerning image motion interpretation and visual surveillance can be found in [2, 1].

The majority of published approaches track vehicles in the 2D *picture domain*. [1] describe experiments regarding aspects of an integrated visual traffic surveillance system. Vehicles are tracked by extracting optical flow vectors and correlating them with causal constraints in a Bayesian Belief Network (BBN). [10] use these geometric results in order to identify 'overtaking' or 'give–way' situations in traffic sequences. [16] have to cope with particularly small object images since the vehicles are usually moving far away from the recording camera. Objects are tracked by computing differences between frames and by updating a background model in order to deal with image changes not originating from object motion. By careful combination of spatial 2D image features (the vehicle's contour and its 2D grey value pattern), [6, 7] track vehicles in the picture domain. [3] perform high level traffic scene interpretation. At the image evaluation level, only simple methods — like background subtraction and pattern matching — are employed in order to roughly characterize object movements based on a few frames.

Some recently published approaches can track vehicles in the 3D *scene domain*. [4] interactively track vehicles in image sequences recorded by hand–held, uncalibrated cameras exploiting physics–based models. Different image sequences showing traffic intersections are evaluated by [12] using a single approach and a single parameter set. They exploit the grey value gradient magnitude, but no motion information is used (except for the initialization step). [11] introduce optical flow matching during tracking in order to cope with low contrast vehicle images and occlusions. Tracking without exploitation of optical flow, but by taking explicitly modelled occlusions into account, has been reported by [8]. [13, 14] provide a motion model including the steering angle of vehicles and a stochastic model for the driving behaviour. Directed edges serve as image measurements. A function assessing the current vehicle pose estimation is maximized by means of a simplex search method. A new *covariance update filter* should capture fast changes in state variables which are not explicitly included in the motion model.

The investigations reported in this literature address specific technical problems and solution approaches. A transition phase from such methodologically

oriented research towards approaches to design and evaluate practically applicable systems has not yet become recognizable.

2 Approach

2.1 Model Based Vehicle Tracking

Tracking of vehicles is initialized in our system by choosing a suitable start frame and computing its optical flow field. Neighboring similar optical flow vectors, which are assumed to correspond to the same moving object, are clustered in the image plane and are then back–projected onto a plane parallel to the road plane by means of the known camera calibration. Exploitation of the mean position, orientation, and magnitude of the back–projected flow vectors results in an initial estimate $\hat{\mathbf{x}}_0^-$ for the vehicle position, orientation, and speed *in the scene domain.* For each object candidate detected in this manner, we have to interactively select an appropriate polyhedral vehicle model. This vehicle model is subsequently projected into the image plane using the initial state estimate $\hat{\mathbf{x}}_0^-$. This state estimate is then updated by matching image features to the projected model as described in the following two subsections. The resulting updated state vector $\hat{\mathbf{x}}_0^+$ forms the basis for predicting the a priori state $\hat{\mathbf{x}}_1^-$ in the next half–frame using a model of circular motion with constant radius. Discrepancies between this motion model and reality are attributed to Gaussian system noise $\mathbf{w} \sim N(\mathbf{0}, Q)$ [5]: $\hat{\mathbf{x}}_{k+1}^- = \mathbf{f}(\hat{\mathbf{x}}_k^+) + \mathbf{w}$, where $\mathbf{f}$ is the system transition function and Q denotes the covariance matrix for the system noise.

In order to obtain spatio–temporal grey value gradients needed for edge element and optical flow extraction, we employ filter kernels especially adapted to *interlaced* video images, enabling us to evaluate both fields of a frame with full frame resolution. Thus, state vector estimation is performed for 50 *half–frames* in each second.

2.2 Matching Based on Edge Elements

One part of the update step of the state estimation is carried out by matching edge elements to model edge segments similar to [17], but adopted to image sequences with particularly small vehicle images. Edge elements $\mathbf{e} = (u, v, \phi)^T$ are defined as grey value transitions at image location $(u, v)^T$ whose gradient magnitude exceeds a threshold and which has a local maximum of the gradient magnitude in gradient direction ϕ. Edge segments of a polyhedral vehicle model are projected into the image plane exploiting the known camera model and employing a hiddenline algorithm. Visible projected model edge segments $\mathbf{m} = (u_m, v_m, \theta, l)^T$ are characterized by the image coordinates $(u_m, v_m)^T$ of the center of the edge segment, its orientation θ, and its length l. Matching of extracted edge elements to projected model edge segments is performed in two steps: (1) associate appropriate edge elements to model edge segments and (2) update the a–priori state estimation by minimizing a distance measure on this association.

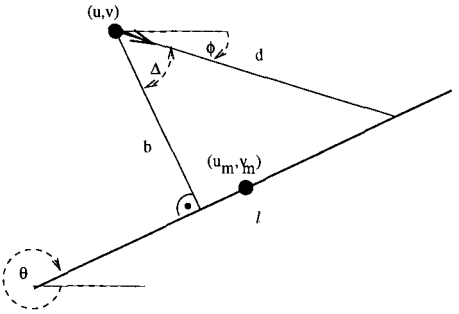

Fig. 1. Distance measure between an edge element $\mathbf{e} = (u, v, \phi)^T$ and a projected model edge segment $\mathbf{m} = (u_m, v_m, \theta, l)^T$ [9,17]. Angles are measured against the horizontal. l denotes the length of the model segment projected into the image plane whereas (u_m, v_m) denote its centerpoint coordinates.

Data Association. The Euclidean distance b between an edge element and a model edge segment (see Fig. 1) induces a Mahalanobis distance where b is assumed to be Gauss–distributed with covariance σ_b^2:

$$\sigma_b^2 = \frac{\partial b(\mathbf{e}, \hat{\mathbf{x}}^-)}{\partial(\mathbf{e}, \mathbf{x})} \cdot \begin{pmatrix} \Sigma_e & 0_{3\times 5} \\ 0_{5\times 3} & P^- \end{pmatrix} \cdot \left(\frac{\partial b(\mathbf{e}, \hat{\mathbf{x}}^-)}{\partial(\mathbf{e}, \mathbf{x})} \right)^T \quad . \tag{1}$$

$\Sigma_e = \mathrm{diag}(\sigma_u^2, \sigma_v^2, \sigma_\phi^2)$ denotes the measurement noise for edge elements and P^- represents the a priori state uncertainty about $\hat{\mathbf{x}}^-$. Edge elements are rejected if their Mahalanobis distance $b(\mathbf{e}, \hat{\mathbf{x}}^-) \frac{1}{\sigma_b^2} b(\mathbf{e}, \hat{\mathbf{x}}^-)$ exceeds the threshold of the $(1-\alpha)$–quantile of the χ^2–distribution. Since the Mahalanobis distance depends on the current state estimation uncertainty P^-, the acceptance area for edge elements in the image plane is larger at the initialization time point than at subsequent time points due to relatively high initial state uncertainties, compared to smaller uncertainties at later time points. In order not to obtain too large acceptance areas incorporating edge elements which do not belong to the actual vehicle image, only certain maximal distances for b are allowed. These limits depend on the length of the corresponding model edge segments: for long model segments, larger Euclidean distances of edge elements are allowed since for long model segments larger distances to edge elements can occur when the model segments do not exactly overlap the edge elements to be associated.

State Estimation Update. In order to incorporate evidence about the *orientation* of edge elements into an assessment of their association with model segments obtained according to the preceding subsection, we employ the following distance measure d between edge elements $(u, v, \phi)^T$ and the corresponding projected model edge segment $(u_m, v_m, \theta, l)^T$ (see Fig. 1):

$$d(\mathbf{e}, \mathbf{x}) = \frac{b}{\cos \Delta} = \frac{-\sin\theta \cdot (u - u_m) + \cos\theta \cdot (v - v_m)}{\cos(\phi - (\theta + \frac{\pi}{2}))} \quad , \tag{2}$$

where $(u_m, v_m, \theta, l)^T$ denotes the projected model edge segment with smallest Mahalanobis distance to $\mathbf{e}$.

In order to update the state vector, a MAP–estimation is performed: given a Gauss–distributed state variable $\mathbf{x}_k$, the a–posteriori distribution at time point k conditioned on the measurements $\mathcal{Z}_k$ up to and including time point k is:

$$p(\mathbf{x}_k|\mathcal{Z}_k) = \frac{1}{p(\mathbf{z}_k|\mathcal{Z}^{k-1})} e^{-\Gamma(\mathbf{x}_k)} \quad . \tag{3}$$

The cost function $\Gamma(\mathbf{x}_k)$ is composed as follows:

$$\Gamma(\mathbf{x}_k) = \underbrace{\frac{1}{2}\sum_{\mathbf{e}} d(\mathbf{e},\mathbf{x}_k)^T \frac{1}{s} d(\mathbf{e},\mathbf{x}_k)}_{=:\Gamma_e} + \underbrace{\frac{1}{2}(\mathbf{x}_k - \hat{\mathbf{x}}_k^-)^T (P^-)^{-1} (\mathbf{x}_k - \hat{\mathbf{x}}_k^-)}_{=:\Gamma_s} \quad , \tag{4}$$

using the distance d of (2) and

$$s := \frac{\partial d(\mathbf{e},\hat{\mathbf{x}}^-)}{\partial \mathbf{e}} \cdot \Sigma_e'(\mathbf{e}) \cdot \left(\frac{\partial d(\mathbf{e},\hat{\mathbf{x}}^-)}{\partial \mathbf{e}}\right)^T \quad .$$

$\Sigma_e'(\mathbf{e})$ denotes the measurement noise Σ_e — see (1) — of edge element extraction, weighted with the gradient magnitude of the edge element $\mathbf{e}$:

$$\Sigma_e'(\mathbf{e}) = \left(\frac{\|\nabla g(\mathbf{e})\|}{\sum_{\mathbf{e}} \|\nabla g(\mathbf{e})\|}\right)^{-\frac{1}{2}} \cdot \Sigma_e \cdot \left(\frac{\|\nabla g(\mathbf{e})\|}{\sum_{\mathbf{e}} \|\nabla g(\mathbf{e})\|}\right)^{-\frac{1}{2}} \quad . \tag{5}$$

The measurement noise $\Sigma_e'(\mathbf{e})$ is thus large for edge elements with low gradient magnitude.

Search for the state vector $\mathbf{x}$ which maximizes the a–posteriori probability $p(\mathbf{x}_k|\mathcal{Z}_k)$ in (3) implies minimizing the cost function Γ in (4) with respect to $\mathbf{x}$. This results in an update step of an Iterated Extended Kalman Filter (IEKF).

2.3 Matching Based on Optical Flow

Optical flow matching is particularly valuable for tracking objects with low contrast images. Sometimes even a human observer can hardly recognize an object in a single frame, but notices it in a *sequence* of frames. We perform optical flow matching according to [11] with some modifications. For each image location $\boldsymbol{\xi}$, the corresponding scene point $\mathbf{x}_w(\mathbf{x})$ on the surface of the instantiated vehicle model is determined. The expected displacement rate $\dot{\boldsymbol{\xi}}$ at this image location is computed according to:

$$\dot{\boldsymbol{\xi}} = \frac{\partial \boldsymbol{\xi}}{\partial t} = \frac{\partial \boldsymbol{\xi}}{\partial \mathbf{x}_c} \frac{\partial \mathbf{x}_c}{\partial \mathbf{x}_w} \frac{\partial \mathbf{x}_w(\mathbf{x})}{\partial t} \quad ,$$

where $\mathbf{x}_w$ denotes the world coordinates of a surface point, $\mathbf{x}_c$ its camera coordinates and $\boldsymbol{\xi}$ its image location. The scene displacement rate $\dot{\mathbf{x}}_w = \frac{\partial \mathbf{x}_w}{\partial t}$ of the object point $\mathbf{x}_w$ is determined by means of the estimated speed and angular velocity of the object.

Data Association Displacement rate vectors $\dot{\boldsymbol{\xi}}$ are matched to optical flow vectors $\mathbf{u}(\boldsymbol{\xi})$ estimated at the same image location, if the optical flow vector has a minimal confidence value and if its orientation and magnitude does not significantly differ from orientation and magnitude of the corresponding displacement rate vector. The idea behind the latter test is that expected displacement rate vectors express the global state of the vehicle while optical flow vectors are computed based on local, noisy grey value distributions. We reject, therefore, optical flow vectors which are not compatible with our expectation. In addition, optical flow vectors are not matched, if the magnitude of the corresponding displacement rate vector becomes too small. In other words: if we expect the image motion to be small or vanish, we only rely on the edge element matching described in the previous subsection. This facilitates the tracking of stopping vehicles.

State Estimation Update Like in Subsection 2.2, the state vector $\mathbf{x}$ is updated by means of a MAP estimation step which results in the minimization of the following cost function — see (4) — where $\boldsymbol{\xi} = \boldsymbol{\xi}(\mathbf{x})$:

$$\Gamma(\mathbf{x}) = \underbrace{\frac{1}{2}\sum(\dot{\boldsymbol{\xi}} - \mathbf{u})^T \left(\Sigma'_{of}(\boldsymbol{\xi})\right)^{-1} (\dot{\boldsymbol{\xi}} - \mathbf{u})}_{=:\Gamma_{of}} + \Gamma_s \quad . \tag{6}$$

The overall cost function $\Gamma(\mathbf{x})$ which has to be minimized when combining edge element and optical flow matching is thus composed as:

$$\Gamma(\mathbf{x}) = \Gamma_e(\mathbf{x}) + \Gamma_{of}(\mathbf{x}) + \Gamma_s(\mathbf{x}) \quad . \tag{7}$$

$\Sigma'_{of}(\boldsymbol{\xi})$ in (6) denotes the measurement noise $\Sigma_{of} = \sigma_{of}^2 \cdot I$ of optical flow estimation weighted by an image location specific confidence value $c(\boldsymbol{\xi})$ which is given as the minimal eigenvalue of the outer gradient product $(g_x, g_y)^T \cdot (g_x, g_y)$ averaged over a local area around $\boldsymbol{\xi}$:

$$\Sigma'_{of}(\boldsymbol{\xi}) = \left(\frac{c(\boldsymbol{\xi})}{\sum\limits_{\boldsymbol{\xi}} c(\boldsymbol{\xi})}\right)^{-\frac{1}{2}} \cdot \Sigma_{of} \cdot \left(\frac{c(\boldsymbol{\xi})}{\sum\limits_{\boldsymbol{\xi}} c(\boldsymbol{\xi})}\right)^{-\frac{1}{2}} \quad . \tag{8}$$

The confidence value is a measure for the grey value structure in a local area around the location where an optical flow has to be estimated. Thus, a larger measurement uncertainty will be associated to optical flow vectors with low confidence value.

3 Results

3.1 Sequence 'nb'

In the 'nb' traffic intersection image sequence, 20 object candidates have been detected automatically by the optical flow segmentation step in half-frame #105,

see Fig. 2, top right. The object candidates 1, 2, and 6 were not worth tracking: the first two belong to the bright truck and trailer configuration with joints for which an articulated model would be needed. The object candidate 6 belongs to a shadow cast by a transporter. The remaining 17 object candidates have been tracked in parallel, see Fig. 3.

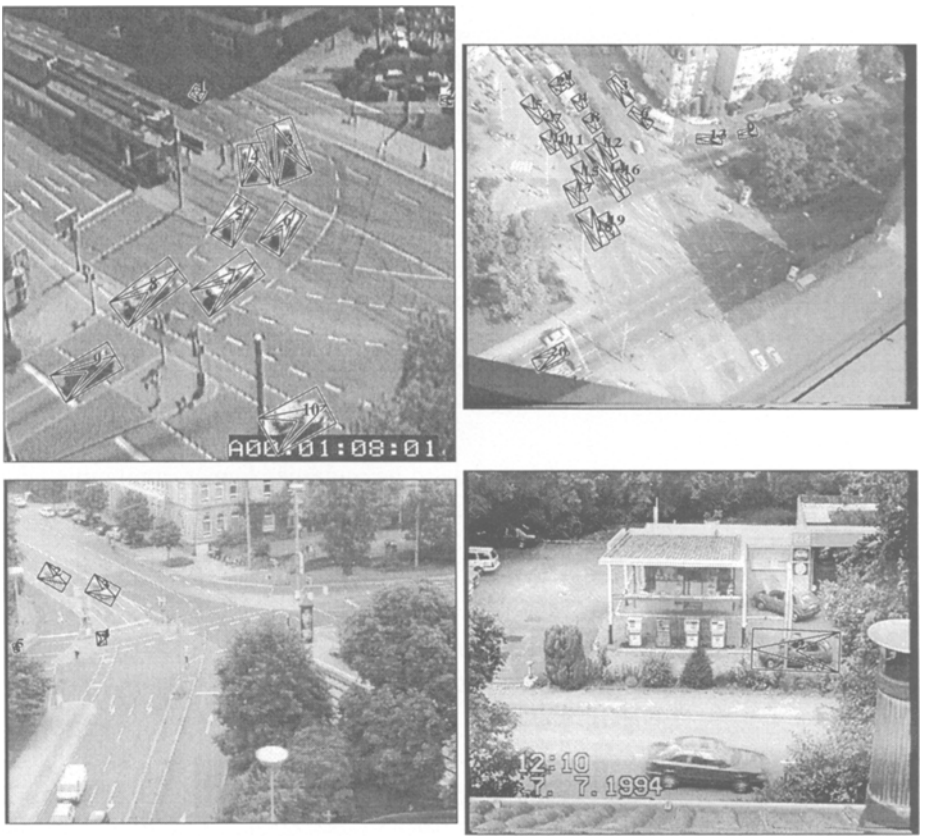

Fig. 2. Object candidates have been automatically extracted by segmenting optical flow fields obtained at a suitable start frame of the sequences 'dt_v' (top left), 'nb' (top right), 'kwbB' (bottom left), and 'gas station' (bottom right).

Objects 3, 4, and 16 have been badly initialized. Object 4 is partially occluded by a bright truck whose occluding effect has not been taken into account. Object 16 is occluded by the tank lorry and lost early due to the bad initial pose estimation. Objects 5, 11, and 15 are tracked until leaving the field of view at the bottom of the image, but their models lag significantly behind the corresponding vehicle images. Since these three objects can be tracked without any lag by only using edge element matching without optical flow, it seems that optical flow is

not able to capture the strong acceleration of these objects. This problem can not be observed for the objects 7, 10, and 17 which, too, are accelerating. But in contradistinction to objects 5, 11, and 15, these objects produce darker images. Object 8, which is hardly visible in the dark shadows cast by the buildings in the center right part of the image, has been tracked successfully until it reaches the right margin of the image. Object 9 which turns right sharply in the upper right part of the image has been lost around the time point #360 when it is occluded by a post which has not been modelled so far. Objects 10, 17, 18 and 19 have been tracked successfully until they leave the field of view at the bottom of the image. The left turning object 12 has been tracked until it leaves the image on the right hand side. Object 13 turns right sharply in the upper right quadrant of the image and has been tracked until disappearing from the image at the top, albeit with a lateral and longitudinal lag. Although the left turning tank lorry (object 14) is an articulated object, it could be tracked with a rigid, single parallelepiped model without modelling the joint. Finally, object 20 which just approaches the intersection in the lower left of the image and then stops, has been tracked successfully.

In this sequence, the invisible buildings positioned near the lower right corner of the image have been modelled approximately in the scene domain in order to determine their shadows. Shadows cast by vehicles driving in this region are not considered, since there is no evidence for such shadow contours in the images.

3.2 Sequence 'dt_v'

The initialization step for half–frame time point #5 of the 'dt_v' sequence (see Fig. 2, top left) delivers 10 object candidates which have been tracked in parallel. Object 2 is not completely in the field of view at the initialization time point, and object 10 is not completely visible due to occlusions by the time stamp in the lower right part of the image. The initial pose estimation, therefore, is quite inaccurate for these two objects. Nevertheless, they are not lost during tracking, although object 10 is even occluded in the bottom right corner of the image by a tree which has not been modelled. Object 1 is moving through a very dark shadow region and can hardly be identified even by human observers in single images during this period. Due to the exploitation of optical flow, it could nevertheless be tracked successfully.

The resulting trajectories after 88 half–frame time points can be seen in Fig. 4.

3.3 Sequence 'kwbB'

Tracking in the 'kwbB' sequence has been initialized at half–frame #10 (objects 2 and 3, see Fig. 2, bottom left), #430 (object 6), #600 (object 12), #780 (objects 13 and 14), #851 (objects 5, 7, 8, and 10), and #2010 (object 1). The resulting trajectories are shown in Fig. 5. Objects 2 and 10 are initialized with a lateral lag. Object 2 is occluded *completely* for some time by a big traffic sign (see Fig. 6), but has been tracked nevertheless until leaving the field of view.

Fig. 3. Tracking results of the 'nb' sequence at half–frame time point #580 after automatic initialization in half–frame #105 (see Fig. 2, top right). The shadow edges of the objects 8, 12, and 14 are automatically excluded when entering the shadow regions of the buildings.

This success must be attributed to the use of optical flow measurements and pose predictions by the Kalman Filter. Object 7, too, is completely occluded by this traffic sign. But unlike object 2, this object stops just behind the traffic sign. The renewed start of object 7 when the traffic light switched to green could, therefore, not be tracked.

3.4 Gas Station

Regarding the gas station sequence depicted in Fig. 2 (bottom right), tracking has been initialized each time when a moving object enters the field of view. For the gas station scene, we had the possibility to measure the real–world coordinates of several static 3D scene components. These measurements of 3D scene points have been interactively associated to their corresponding image locations, resulting in a more robust camera calibration than in the case of the intersection scenes where only 2D maps were available. The gas station sequence is characterized by larger vehicle images compared to the intersection scenes, significant occlusions by static and moving scene components, changing illumination conditions during recording, low contrast vehicle images (especially when objects move under the canopy of the gas station building), and by more complex driving maneuvers (stopping, starting, overtaking, backing up).

Fig. 4. Tracking results after evaluation of 88 half–frames of the 'dt_v' sequence.

While there was no need to explicitly treat occlusions in the intersection sequences after incorporating optical flow into the measurements, this is not sufficient for the gas station sequence. Occlusions by static scene components, like posts, bushes, or petrol pumps, and by moving vehicles occur. Unlike in most of the intersection scenes investigated, partially occluded vehicles remain stationary for long time intervals so that optical flow information will not help in these cases. We, therefore, explicitly model occlusion in the 3D scene domain by providing a model of the gas station, similar to [8]. Occluding moving scene objects are tracked automatically in the scene domain and are considered when tracking the occluded objects. All image features belonging to (known) occluding scene components are excluded when vehicles have to be matched. In contradistinction to [8], the order in which the objects have to be tracked (occluding objects must be tracked prior to occluded objects) is determined automatically for each half–frame time point. There is no need anymore to precompute the trajectories of occluding scene components before tracking occluded objects. This facilitates to track all object candidates in an image sequence in parallel, nevertheless taking mutual occlusions into account.

Fig. 5. Estimated trajectories at half–frame time point #2820 of the 'kwbB' sequence.

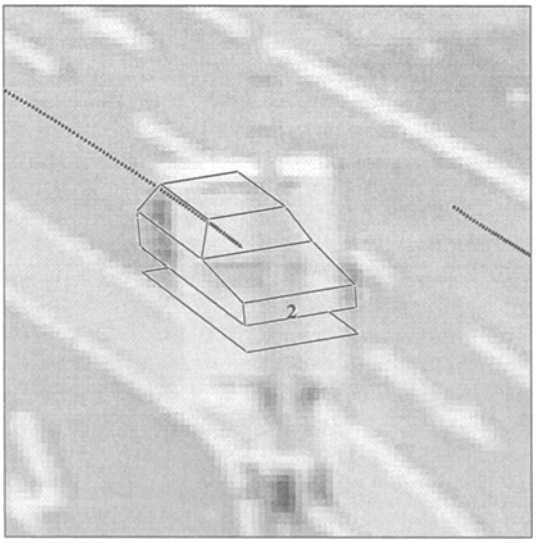

Fig. 6. Temporary, but complete occlusion of object 2 at half–frame #90 due to a traffic sign. Optical Flow and the Kalman Filter pose prediction facilitate nevertheless to track this vehicle in its *continuous motion* even through the period of (practically complete) transitory occlusion.

Objects 1, 2, and 4 (see Fig. 7) as well as objects 5 and 11 (see Fig. 8), respectively, have been tracked in parallel, automatically considering their mutual occlusions and the occlusions by static scene components which have been modelled in the 3D scene domain. Objects 2 and 4 are tracked well, though heavy occlusions occur, especially during tracking of object 4 which overtakes the standing object 1 on the back lane. The model match for object 1 becomes worse when it backs up beginning with half–frame #1800 in order to evade object 4 on the back lane. Similar considerations hold for object 10 when it proceeds after filling up in order to evade object 4 at half–frame #3200. Object 5 is tracked successfully on its way to the rightmost petrol pump on the back lane. The match becomes worse when the vehicle backs up after remaining stationary for about 1200 half–frames and subsequently evades object 11 standing in front. But object 5 is not lost until leaving the field of view. Although object 11 has been badly initialized automatically, it could be tracked somewhat while overtaking the standing object 5 and while standing behind the left post. Beginning with half–frame #6000, object 11 backs up in order to park in front of object 5. In this phase, the vehicle image is not lost, but the model match becomes quite inaccurate. Object 11 is definitely lost while proceeding to the left exit of the gas station beginning with half–frame #7000. Object 6 has been successfully tracked while proceeding to the leftmost petrol pump on the front lane, standing, backing up to the center petrol pump, and standing once again. The match becomes worse only while object 6 subsequently proceeds to the left exit due to significant occlusions by bushes and the (not modelled) left tree.

4 Performance

Table 1 gives an overview of the results obtained by tracking 56 object candidates in five different image sequences. The results of the intersection image sequence 'et', comprising about 90 half–frames and 11 moving objects, have not been presented in a figure, due to space limitations.

In total, only five objects have been completely lost, mainly either due to inaccurate initial pose estimations provided by the segmented optical flow field and/or due to occlusions. 34 object candidates could be successfully tracked, while 17 object candidates could be tracked, but only with significant lateral and/or longitudinal lags between estimated and actual vehicle position.

The main problems regarding the unsatisfactory as well as the completely failed tracking results seem to be initialization and occlusion. In nearly all cases, inaccurate initializations can be attributed to occlusions or to cases where vehicles just enter the field of view and are not yet completely visible. Indeed, we are not yet using any occlusion information at the initialization step. But there are also several cases of a perfectly initialized tracking which later failed due to occlusions, especially in cases where significantly occluded vehicles remain stationary for a while.

Please note that Table 1 only provides a quite aggregated representation of the evaluation since it does not contain the *length* of each object's trajectory.

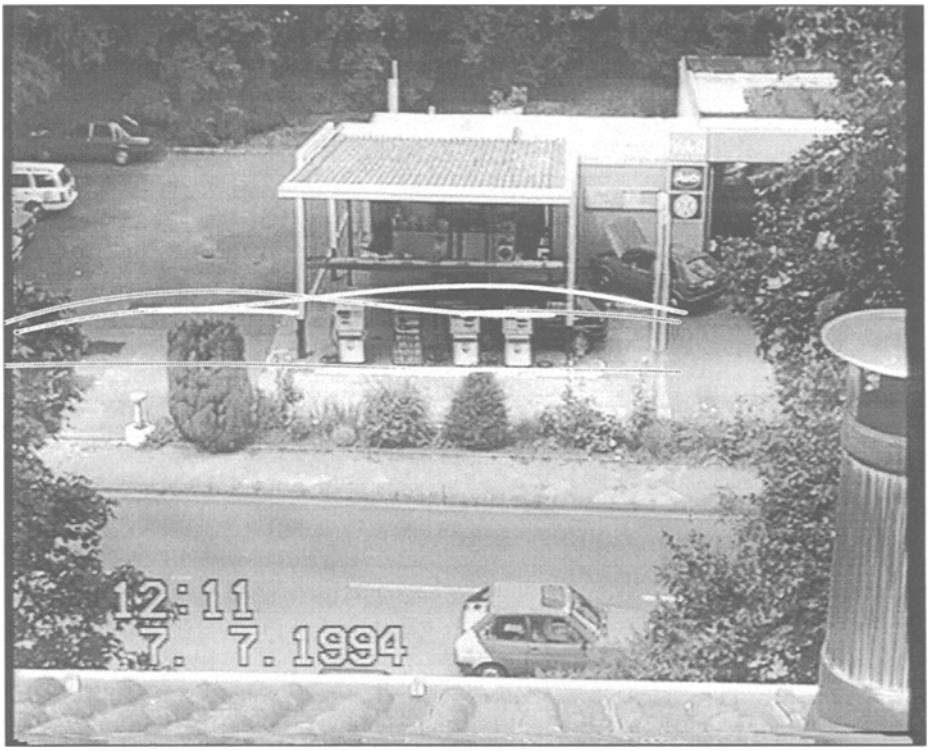

Fig. 7. Estimated trajectories of objects 1, 2, and 4 at half-frame time point #4905 of the gas station sequence.

Table 1. Tracking rates and failure reasons for the five video image sequences considered in this contribution.

Sequence	'et'	'nb'	'dt_v'	'kwbB'	gas station	total
# half-frames processed	88	600	88	3.100	8.000	11.876
# object candidates	11	17	10	11	7	56
# successfully tracked	8	9	8	7	2	34
# satisfactorily tracked	3	5	2	3	4	17
# completely lost		3		1	1	5
reason for failures (multiple reasons per object possible):						
# bad initialization	2	3	2	2	1	9
# low contrast image					4	4
# occlusion	1	4	2	1	4	11
# too strong acceleration		3				3
# too strong turning		1		1		2

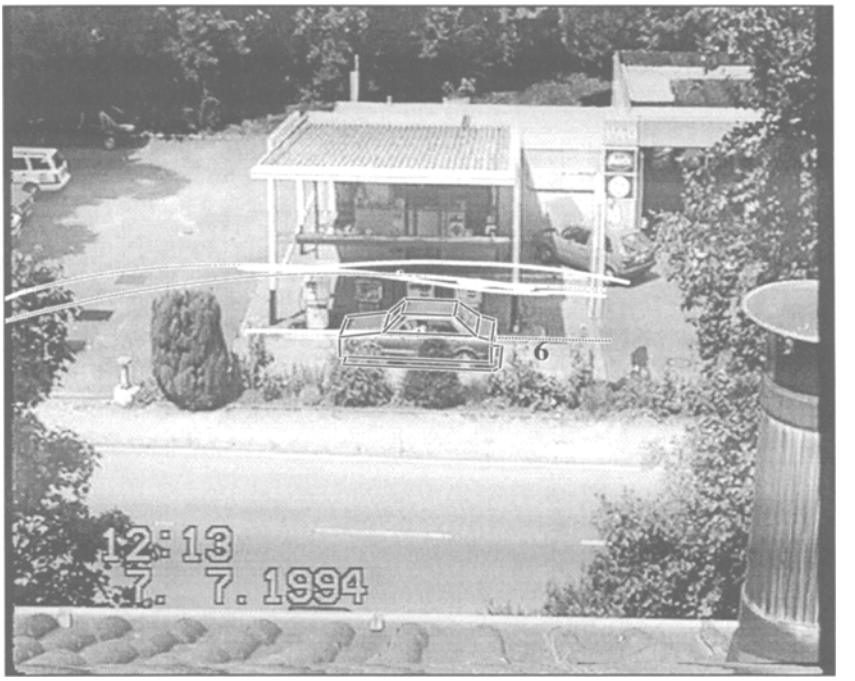

Fig. 8. Estimated trajectories of objects 5, 6, 10, and 11 at half-frame time point #8450. Note that object 6 is leaving the gas station premise beginning with this time point. This is the reason why the front trajectory does not yet extend to the left side of the image frame shown here.

If we consider the state prediction and update step per object and per half-frame as one processing unit ('tracking step'), we obtain a total of over 34.000 single tracking steps. We are, however, humbly aware of the fact that we have to evaluate at least an order of magnitude more image sequences in order to obtain a more reliable assessment of our tracking approach.

Although we do not focus on real-time evaluation in our vehicle tracking system, we were forced to drastically (by a factor of 10) reduce computation times in order to facilitate the evaluation of several image sequences each containing up to 20 moving objects within reasonable time. This now results in tracking times from between 3 and 7 sec (depending on the object image size) per object and half-frame on a SUN ULTRA I 167 MHz workstation including all input and output operations, like loading grey value images over a network.

5 Conclusion

This contribution adapts an edge element matching approach, originally developed for real-time tracking of large object images, and combines it with optical flow matching as well as 3D occlusion modelling. Significant modifications of

different original approach versions were necessary in order to achieve robust tracking results for small and low contrast vehicle images in different image sequences: (1) employment of an *Iterated* Extended Kalman Filter in order to deal with larger discrepancies between a–priori state estimations and the true state, (2) incorporating the gradient magnitude into the measurement noise for edge elements in order to become independent of critical thresholds for gradient magnitudes, (3) limiting the area of acceptance for edge elements depending on the length of projected model edge segments, (4) switching–off optical flow matching for all image locations where no motion is expected in order to track stopping objects, (5) using bias–corrected optical flow in order to avoid systematic underestimations of optical flow vector magnitudes affecting the speed estimation of moving objects, (6) tracking multiple objects in parallel and determine automatically the tracking order at each particular point in time (occluding objects have to be processed prior to occluded objects), (7) reducing drastically the computation times of optical flow matching in order to obtain reasonable tracking times even for long image sequences and multiple object configurations.

The results reported here have been obtained by an *experimental* system where we focus on tracking most of the moving objects in different image sequences with a single approach and one parameter set instead of perfectly tracking single objects by individually adjusting the parameters to each moving object. In the research reported here, attention is neither directed towards vehicle classification nor towards real time evaluation.

The extended evaluation performed in this contribution now shifts us to a position where we can identify remaining key problems which still prevent us from constructing a robust traffic surveillance application as stated in the introductory remarks. These problems can be classified into three categories: (1) structural tracking and modelling problems (lost objects due to occlusion and strong acceleration or turning maneuvers), (2) initialization problems (better initial pose estimations required, elimination of 'false alarm' object candidates like pedestrians, automatic selection of appropriate initialization time points, automatic selection of appropriate models), and (3) real–time evaluation restrictions. The results reported in this contribution suggest the following strategy: First, consider the tracking and modelling problems which are not caused by unsuitable initializations. After that, develop initialization approaches delivering better initial pose estimations so that a sufficient majority of moving objects can be tracked. Subsequently, a fully automatic initialization, including frame and model selection, has to be established. Only when the evaluation system will have reached this stage, we intend to focus on the real–time aspect.

References

1. H. Buxton, S. Gong: *Visual Surveillance in a Dynamic and Uncertain World.* Artificial Intelligence **78** (1995) 431–459.
2. C. Cédras, M. Shah: *Motion–Based Recognition: A Survey.* Image and Vision Computing **13**:2 (1995) 129–155.

3. S. Dance, T. Caelli, Z.-Q. Liu: *A Concurrent, Hierarchical Approach to Symbolic Scene Interpretation.* Pattern Recognition **29**:11 (1996) 1891–1903.
4. W.F. Gardner, D.T. Lawton: *Interactive Model-Based Vehicle Tracking.* IEEE Transactions on Pattern Analysis and Machine Intelligence **18**:11 (1996) 1115–1121.
5. A. Gelb: *Applied Optimal Estimation.* The MIT Press, Cambridge / MA, 1974.
6. S. Gil, R. Milanese, and T. Pun: *Combining Multiple Motion Estimates for Vehicle Tracking.* Proc. Fourth European Conference on Computer Vision 1996 (ECCV '96), 15–18 April 1996, Cambridge / UK; B. Buxton and R. Cipolla (Eds.), Lecture Notes in Computer Science **1065** (Vol. II), Springer–Verlag, Berlin, Heidelberg 1996, pp. 307–320.
7. S. Gil, R. Milanese, and T. Pun: *Comparing Features for Target Tracking in Traffic Scenes.* Pattern Recognition **29** (1996) 1285–1296.
8. M. Haag, Th. Frank, H. Kollnig, H.–H. Nagel: *Influence of an Explicitly Modelled 3D Scene on the Tracking of Partially Occluded Vehicles.* Computer Vision and Image Understanding **65**:2 (1997) 206-225.
9. F. Heimes, H.–H. Nagel, Th. Frank: *Model-Based Tracking of Complex Innercity Road Intersections.* Mathematical and Computer Modelling, 1998, *in press.*
10. R. Howarth, H. Buxton: *Visual Surveillance Monitoring and Watching.* Proc. Fourth European Conference on Computer Vision 1996 (ECCV '96), 15–18 April 1996, Cambridge / UK; B. Buxton and R. Cipolla (Eds.), Lecture Notes in Computer Science **1065** (Vol. II), Springer–Verlag, Berlin, Heidelberg 1996, pp. 321–334.
11. H. Kollnig, H.–H. Nagel: *Matching Objects to Segments from an Optical Flow Field.* Proc. Fourth European Conference on Computer Vision 1996 (ECCV '96), 15–18 April 1996, Cambridge / UK; B. Buxton and R. Cipolla (Eds.), Lecture Notes in Computer Science **1065** (Vol. II), Springer–Verlag, Berlin, Heidelberg 1996, pp. 388–399.
12. H. Kollnig, H.–H. Nagel: *3D Pose Estimation by Directly Matching Polyhedral Models to Gray Value Gradients.* International Journal of Computer Vision **23**:3 (1997) 283–302.
13. S.J. Maybank, A.D. Worrall, G.D. Sullivan: *A Filter for Visual Tracking Based on a Stochastic Model for Driver Behaviour.* Proc. Fourth European Conference on Computer Vision 1996 (ECCV '96), 15–18 April 1996, Cambridge / UK; B. Buxton and R. Cipolla (Eds.), Lecture Notes in Computer Science **1065** (Vol. II), Springer–Verlag, Berlin, Heidelberg 1996, pp. 540–549.
14. S.J. Maybank, A.D. Worrall, G.D. Sullivan: *Filter for Car Tracking Based on Acceleration and Steering Angle.* Proc. of the 7th British Machine Vision Conference (BMVC '96), 9–12 September 1996, Edinburgh, England, R. B. Fisher and E. Trucco (Eds.), Vol. 2, ISBN 0 9521898 5 2, 1996, pp. 615–624.
15. H.–H. Nagel, M. Haag: *Bias-Corrected Optical Flow Estimation for Road Vehicle Tracking.* International Conference on Computer Vision (ICCV '98), 4–7 January 1998, Bombay/India, Narosa Publishing House New Delhi a. o. 1998, pp. 1006–1011.
16. M.K. Teal, T.J. Ellis: *Spatial–Temporal Reasoning Based on Object Motion.* Proc. of the 7th British Machine Vision Conference (BMVC '96), 9–12 September 1996, Edinburgh, England, R. B. Fisher and E. Trucco (Eds.), Vol. **2**, ISBN 0 9521898 5 2, 1996, pp. 465–474.
17. M. Tonko, K. Schäfer, F. Heimes, H.–H. Nagel: *Towards Visually Servoed Manipulation of Car Engine Parts.* Proc. IEEE International Conference on Robotics and Automation, Albuquerque/NM, 20–25 April 1997, R. W. Harrigan (Ed.), IEEE Computer Society Press, Los Alamitos/CA, 1997, pp. 3166–3171.