Integrating Behavior-based Prediction for Tracking Vehicles in Traffic Videos

A. Fexa and H.–H. Nagel*

Institut für Algorithmen und Kognitive Systeme
Universität Karlsruhe (TH)
76128 Karlsruhe, Germany

Abstract. Road vehicles usually remain within marked lanes. We study this hypothesis in particular to track road vehicles through extended periods of occlusion *without*, however, relying on 3D-models of occluding foreground bodies. A potential onset of occlusion is detected by a fuzzy conjunction of large, *facet-specific* color changes and a low ratio of the number of pixels with a prediction-compatible Optical-Flow (OF) vector relative to the total number of pixels within a facet of the 3D-polyhedral vehicle model. Experimental results for the entire approach are presented.

1 Introduction

In addition to an application perspective, tracking road vehicles in traffic videos offers a wide range of *challenges for methodological investigations*. Data-driven approaches such as background subtraction may serve well, e.g., for vehicle counting on motorway segments with low traffic density under favourable weather conditions. Reliably tracking *all* vehicles at a congested innercity intersection, however, still exceeds the capabilities of State-of-the-Art 3D-model-based tracking approaches.

This latter type of approach normally incorporates partial a-priori knowledge about the image plane appearance of a vehicle in the form of a polyhedral vehicle model, without any prior assumptions about color and texture of model-facets. In conjunction with a short-term motion model, usually in the form of constant-speed straight-line or circular motion, such a coarse polyhedral approximation allows to predict how the shape of a vehicle's image-plane projection depends on variations of 3D-vehicle-pose relative to the camera.

A complication occurs if vehicles are (partially) occluded as seen from the center of projection of a recording video camera. Simple motion-models no longer suffice because vehicle maneuvers may change while a vehicle is occluded. It thus becomes desirable to formulate assumptions about sequences of vehicle maneuvers, i.e. about vehicle *behavior*. The investigation to be reported here addresses implications of an attempt to incorporate a *behavior-based prediction step* into a 3D-model-based vehicle tracking system for traffic videos.

* Corresponding author

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

Such a systems approach will have to cope not only with tracking individual vehicles, but with the full range of pertinent (sub-)tasks: (i) detection of vehicle images, (ii) choice and parameterisation of an appropriate 3D vehicle representation, (iii) selection of pose and motion parameters for this vehicle representation, i.e. initialisation of 3D-model-based tracking, (iv) tracking proper, (v) detection and handling of (temporary, partial) occlusion of a tracked vehicle, (vi) detection of tracking failures as distinguished from vehicles being parked or leaving the field of view, and (vii) reporting all results in an easily inspectable format. The effort to cope with the latter aspect must not be underestimated once such a system has to process tens of thousands of video frames.

A study covering the multitude of effects which may influence the outcome of a tracking experiment would exceed the space available here. The subsequent presentation will thus concentrate on the effects associated with integrating a behavior-based prediction step and will only sketch treatment of other aspects.

2 Relevant Publications

Conte et al. [2] report on tracking images of vehicles and people even through partial occlusions. A hierarchical graph representation of connected foreground pixels allows to establish a kind of 'variable-resolution appearance-model' which facilitates frame-to-frame tracking of region characteristics even if foreground-regions merge or split. This capability is exploited to handle partial occlusions (i.e. to track part of the pre-occlusion region characteristics). We refer to [2] for a well-written survey of related *data-driven* tracking approaches which sort-of updates the discussion in [10]. The systems approach reported in [11, Section 3.1] relies on refined background subtraction in a motion-detection step. Resulting foreground regions, i.e. 2D-image-plane representations of moving bodies, are tracked by a feature-based approach in the image plane. Tracking results obtained thereby independently from each video stream provided by a multi-camera setup are subsequently associated with 3D-representations for different body-categories. These authors explicitly acknowledge the difficulty to track moving bodies through occlusions [11, Section 6, first paragraph].

In distinction to the system reported in [11], we use a *3D-model-based* tracking approach for a *monocular* video. Due to space limitations, we restrict the subsequent treatment of publications to questions which are directly associated with our subject, for example a survey of object motion and behaviors which comprises, however, surprisingly few references to 3D-model-based vehicle tracking approaches [6]. Our own search for relevant archival publications confirms this finding and extends its validity even up to the immediate past.

A recent development exploits a probabilistic representation of the (polygonal) perspective image plane projection of a 3D-polyhedral road vehicle model in order to update the vehicle state vector based on an Expectation Maximization (EM) approach – see [8,9] and the literature quoted therein. A comparison with an edge-based tracking alternative does not result in significant performance dif-

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

ferences for the set of real-world road traffic videos taken into account, provided both approaches are evaluated under equivalent conditions [3].

Whereas the publications mentioned in the preceding paragraph are not concerned with occlusion handling, the approach described in [1] (and preceding related publications quoted therein) directly addresses the problem to switch between motion-based and behavior-based prediction, albeit *relying on explicit 3D-models* of stationary scene components which occlude a vehicle to be tracked. This greatly simplifies the task to determine the onset of (dis-)occlusion, using standard hidden-line removal algorithms. In general, appropriate 3D-models for possibly occluding stationary scene components will not be available. It thus is of interest to search for alternative assumptions allowing to bypass the a-priori provision of such explicit 3D-models for occluding stationary scene components.

3 Structure of the Underlying 3D-model-based System

In order to cope with tracking road vehicles through major occlusions, it has been suggested recently in [4] to explore a combination of solution approaches to the following subproblems: (i) detection – without reliance on explicit 3D-models of opaque bodies in the scene – that the vehicle currently being tracked becomes (partially) occluded; (ii) switching to a prediction mode based on expected vehicle *behavior* as opposed to a mere 'blind continuation' based on the last pre-occlusion state vector estimate; (iii) re-detection of this vehicle when it re-emerges from behind the occluding body, followed by its state vector update and switching back to the standard 'motion-based' prediction mode which proceeds from the current updated state vector estimate. These ideas provided a starting point for substantial adaptations suggested by experience gathered during experiments with a larger ($\sim$ 15000 frames) road traffic video sequence.

3.1 3D-model-based vehicle tracking

The vehicle state in the 3D-scene will be represented by a '*basic*' state vector $\boldsymbol{x}_{basic} = (p_x, p_y, \theta, v, \psi)^T$ where $(p_x, p_y)^T$ denotes the position of the vehicle reference point in the groundplane of the 3D-scene coordinate system (usually taken to be the road plane), θ denotes the vehicle orientation with respect to a reference direction in the 3D-scene groundplane, v the 3D-speed of the vehicle to be tracked, and ψ the steering angle of its front wheels ($\psi = 0$ for driving straight ahead). Prediction and update of these state vector components is handled in analogy to already published 3D-model-based vehicle tracking approaches and – due to space limitations – will not be explained here in detail again.

The 'basic' state vector will now be extended stepwise in order to obtain estimates for color and motion characteristics at each frame-time t.

3.2 State-vector extension by a *facet-specific* color representation

Experience has shown that vehicle color is difficult to estimate for small vehicle images like those illustrated in Figures 2 and 3. The 3D-polyhedral vehicle model

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

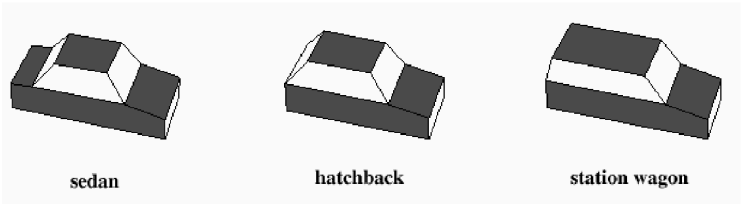

Fig. 1. Generic 3D-polyhedral vehicle models without wheelarches. A pixel will be taken into account for vehicle color determination only if this pixel is covered by one of the facets marked in dark gray.

used for model-based tracking offers the possibility to restrict color estimation to those facets which are expected to give the most reliable color cues, namely the hood, the roof, the trunk (in case of a sedan), and the lower parts of both sides – see Figure 1. We thus exclude the front and rear side from color estimation because these facets carry diverse lights and other parts like the cooling air intake which usually do not exhibit the vehicle color. In addition, all window facets are excluded because they reflect light and thus show the color of the vehicle surroundings rather than that of the vehicle itself.

Let Ω_i denote facet i whose color has to be estimated. Let $\boldsymbol{\xi}$ denote a pixel position in the image plane which is covered by the projection of Ω_i into the image plane according to the current state estimate $\hat{\boldsymbol{x}}_{basic}$. Let $\mathbf{q_{color}}(\boldsymbol{\xi}) = (q_R(\boldsymbol{\xi}), q_G(\boldsymbol{\xi}), q_B(\boldsymbol{\xi}))^T$ denote the RGB-color values recorded at pixel position $\boldsymbol{\xi}$. The current color $\boldsymbol{\mu}_{\Omega_i,current}$ for facet Ω_i is determined according to

$$\boldsymbol{\mu}_{\Omega_i,current} = \frac{1}{m} \sum_{\boldsymbol{\xi} \in \Omega_i} \mathbf{q_{color}}(\boldsymbol{\xi}) \tag{1}$$

where m denotes the number of pixels covered by the projection of Ω_i into the image plane. Ray-tracing from the camera center of projection through the image plane into the scene is used to determine whether or not pixel location $\boldsymbol{\xi}$ has to be taken into account for estimating the color of facet Ω_i. We now extend the 'basic' state vector $\boldsymbol{x}_{basic}$ by incorporating a color vector $\boldsymbol{\mu}_{\Omega_i}$ for each facet into an extended state vector $\boldsymbol{x} = (p_x, p_y, \theta, v, \psi, \boldsymbol{\mu}_{\Omega_1}^T, ..., \boldsymbol{\mu}_{\Omega_n}^T)^T$ with $(5 + 3n)$ components where n denotes the number of facets taken into account for a particular vehicle type, for example 4 in case of a hatchback.

The facet color $\boldsymbol{\mu}_{\Omega_i}$ is assumed to remain constant during the period between frame-times t and $t+1$ for two consecutively recorded video frames. The Kalman-Filter estimate $\boldsymbol{\mu}^+_{k-1,\Omega_i}$ obtained for video-frame $k-1$ will thus constitute the expected color observation $\mathbf{h}_{k,\Omega_i}(\boldsymbol{x})$ for this facet at the next frame k:

$$\mathbf{h}_{k,\Omega_i}(\boldsymbol{x}) \equiv \boldsymbol{\mu}^+_{k-1,\Omega_i}. \tag{2}$$

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

3.3 Facet-specific Optical-Flow-Coverage (OFC)

It is well-known that Optical Flow (OF) can be estimated *locally* as the image-plane projection of the eigenvector related to the smallest eigenvalue of the Grayvalue Structure Tensor (GST), provided its third component is normalized to 1. The ratio of the number of prediction-compatible OF-vectors within a model-projection into the image plane and the total number of pixels covered by such a projection is denoted as the '*OF-Coverage (OFC)*' of a model projection according to the current state estimate. Detailed investigations have shown that the OFC can provide a cue for detection of vehicles (re-)appearing in the field of view of a recording video camera [7].

In analogy to the introduction of facet-specific color-estimates, we further extend the state vector $\boldsymbol{x}$ by a facet-specific OFC q_{OFC,Ω_i} for each facet Ω_i ($i = 1, ..., n$), i.e. by a vector $\boldsymbol{q}_{OFC} = (q_{OFC,\Omega_1},\ \dots\ ,\ q_{OFC,\Omega_n})^T$. Assuming that these q_{OFC,Ω_i} remain essentially constant from a video frame to the next one, we may consider the updated Kalman-estimate $\boldsymbol{q}^+_{OFC,k-1}$ obtained at the preceding frame $k-1$ as the *expected* observation $\boldsymbol{h}^{(OFC)}_k(\boldsymbol{x})$ for the current frame k. We obtain the measurement equation for the OFC-part as

$$\boldsymbol{z}^{(OFC)}_k \ = \ \boldsymbol{h}^{(OFC)}_k(\boldsymbol{x}) \equiv \boldsymbol{q}^+_{OFC,k-1}. \tag{3}$$

3.4 Kalman-Filter (KF) handling of facet-specific components

Let the scalar q denote any color component of the state vector, let $\dot{q} = \partial q/\partial t$ and let $\boldsymbol{x}_q = (q,\ \dot{q})^T$ with an associated system model

$$\begin{pmatrix} q^-_k \\ \dot{q}^-_k \end{pmatrix} = \begin{pmatrix} 1 & \tau \\ 0 & 1 \end{pmatrix} \begin{pmatrix} q^+_{k-1} \\ \dot{q}^+_{k-1} \end{pmatrix} \ + \ a_{k-1} \cdot \begin{pmatrix} \frac{\tau^2}{2} \\ \tau \end{pmatrix}, \tag{4}$$

where τ $(= 0,02sec)$ is the time between two frames and a_{k-1} is a normally distributed time-varying acceleration with mean 0 and variance σ_a^2. This implies the following process noise covariance:

$$\boldsymbol{Q} = E\left\{ a \begin{pmatrix} \frac{\tau^2}{2} \\ \tau \end{pmatrix} \cdot a \begin{pmatrix} \frac{\tau^2}{2} & \tau \end{pmatrix} \right\} = E\left\{a^2\right\} \begin{pmatrix} \frac{\tau^4}{4} & \frac{\tau^3}{2} \\ \frac{\tau^3}{2} & \tau^2 \end{pmatrix} = \sigma_a^2 \begin{pmatrix} \frac{\tau^4}{4} & \frac{\tau^3}{2} \\ \frac{\tau^3}{2} & \tau^2 \end{pmatrix}. \tag{5}$$

The following measurement model has been used

$$z_{q,k} \ = \ \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} q^-_k \\ \dot{q}^-_k \end{pmatrix} \ + \ w_k, \tag{6}$$

where w_k is a normally distributed random variable with mean 0 and variance σ_w^2. This approach implies that the temporal derivative of q does *not* need to be determined *explicitly from image data*.

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

The dimension of the complemented state vector thus increases from 5 to $5 + (\#\text{ of 'admitted facets'}) \cdot (3colors * 2components + 1) \leq 40$. Because a proper update of the components of $\boldsymbol{x}_{basic}$ requires a non-linear optimization method (Gauss-Newton or Levenberg-Marquard), computation time and storage requirements will threaten to turn such an approach practically unattractive. Closer inspection shows, however, that the added state vector components are needed exclusively for occlusion detection and can be updated using a linear KF (see, e.g., [5]): system and measurement model for prediction and update of occlusion components are linear functions, and the position of an actor (i.e. $\boldsymbol{x}_{basic}$) can be estimated *independently* of facet-specific components for color and OF-coverage. An update of these components prior to the very last iteration of the non-linear KF for $\boldsymbol{x}_{basic}$ is unnecessary, one always reaches the minimum in a single iteration given a new position estimate for an actor. Because a non-linear optimization is performed on a much smaller state vector, computation time is significantly reduced.

Furthermore, occlusion components for one facet are independent of occlusion components for a different facet. The multi-component state vector can thus be split into state vectors which are specific for each facet, thereby turning the state covariance matrix blockdiagonal, with one block corresponding to $\boldsymbol{x}_{basic}$ and an additional block for each facet. Such 'splitting' reduces memory requirements, without any loss of the quality of results.

3.5 Fuzzy combination of OFC and Color Change Cues

The $OFC \in [0, 1]$ can be treated immediately as a fuzzy predicate. Determination of an OF-vector requires that a sizable environment around the pixel location in question has to be taken into account. OFC thus begins to drop already somewhat prior to the onset of geometrical occlusion. OFC recovers to the level prevailing prior to the onset of occlusion, moreover, only after the occlusion proper has terminated. It thus appears reasonable to demand that the OFC-estimate will already be small when an occlusion proper starts.

On the other hand, color changes are expected to be largest when occlusion starts, i.e. that the temporal derivative of at least one color component has to become large at the onset of occlusion. The *temporal derivatives* of all *three color components* have been combined into a fuzzy predicate 'Color Change Cue (CCC)' which quickly drops for a sizeable temporal derivative of at least one color component:

$$\mathrm{CCC}_{\Omega_i} = e^{-\frac{\dot{\boldsymbol{\mu}}_{\Omega_i}^T \dot{\boldsymbol{\mu}}_{\Omega_i}}{\sigma^2_{CCC_{\Omega_i}}}}, \tag{7}$$

where $\sigma^2_{CCC_{\Omega_i}}$ $(= 150^2)$ has been determined interactively.

It is postulated that a facet begins to become occluded when its OFC is small and its CCC simultaneously drops below a threshold. The fuzzy conjunction of these two conditions is implemented by testing whether the larger of the two has dropped below a given threshold whose exact value (e.g., 0.3) does not appear

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

to be critical. A vehicle is said to have become occluded if *more than half of its visible facets* have become occluded.

4 Switching Between Motion-based and Behavior-based Prediction

Let the term '*agent*' refer to a vehicle whose image has been detected and (partially) tracked in a road traffic video. The *behavior* of an agent is represented as a *sequence of situated actions*. The set of such sequences which has to be taken into account for a particular scenario can be specified as the set of all paths through a directed hypergraph. A node of such a graph combines conditions – specified in form of a (frame-)time-dependent logic formula – with the action(s) which the agent is expected to perform, provided

1. this node is currently visited by a hypergraph traversal algorithm and
2. its conditions can be satisfied by data which have been extracted from the current video frame and have been converted to conceptual representations.

Such a hypergraph can be generated interactively, using a special tool developped for this purpose.

In principle, one or more of all feasible vehicle maneuvers could occur in the action-part of a hypergraph-node. So far, the maneuvers '`start_up`', '`accelerate`', '`drive_at_constant_speed`', '`follow_the_current_lane`', '`slow_down`', '`stop`', '`retain_a_safe_distance_from_preceding_vehicle`' have been included. Some of these maneuvers require a lane model for the road scene recorded by the video stream to be evaluated, for example the maneuver '`follow_the_current_lane`'.

The crucial extension of our system approach in comparison with a 'normal' KF consists in the capability to substitute a suitably parameterized maneuver for the 'motion-model' part of the KF, i.e. for the '*behavior-based*' prediction phase. In case of a visible vehicle, the maneuver '`drive_at_constant_speed`' is used with parameters – speed, steering angle – to be estimated by the standard KF-process (see Section 3.1). In the experiments to be discussed below, this maneuver is replaced by '`follow_the_current_lane`' once the onset of occlusion has been detected according to Section 3.5. In combination with the a-priori known lane model, this maneuver facilitates to track a vehicle even if the lane is curved.

5 Experiments

Earlier 3D-model-based tracking experiments, for example in [3], relied on an interactive initialisation phase in order to disentangle the effects of tracking failures proper from those due to suboptimal initialisations. Our experiments, however, use a new *fully automatic* initialisation based on the detection of coherently moving (near constant-size) OF-segments comprising an edge-element structure which is compatible with the projected 3D-polyhedral vehicle representation [7].

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

Fig. 2. Tracking of a vehicle which becomes occluded by a traffic sign (left panel: without occlusion detection; center panel: with occlusion detection and *motion*-based prediction; right panel: with occlusion detection and *behavior*-based prediction). See Section 5 for further discussion.

Space limitations do not allow to discuss this automatic initialisation here in detail. It has been used, too, in order to detect and *re*-initialise the vehicle once it (appears to) emerge from behind an occluding foreground body.

The first example demonstrates the advantage of occlusion detection and – to some extent – of behavior-based prediction. The left panel shows a vehicle which becomes occluded by a traffic sign. The undetected onset of occlusion confused the tracker and the tracking failed due to incorrect measurements caused by the occluding traffic sign. The center panel shows the result obtained *with* (successful) occlusion detection and *motion*-based prediction. The vehicle could be tracked in spite of some problems, for example the large 'jump' to the lane center where the vehicle was re-initialized. The right panel shows a screenshot obtained at the time of occlusion detection with *behavior*-based prediction. The occluded facets are marked by a black-white texture. Because the position of the vehicle, which had been corrupted by late occlusion detection, was corrected by the behavior-based prediction already during the occlusion, the vehicle could be tracked without any difficulties.

Whereas the actor used in the previous example follows a straight lane where motion-prediction still sufficed (in spite of some difficulties), the following example of a vehicle which becomes occluded by a tree demonstrates in particular the advantage of *behavior*-based prediction. The left screenshot of Figure 3 shows tracking of a vehicle without occlusion detection. This tracking attempt failed due to incorrect measurements caused by the occluding tree. The center panel displays the tracking result *with* occlusion detection and *motion*-based prediction. The occlusion by the tree was detected successfully, and the tracking continued for a while. Because the lane is curved, however, the agent model eventually left the lane and tracking failed as well. The right panel illustrates the result obtained *with* occlusion detection and *behavior*-based prediction. Because the behavior-based prediction follows the (curved) lane, the agent was tracked successfully until it left the field of view.

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

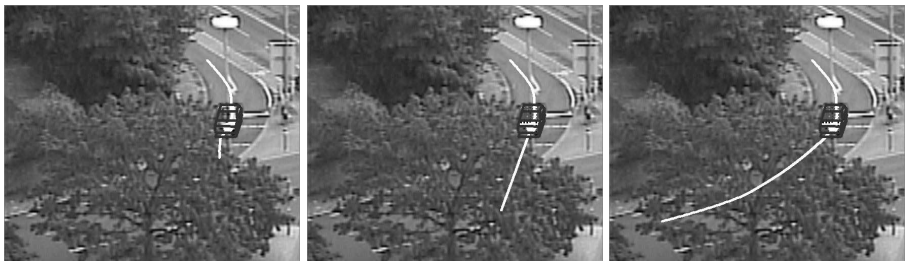

Fig. 3. Tracking of a vehicle which becomes occluded by a tree (left panel: without occlusion detection; center panel: with occlusion detection and *motion*-based prediction; right panel: with occlusion detection and *behavior*-based prediction). See Section 5 for further discussion.

6 Conclusions

The research reported here starts from the observation that tracking an occluded vehicle requires at least one hypothesis about which maneuver(s) such a vehicle will perform while it can not be observed. The simplest such assumption – namely that the vehicle will continue at constant speed straight ahead in the same direction observed immediately prior to the onset of an occlusion – will in general not suffice. The next simplest hypothesis postulates that the vehicle will continue at constant speed within the same lane as before. This hypothesis requires knowledge about the lane structure of the roads within the field of view. Although such knowledge can in principle be derived from digital maps which are widely available already for navigation support in driver assistance systems, it again will turn out to be insufficient, e.g., if the vehicle has to follow a preceding one: in this case, the driver of the trailing vehicle will have to switch the longitudinal control regime from velocity control to distance control in order to avoid bumping into the leading vehicle. The reader may notice that these control regimes are not characterized by quantitative notions, but by conceptual ones.

The approach pursued here thus has been *based on modeling vehicle behavior* at a qualitative, conceptual level of representation rather than on a quantitative (geometrical trajectory) one. This generalisation facilitates tracking vehicles even through longer occlusions provided the onset of vehicle occlusion by *unknown* opaque bodies in the scene can be detected. Likewise, the vehicle has to be redetected as soon as possible when it becomes disoccluded again. For this task, the same process is used as for the automatic initialisation of tracking once a vehicle enters the field of view of the recording camera. The onset of occlusion is assumed to be detectable by a fuzzy combination of depressed OF-coverage and a sharp Color-Change. Given the small vehicle images in the videos used for testing, the color estimate for many vehicles is corrupted by wheelarches, windows, etc. such that color changes tend to become ambiguous occlusion cues. Exploiting the known facet structure of the polyhedral vehicle model, however,

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

allows to extract a much cleaner Color-Change-Cue by analysing only the color of certain parts of the visible vehicle body. The seemless integration of these component heuristics into a 3D-model-based vehicle tracking system can cope with some otherwise unmanageable tracking challenges. The advantage of this model-based approach is seen in the incentive to diagnose system failures as being due to insufficient modeling rather than to insufficient parameter tuning.

Acknowledgement

This research has been supported in part by the European Union Project HERMES (6thFP-IST-027110).

References

1. M. Arens and H.-H. Nagel. Quantitative Movement Prediction Based on Qualitative Knowledge about Behavior. *KI Künstliche Intelligenz*, 2/05:5–11, May 2005.
2. D. Conte, P. Foggia, J.-M. Jolion, and M. Vento. A Graph-based Multi-resolution Algorithm for Tracking Objects in Presence of Occlusions. *Pattern Recognition*, 39(4):562–572, April 2006.
3. H. Dahlkamp, A.E.C. Pece, A. Ottlik, and H.-H. Nagel. Differential Analysis of Two Model-Based Vehicle Tracking Approaches. In C. E. Rasmussen, H. H. Bülthoff, M. A. Giese, and B. Schölkopf, editors, *Pattern Recognition, Proc. 26th DAGM-Symposium (DAGM'04)*, volume 3175 of *Lecture Notes in Computer Science*, pages 71–78, Tübingen, Germany, 30 August–1 September 2004. Springer Berlin · Heidelberg · New York, NY 2004.
4. A. Fexa, H.-H. Nagel, A. Ottlik, and J. Siecke. Behavior-based 3D-Tracking During Unmodelled Occlusion. In M. Vincze and L. Paletta, editors, *2nd International Cognitive Vision Workshop (in conjunction with ECCV-2006)*, pages 1–8 (on the Workshop–CD), Graz, Austria, 13 May 2006.
5. M.S. Grewal and A.P. Andrews. *Kalman Filtering – Theory and Practice*. Prentice-Hall, Inc., Englewood Cliffs/NJ, 1993.
6. W. Hu, T. Tan, L. Wang, and S. Maybank. A Survey on Visual Surveillance of Object Motion and Behaviors. *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews*, 34(3):334–352, August 2004.
7. A. Ottlik. *Zur modellgestützten Initialisierung von Fahrzeugverfolgungen in Videoaufzeichnungen*, volume 291 of *Dissertationen zur Künstlichen Intelligenz (DISKI)*. Akademische Verlagsgesellschaft Aka GmbH, Berlin, Germany, 2005. Dissertation, Februar 2005, Fakultät für Informatik der Universität Karlsruhe (TH).
8. A.E.C. Pece. Contour Tracking Based on Marginalized Likelihood Ratios. *Image and Vision Computing*, 24(3):301–317, March 2006.
9. A.E.C. Pece and A.D. Worrall. A Comparison Between Feature-based and EM-based Contour Tracking. *Image and Vision Computing*, 24(11):1218–1232, November 2006.
10. A. Senior, A. Hampapur, Y.-L. Tian, L. Brown, S. Pankanti, and R. Bolle. Appearance Models for Occlusion Handling. *Image and Vision Computing*, 24(11):1233–1243, November 2006.
11. D. Thirde, M. Borg, J. Ferryman, F. Fusier, V. Valentin, F. Brémond, and M. Thonnat. A Real-Time Scene Understanding System for Airport Apron Monitoring. In *Proc. Fourth IEEE International Conference on Computer Vision Systems*, New York, NY, 5–7 January 2006. IEEE Computer Society. ISBN 0-7695-2506-7.

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de