Geometrically Constrained Level Set Tracking for Automotive Applications

Esther Horbert, Dennis Mitzel, Bastian Leibe

UMIC Research Centre RWTH Aachen University, Germany

Abstract. We propose a new approach for integrating geometric scene knowledge into a level-set tracking framework. Our approach is based on a novel constrained-homography transformation model that restricts the deformation space to physically plausible rigid motion on the ground plane. This model is especially suitable for tracking vehicles in automotive scenarios. Apart from reducing the number of parameters in the estimation, the 3D transformation model allows us to obtain additional information about the tracked objects and to recover their detailed 3D motion and orientation at every time step. We demonstrate how this information can be used to improve a Kalman filter estimate of the tracked vehicle dynamics in a higher-level tracker, leading to more accurate object trajectories. We show the feasibility of this approach for an application of tracking cars in an inner-city scenario.

1 Introduction

Object tracking from a mobile platform is an important problem with many potential applications. Consequently, many different approaches have been applied to this problem in the past, including tracking-by-detection [1–4], model-based [5, 6], template-based [7, 8] and region-based [9, 10] methods. In this paper, we focus on the latter class of approaches, in particular on level-set tracking, which has shown considerable advances in recent years [9, 11].

Level-set tracking performs a local optimization, iterating between a segmentation and a warping step to track an object's contour over time. Since both steps only need to be evaluated in a narrow band around the currently tracked contour, they can be implemented very efficiently [9]. Still, as all appearance-based approaches, they are restricted in the types of transformations they can robustly handle without additional knowledge about the expected motions.

In this paper, we investigate the use of geometric constraints for improving level-set tracking. We show how geometric scene knowledge can be directly integrated into the level-set warping step in order to constrain object motion. For this, we propose a constrained-homography transformation model that represents rigid motion on the ground plane. This model is targeted for tracking vehicles in an automotive scenario and takes advantage of an egomotion estimate obtained by structure-from-motion (SfM).

An advantage of our proposed approach, compared to pure 2D tracking, is that it restricts the deformation space to physically plausible rigid-body motions, thus increasing the robustness of the estimation step. In addition, the 3D

transformation model allows us to directly infer the tracked object's detailed 3D motion and orientation at every time step. We show how this information can be used in a higher-level tracker, which models the vehicle dynamics in order to obtain smooth and physically correct trajectories. The additional measurements provided by our geometrically constrained level set tracker make the estimation more robust and lead to smoother trajectories. We demonstrate our approach on several video sequences for tracking cars on city roads under viewpoint changes.

The paper is structured as follows. The next section gives an overview of related work. Section 2 then presents the details of our proposed level-set tracking approach and Section 3 shows how its results can be integrated with a high-level tracker. Section 4 finally presents experimental results.

Related Work. Tracking-by-detection approaches have become very popular recently, since they can deal with complex scenes and provide automatic re-initialization by continuous application of an object detector [1–4]. However, for elongated objects with non-holonomic motion constraints, the raw detection bounding boxes often do not constrain the object motion sufficiently, making robust trajectory estimation difficult. Model-based tracking approaches try to obtain more information about the tracked objects by estimating their precise pose [5, 6]. However, they require a 3D model of the target object, which makes it hard to apply them for complex outdoor settings where many different objects can occur. The complexity can be reduced by limiting pose estimation to a planar region, for which efficient template-based tracking schemes can be used [7]. By decomposing the homography estimated from the template deformation, information about the 3D object motion can be obtained [8]. However, this approach heavily relies on sufficient texture content inside the tracked region, which restricts it mainly to tracking fiducial regions.

In the context of region-based tracking, little work has been done in order to incorporate dedicated 3D scene constraints. [12] explore affine motion models in order to track multiple regions under 3D transformations. [13] and [14] propose different ways of combining level-set tracking with direct 3D pose estimation. However, they both assume a detailed 3D model of the target object to be available, which is not the case in our application. [10] propose a globally optimal approach for contour tracking which is also applied to an automotive scenario, but this approach does not use knowledge about the geometric meaning of the changed contour.

2 Approach

2.1 Level-Set Tracking

We use a probabilistic level-set framework for segmentation and tracking similar to the one introduced by [11]. The target object is first segmented, then tracked through the subsequent image frames. In the following, *background* denotes the area around and *foreground* the area containing the object. The object's contour is represented implicitly by the zero level-set of the embedding function $\mathbf{\Phi}(\mathbf{x})$

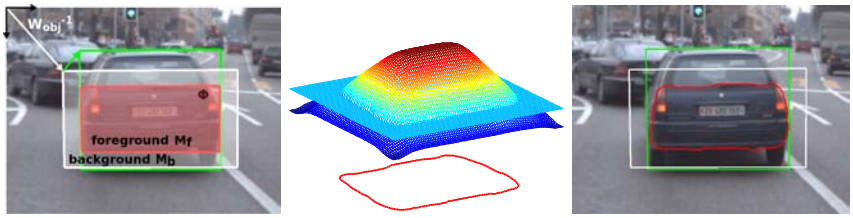


Fig. 1. Detection box (green), foreground initialization (red) and object frame (white) with foreground and background pixels and the corresponding evolved level set embedding function $\mathbf{\Phi}$.

(see Fig. 1). The color $\mathbf{y}$ of pixels $\mathbf{x}$ is used to build foreground and background models M_f and M_b, in our case color histograms.

Segmentation. To obtain a segmentation of an object, the level set is evolved starting from an approximate initialization, *e.g.* a bounding box provided by an object detector. We use a variational formulation with three terms which penalize the deviation from M_f and M_b [11], the deviation from a signed distance function [15] (a constraint on the shape of the embedding function), and the length of the contour (to reward a smoother contour, similar so [15]). Eq. 1 shows the gradient flow used to optimize the segmentation:

$$\frac{\partial P(\mathbf{\Phi},\mathbf{p}|\mathbf{\Omega})}{\partial \mathbf{\Phi}} = \underbrace{\frac{\delta_\epsilon(\mathbf{\Phi})(P_f - P_b)}{P(\mathbf{x}|\mathbf{\Phi},\mathbf{p},\mathbf{y})}}_{\substack{\text{deviation from}\\ \text{fg/bg model}}} - \underbrace{\frac{1}{\sigma^2}\left[\nabla^2\mathbf{\Phi} - div\left(\frac{\nabla\mathbf{\Phi}}{|\nabla\mathbf{\Phi}|}\right)\right]}_{\substack{\text{deviation from}\\ \text{signed distance function}}} + \underbrace{\lambda\delta_\epsilon(\mathbf{\Phi})div\left(\frac{\nabla\mathbf{\Phi}}{|\nabla\mathbf{\Phi}|}\right)}_{\text{length of contour}} \quad (1)$$

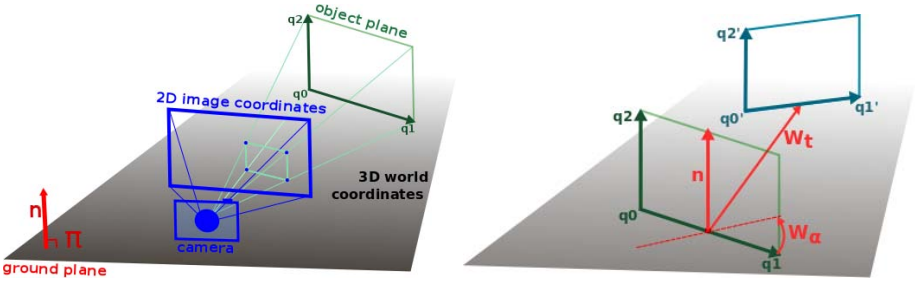
where $P(\mathbf{x}_i|\mathbf{\Phi},\mathbf{p},\mathbf{y}_i) = H_\epsilon(\mathbf{\Phi}(\mathbf{x}_i))P_f + (1 - H_\epsilon(\mathbf{\Phi}(\mathbf{x}_i)))P_b$, ∇^2 is the Laplacian operator, H_ϵ is a smoothed Heaviside step function, δ_ϵ a smoothed Dirac delta function and $\mathbf{\Omega}$ denotes the pixels in the object frame.

P_f and P_b are the pixel-wise posteriors for pixels' probabilities of belonging to the foreground and background. During segmentation, M_f and M_b are rebuilt in every iteration; during the later tracking stage the models are only slightly adapted to achieve high robustness while still adapting to lighting changes.

Tracking. In the following frames the obtained contour is tracked by performing a rigid registration, *i.e.* by warping its reference frame to another position without changing the contour's shape. Similar to inverse compositional image alignment [16] the content of the new frame is warped such that it looks more like the old frame. The inverse of the resulting warp with parameters $\Delta\mathbf{p}$ can in turn be used to warp the contour onto the new frame.

$$\Delta\mathbf{p} = \left[\sum_{i=1}^{N} \frac{1}{2P(\mathbf{x}_i|\mathbf{\Phi},\mathbf{p},\mathbf{y}_i)}\left[\frac{P_f}{H_\epsilon(\mathbf{\Phi}(\mathbf{x}_i))} - \frac{P_b}{(1-H_\epsilon(\mathbf{\Phi}(\mathbf{x}_i)))}\right]\mathbf{J}^\mathsf{T}\mathbf{J}\right]^{-1} \times \sum_{i=1}^{N} \frac{(P_f - P_b)\mathbf{J}^\mathsf{T}}{P(\mathbf{x}_i|\mathbf{\Phi},\mathbf{p},\mathbf{y}_i)} \quad (2)$$

with $\mathbf{J} = \delta_\epsilon(\mathbf{\Phi}(\mathbf{x}_i))\nabla\mathbf{\Phi}(\mathbf{x}_i)\frac{\partial\mathbf{W}}{\partial\Delta\mathbf{p}}$, where $\frac{\partial\mathbf{W}}{\partial\Delta\mathbf{p}}$ is the Jacobian of the warp.



(a) Coordinate Systems (b) Transformation Model

Fig. 2. Visualization of the coordinate systems and the proposed transformation model used in our approach.

2.2 Geometric Transformation Model

In addition to the image based tracking approach as described above, we model the 3D position of a tracked object. This allows us to make assumptions about an object's movement by the projective distortions that arise on the 2D image.

Coordinate Systems. Figure 2(a) shows the used coordinate systems. The image itself consists of a number of pixels with 2D coordinates. The colors of these pixels correspond to points in the world which were projected onto the image plane. The 3D coordinates of those points cannot however be inferred directly from one image without additional depth information. We use a ground plane, which was obtained with structure-from-motion (SfM), to estimate the base point of a detected object. We approximate the object to be a plane in world coordinates that is orthogonal to the ground plane. This object plane can be described with a point $\mathbf{q}_0$ and two direction vectors $\mathbf{q}_1$ and $\mathbf{q}_2$.

$$\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \\ w_i \end{bmatrix} = \mathbf{P}\mathbf{x}_w = \mathbf{P}\mathbf{Q}\mathbf{x}_o, \quad \text{with } \mathbf{Q} = \begin{bmatrix} \mathbf{q}_1 & \mathbf{q}_2 & \mathbf{q}_0 \\ 0 & 0 & 1 \end{bmatrix}, \ \mathbf{x}_o = \begin{bmatrix} x_o \\ y_o \\ 1 \end{bmatrix}, \ \mathbf{x}_w = \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \quad (3)$$

where $\mathbf{x}_o$ is a 2D point on the object plane and $\mathbf{x}_w$ are its corresponding world coordinates. The point $\mathbf{x}_i$ in the image that corresponds to this world point can be obtained by projection with the camera matrix $\mathbf{P}$.

3D Transformation Model. The level-set tracking framework requires us to specify a family of warping transformations $\mathbf{W}$ that relate the previous object reference frame to the current one. In the following, we show how this warp can be used to incorporate scene knowledge by enforcing geometric constraints on the object motion.

Our target scenario is an automotive application where the goal is to track other vehicles' motions relative to our own vehicle. In this scenario, we can assume that the tracked object parts are approximately planar and that the target objects move rigidly on the ground plane. This means that their 3D shape will not change between two frames; only their position relative to the camera

will. The resulting projective distortions in the image can therefore be modeled by a homography. However, an unconstrained homography has many degrees of freedom, which makes it hard to keep the tracking approach robust. Instead, we propose to use the available scene knowledge by modeling $\mathbf{W}$ as a *constrained homography* that requires fewer parameters and can be estimated more robustly.

Figure 2(b) illustrates our proposed transformation model. We represent object motion by a 3D homography, consisting of a rotation $\mathbf{W}_\alpha$ around an axis orthogonal to the ground plane and a translation $\mathbf{W}_t$ along the vector $\mathbf{t} = [t_x, t_y, t_z]^\mathsf{T}$. In order to compare the object points with the stored level-set contour of the previous frame, we then project the object into the image using an estimated camera matrix $\mathbf{P}$ obtained by SfM. Finally, we compute a 2D homography $\mathbf{W}_{obj}$ which warps the content of the object window (defined by the projections of its four corner points) onto the level-set reference frame.

$$\mathbf{W} = \mathbf{W}_{obj}\ \mathbf{P}\ \mathbf{W}_t \mathbf{W}_\alpha \mathbf{Q} \begin{bmatrix} x_o \\ y_o \\ 1 \end{bmatrix} \tag{4}$$

$\mathbf{W}_\alpha$ can be computed as a sequence of several transformations: a translation $\mathbf{T}_P$ moving the rotation axis into the origin; a rotation $\mathbf{R}_{xz}$ into the xz-plane; another rotation $\mathbf{R}_z$ onto the z-axis; and finally a rotation $\mathbf{R}_z(\alpha)$ about the z-axis with the desired angle α, followed by the inverse of the first three steps.

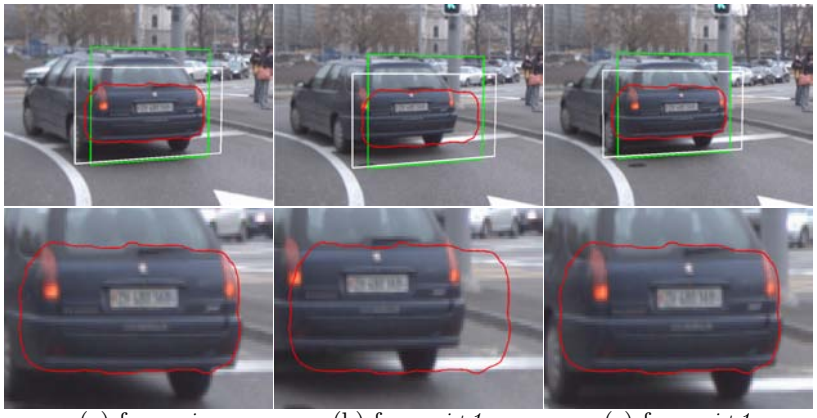
$$\mathbf{W}_\alpha = \mathbf{T}_P^{-1}\mathbf{R}_{xz}^{-1}\mathbf{R}_z^{-1}\mathbf{R}_z(\alpha)\mathbf{R}_z\mathbf{R}_{xz}\mathbf{T}_P, \qquad \mathbf{W}_t = \begin{bmatrix} \mathbf{I} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \tag{5}$$

In the above formulation, we have assumed a general translation $\mathbf{W}_t$. In principle, this could be restricted further to only allow translations parallel to the ground plane. However, the estimated ground plane is not always completely accurate and in any case does not account for uneven ground, especially at farther distances. We have therefore found that allowing a small movement component in the direction of the ground plane normal is necessary to achieve robustness.

Optimizing for the Transformation. The tracking framework uses the Gauss-Newton method to optimize the warp between two image frames. This requires the Jacobian of the overall warp $\mathbf{W}$, which contains the partial derivatives of $\mathbf{W}$ with respect to the parameters α, t_x, t_y and t_z.

$$\frac{\partial \mathbf{W}}{\partial \Delta \mathbf{p}} \quad \text{with} \quad \Delta\mathbf{p} = \begin{bmatrix} \alpha\ t_x\ t_y\ t_z \end{bmatrix}^\mathsf{T} \tag{6}$$

The parameters $\Delta\mathbf{p}$ available for optimization restrict the possible movements of the contour and the gradient $\frac{\partial \mathbf{W}}{\partial \Delta \mathbf{p}}$ indicates the effect a certain parameter value has on the position of the contour in the image. (6) is substituted into (2) and is evaluated for every point $\mathbf{x}$ in the band around the contour which is determined by $\delta_\epsilon(\mathbf{\Phi})$. In this way, pixel locations with a low probability of belonging to the foreground contribute a warp towards the outside of the contour and vice versa. A lower probability results in a larger step and the total step size is thus determined automatically. The algorithm has converged when the step size has become sufficiently small.



(a) frame i (b) frame $i+1$ (c) frame $i+1$

Fig. 3. (a) 3D position and synthetic view of object frame. (b) Before warp: 3D position as in frame i. (c) After warp: tracked 3D position. Notice how the contour moved in 3D and its projection onto the image plane changed accordingly; its shape did not change.

Final Tracking Algorithm. Putting the above steps together, we can summarize the proposed tracking algorithm as follows:

1. Initialize the object position, *e.g.* using a detection bounding box.
2. Apply the level-set segmentation (in our implementation for 200 iterations).
3. Compute the object plane's world coordinates $\mathbf{Q}$ by projecting the detection box base points onto the ground plane. (This box does not need to be aligned with the image borders and can also be used to initialize rotated objects).
4. For the following frames: Track the object's shape, *i.e.* compute $\Delta\mathbf{p}$ for the warp between two images i and $i+1$:
   (a) Assume the object is still located at the same 3D position. Interpolate synthetic views for both object frames (Fig. 3(a), 3(b)) with $\mathbf{W}^{i}_{obj}$,$\mathbf{W}^{i+1}_{obj}$.
   (b) Use eq. (2) with eq. (6) to find a set of parameters $\Delta\mathbf{p}$ such that the contour in the warped object frame $i+1$ better matches object frame i.
   (c) Use the inverse of the estimated homography $(\mathbf{W}_t\mathbf{W}_\alpha)^{-1}$ to warp the modeled 3D coordinates of the object and obtain a new 3D position estimate of the object in frame $i+1$ (Fig. 3(c)). (*E.g.* if image $i+1$ needs to be warped "closer" to the camera in order to look like image i, the object in fact moved to the back.)
   (d) Use the new 3D coordinates to obtain an improved synthetic view of the object in frame $i+1$.
   (e) Repeat steps (b) to (d) until the step size is small enough: $\|\Delta\mathbf{p}\| < \epsilon$.
   (f) Apply the level-set segmentation for 1 iteration to update the contour.

The results of this procedure are a level-set contour and a bounding box for each frame, as well as the estimated 3D position and orientation of the object.

3 Integration with a High-Level Tracker

The level-set tracking approach described in the previous section can robustly follow an individual object over time. However, it requires an initialization to pick

out objects of interest, and it does not incorporate a dynamic model to interpret the observed motion. For this reason, we integrate it with a high-level tracker. In this integration, the task of the level-set tracker is to generate independent *tracklets* for individual objects, which are then integrated into a consistent scene interpretation with physically plausible trajectories by the high-level tracker.

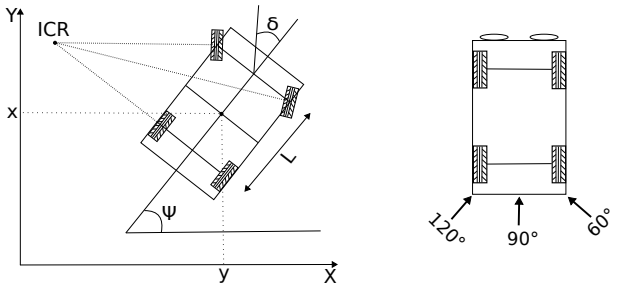
System Overview. We apply a simplified version of the robust multi-hypothesis tracking framework by [3]. Given an estimate of the current camera position and ground plane location from SfM, we collect detected vehicle positions on the ground plane over a temporal window. Those measurements are then connected to trajectory hypotheses using Extended Kalman Filters (EKFs). Each trajectory obtains a score, representing the likelihood of the assigned detections under the motion and appearance model (represented as an RGB color histogram). As a result, we obtain an overcomplete set of trajectory hypotheses, from which we select the best explanation for the observed measurements by applying model selection in every frame (Details can be found in [3]).

Motion Model. For modeling the vehicle trajectories, we use an EKF with the Ackermann steering model (*e.g.* [17]), as shown in Fig. 4(a). This model incorporates a non-holonomic motion constraint, enforcing that the velocity vector is always perpendicular to the rear wheel axis. The state vector is given as $\mathbf{s}_t = [x_t, y_t, \psi_t, v_t, \delta_t, a_t]$, where (x, y) is the position of the car, ψ the heading angle, δ the steering angle, v the longitudinal velocity, and a the acceleration. The prediction step of the Kalman filter is then defined as follows:

$$\mathbf{s}_{t+1} = \begin{pmatrix} x_t + v_t\cos(\psi_t)\Delta t + \frac{1}{2}a_t\cos(\psi_t)\Delta t^2 \\ y_t + v_t\sin(\psi_t)\Delta t + \frac{1}{2}a_t\sin(\psi_t)\Delta t^2 \\ \psi_t + \frac{v_t}{L}\tan(\delta)\Delta t \\ v_t + a_t\Delta t \\ \delta_t \\ a_t \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ n_\delta \\ n_a \end{pmatrix} . \tag{7}$$

L is the distance between the rear and front wheel axes and is set to a value of 3.5m. Multi-vehicle tracking-by-detection using a similar motion model was demonstrated by [4] based on a battery of object detectors with discretized viewing angles. We use a similar coarse discretization of the viewing angle with three separate, HOG-style [18] vehicle detectors in order to initialize our level-set tracker (Fig. 4(b)). However, after this initialization, we only use the observations provided by the low-level tracker and integrate them into the motion model.

Discussion. Our proposed approach has several advantages. Compared to a pure tracking-by-detection approach, the level-set tracker yields much finer-grained measurements of the viewing angle at which the target vehicle is seen. In addition, the level-set tracker can continue tracking objects even when they partially leave the image and the object detector would fail. Compared to a level-set tracker with a simpler 2D transformation model (*i.e.*, just using translation and scale, without ground plane constraints), our model has the advantage of being able to estimate the target vehicle's location *and* its current orientation.



(a) (b)

Fig. 4. (a) Ackermann steering model used for modeling the motion of the vehicle (assumes rolling without slippage). (b) Discretization of the the detected viewing angles.

This orientation estimate is beneficial in two respects. It allows us to extrapolate from the tracked car trunk location and infer the true object center, resulting in better position estimates (which is especially important for elongated objects such as cars). And it enables the use of orientation as *observed quantity* in the motion model, resulting in better predictions. All of those factors contribute to more robust tracking performance, as will be demonstrated in the next section.

4 Experimental Results

Data. We demonstrate our approach on three parts of a challenging sequence from the Zurich Mobile Car corpus, generously provided by the authors of [4]. The sequence was captured using a stereo setup (13-14 fps and 640×480 resolution) mounted on top of a car. We use SfM and ground plane estimates provided with this data set, but restrict all further processing to the left camera stream.

Qualitative Results. Fig. 5 shows qualitative results of our approach on three test sequences which contain cars turning corners, demonstrating its capability to accurately track vehicles under viewpoint changes. (The corresponding result videos are provided on `www.mmp.rwth-aachen.de/projects/dagm2010`). As can be seen, the estimated vehicle orientation from the level-set tracker enables the high-level tracker to compute smooth vehicle trajectories.

Comparison with Baseline Approach. Fig. 6 presents a comparison of our 3D estimation approach with the results of our level-set tracker using only a 2D (translation + scale) transformation model. As can be seen from those results, our approach achieves better tracking accuracy and manages to closely follow the target vehicles despite considerable viewpoint changes. In contrast, the 2D baseline method slips off the car in all cases, since the 3D position of the car's center is incorrectly estimated, resulting in a wrong trajectory.

5 Conclusion

In conclusion, we have presented an approach for incorporating geometric scene constraints into the warping step of a level-set tracker. Our approach allows to

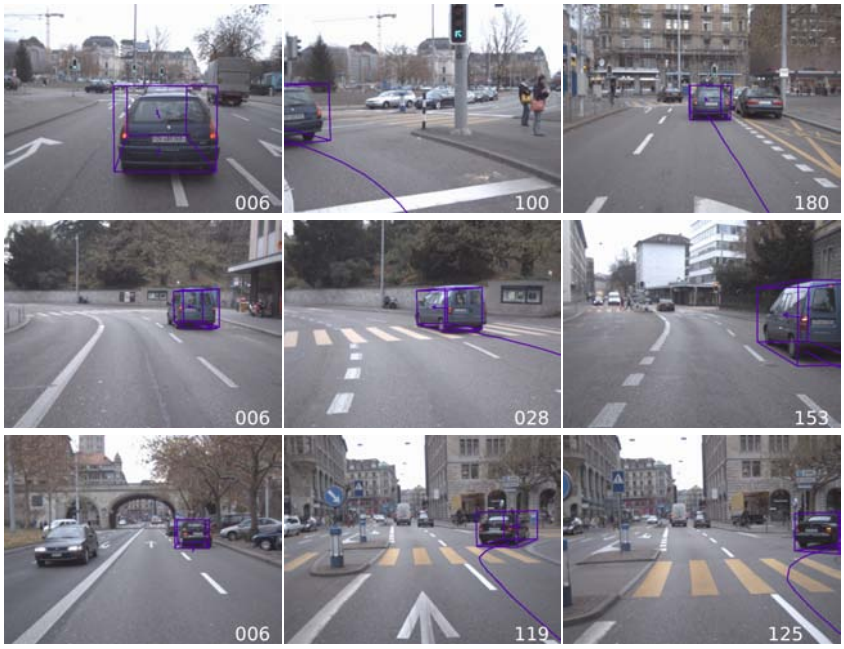


Fig. 5. Tracking results of our approach on three test sequences. The integrated 3D estimation results of the high-level tracker show that it is able to accurately follow cars turning corners and to produce smooth trajectories.

estimate both the location and orientation of the tracked object in 3D, while at the same time restricting the parameter space for more robust estimation. As we have shown, the estimation results can be used to improve the performance of a higher-level multi-hypothesis tracker integrating the measurements with vehicle dynamics into physically plausible trajectories. A possible extension could be to incorporate detections for different vehicle orientations, as well as stereo depth information, in order to initialize tracking also for other vehicle viewpoints.

Acknowledgments. This project has been funded, in parts, by the EU project EUROPA (ICT-2008-231888) and the cluster of excellence UMIC (DFG EXC 89). We thank C. Bibby and I. Reid for valuable comments for the level-set tracking and for making their evaluation data available.

References

1. Betke, M., Haritaoglu, E., Davis, L.S.: Real-time multiple vehicle detection and tracking from a moving vehicle. MVA **12** (2000) 69–83
2. Gavrila, D., Munder, S.: Multi-Cue Pedestrian Detection and Tracking from a Moving Vehicle. IJCV **73**(1) (2007) 41–59
3. Leibe, B., Schindler, K., Van Gool, L.: Coupled Object Detection and Tracking from Static Cameras and Moving Vehicles. PAMI **30**(10) (2008) 1683–1698
4. Ess, A., Leibe, B., Schindler, K., Van Gool, L.: Robust Multi-Person Tracking from a Mobile Platform. PAMI **31**(10) (2009) 1831–1846

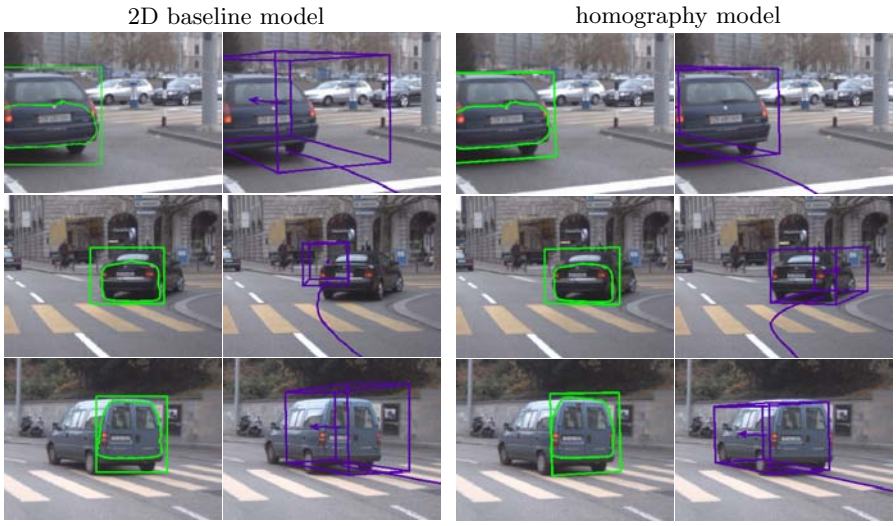


Fig. 6. Comparison with the results of a 2D baseline model. (Left columns) Level-set tracking results; (Right columns) High-level tracker's results. The lacking orientation estimate causes the high-level tracker to slip off the vehicles during viewpoint changes.

5. Koller, D., Daniilidis, K., Nagel, H.: Model-Based Object Tracking in Monocular Image Sequences of Road Traffic Scenes. IJCV **10**(3) (1993)
6. Dellaert, F., Thorpe, C.: Robust Car Tracking using Kalman filtering and Bayesian templates. In: CITS. (1997)
7. Chateau, T., Jurie, F., Dhome, M., Clady, X.: Real-Time Tracking Using Wavelet Representation. In: DAGM. (2002)
8. S.Benhimane, Malis, E., Rives, P.: Vision-based Control for Car Platooning using Homography Decomposition. In: ICRA. (2005) 2161–2166
9. Cremers, D., Rousson, M., Deriche, R.: A Review of Statistical Approaches to Level Set Segmentation Integrating Color, Texture, Motion and Shape. IJCV **72** (2007) 195–215
10. Schoenemann, T., Cremers, D.: A Combinatorial Solution for Model-based Image Segmentation and Real-time Tracking. PAMI (2009)
11. Bibby, C., Reid, I.: Robust Real-Time Visual Tracking using Pixel-Wise Posteriors. In: ECCV. (2008)
12. Fussenegger, M., Deriche, R., Pinz, A.: Multiregion Level Set Tracking with Transformation Invariant Shape Priors. In: ACCV. (2006)
13. Brox, T., Rosenhahn, B., Weickert, J.: Three-dimensional shape knowledge for joint image segmentation and pose estimation. In: DAGM. (2005)
14. Prisacariu, V., Reid, I.: PWP3D: Real-time segmentation and tracking of 3D objects. In: BMVC. (2009)
15. Li, C., Xu, C., Gui, C., Fox, M.: Level Set Evolution without Re-initialization: A New Variational Formulation. In: CVPR. (2005)
16. Baker, S., Matthews, I.: Lucas-Kanade 20 Years On: A Unifying Framework. IJCV **69** (2004) 221–255
17. Muir, P., Neuman, C.: Kinematic Modeling of Wheeled Mobile Robots. J. Robotic Systems **4**(2) (1987) 281–340
18. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR. (2005)