Abstract—Vision-based driver assistance systems have great potential for preventing fatalities. This work addresses the problem of 3D monocular vehicle tracking and turn rate estimation in situations where vehicles need to be tracked along intersections and curves. To estimate the tracked vehicle’s turn rate, an approach based on image feature correspondences and a simplified geometric vehicle model is used. The model is robustly and efficiently fitted to the matched image features using an improved RANSAC scheme that automatically enforces physically plausible vehicle motions and speeds up the overall system at the same time. Temporal integration of the computed turn rates is performed by an Extended Kalman Filter with the bicycle motion model. Experiments with real world data show the applicability and robustness of the proposed concepts.

I. INTRODUCTION

Vision-based driver assistance systems have great potential for preventing fatalities. While most of the focus in recent years lay on the development of systems for highways or rural areas, future systems will have to cope with the more complex situations in urban areas.

This work addresses the problem of 3D monocular vehicle tracking and turn rate estimation in situations where vehicles need to be tracked along intersections and curves. The main contribution of this paper is a novel, robust and efficient method to estimate turn rates of vehicles from monocular images. The presented approach makes use of geometric relations between vehicle views and uses image feature correspondences to compute the relative vehicle motion between them. The calculated values are then used to improve a 3D vehicle tracker.

The remainder of this paper is organized as follows: Sec. I-A provides a short overview of the proposed system, Sec. I-B summarizes the related work. In Sec. II the approach to monocular turn rate estimation is introduced. The 3D vehicle tracker is introduced in Sec. III. Finally, experimental results on real world data are presented (Sec. IV) and a conclusion is drawn (Sec. V).

A. System overview

This paper introduces a system that robustly estimates the turn rate of vehicles by reconstructing the transformation of their rear section views in consecutive images. After finding the initial regions of interest (ROIs) using object detection [1],[2], image correspondences between the views are extracted and used to perform a robust, RANSAC-based model fitting by enforcing motion constraints on the estimated angular rates. At the end, the rates are integrated into an Extended Kalman Filter (EKF) based framework using the bicycle motion model [4], one view onto the other. In order to increase robustness and speed of the measurement process, candidate homographies are constrained to physically plausible vehicle motions. In a second step, the motion of the observing vehicle is compensated so that the true target angular rate \( \dot{\phi} \) is reconstructed from \( H \), even when both—target and the observing vehicle—are moving.

Temporal integration is performed by an Extended Kalman Filter (EKF) based 3D vehicle tracker [3], which smooths the measured turn rates over time using the bicycle motion model [4]. Fig. 1 illustrates the proposed architecture.

B. Related work

Video-based vehicle tracking and orientation estimation is a widely researched topic in the community (see [5] for an overview). However, most of the existing work focuses on highway scenarios [6],[7] or requires precisely known target geometries [8],[9],[10].

Approaches that focus on inner city or rural road scenarios either use depth information (e.g. from stereo vision [11]), or rely on classifiers [12],[13],[14],[15],[16] which are subject to classification errors and provide only very coarse orientation estimates.

Horbert et al. [17] propose a segmentation-based system to estimate the image distortion induced by vehicle motion.
However, the final viewpoint estimate has to be discretized in order to retrieve the real world angle from the image transformation.

Recently, Peng et al. [18] proposed a monocular system that is able to provide the 3D pose of preceding vehicles by incrementally building a target-local coordinate frame for pose estimation. However, this heavily depends on the accuracy of the estimated points with respect to the true position on the observed vehicle and may lead to errors in pose estimation that cannot be corrected as the assumed target frame is used as input for the next images.

II. VEHICLE TURN RATE ESTIMATION

The proposed system uses a geometrically constrained vehicle model to infer tracked vehicles’ turn rates by matching feature points in successive frames (Sec. II-A). After feature matching is completed, the vehicle model is robustly and efficiently fitted to the feature correspondences by constraining the vehicle motion to physically plausible ones (Sec. II-B) and taking the camera motion into account (Sec. II-C).

A. Geometric vehicle model

Since vehicle rear sections are almost planar, the relation between two views of the same rear section is captured by a homography \( H \in \mathbb{P}^{2 \times 2} \). Let \( x_1, x_2 \in \mathbb{P}^2 \) be corresponding points on the vehicle’s rear section in the first and second image, respectively. Then

\[
x_2 = \lambda H x_1
\]

or, equivalently,

\[
x_2 \times H x_1 = 0
\]

holds for a scale factor \( \lambda \in \mathbb{R} \). Given four non-collinear feature correspondences \( x_1 \leftrightarrow x_2, H \) can be computed up to scale using the Direct Linear Transformation (DLT) [19].

Once \( H \) is known, the observed target motion can be recovered in the form

\[
H = R_A + t_A n_A^T, \quad \Lambda = 1, 2,
\]

where \( R_A \) is the rotation, \( t_A \) is the depth-normalized translation and \( n_A \) is a normal vector on the mapped plane [20].

Note that if the distance \( d \) to the plane were known, the target translation could be fully reconstructed as \( t^* = dt_A \). Also note that \( H \) cannot be uniquely decomposed into \( R_A \) as there are two possible solutions for most setups [20]. The correct solution out of the two is chosen in subsequent processing steps (Sec. II-D).

B. Robust model fitting with target motion constraints

In practice, image feature correspondences are perturbed by noise and frequent matching outliers, so model fitting, i.e. finding \( H = R_A + t_A n_A^T \), has to be performed robustly. Thus, the RANSAC scheme [21] is employed.

To increase robustness and speed up RANSAC, the technique from [22] is used to enforce that \( H \) encodes a rigid body transformation. Furthermore, \( H \) is constrained to represent only physically plausible target motions. Since vehicles cannot freely rotate around every axis, the greatest portion of the observed rotation must be around the vehicle’s upright axis [4]. Thus, the rotation axis \( r \) of \( R_A \) has to be orthogonal to the ground plane (Fig. 2). Let \( g \) be a normal vector on the ground plane under the target vehicle. With \( g \), we use a threshold \( t_\alpha \) to enforce a plausible \( H \) by defining

\[
\alpha = \angle(r, g),
\]

\[
H \text{ valid } \iff \alpha \leq t_\alpha.
\]

Note that \( g \) is not known in general. Horbert et al. [17] use Structure from Motion (SfM) to estimate \( g \). We, instead, set \( g \) to be a normal to the ground plane under the ego vehicle and relax \( t_\alpha \) to avoid the computational cost of SfM and keep the overall system real-time capable.

Constraining \( H \) has two advantages. First, the consensus set has to be build only for those homographies that encode a valid motion. Thus, homographies stemming from matching errors can be rejected early, speeding up the overall system by an order of magnitude [22]. Second, the system gains robustness because in situations where only a few correct correspondences are found, no severely disturbed \( \varphi \) is used as input to the EKF tracker.

C. Ego motion compensation

So far, the camera was assumed to be static. However, the camera motion influences the observed angle \( \varphi \), and thus has to be accounted for. Unfortunately, in presence of both target and camera motion, an efficient 3D reconstruction is not possible in general without further assumptions [19]. However, since the rotation of vehicles does not encompass all three rotation axes, a simplified model that allows the computation can be used.

Let \( R_e \) be the rotation of the ego vehicle, computed using Visual Odometry [24] or known from the vehicle’s inertial sensors. To reconstruct the true target motion \( R^* \), the target rotation only induced by the motion of the observing vehicle needs to be known. Let \( X \) be this induced rotation (Fig. 3). Then

\[
R^* = XR_A
\]

holds. Let \( B \) be the target pose relative to the ego vehicle for the second image. Then, the relationship

\[
X = BR_eB^{-1}
\]

could be used to compute \( X \) with a known \( B \). However, estimation of \( B \) is not necessary to compensate for \( R_e \).

\[\text{1Although [24] is about visual odometry using stereo images, the corresponding source code, available at http://www.cvlibs.net/supports monocular camera motion estimation as well.}\]
because \( B \) is a general affine transformation. Thus, \( BR_eB^{-1} \) forms a conjugate rotation to \( R_e \) \[\text{(19)}\]. Since \( R_e \) is a rotation matrix, the eigenvalues of \( R_e \) are
\[
eig (R_e) = \{1, e^{i\beta}, e^{-i\beta}\}
\] (8)
for the rotation angle \( \beta \). Because of the conjugate relationship between \( R_e \) and \( BR_eB^{-1} \), the eigenvalues of \( X \) are
\[
eig (X) = \eig (BR_eB^{-1}) = \{\omega, \omega e^{i\beta}, \omega e^{-i\beta}\}
\] (9)
for a scale factor \( \omega \in \mathbb{R} \) \[\text{(19)}\]. Thus, the rotation angle \( \beta \) is preserved under the conjugate relationship and
\[
\varphi_X = \varphi_{R_e}
\] (10)
holds for the rotation angles \( \varphi_{R_e} \) and \( \varphi_X \) of \( R_e \) and \( X \).

Since vehicles mainly rotate around their upright axis, the rotation axes of \( R_e \), \( X \) and \( R_A \) have to be roughly the same. Hence, to compensate for \( R_e \), it is sufficient to combine the two rotations \( R_A \) and \( R_e \) by adding their rotation angles. Thus, we can compute the compensated target turn rate \( \dot{\varphi} \) as
\[
\dot{\varphi} = \varphi_X + \varphi_{R_e} = \varphi_{R_e} + \varphi_{R_A}.
\] (11)

D. Choosing the correct homography decomposition

There are two possible solutions to decompose \( H \) into rotation and translation (Sec. II-A). To choose the correct decomposition, Burke and Brink [25] propose to compare the computed \( \varphi \) to the current system state and choose the value that best matches it. However, this may lead to large errors, as false turn directions cannot be corrected in subsequent measurement cycles.

Instead of choosing the decomposition from context, we use both solutions throughout the overall processing chain and discard solutions when the required motion constraints do not hold for them. In most cases (\( \geq 90 \% \)), only one valid solution remains at the end. In cases when there are two remaining solutions, the smaller \( \varphi \) is chosen because slow turn manoeuvres are more likely to occur in everyday traffic. Experiments have shown that the choice of \( \varphi \) does not alter the overall system performance significantly in these cases as the angles seem to be very similar when they both fulfill the constraints. Thus, the EKF tracker effectively compensates for the disturbed turn rate.

Fig. 3. Overview of the situation for the ego motion compensation. (a) \( R_A \) is estimated as if the camera were static between the images. \( A, B \) denote the target pose relative to the ego vehicle for the first and second image, respectively. (b) In order to reconstruct the true target rotation \( R^* \) from \( R_A \), the rotation \( X \) induced by the ego rotation \( R_e \) needs to be recovered after \( R_A \) is estimated.

III. INTEGRATION WITH A HIGH LEVEL TRACKER

To integrate the computed \( \varphi \) over time, an EKF framework with the measurements
\[
z = \begin{pmatrix} x_i, y_i, w, \varphi \end{pmatrix}^T
\] (12)
and the vehicle state
\[
x = \begin{pmatrix} X, Y, v, \varphi, \dot{\varphi} \end{pmatrix}^T
\] (13)
is used. \( (x_i, y_i)^T \) and \( w \) denote the bottom center location and the width (all in pixels) of the ROI, respectively. Fig. 4 illustrates how the measurements are formed from ROIs. \( (X, Y)^T \) is the target location in the ego vehicle coordinate frame (Fig. 2), \( v \) denotes the target velocity, and \( \varphi \) and \( \dot{\varphi} \) are the target orientation and turn rate, respectively.

State prediction is performed with the bicycle motion model [4]. For the EKF update step, the measurement function \( z = h(x) \) is a perspective re-projection of the vehicle with the width \( W \). Note that this is basically equivalent to an inverse perspective mapping (IPM) using the theorem of intersecting lines; i.e. the vehicle’s distance from the camera \( d \) is set to be
\[
d = \frac{W f}{w_{px}} \cos (\varphi),
\] (14)
where \( f \) is the camera’s focal length, \( w_{px} \) is the width of the ROI in pixels, and \( \varphi \) is the vehicle orientation.

IV. EVALUATION

A. Experimental setup

To test the proposed concepts, real world data sequences were recorded. The target ground truth is obtained by a high precision localization unit based on an IMU\(^2\) and global position information from RTK/GPS\(^3\) (turn rate accuracy \( < 0.5 \) degrees) running in the target vehicle. Ground truth motion to image synchronization is performed via the GPS clock running synchronously in both vehicles. The test scenes are explained in more detail in Sec. IV-B.

The images are of size \( 752 \times 480 \) pixels from an intrinsically and extrinsically calibrated camera running at 16 frames per second. Object detection is performed with a boosted detector as described in [12]. Fig. 4(a) shows an exemplary ROI. To get correspondences between images, the standard Kanade-Lucas-Tomasi (KLT) feature tracker [26],[27] is used. To avoid problems caused by non-planar

---

\(^2\)Inertial Measurement Unit

\(^3\)Real Time Kinematic/Global Positioning System
Fig. 5. Sequence 2 (top row) and sequence 3 (bottom row) of the evaluation sequences (original picture size 752 × 480 pixels). As can be seen, especially in sequence 3, the resolution available to our algorithm is very low due to the small ROI size. In sequence 3, only about 20 × 20 pixels are available to extract feature correspondences. Fig. 6 depicts the ROIs available in the two right images of the bottom row.

Fig. 6. The proposed turn rate estimation is able to cope with very small ROIs or low resolution images. (a) and (b) show two ROIs from sequence 3.

vehicle rear sections, feature detection is only performed in the lower half of the square ROI from the object detector (Fig. 4(b)). The resulting ROI is divided into four equally sized regions, in which features are detected separately to ensure good spatial support. Feature detection is re-iterated with a lowered detection threshold until a sufficient amount of feature correspondences have been found.

Note that there is a multitude of alternative feature detectors available in the literature (see [28] for an overview) that could have been used instead of the KLT feature tracker. However, this paper focuses on the geometric vehicle model and the improved RANSAC scheme, both of which are independent of specific image features. An experimental evaluation of different feature detectors remains for future work.

B. Test sequences

We chose three different test sequences to test the system; all three cover different aspects. The first sequence is an intersection situation wherein the target vehicle performs a turn to the left. To keep this case simple, the observing vehicle does not move. The second sequence is a similar situation with the ego vehicle moving. The third sequence is longer and has larger discrepancies between target and ego motion because the ego vehicle performed small turns to generate a sinus-like trajectory while the target vehicle was driving straight. In addition, the ROI is only about 20 pixels high when the target vehicle starts a sharp right turn. Thus, this sequence acts as a test for the applicability of the proposed concepts in situations where only a few, very noisy feature correspondences can be found, e.g. when the camera has a low resolution. Fig. 5 shows exemplary pictures taken from sequences 2 and 3.

C. Results

Figures 7, 8 and 9 show the results of the algorithm on the three test sequences. As can be seen, the ground truth turn rate (red, dashed) is correctly approximated by the computed values (blue, solid) and the target turn is clearly visible in the plots. The computed values contain noise stemming from matching errors, inaccurate feature correspondences and the low resolution of the used camera hardware. However, this noise can be compensated by the EKF tracker. Note that the computed values are not smoothed or averaged over time.

Sequence 1 (Fig. 7) and sequence 2 (Fig. 8) show very similar behavior; the ego motion present in sequence 2 does not alter the turn rate estimation, thus the simplifications made to allow ego motion compensation do not harm the overall system performance.

Sequence 3 (Fig. 9) shows a turn in greater distance. As can be seen, the measurement noise is greater than for the first two sequences. This can be explained by the smaller ROI that makes it difficult to extract stable and correct feature correspondences (see Fig. 6). Nevertheless, the target’s turn direction is correctly estimated and the values show the correct magnitude when smoothed.

Figures 10, 11 and 12 show the temporally integrated target orientation from the EKF tracker (blue, solid) and the corresponding ground truth (red, dashed). Clearly, the measured turn rate estimates are correctly integrated over time and the target vehicle orientation is estimated with high accuracy, given the low resolution of the input images, the small ROI size, and the noisy feature correspondences.

In order to assess the effect of the turn rate measurements, we omitted the estimates of \( \dot{\varphi} \), i.e. we set the measurements to be \( \hat{z} = (x, y, w, )^T \), and re-ran the EKF tracker. The results are illustrated in Figures 10, 11 and 12 (green, dotted). Note that the measurement function \( \hat{z} = h(x) \) suffers from a projective ambiguity in this case as multiple state vectors
Fig. 7. Ground truth (red, dashed) and computed raw target turn rates (blue, solid) in sequence 1. Clearly, the target turn direction is correctly estimated by the proposed RANSAC scheme; and the computed values, albeit very noisy, correctly approximate the ground truth turn rate. Note that the computed values are not smoothed or averaged over time.

Fig. 8. Ground truth (red, dashed) and computed raw target turn rates (blue, solid) in sequence 2. Clearly, the target turn direction is correctly estimated by the proposed RANSAC scheme; and the computed values, albeit very noisy, correctly approximate the ground truth turn rate. Note that the computed values are not smoothed or averaged over time.

Fig. 9. Ground truth (red, dashed) and computed raw target turn rates (blue, solid) in sequence 3. Compared the the other test sequences, the computed turn rates contain more noise, especially when the target begins its right turn. This can be explained by the small size of the ROI available at the end of sequence 3. At the beginning of the right turn, the ROIs is only $20 \times 20$ pixels in size (see Fig. 6). Nevertheless, the computed values correctly indicated the target’s turn direction. Note that the computed values are not smoothed or averaged over time.

Fig. 10. Ground truth (red, dashed) and estimated (blue, solid) target orientation in sequence 1. As can be seen, the computed turn rates (Fig. 7) can be used to accurately estimate the target orientation through temporal integration with the bicycle motion model. The green, dotted curve indicates the orientation estimated only from ROIs, i.e. without the proposed turn rate estimation.

Fig. 11. Ground truth (red, dashed) and estimated (blue, solid) target orientation in sequence 2. As in Fig 10, the computed turn rates (Fig. 8) can be used to accurately estimate the target orientation through temporal integration with the bicycle motion model. The green, dotted curve indicates the orientation estimated only from ROIs, i.e. without the proposed turn rate estimation.

Fig. 12. Ground truth (red, dashed) and estimated (blue, solid) target orientation in sequence 3. The sinus-like trajectory of the ego vehicle is clearly visible in the ground truth since the target orientation is relative to the ego vehicle. Thus, the target orientation reflects the orientation changes of the ego vehicle. As can be seen, the estimated vehicle orientation lags a few images compared to the ground truth orientation. This is caused by the bicycle motion model which assumes a constant turn rate. Nevertheless, the target vehicle is correctly followed, even with the noisy input measurements (Fig. 9). The green, dotted curve indicates the orientation estimated only from ROIs, i.e. without the proposed turn rate estimation.
x are mapped to a similar measurement z (Fig 13). This ambiguity explains the sudden turns visible in the plots—smaller ROI sizes get misinterpreted as target turn. As can be seen easily from these results, a turn rate information is necessary for monocular 3D vehicle tracking, as the ambiguity in $h(x)$ cannot be resolved from ROIs alone.

V. CONCLUSION AND FUTURE WORKS

The paper presents a method to estimate the turn rate of vehicles from monocular images. Using feature correspondences in subsequent frames, our approach estimates homographies that describe the image distortion of rear section views in subsequent frames. These homographies are then used to infer the observed vehicle motion in 3D. We developed an improved RANSAC scheme to robustly and efficiently match a simplified vehicle model to the feature correspondences. To perform temporal integration of the computed turn rates, an EKF based 3D vehicle tracker with the bicycle motion model is used.

Experiments on real world data with highly accurate ground truth showed that the proposed concepts allow to estimate turn rates from monocular video sequences only. Even from ROIs of size $20 \times 20$ pixels and with strong ego motion, the computed values correctly indicated the target’s turn and allowed the EKF tracker to follow the vehicle along its trajectory.

In future, we plan to combine recent advances in multi-view vehicle detection [29],[12],[14] with the proposed concepts to further improve the vehicle tracker.

REFERENCES