Autonomous Convoy Driving by Night: The Vehicle Tracking System

Carsten Fries and Hans-Joachim Wuensche
Department of Aerospace Engineering
Institute for Autonomous Systems Technology
University of the Bundeswehr Munich, Germany
Email: carsten.fries@unibw.de

***Abstract*—Previous publications of our institute describe a robust vehicle tracking system for daylight conditions. This paper presents an improved vehicle tracking system which is able to detect and track the convoy leader also by twilight and night. The primary sensor equipment consists of a daytime camera, a LiDAR and an inertial navigation system. An expansion with a thermal and a lowlight camera was necessary to be robust against any illumination conditions. The system is capable of estimating the relative 3D position and orientation, the velocity and the steering angle of a convoy leader precisely in real-time. This makes it possible to follow the convoy leader's track. Another novelty is coupling a Kalman filter with a particle filter for higher stability and accuracy in vehicle tracking. The tracking system shows excellent functionality while driving more than 50 km fully autonomously in urban- and unstructured environments at night.**

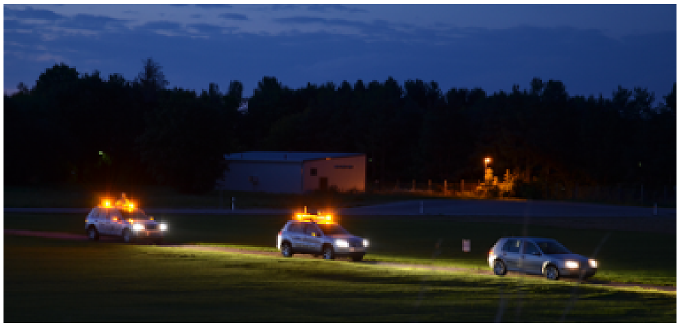


Fig. 1: Autonomous convoy driving at night. A VW Touareg following a VW Tiguan which follows a manually-driven VW Golf. Orange lights indicate an activated autonomous mode.

I. Introduction

The industrial and scientific interest in self-driving cars has grown constantly in recent years. There are various fields of application with potential advantages of (semi-) autonomous cars. Compared to human drivers, the technical components work twenty-four seven without decreasing reliability, can react faster and observe up to the whole 360 degree environment. Some benefits are fewer traffic collisions, a better traffic flow, higher fuel economy and free time for the human drivers [1], [2].

This paper introduces a vehicle tracking system for an autonomous convoy that is functional at night (see Fig. 1). It is an extension of our tracking system which was successfully demonstrated during daylight conditions [3], [4]. Compared to the previous version, the main improvements are:

- Nighttime vehicle detection by using more than one camera sensor (lowlight, thermal and daylight cameras).
- Higher stability and accuracy in vehicle tracking by coupling a Kalman filter [5] with a particle filter [6].

The outline of the paper is as follows: In the next section we describe the related work on vehicle detection and tracking methods. A preprocessing step for the presented approach is explained in Section III. Section IV describes the vehicle tracking system in detail. Experimental results collected while driving in urban and non-urban environments are illustrated in Section V. Finally, conclusions of this study and future work are given in the last section.

II. Related Work

The related work focuses on publications about solutions for vehicle detection and tracking at night.

Most authors use one sensor [7]–[12]. Some combine (different) sensors to achieve an increased measurement accuracy and robustness against varying weather conditions [3], [4], [13], [14]. Typically used sensors are daylight and infrared cameras, RADARs and LiDARs. The purchase price of LiDAR and thermal sensors is much higher than the price for RADARs and daylight cameras. Additional advantages of daylight cameras are the lower power consumption and the higher resolution. A disadvantage is that the image quality strongly depends on light and weather conditions. In contrast, RADAR and LiDAR sensors are more robust to light and weather changes and they provide depth information with a detection range of up to around 200 m.

The sensor equipment is placed on board of vehicles or stationary on infrastructure buildings. The stationary tracking systems are primarily designed for traffic surveillance. In contrast, on-board systems are generally used to prevent collisions and for intelligent headlights control [15]. Stationary sensors allow to remove the motionless background scene from the foreground (traffic participants). This method is called frame differencing or background subtraction [16]. Frame differencing is only applicable in case of constant illumination conditions [9]. But vehicle lights produce varying lighting effects especially at night. For that reason, Chen et al. [9] decided to detect headlights and taillights of cars and motorbikes in a different way. They use a daylight camera and extract the light blobs by applying multilevel histogram thresholding. A non-

978-1-4799-8757-3/15/$31.00 ©2015 IEEE

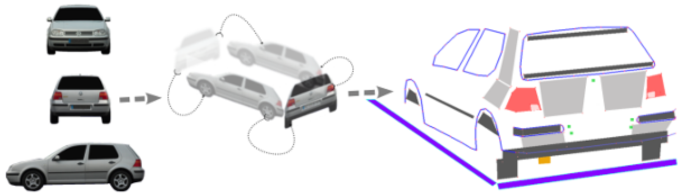


Fig. 2: Building a 3D feature model incl. vertices (green dots), edges (blue lines), colored / thermal regions (quadrangles) and occupied / non-occupied ground-plane cells. Purple quadrangles on the ground-plane correspond to non-occupied cells. Occupied cells are under the vehicle and not visualized.

stationary vehicle tracking system is described in [10]. Their sensor equipment consists of a black and white micro-camera which is mounted on-board a car in the windshield area. They detect vehicles with the same driving direction and oncoming vehicles in real-time. Any vehicle detection induces a transition of the own vehicle headlights from high beams to low beams. The image processing starts with adaptive thresholding to extract bright blobs in the image. Afterwards, blobs are filtered out in case of wrong geometric characteristics. The remaining blobs are used for vehicle detection and tracking with a Kalman filter. Contrary to image-based head or tail light detection, a LiDAR-based perception system is introduced in [11]. Their algorithm segments point clouds in fragments by consideration of the point distances. The segmentation is followed with a classification step which is based on contours. E.g. a L-shape corresponds to a car.

Most mentioned publications do not have an accurate position estimation of traffic participants. They need only rough 2D position and velocity information for their area of application. In the field of autonomous convoy driving, a precise 3D vehicle position, orientation and the velocity of each local convoy leader is required. Otherwise, the follower is not driving on the leader's track, which implicates some hazard traffic situations. E.g. wrong lateral or longitudinal distance estimation increase the risk of traffic or infrastructure collisions.

Himmelsbach et al. [12] process raw point cloud data to estimate the relative 3D position and orientation of a convoy leader and its velocity. Another approach uses LiDAR and camera information for this. Manz et al. [3] and [4] build a detailed 3D feature model of each potential convoy leader type in an offline step. A particle filter generates 3D vehicle hypotheses. Each hypothesis is evaluated with the 3D feature model and the optimal 3D position and orientation of the convoy leader is determined. There exist other approaches with predefined 3D feature models, e.g. [7] uses 3D edge models of vehicle windshields and [17] creates complete 3D wireframe models of different vehicle types.

III. Building 3D Feature Models

Our vehicle detection algorithm requires a 3D feature model for each vehicle type which describes the vehicle-specific features. The model generation is an offline preprocessing step and consists of:

- Capturing three orthogonal views of vehicle sides
- Image scaling and sorting
- Selection of significant features: Vertices, edges, colored / thermal regions and occupied / non-occupied ground-plane cells
- Positioning the features to their correct 3D position

The generation process is performed manually. Figure 2 visualizes the single steps for creation of a 3D feature model. In this process, it is important to select the significant features for different sensor types. This induces robustness against varying illumination and weather conditions. Vertices, edges and colored regions are appropriate for low/day-light cameras and thermal regions are appropriate for infrared cameras. Occupied and non-occupied ground-plane cells are practical in case that LiDAR data is used to generate occupancy grids. Some typical vehicle features are two red rear lights, two bright driving lights, a hot area on the exhaust, dark regions under the vehicle and on tires, four vertices on the license plate border and edges which lie on intensity transitions.

IV. Vehicle Tracking System

The goal of the vehicle tracking system is a continuous precise estimation of the relative cylindrical translation $\{\text{distance } r, \text{angle } \alpha, \text{height } h\}$ and orientation $\{\text{yaw } \psi, \text{pitch } \theta, \text{roll } \phi\}$, the velocity v and the steering angle λ of a convoy leader. The algorithm is based on the combination of a multidimensional particle filter and a Kalman filter. In the first step, the particle filter estimates the mentioned information about the convoy leader. Some of these measurement values are the input for the Kalman filter which implies more robustness against measurement errors produced from illumination and weather changes.

The following sections describe that combination of a particle filter and a Kalman filter and their state initialization. We refer to [5] and [6] for a general application-independent description of both filters.

A. Filter Initialization

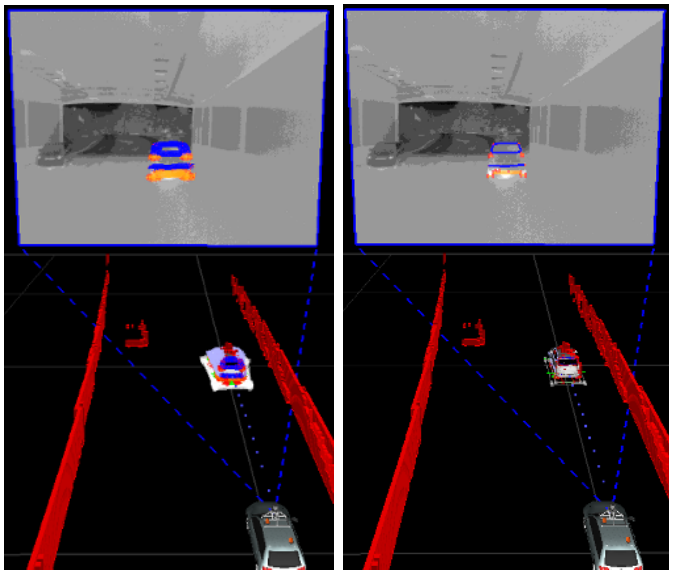
In case that another vehicle detection application provides information about the leading vehicle, e.g. our fast template-based vehicle detection algorithm [4], [8], we initialize both filters with this data. Otherwise we set $\lambda = 0$, v to the ego velocity and the relative pose $\boldsymbol{P}_{\text{ego}\rightarrow\text{obj}} = \{r, \alpha, h, \psi, \theta, \phi\}$ to $\{10, 0, 0, 0, 0, 0\}$. This pose is the most likely situation at convoy start. Furthermore, adding of Gaussian noise to all particle states distributes these vehicle hypotheses in an appropriate way.

B. Particle Filter

A particle filter is a closed loop system with a continuous comparison between prediction and measurement. The following two steps describe the control system.

1) Prediction: The multidimensional particle filter predicts the eight state values $\boldsymbol{x} = \{\boldsymbol{P}_{\text{ego}\rightarrow\text{obj}}, v, \lambda\}$ through a kinematic bicycle model and by consideration of the estimated egomotion [18] from time step $t-1$ to t. $\boldsymbol{P}_{\text{ego}_\text{t}\rightarrow\text{obj}_\text{t}}$ is predictable through the multiplication of inverse egomotion $\boldsymbol{P}_{\text{ego}_t\rightarrow\text{ego}_{t-1}}$, last relative pose $\boldsymbol{P}_{\text{ego}_{t-1}\rightarrow\text{obj}_{t-1}}$ and object motion $\boldsymbol{P}_{\text{obj}_{t-1}\rightarrow\text{obj}_t}$. The object motion is determinable by consideration of predicted v and λ through an Ackermann steering model.

978-1-4799-8757-3/15/$31.00 ©2015 IEEE

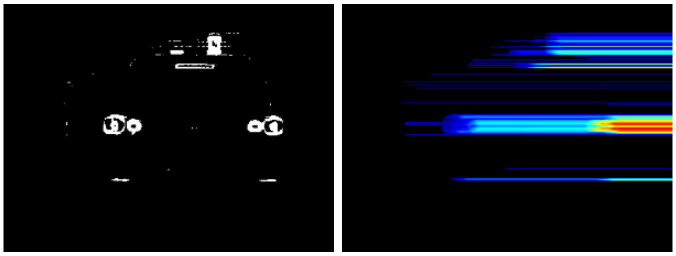


(a) (b)

Fig. 3: Particle filter prediction and projection of all particles to the image plane of the thermal camera: (a) All particle hypotheses. (b) Particle with highest weight. The upper part of the image visualizes the thermal image plane. The lower image part shows the 3D world inclusive the $2\frac{1}{2}$D occupancy grid (red cuboids).

2) Update: The update step determines the measurement values $\boldsymbol{y} = \boldsymbol{P}_{\text{ego}\to\text{obj}}$ to correct the prediction from Section IV-B1. Therefore, all n particles, or rather pose hypotheses $\boldsymbol{P}_{1\ldots n}$ of the tracked vehicle, are projected into the image plane (see Fig. 3(a)). Then, the model features are used to evaluate the hypotheses. Each hypothesis $\boldsymbol{P}_i \in \{\boldsymbol{P}_1 \ldots \boldsymbol{P}_n\}$ is weighted based on the feature congruency between the projected model features and the current image data. In order to apply this feature matching, it is necessary to extract all feature types in the current camera image. Instead of processing the full image, we recommend to apply the feature extraction only in an area of interest. This reduces the processing time enormously. The feature extraction works as follows:

- *Vertices*
  Corner detection is done by calculation of the minimal eigenvalues of the image gradient covariance matrix.
- *Edges*
  The Canny edge operator [19] produces raw edge data for each camera image. Criteria to weight each hypothesis are the distance, the orientation difference and the intensity difference between edges. The distance of two edges is calculable using the Euclidean distance. Pairwise discretization of two edges results in a set of point correspondences. Then the distance of two edges is the arithmetic mean of point distances. The pixel values correspond to the edge intensity.
- *Colored/Thermal regions*
  Colored and thermal areas are defined by $1 \ldots m$ quadrangles to evaluate hundreds of hypotheses efficiently. Typical color regions are white, black, amber and red. A smoothed HSV color space is suitable to work with color information. Gonzalez and Woods describe in [20] the transformation from a RGB image into the HSV color space. We cascade this color space transformation with a fast smoothing algorithm to reduce pixel noise. The next step is to generate a binary image mask for each color, respective hue range (see Fig. 4(a)). Simple thresholding is a possible method for that. E.g. a pixel value (u, v) with hue value $h \in [0 \ldots 360]$, saturation value $s \in [0 \ldots 1]$ and luminance value $v \in [0 \ldots 1]$ is classified as a red pixel if the following three nighttime-specific conditions are valid:
  $$h \leq 40 \text{ or } h \geq 320 \tag{1}$$
  $$s \geq 0.36 \tag{2}$$
  $$v \geq 0.36 \tag{3}$$
  A large s and v range allows color classification in different illuminations. The processing of thermal regions is similar. Image thresholding of a specific temperature value generates a binary image mask for this temperature.
  Each binary mask is usable to evaluate the feature congruence between projected model features and the current image data. Concerning the computational efficiency of hundreds of particles, it is advisable to calculate line integral images ([21], [22]) for analyzing the feature congruence. The construction of line integral images $\boldsymbol{I}_{\text{L}}$ out of binary images $\boldsymbol{I}_{\text{B}}$ is processible in parallel. Each pixel (u, v) receives the cumulative sum of pixel values starting from the beginning of the row:
  $$\boldsymbol{I}_{\text{L}}(u, v) = \sum_{i=1}^{\text{u}} \boldsymbol{I}_{\text{B}}(i, v) \tag{4}$$
  In equation (4), u and v correspond to the column and row of an image. The Figure 4(b) shows one line integral image with a summation of pixel values from left to right. It is visualized as a false color image to illustrate the summation.
  The evaluation of color and thermal quadrangles with a line integral image requires a discretization of each quadrangle to lines $\boldsymbol{l}_{1\ldots l}$. Estimating the sum of pixels
  $$\sum_{i=v_{\text{start}}}^{v_{\text{end}}} \boldsymbol{I}_{\text{L}}(u_{\text{end}}, i) - \boldsymbol{I}_{\text{L}}(u_{\text{start}}, i)$$



(a) (b)

Fig. 4: Color filtering for the red tone: (a) Binary image mask after thresholding. (b) Line integral image as false color image to illustrate the summation of pixel values.

978-1-4799-8757-3/15/$31.00 ©2015 IEEE

for one quadrangle needs only l subtractions and additions. u_{start} and u_{end} denotes the first and last image column coordinate of a line. v_{start} and v_{end} denotes the first and last image row of a quadrangle. Each pose $\boldsymbol{P}_i \in \{\boldsymbol{P}_1 \ldots \boldsymbol{P}_n\}$ is weighted based on the sum of pixels. A hypothesis gets a high weight if the pixel sum is high.

- *Occupied/non-occupied cells*
The occupancy grid has a size of $100 \times 100\,\mathrm{m}$ and consists of $20 \times 20\,\mathrm{cm}$ cells. A cell is classified as occupied if a laser beam hits this cell. A probability value for each cell is predictable by the accumulation of obstacle informations over time. The longer the time horizon of the data accumulation, the more shadows of moving objects emerge. For that reason, we do not recommend obstacle accumulation to prevent a negative influence on vehicle detection.
Each pose $\boldsymbol{P}_i$ receives a higher weight if the predicted occupancy (ground area under the 3D vehicle hypothesis) matches the measured occupancy and vice versa.

The mentioned features are used to weight the particles and hypotheses respectively. This enables the correction of the predicted state values and resampling of the particles. Following the update step, the control system restarts with the prediction of the state values for the next time step according to a kinematic vehicle model.

Our previous publication [4] showed that a good overall estimation of the pose $\boldsymbol{P}_{\text{ego}\rightarrow\text{obj}}$ is the arithmetic mean value from the best $10\,\%$ of the entire particle set. In contrast to this, we achieve higher stability and accuracy by using the particle with the highest weight as an input to a Kalman filter.

There are two possibilities to handle multi-camera support with a particle filter:

- Each camera is assigned to a separate particle filter which is recommendable in case of imprecise extrinsic camera calibration. For each particle filter, the particle with the highest weight is used as an input for a sequential Kalman filter innovation [23]. In doing so, the error of the extrinsic camera calibration is predictable over time.
- In case of an accurate extrinsic calibration between the cameras, a more computational efficient method is using a single particle filter for all cameras. In doing so, the particle-to-camera correlation is uniformly distributed.

C. Kalman Filter

A Kalman filter is a feedback estimation system with a continuous comparison between prediction and measurement. We use a Scaled Unscented Kalman Filter (SUKF) [24] for nonlinear state estimation.

1) Prediction: The Kalman filter predicts the eight state values $\boldsymbol{x} = \{\boldsymbol{P}_{\text{ego}\rightarrow\text{obj}}, v, \lambda\}$ by using a kinematic bicycle model and the egomotion [18] from time step $t-1$ to t.

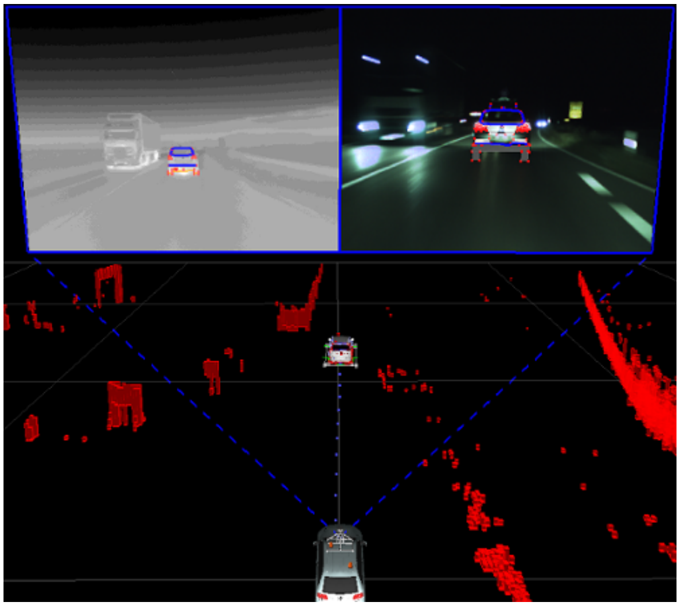


Fig. 5: Tracking the convoy leader to follow autonomously. Obstacles from the $2\frac{1}{2}$D occupancy grid are colored as red cuboids. Thermal and daylight camera visualizes the 3D-to-2D projection of the Kalman filter pose estimation.

2) Update: The measurement value $\boldsymbol{y} = \boldsymbol{P}_{\text{ego}\rightarrow\text{obj}}$ for the Kalman filter is the particle with the highest weight. Each measurement induces a filter innovation which corrects the predicted state values $\boldsymbol{x}^{\star}$: Relative pose $\boldsymbol{P}_{\text{ego}\rightarrow\text{obj}}$, object velocity v and steering angle λ (Ackermann steering geometry). The incoming measurement $\boldsymbol{y}$, its prediction $\boldsymbol{y}^{\star}$ and the Kalman gain matrix $\boldsymbol{K}$ are used for this correction:

$$\hat{\boldsymbol{x}} = \boldsymbol{x}^{\star} + \boldsymbol{K} \cdot [\boldsymbol{y} - \boldsymbol{y}^{\star}] \tag{5}$$

A detailed description of the Kalman filter innovation can be found in [24].

V. Experimental Results

This section presents the vehicle tracking results while driving in an autonomous convoy (see Fig. 5).

The maximum amount of convoy participants were three vehicles during twilight and two vehicles at night. A VW Touareg called MuCAR-3 (Munich Cognitive Autonomous Robot Car 3rd Generation [25]), a VW Tiguan (MuCAR-4) and a VW Golf 4 were part of the vehicle pool. Figure 1 shows these vehicles by twilight. The Touareg and Tiguan can drive fully autonomously and the Golf 4 is used as a manually-driven convoy leader. The hardware equipment of MuCAR-4 is similar to MuCAR-3:

- Our inertial navigation system (INS) consists of an inertial measurement unit (IMU) coupled to a Differential GPS with Real Time Kinematic (RTK) to enhance positional precision to approximately 2 cm.
- A movable camera platform [26] equipped with three daylight cameras and one lowlight camera is mounted behind the windshield. The daylight cameras have 1296 x 966 pixels and provide by daytime 20 fps. The

	$r\,(\mathrm{m})$	$\alpha\,(^\circ)$	$h\,(\mathrm{m})$	$\psi\,(^\circ)$	$\theta\,(^\circ)$	$\phi\,(^\circ)$	$v\,(\frac{\mathrm{m}}{\mathrm{s}})$
$\max_{\mathrm{night}}$	32.7	37.9	0.76	84.6	2.41	0.94	20.7
$\epsilon_{\mathrm{night}}$	0.34	0.87	0.08	2.19	1.47	0.67	0.32
$\epsilon_{\mathrm{twilight}}$	0.31	0.56	0.06	1.82	1.21	0.50	0.29
ϵ_{day}	0.31	0.30	0.06	1.45	1.16	0.45	0.27
$\epsilon_{\mathrm{day}_{\mathrm{old}}}$	0.47	0.32	0.06	1.50	1.16	0.45	0.38

TABLE I: Tracking accuracy compared to ground truth: Evaluated values are the relative position $\{r, \alpha, h\}$ and orientation $\{\psi, \theta, \phi\}$ in cylindrical coordinates and the object velocity v. $\max_{\mathrm{m}}$ denotes the maximal state values and $\epsilon_{\mathrm{night}}$ the RMSE of vehicle tracking by night. $\epsilon_{\mathrm{twilight}}$ and ϵ_{day} denotes the error by twilight, respectively daytime. $\epsilon_{\mathrm{day}_{\mathrm{old}}}$ denotes the RMSE from previous publication [4].

daylight cameras provide 5-10 fps in lowlight conditions because of the increased exposure time. In contrast to the daylight cameras, the lowlight camera with 1280 x 960 pixels and a dynamic range of 85 dB runs continuously with 10 fps.

- Two sensors are positioned on the roof. The first is an uncooled thermal camera with 640 x 480 pixels and a spectral range of 7.5 to 13 µm. The measurement range starts at -20°C and goes up to 150°C with an absolute measurement precision of ±2°C. The second sensor is a LiDAR which has a 360° horizontal field of view, a 26.8° vertical field of view and a 10 Hz rotation rate.

The software runs on a computer which has an Intel Xeon L5640 Dual CPU Hexa Core and is allowed to use four cores of them.

Each vehicle has an INS and a radio component to transmit GPS coordinates between vehicles. This vehicle-to-vehicle communication was exclusively used for generating ground truth data $\boldsymbol{P}_{\mathrm{GT}_{\mathrm{ego}\to\mathrm{obj}}}$ through a transformation from ego to the object position. Table I lists the measurement accuracy by comparison of the presented tracking algorithm and ground truth data. In total, the driving length was 5 km on the road and through unstructured terrain. In doing so, MuCAR-3 handled the throttle, brake and steering fully autonomously. The tracking algorithm runs in real-time with a precise pose estimation allowing convoy velocities of up to 28 $\frac{\mathrm{m}}{\mathrm{s}}$ during twilight and 21 $\frac{\mathrm{m}}{\mathrm{s}}$ at night.

The performance of the system was tested on a challenging 20 km route in suburbs around Munich, Germany. During the 34 minutes autonomous drive no manual interventions were required. In this 2-car convoy run, MuCAR-4 was used as a manually-driven convoy leader, avoiding (easy) highways, and concentrating on urban and non-urban environments, all in normal public traffic. MuCAR-3 was the autonomous follower. Figure 6 and 7 show a 12 km part of the entire route. More visual impressions are available on Youtube: http://youtu.be/Q8wtJ_YOg64
The tracking system can cope with rainy weather, sharp turns, tunnels (see Fig. 3 and 7(b)), gates and partial occlusion like bushes, high grass or windshield wipers. In addition, it is real-time capable with an overall cycle time between 20 and 60 ms.

VI. Conclusion and Future Work

A. Conclusion

This paper describes a vehicle tracking system for autonomous driving in a convoy. Determining an accurate relative vehicle pose, velocity and steering angle of a specific convoy participant is essential for exact path following. We could increase the vehicle tracking stability and accuracy of our previous publication [4] by coupling a particle filter with a Kalman filter.

The vehicle tracking system is based on detailed 3D feature models which consist of significant vehicle specific features. The model generation is an offline preprocessing step where each feature is chosen manually depending on the sensor type.

Our vehicle tracking system supports multiple cameras and extends the daylight functionality from [3], [4] with twilight and night-time operability. The integration of a lowlight and thermal camera was necessary to handle vehicle tracking during twilight and night-time illumination. A 360° LiDAR sensor provides depth information and is required if the (local) convoy leader is driving out of the camera field of view.

We evaluated the tracking system on real-world data by driving autonomously in urban and unstructured terrain. MuCAR-3 drove more than 50 km in an autonomous convoy consisting of two or three vehicles. The real-time tracking system proved to be reliable by night and during different weather conditions.

B. Future Work

Our Kalman filter works not optimally and needs further investigations. Kalman filtering usually requires white measurement and process noise. The assumption of white noise is not fulfilled by using the particle with the highest weight as an input to a Kalman filter.

ACKNOWLEDGMENTS

The authors gratefully acknowledge funding by the Federal Office of Bundeswehr Equipment, Information Technology and In-Service Support (BAAINBw).

References

[1] J. M. Lutin, A. L. Kornhauser, and E. Lerner-Lam, "The Revolutionary Development of Self-Driving Vehicles and Implications for the Transportation Engineering Profession," *Institute of Transportation Engineers (ITE) Journal*, vol. 83, pp. 28–32, 2013.

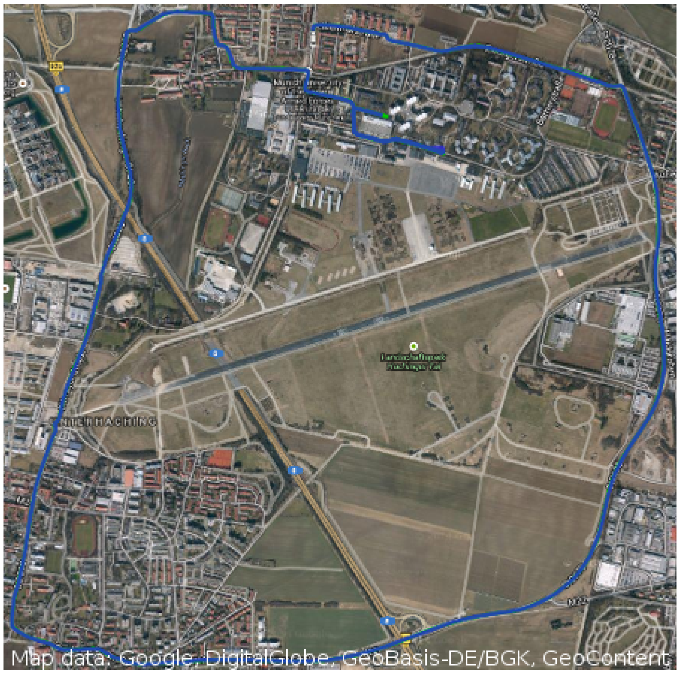
[2] T. Robinson, E. Chan, and E. Coelingh, "Operating Platoons on Public Motorways: An Introduction to The SARTRE Platooning Programme," in *17th ITS World Congress*, Korea, 2010.

[3] M. Manz, T. Luettel, F. von Hundelshausen, and H.-J. Wuensche, "Monocular Model-Based 3D Vehicle Tracking for Autonomous Vehicles in Unstructured Environment," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.

[4] C. Fries, T. Luettel, and H.-J. Wuensche, "Combining Model- and Template-based Vehicle Tracking for Autonomous Convoy Driving," in *IEEE Intelligent Vehicles Symposium (IV)*, 2013.

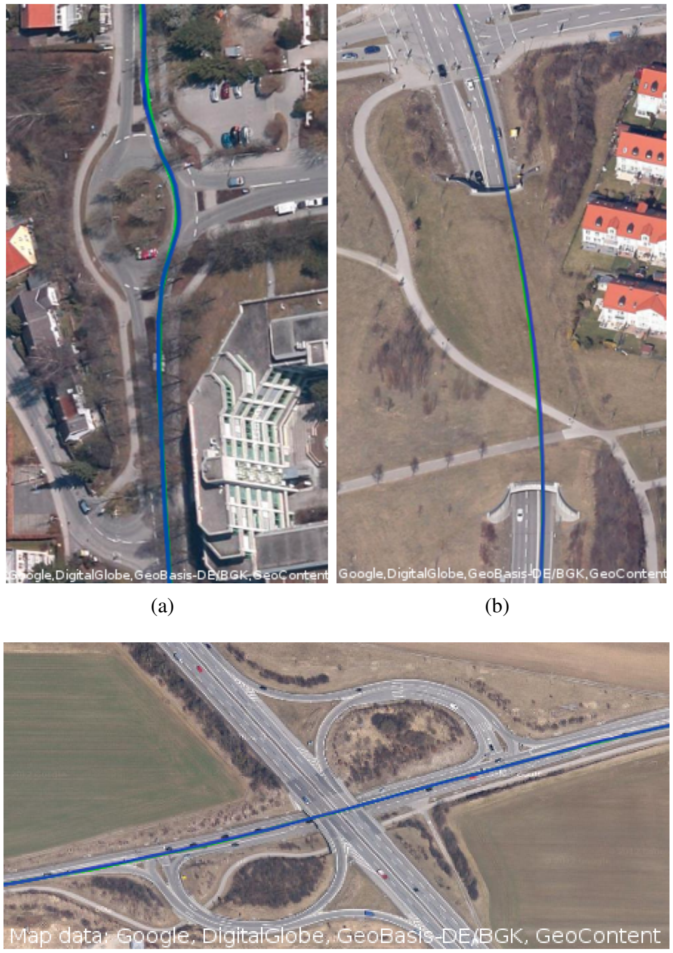
[5] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME – Journal of Basic Engineering*, no. 82 (Series D), pp. 35–45, 1960.

[6] N. Gordon, D. Salmond, and A. F. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *Radar and Signal Processing, IEE Proceedings F*, vol. 140, no. 2, pp. 107–113, 1993.



Map data: Google, DigitalGlobe, GeoBasis-DE/BGK, GeoContent

Fig. 6: Overview of a 12 km driven path while driving autonomously by night. Egomotion is colored with green lines. Vehicle tracking is visualized with blue lines.


Google, DigitalGlobe, GeoBasis-DE/BGK, GeoContent

(a)

Google, DigitalGlobe, GeoBasis-DE/BGK, GeoContent

(b)

Map data: Google, DigitalGlobe, GeoBasis-DE/BGK, GeoContent

(c)

Fig. 7: Coped challenging road situations: (a) Congested Rotary. (b) Long tunnel. (c) Motorway junction. Egomotion is colored with green lines. Vehicle tracking is visualized with blue lines.

[7] K. Robert, “Video-based traffic monitoring at day and night - vehicle features detection tracking,” in *12th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Australia, October 2009.

[8] C. Fries and H.-J. Wuensche, “Monocular Template-based Vehicle Tracking for Autonomous Convoy Driving,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2014.

[9] Y.-L. Chen, B.-F. Wu, H.-Y. Huang, and C.-J. Fan, “A real-time vision system for nighttime vehicle detection and traffic surveillance,” *IEEE Transactions on Industrial Electronics*, vol. 58, pp. 2030–2044, 2011.

[10] P. F. Alcantarilla, L. M. Bergasa, P. Jiménez, M. Á. Sotelo, I. Parra, D. F. Llorca, and S. Mayoral, “Night time vehicle detection for driving assistance lightbeam controller,” in *IEEE Intelligent Vehicles Symposium (IV)*, June 2008, pp. 291–296.

[11] R. Dominguez, E. Onieva, J. Alonso, J. Villagra, and C. Gonzalez, “Lidar based perception solution for autonomous vehicles,” in *11th International Conference on Intelligent Systems Design and Applications (ISDA)*, November 2011, pp. 790–795.

[12] M. Himmelsbach, T. Luettel, and H. Wuensche, “Real-time object classification in 3d point clouds using point feature histograms,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2009, pp. 994–1000.

[13] R. Schweiger, H. Neumann, and W. Ritter, “Multiple-cue data fusion with particle filters for vehicle detection in night view automotive applications,” in *IEEE Intelligent Vehicles Symposium (IV)*, June 2005.

[14] S. Kim, S. young Oh, J. Kang, Y. Ryu, K. Kim, S.-C. Park, and K. Park, “Front and rear vehicle detection and tracking in the day and night times using vision and sonar sensor fusion,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, August 2005.

[15] R. O’Malley, E. Jones, and M. Glavin, “Rear-lamp vehicle detection and tracking in low-exposure color video for night conditions,” *Transactions on Intelligent Transportation Systems*, vol. 11, no. 2, pp. 453–462, 2010.

[16] L. Guo and X. H. Shen, “Vehicle detection method under night circumstance,” in *Applied Mechanics and Materials*, 2013.

[17] J. Lou, T. Tan, W. Hu, H. Yang, and S. J. Maybank, “3-D Model-Based Vehicle Tracking,” *IEEE Transactions on Image Processing*, 2005.

[18] T. Luettel, M. Himmelsbach, F. von Hundelshausen, M. Manz, A. Mueller, and H.-J. Wuensche, “Autonomous Offroad Navigation Under Poor GPS Conditions,” in *3rd Workshop On Planning, Perception and Navigation for Intelligent Vehicles (PPNIV), IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2009.

[19] J. Canny, “A Computational Approach to Edge Detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986.

[20] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. USA: Prentice Hall, 2008.

[21] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001.

[22] ——, “Robust Real-time Object Detection,” in *International Journal of Computer Vision*, 2001.

[23] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. Wiley-Interscience, 2006.

[24] S. J. Julier, “The Scaled Unscented Transformation,” in *American Control Conference*, vol. 6, 2002, pp. 4555–4559.

[25] M. Himmelsbach, F. von Hundelshausen, T. Luettel, M. Manz, A. Mueller, S. Schneider, and H.-J. Wuensche, “Team MuCAR-3 at C-ELROB 2009,” in *1st Workshop on Field Robotics, Civilian European Land Robot Trial*, 2009.

[26] A. Unterholzner and H.-J. Wuensche, “Hybrid Adaptive Control of a Multi-Focal Vision System,” in *IEEE Intelligent Vehicles Symposium (IV)*, USA, 2010, pp. 534–539.