Geometric-Model-Free Tracking of Extended Targets Using 3D-LIDAR-Measurements

Philipp Steinemann[a], Jens Klappstein[a], Jürgen Dickmann[a],
Felix von Hundelshausen[b], Hans-Joachim Wünsche[b]
[a]Daimler AG, [b]University of the Bundeswehr München

ABSTRACT

Tracking of extended targets in high definition, 360-degree 3D-LIDAR (Light Detection and Ranging) measurements is a challenging task and a current research topic. It is a key component in robotic applications, and is relevant to path planning and collision avoidance.

This paper proposes a new method without a geometric model to simultaneously track and accumulate 3D-LIDAR measurements of an object. The method itself is based on a particle filter and uses an object-related local 3D grid for each object. No geometric object hypothesis is needed. Accumulation allows coping with occlusions.

The prediction step of the particle filter is governed by a motion model consisting of a deterministic and a probabilistic part. Since this paper is focused on tracking ground vehicles, a bicycle model is used for the deterministic part. The probabilistic part depends on the current state of each particle. A function for calculating the current probability density function for state transition is developed. It is derived in detail and based on a database consisting of vehicle dynamics measurements over several hundreds of kilometers. The adaptive probability density function narrows down the gating area for measurement data association.

The second part of the proposed method addresses weighting the particles with a cost function. Different 3D-grid-dependent cost functions are presented and evaluated.

Evaluations with real 3D-LIDAR measurements show the performance of the proposed method. The results are also compared to ground truth data.

Keywords: LIDAR, vehicle tracking, local grid

1. INTRODUCTION

Object detection and tracking is a key component of modern driver assistance systems. As the measurement resolution of modern sensors becomes finer, more simultaneous measurements of the same object are made. This progress is well-marked when using 3D laser scanners. 3D laser scanners are able to get hundreds of measurements of one object in one measurement cycle. However the object is not sampled densely, therefore no exact point-to-point correspondence in consecutive measurements can be calculated to obtain the rigid body motion.

To deal with the detailed measurements of one object, a particle filter (Chapter 3) is combined with a local 3D grid (Chapter 4) for each object. This requires segmentation of the measurements as a first step. The segmented data is obtained using a 2.5D algorithm similar to that proposed by Michael Himmelsbach et al. [4]. For measurement integration in the local 3D grid, a confidence value consisting of combined cost functions is used (Chapter 4). The tracking is accomplished by a particle filter using the confidence value of the local 3D grids in combination with a motion model (Chapter 5) and data evaluated from vehicle dynamics (Chapter 6). Chapter 7 shows the results and Chapter 8 gives a conclusion. The following chapter gives a brief introduction to similar works.

2. RELATED WORK

Grid-based approaches are commonly used when dealing with environment perception and representation. All sensor measurements are saved and accumulated in a map. For static object representation, 2D occupancy grid map approaches introduced by Alberto Elfes [2] are widespread. With the development of 3D sensors and higher computing power, more and more grid based 3D approaches have been published [8].

Laser Radar Technology and Applications XVII, edited by Monte D. Turner, Gary W. Kamerman,
Proc. of SPIE Vol. 8379, 83790C · © 2012 SPIE · CCC code: 0277-786X/12/$18 · doi: 10.1117/12.919011

Austin Eliazar [3] proposes a grid-based approach called DP-SLAM for simultaneous localization and mapping. This approach uses a particle filter to maintain a joint probability distribution over maps and robot positions.

Frank Moosmann et al. [7] accumulate measurements of objects in a local coordinate system for each object. The registration of new measurements to the already integrated measurements in the object related local coordinate system is done using an ICP algorithm [10]. This step is called refinement. The refined motion estimate is then used for a Kalman filter update.

Jan Effertz [1] describes a local map for each object to accumulate its contour in 2D. For the sake of data reduction, the measurements are processed and a polygon representing the 2D contour is calculated. Only the reduced data of the contour is integrated into the local map.

3. PARTICLE FILTER

A particle filter is used to track an object and accumulate its measurements in a local grid. For a detailed description of particle filters, refer to [9] and [5].

Each of the nP particles of the particle filter represents one local grid with its internal states. The state vector $\vec{x}$ is defined in Equation (1) and corresponds to the states of the used motion model. x and y represent the position of the object in global coordinates, v is the velocity, φ the steering angle, and ψ the orientation in the global coordinate system.

$$\vec{x} = [x, y, v, \varphi, \Psi]^T \tag{1}$$

The motion of the particles depends on the current state of $\vec{x}$. The parameters velocity v and steering angle φ used for the probabilistic motion are sampled from the distributions described in Chapter 6.

Weighting and resampling of the particles is done using the confidence value described in Chapter 4.

Chapter 7 shows the results of tracking and accumulating the measurements of one object with the particle filter approach.

4. LOCAL GRID

A local grid is defined as a 3D grid related to a single rigid object. It is assumed that the length l, width w, and height h of the grid are all multiples of the cell size κ (here 0.1 m). Equation (2) shows the mathematical definition of a grid. G defines the center of each cell. A cell itself is defined as the cube with edge lengths κ and the center according to G (Figure 1). Function G counts the number of measurements falling into each cell. This is possible because the beam divergence of the sensor is small compared to the used cell size κ. Initially, all values of G are set to zero. When integrating new measurements into the grid, several cost functions are calculated and combined to one single confidence value. The confidence value is used later to weight the particles in the particle filter. This means that one local grid is a hypothesis of the position and orientation of a tracked object. The particle filter uses several hypotheses for tracking, so numerous local grids are needed. This needs to be kept in mind when defining the cost functions, because values of all hypotheses are needed for normalization.

$$G : \Gamma \to R^3 \tag{2}$$
$$\Gamma := \{(i\kappa, j\kappa, q\kappa) \mid i \in N_x, j \in N_y, q \in N_z\},$$
$$N_x = \{1,...,n_x\},\ N_y = \{1,...,n_x\},\ N_z = \{1,...,n_z\}$$
with
$$n_x = \frac{l}{\kappa},\ n_y = \frac{w}{\kappa},\ n_z = \frac{h}{\kappa}$$

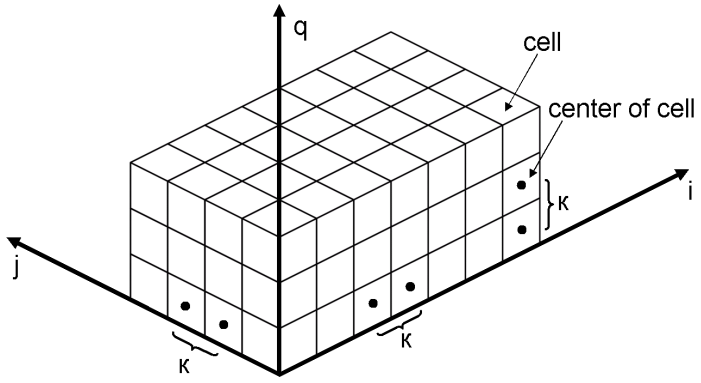

Figure 1. Definition of the local grid.

4.1 Cost functions

The following cost functions score the integration of the measurements into the grid. In the subsequent subchapter, the cost functions are compared and evaluated for their use. Each cost function is defined in the interval [0 1], where zero means low cost and one means high cost. Each cost function is illustrated by a schematic representation.

Integration ratio (IR)

Since the local grid is rigid, a cost function counting the number of measurements integrated into the grid is defined. The more measurements fall into the grid, the better the results. This is defined in Equation (3), with nI being the number of integrated measurements. N is the total number of measurements. Figure 2 shows a schematic representation.

$$c_{IR} = 1 - \frac{nI}{N} \tag{3}$$

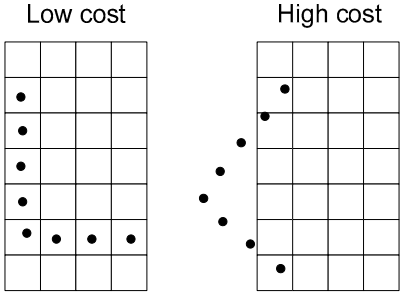

Figure 2. Principle of the cost function c_{IR}. Measurements are highlighted as black dots.

Occupation ratio (OR)

Nearly the same areas of the objects are measured in consecutive measurements. This leads to the cost function of the occupation ratio. It means that newly inserted measurements have to fall into cells already occupied by previous measurements. The more measurements that fall into already occupied cells, the better the results. nO is the sum of all measurements falling into already-occupied cells. The cost function is weighted according to a weighting factor μ. μ is the maximum of occupied cells over all hypotheses of the particle filter. It follows the cost function in Equation (4). Figure 3 shows the schematic representation.

$$e_{OR} = \frac{nO}{\mu} \tag{4}$$

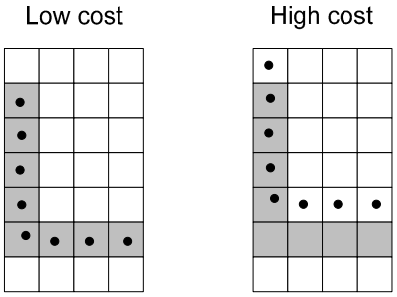

Figure 3. Principle of the cost function c_{OR}. Measurements are highlighted as black dots. Already-occupied cells are highlighted in gray.

Occupation value ratio (OVR)

The assumption that nearly the same areas of the objects are measured in consecutive measurements applies to OVR as well. This requires the cost function to be a maximum (Equation (5)) for the sum of measurements already integrated into the cells where new measurements should be integrated. sI is the sum of all previous measurements. NI is the maximum of the sum of all integrated measurements over all hypotheses of the particle filter. Figure 4 shows the schematic representation.

$$c_{OVR} = 1 - \frac{sI}{NI} \tag{5}$$

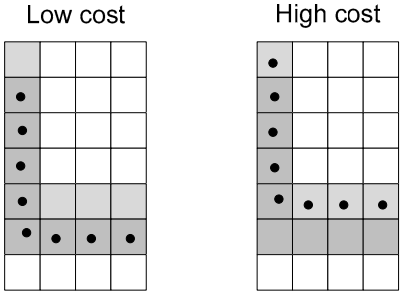

Figure 4. Principle of the cost function c_{OVR}. Measurements are highlighted as black dots. Already-occupied cells with many measurements are highlighted in gray. Cells occupied with fewer measurements are highlighted in lighter gray.

Distance to occupied cells (DTOC)

Unlike the cost functions OR and OVR, DTOC does not only evaluate the exact match of a measurement to an occupied cell. DTOC sums up the shortest (dMin) distances from the measurements to the next occupied cells and weights it with the maximum over all hypotheses (dMax) of the particle filter. The cost function is defined in Equation (6). This is a very costly cost function, as there are a lot of distances to calculate. Figure 5 shows the schematic representation.

$$c_{DTOC} = \frac{dMin}{dMax} \tag{6}$$

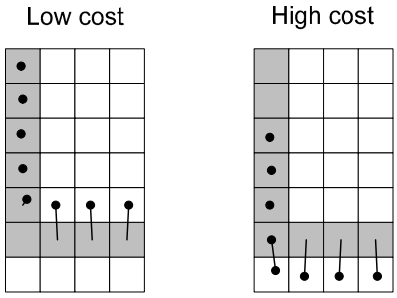

Figure 5. Principle of the cost function c_{DTOC}. Measurements are highlighted as black dots. Already-occupied cells are highlighted in gray. Distances to the next occupied cells are noted by black lines.

Center distance (CD)

The center of gravity cog (x_{cog}, y_{cog}, z_{cog}) of the measurements should lie inside the local grid (Equation (7)). Function inGrid returns '1' if cog lies inside of the grid and '0' otherwise. This cost function emphasizes IR and OR. Figure 6 shows the schematic representation.

$$e_{CD} = \begin{cases} 1 & for \;\; inGrid(cog) == 0 \\ 0 & else \end{cases} \tag{7}$$

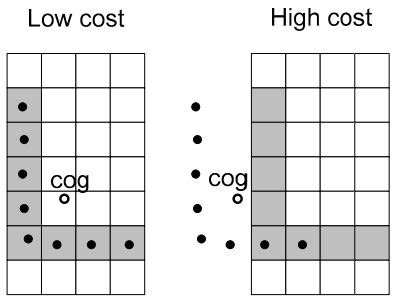

Figure 6. Principle of the cost function c_{CD}. Measurements are highlighted as black dots. Already-occupied cells are highlighted in gray. The center of gravity cog is shown as a black circle.

4.2 Comparison of cost functions and definition of confidence value

The presented cost functions are all designed so they do not need any assumptions of the object accumulating in the grid. As stated previously in the description of the cost functions, some are similar to another. In this subchapter, the cost functions will be compared and the best ones will be selected for further evaluations.

Figure 7 shows the values of the different cost functions for two hundred hypotheses of how to integrate the measurements into the local grid. This means an arbitrary variation from the measurements for the position and rotation of the grid. To allow comparison with the combined value for all cost functions, Figure 7 also shows the mean of all cost functions (Equation 8). This is called the confidence value. The hypotheses are sorted in ascending order according to their confidence value.

$$c = \frac{c_{IR} + c_{OR} + c_{OVR} + c_{DTOC} + c_{CD}}{5} \tag{8}$$

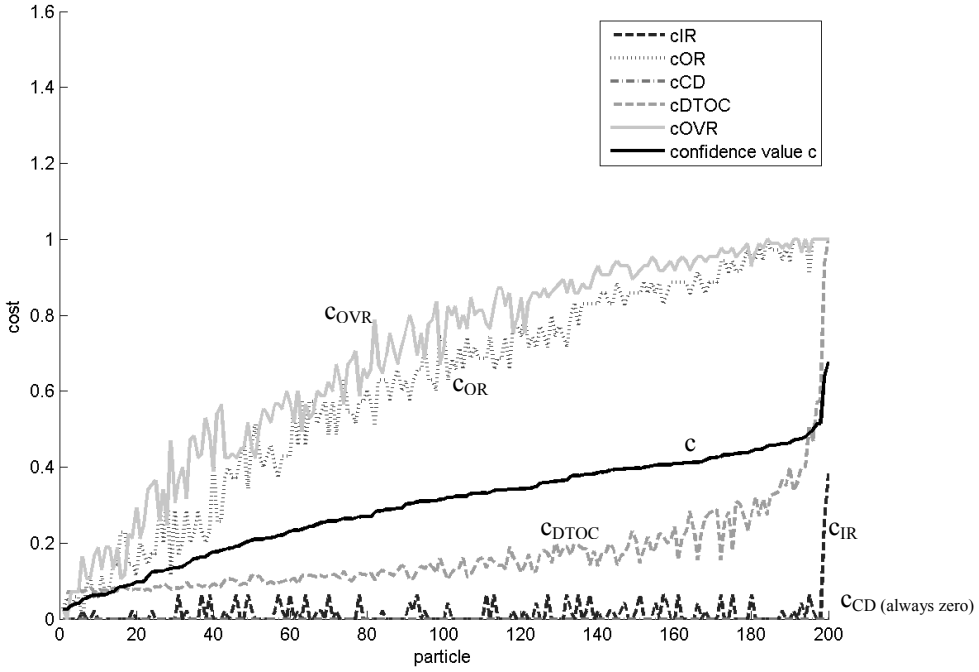

Figure 7. Plot of the cost functions and the confidence value.

The evaluation over several measurement integrations shows that not every cost function influences the confidence value equally. Figure 7 shows that the cost functions c_{OR} and c_{OVR} influence the confidence value the most. The influence of c_{IR} is not very high, but it is important to have a number for how many points are actually integrated. Only these three functions are selected for further processing. c_{CD} seems less important because it is only significant for extreme hypotheses and is never used in this example. c_{DTOC} is not used because it is very costly to calculate.

The reduced confidence value is used for weighting the particles in the particle filter. It is defined as

$$c_R = \frac{c_{IR} + c_{OR} + c_{OVR}}{3} \tag{9}$$

5. MOTION MODEL

All local grids are connected to a motion model. To move each local grid to its next position, a bicycle model approximates the vehicle motion. Figure 8 shows the principle of the bicycle model with the grid and the four wheels of the vehicle. The size of the local grid determines the length l of the bicycle model. Two inputs are used to control the motion: the translational velocity v_R of the rear wheel, and the steering angle φ. The vehicle acceleration and steering wheel angle velocity are modeled by noise factors s_{ar} and s_{vs} which are sampled from distributions introduced in the next chapter. The model constrains the motion of the vehicle and assures that it cannot move sideways. It is assumed that the vehicle only moves on a circular path. M denotes the instant center of rotation. C is the center of gravity of the vehicle, assumed to be in the middle of the local grid at $\frac{l}{2}$ and in global coordinates at (x_C, y_C). v_C denotes the velocity of C. ψ is the orientation of the local grid according to the global coordinate system. It follows that

$$d = \frac{l}{\tan(\varphi)},\ \alpha = \arctan\left(\frac{\tan(\varphi)}{2}\right),\ r = \sqrt{d^2 + \left(\frac{l}{2}\right)^2},\ v_C = \frac{v_R}{d} r = v_R \sqrt{1 + \frac{\tan^2(\varphi)}{4}} \quad (10)\ (11)\ (12)\ (13)$$

$$\dot{x} = v_C \cos(\Psi + \alpha) = v_R \sqrt{1 + \frac{\tan^2(\varphi)}{4}} \cos\left(\Psi + \arctan\left(\frac{\tan(\varphi)}{2}\right)\right) \quad (14)$$

$$\dot{y} = v_C \sin(\Psi + \alpha) = v_R \sqrt{1 + \frac{\tan^2(\varphi)}{4}} \sin\left(\Psi + \arctan\left(\frac{\tan(\varphi)}{2}\right)\right) \quad (15)$$

$$\dot{v}_R = 0 + s_{ar} \quad (16)$$

$$\dot{\varphi} = 0 + s_{vs} \quad (17)$$

$$\dot{\Psi} = \frac{v_R \tan(\varphi)}{l} \quad (18)$$

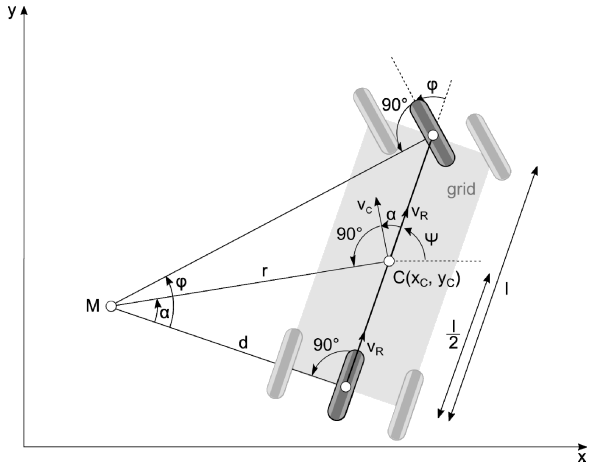

Figure 8. Principle of a bicycle model.

6. VEHICLE DYNAMICS

The evaluation of vehicle dynamics is based on a database consisting of vehicle dynamics measurements over several hundreds of kilometers. The roads used for the database were highways from Italy, Austria and primarily Germany. Some rural roads are also included.

The typical velocity changes are evaluated first. This means that for each vehicle velocity, the probability for the next acceleration within one measurement cycle of the LIDAR needs to be calculated (cycle time: 0.1s). Mostly the distribution is assumed to be Gaussian and some mathematically determined limits are used (e.g. [6]). Figure 9 shows the measured values for the probability of the next acceleration. It shows a histogram where each line (velocity) is normalized so that their sum is one. It also shows that the assumption of Gaussian distributions is applicable. The distributions are used by the particle filter to sample the next probable acceleration for the object.

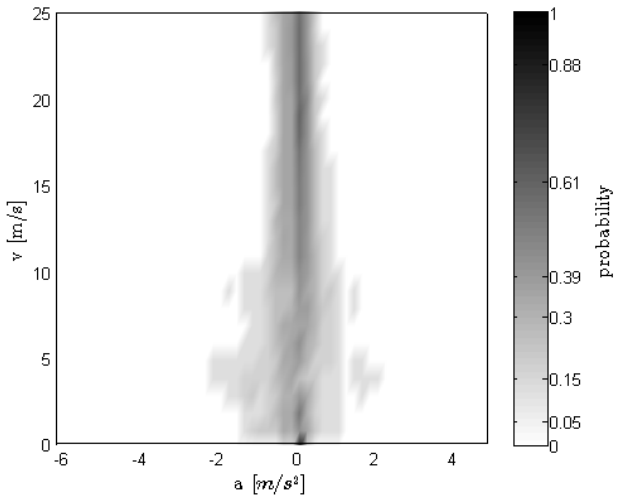

Figure 9. Distribution of the acceleration and deceleration depending on the current velocity

The approach presented above can also be used to evaluate the steering angles. The lateral acceleration a_L is used to calculate the steering angle used for the model in Chapter 5. φ is calculated as follows:

$$\varphi = \operatorname{atan}\left(\frac{l}{v_R^{\,2}} a_L\right) \tag{19}$$

Figure 10 shows the histogram of the steering angle for the current velocity. Again, each line (velocity) is normalized so that their sum is one. According to the Gaussian distribution assumption above, this is valid for the steering angles as well but depends on the current velocity. To use this distribution for the motion model it needs to be divided by the sensor cycle time to get the distribution of the steering wheel angle velocity.

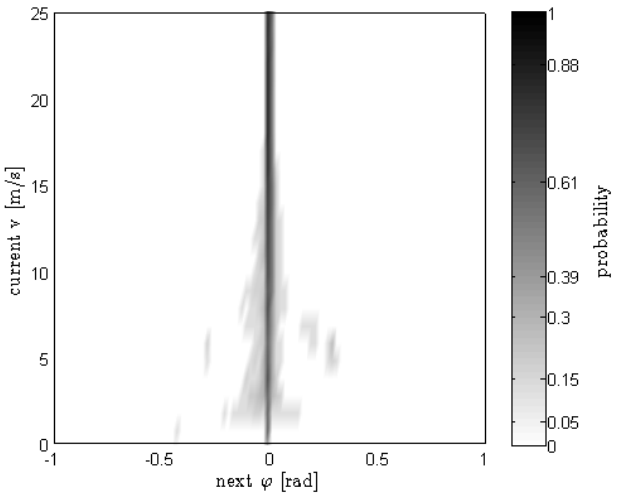

Figure 10. Distribution of the steering wheel angle depending on the current velocity

The histograms can also be used to support the data association step needed when associating measurements to the local grids. This can be done by incorporating the probabilities provided by the histograms.

7. EXPERIMENTS

To show the results of the proposed algorithm, data from a robot car are used for comparison. The robot car is equipped with a high precision inertial measurement unit and a system that controls the steering angle, acceleration and deceleration. It is programmed to drive a maneuver similar to a figure eight. This scenario covers occlusions as well. Since the object is seen from different angles, some portions of the object are occluded at any given time.

During the experiments, it turned out that the special case of driving a trajectory such as a figure eight is not covered in the vehicle dynamics of Chapter 6. Therefore the distribution of the steering angles is extended to cover a wider range of angles at lower velocities.

Because the robot car is black, many laser beams are absorbed and not reflected. Figure 11 shows a measurement example and an image of the robot car.

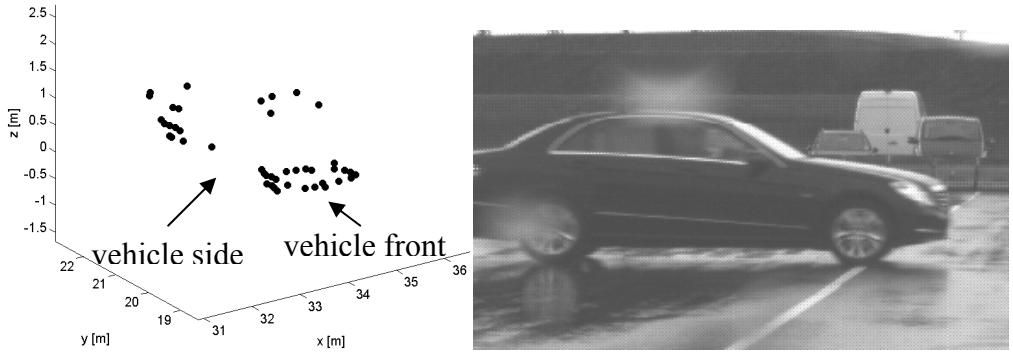

Figure 11. Measurement example and video image of the robot car.

The number of particles for the particle filter is set to 200 and the length, width, and height of the local grid are set arbitrarily to 5 m, 4 m and 2 m, respectively. The length, width and height of the robot car are: 4.8 m, 1.8 m and 1.4 m, respectively.

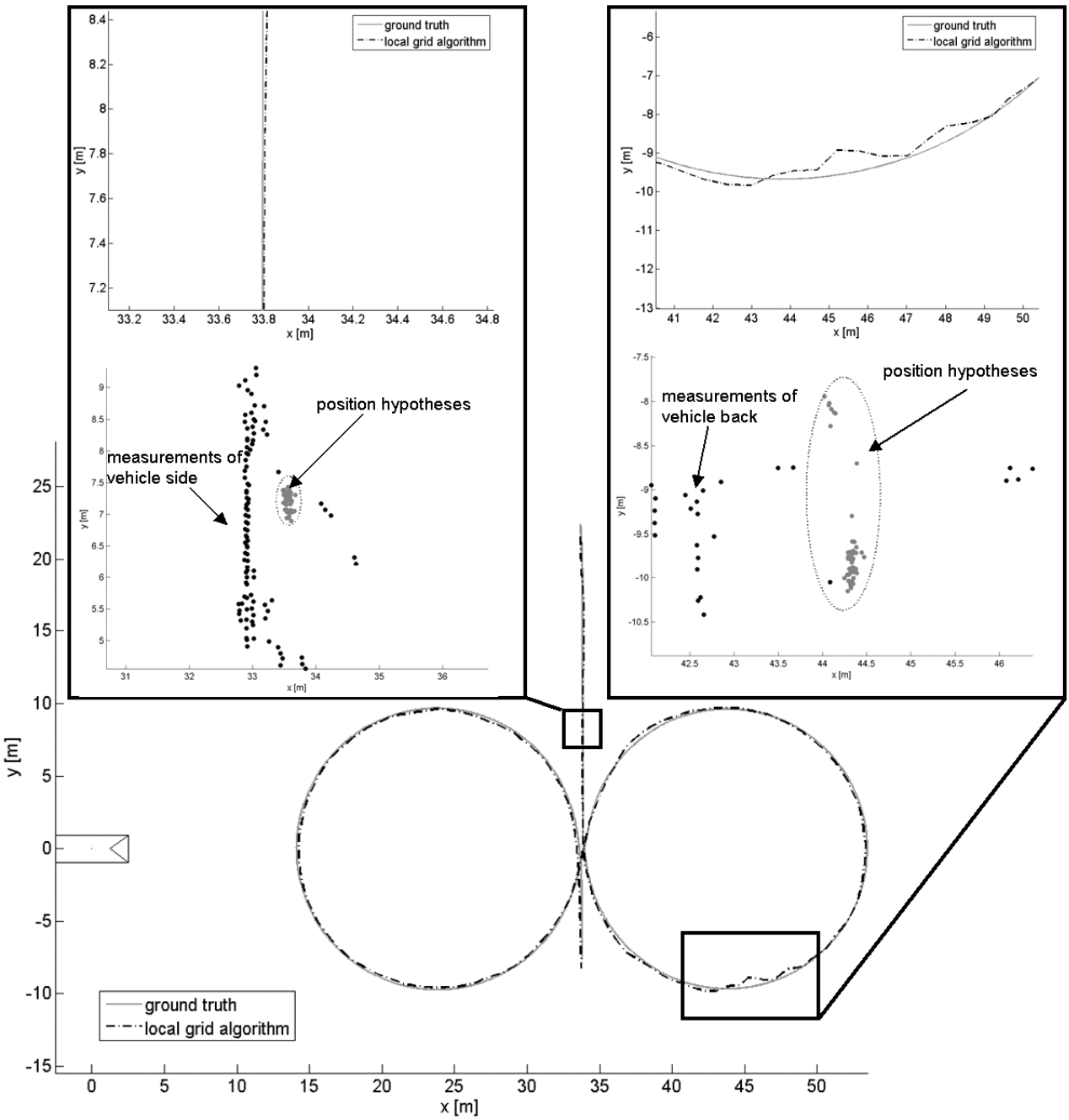

Figure 12. Comparison of the ground truth trajectory with the estimated trajectory and two exemplary distributions of the position hypotheses.

Figure 12 shows the comparison of the estimated position of the object with the ground truth of the inertial measurement unit. The sensor is at position (0, 0). The mean error is 0.11 m. A larger position error is detected at position (45, -9). The evaluation of the raw data shows that few measurements from the back of the vehicle are reflected. It is possible to integrate them with several hypotheses. This is represented by a high distribution of the particles expressing the uncertainty.

Figure 13 compares the estimated velocity with the ground truth of the inertial measurement unit. The mean error is 0.18 m/s.

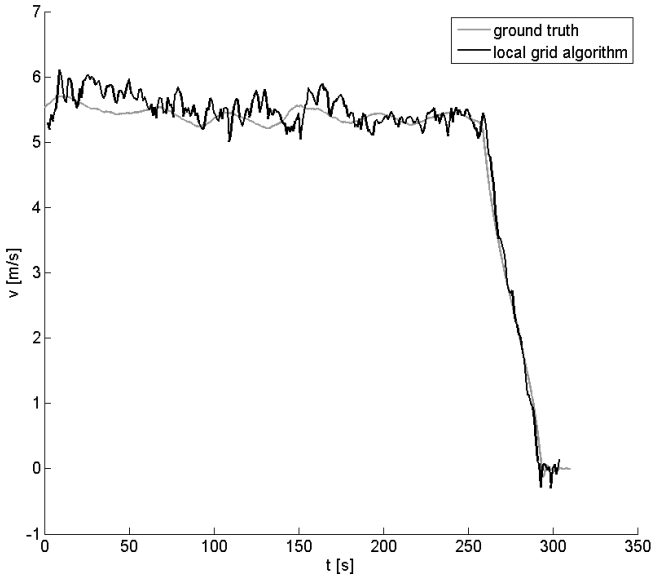

Figure 13. Comparison of the ground truth velocity with the estimated velocity.

In Figure 14, the accumulated measurements of the local grid are shown. A threshold is used to show only cells above a predefined value.

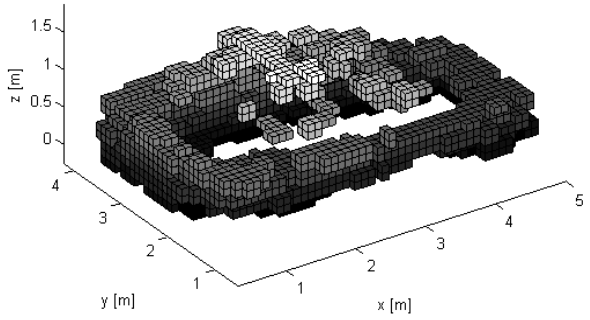

Figure 14. Accumulated measurements in the local grid.

8. CONCLUSION

A method for tracking extended targets with local 3D grids has been proposed. The advantage of using local 3D grids is that no geometric model hypotheses are needed to track and accumulate 3D-LIDAR measurements. The method itself is based on a particle filter. The probabilistic motion used for the particle filter is derived from a database of real vehicle dynamics data. Different cost functions for weighting the particles have been evaluated.

The experiments show that the method is capable of tracking targets even in difficult scenarios. The estimation of the position as well as of the velocity is very accurate.

Future work will include dealing with multiple objects and improving the performance for real-time usage.

REFERENCES

[1] Effertz, J., "Autonome Fahrzeugführung in urbaner Umgebung durch Kombination objekt- und kartenbasierter Umfeldmodelle", PhD thesis, Technische Universität Carolo-Wilhelmina zu Braunschweig (2009).

[2] Elfes, A., "Using Occupancy Grids for Mobile Robot Perception and Navigation", IEEE Computer 22(6), 46-57 (1989).

[3] Eliazar, A., "DP-SLAM", PhD thesis, Duke University (2005).

[4] Himmelsbach, M., Müller, A., Lüttel, T. and Wünsche, H.-J., "LIDAR-based 3D Object Perception", Proceedings of 1st International Workshop on Cognition for Technical Systems (2008).

[5] Isard, M. and Blake, A., "CONDENSATION—Conditional Density Propagation for Visual Tracking", International Journal of Computer Vision 29, 5-28 (1998).

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

[7] Moosmann, F. and Fraichard, T., "Motion Estimation from Range Images in Dynamic Outdoor Scenes", Proceedings of the IEEE International Conference on Robotics and Automation, 142-147 (2010).

[8] Schmid, M. R., Maehlisch, M., Dickmann, J. and Wuensche, H.-J., "Dynamic Level of Detail 3D Occupancy Grids for Automotive Use", Proceedings of IEEE Intelligent Vehicles Symposium, 269-274 (2010).

[9] Thrun, S., Burgard, W. and Fox, D., [Probabilistic Robotics] (Intelligent Robotics and Autonomous Agents series), The MIT Press (2005).

[10] Yang, C. and Medioni, G., "Object modelling by registration of multiple range images", Image and Vision Computing 10(3), 145 - 155 (1992).