Simultaneous Tracking and Shape Estimation with Laser Scanners

Markus Schütz∗, Nils Appenrodt†, Jürgen Dickmann‡, Klaus Dietmayer§

∗Daimler AG, Ulm, Germany
Email: markus.m.schuetz@daimler.com
†Daimler AG, Ulm, Germany
Email: nils.appenrodt@daimler.com
‡Daimler AG, Ulm, Germany
Email: juergen.dickmann@daimler.com
§University of Ulm, Germany
Email: klaus.dietmayer@uni-ulm.de

Abstract—Advanced driver assistance systems and the environment perception for autonomous vehicles will benefit from systems robustly tracking objects while simultaneously estimating their shape. Unlike many recent approaches that represent object shapes by approximated models such as boxes or ellipses, this paper proposes an algorithm that estimates a free-formed shape derived from raw laser measurements. For that purpose local occupancy grid maps are used to model arbitrary object shapes. Beside shape estimation the algorithm keeps a stable reference point on the object. This will be important to avoid apparent motion if the observable part of an object contour changes. The algorithm is part of a perception system and is tested with two 4-layer laser scanners.

I. INTRODUCTION

Today’s dense traffic situation requires more and more the support of advanced driver assistance systems. An improved environment perception, especially the detection and tracking of the surrounding objects, becomes crucial for multiple safety applications.

Tracking algorithms, that are designed to estimate the dynamic states of measured objects, are a well-explored field in literature. The common approach is to use the Gaussian approximation to model the uncertainty of the state and use a Gaussian filter for estimation, e.g. a standard or an extended Kalman filter. Those filters are generally used to infer dynamic states of the objects, such as the position and the velocity of objects, from noisy measurements. In those tracking systems a single measurement updates a track. Since a laser scan provides usually several reflections per object, it is crucial to find a suitable common reference point for all these reflections that represents the object’s dynamic state best. A common approach is to use the mean value of these reflections as a measurement to update the track. Fig. 2 shows a major problem of those approaches: the moving reference point that leads to an apparent motion when the observable part of an object contour changes e.g. when the perspective changes from that an object is seen. More advanced approaches search for stable features of an object such as corners and use one of these as the reference point (that must be able to change suddenly when the object turns in a way that this corner is not visible anymore). These model-based approaches are mostly used for a specific kind of object, such as cars. For example box-models are often used to describe the shape of a car. Beside the use of the corners as stable features, this model also allows to estimate the size of the vehicle. Often, this box-model assumption is a suitable approximation. However, there exist scenarios in which a more accurate representation of the shape is important, especially when no specific types of objects can be expected, but accurate shapes have to be estimated.

An important application for this requirement are pre-crash scenarios. Here it is crucial to detect if an accident is unavoidable. A bounding box model easily overrates the danger in such cases and produces false alarms. Furthermore, a tracking system that is not restricted to cars and its typical shape is eligible.

The presented approach in this paper aims to tackle these two problems: on the one side keeping a stable reference point to achieve accurate tracking results and on the other side estimating a detailed shape that allows to overcome limitations caused by abstract approximations.

To achieve these goals, we use occupancy grid maps to store associated measurements and represent the shape of the object. The occupancy grids are normally used to map the static environment of a vehicle or robot. In contrast to this application the occupancy grids in the presented approach are fixed to the objects, i.e. they are object-local.

This approach entails the limitation that the shape of the objects has to be static while tracking. The major benefit is the complexity of shapes that can be accurately modeled. Furthermore, a fixed anchor point on the shape is available that prevents the errors caused by a moving reference point as shown in Fig. 2.

Two Sick LD-MRS400001 laser scanners, mounted at the front of our test car (see Fig. 1), provide the input data for the developed algorithm.

Cooperative traffic (e.g. Vehicle-2-vehicle communication) is not considered since the system is designed to work with generic shaped objects that does not necessarily provide this data (bicycles, pedestrians, common vehicles). Also a highly accurate inertial navigation platform fused with a D-GPS
II. RELATED WORK

Different attempts have been made in literature to combine occupancy grid maps and tracking algorithms on an object level. While occupancy grid maps, first developed by Elfes [1], are a good approach to model static environments on a raw level of abstraction, tracking algorithms estimate the dynamic state of moving objects on a high level.

The combination of these two models is mostly achieved by making the grid cells dynamic, e.g. by storing velocity data in each grid cell. Examples for these kind of approaches can be found in Coué et al. [2], where a 4-D occupancy grid map (position and velocity in 2-D) is developed. The beside the occupation of each cell, the velocity components can be estimated. Also, Danescu et al. [3] shows an approach where the occupation within each cell consists of particles that can move from cell to cell. Tracking is based on the estimated particle motion.

Instead of a global grid map with dynamic information contained in its cells, the approach we propose aims to move the whole grid assigned to an object. The following works also build up a shape or map that is moved over time. Effertz [4] uses a local map to achieve the contour of an object. Instead of integrating the associated raw sensor data, the data is abstracted and the resulting polygon is integrated into a map. This approach needs an additional association step since a feature map is created. Steinemann et al. [5] use a highly accurate Velodyne HDL-64e 3D-laser scanner to accumulate a local map of a passenger car in 3D. Several deterministic cost functions are combined to determine the best object estimate. Moosmann et al. [6] accumulate a local point cloud using the well-known optimization algorithm ICP. New sensor data is matched to already accumulated data of the object from the past in a so-called refinement step. The resulting translation and rotation of this refinement step is used as an input to the Kalman filter update step.

III. PERCEPTION SYSTEM

The tracking system is integrated in the perception system presented at [7].

The tracking is accomplished in the vehicle coordinate system which has its origin at the center of the rear wheel axis, with the x-axis to the front and the y-axis to the left of the vehicle.

A localization module based on a global occupancy map improves the proprioceptive measurements provided by the ego vehicle. The laser scanners has four layers with a vertical field of view of $3.2^\circ$ between the topmost and lowermost layer. Not all of them can be used, because of the pitch and roll motion of the ego vehicle while driving and the unevenness of the road. Because this lack of 3D-information, the tracking is performed in two dimensions.

The measurements used to track the objects first have to be distinguished from non-relevant data. Literature does not regard the extraction of dynamic objects as trivial, however several approaches can be found to cope with the detection of dynamic objects, e.g. [8], [9], [10].

This work assumes that these measurements caused by an object are associated correctly. However the algorithm tends to be robust against association error to a certain degree. For experiments the global nearest neighbor association was applied after segmenting the data with the DBSCAN algorithm in polar coordinates.

IV. MATHEMATICAL APPROACH

Each object can be thought of to be divided into two parts. On the one hand a dynamic state, which is similar to the state of a standard tracking algorithm. It describes the state vector that usually changes over time, such as the position and the velocity. This part will be described in IV-A. On the other hand there is the static part, consisting of the shape and the offset of the anchor point and the center of gravity of the object. Static in this context means that the true shape and offset of the real object are constant, i.e. do not change over time, but are initially unknown and have to be estimated. This part is described in IV-B.

A. Dynamics Model

As stated in Sec. IV the modeling of a tracked object is divided into a dynamic state representation and a static shape representation.

The dynamic part consists of the following state variables

$$X_k = [x_k, y_k, \Psi_k, v_k, \omega_k]^T$$  \(1\)
where \( k \) denotes the current time step. \( x_k, y_k \) and \( \Psi_k \) refer to the anchor point’s position and orientation of the vehicle in the 2-D plane. \( v_k \) is the velocity in the movement direction \( \Psi_k \) at the center of gravity (see Fig. 3). Orientation and direction of the velocity coincide in this model. Its yaw-rate is denoted by \( \omega_k \).

A Constant-Turn-Rate/Constant-Velocity-model is used to predict the motion of the object. Since the anchor coordinates \( x_k \) and \( y_k \) can eventually be positioned in arbitrary relationship to the object, they cannot serve as the representation point of the motion. To undertake the prediction, the position of the center of gravity has to be calculated, by recalculating it using the offset values \( c_{x,k} \) and \( c_{y,k} \) that are also estimated by the system (see Sec. IV-B2). Since the offset values are kept in object local coordinates, the calculation of the center of gravity is defined by:

\[
\begin{bmatrix}
\hat{x}_k \\
\hat{y}_k
\end{bmatrix} =
\begin{bmatrix}
x_k \\
y_k
\end{bmatrix} +
\begin{bmatrix}
\cos \Psi_k & -\sin \Psi_k \\
\sin \Psi_k & \cos \Psi_k
\end{bmatrix}
\begin{bmatrix}
c_{x,k} \\
c_{y,k}
\end{bmatrix}
\]

(2)

The actual prediction is then performed by evaluating the following equations if it can be assumed that \( \omega_k > 0 \)

\[
\begin{align*}
\hat{x}_{k+1} &= \hat{x}_k + \frac{v_k}{\omega_k} (\sin (\Psi_k + \omega_k \Delta T) - \sin (\Psi_k)) \\
\hat{y}_{k+1} &= \hat{y}_k + \frac{v_k}{\omega_k} (\cos (\Psi_k) - \cos (\Psi_k + \omega_k \Delta T)) \\
\Psi_{k+1} &= \Psi_k + \omega_k \Delta T \\
v_{k+1} &= v_k \\
\omega_{k+1} &= \omega_k
\end{align*}
\]

(3) - (7)

For very small \( \omega_k \) the model degenerates to a constant-velocity model.

\[
\hat{x}_{k+1} = \hat{x}_k + v_k \Delta T \cos (\Psi_k) \\
\hat{y}_{k+1} = \hat{y}_k + v_k \Delta T \sin (\Psi_k) \\
\Psi_{k+1} = \Psi_k + \omega_k \Delta T \\
v_{k+1} = v_k \\
\omega_{k+1} = \omega_k
\]

(8) - (12)

After the evaluation the anchor point is calculated by inverting Eq. (2).

Additionally the states are ego motion compensated, i.e. any motion resulting from the moving ego vehicle is removed. The position and orientation are represented in the ego vehicle coordinate system, the velocity and yaw rate are represented over ground.

B. Shape Model

Object shapes can differ in many aspects and generally can be complex.

In our approach the object’s shape model \( S_k = (M_k, C_k) \) consists of a local occupancy grid map

\[
M_k = \{m_1, m_2, ..., m_N\}
\]

(13)

consisting of grid cells \( m_i \) and an offset vector

\[
C_k = \begin{bmatrix} c_{x,k} & c_{y,k} \end{bmatrix}^T
\]

(14)

that is described in Sec. IV-B2.

1) Occupancy Grid Map: The shape of the object at time \( k \) is represented by the occupancy map \( M_k \).

Shape in this context means the contour of an object seen from the top. For automotive applications the height of an object is not important.

The mapping approach allows various free-form shapes, while not demanding more computational resources with increasing complexity of the object shape. Predefined features in the model are not required since the raw sensor data is accumulated.

Major drawback is that shapes have to be inflexible while tracking, i.e. the shape is assumed to be static. However the algorithm can adapt to small changes over time.

In a occupancy grid map, a 2D-space (normally the environment of the vehicle) is divided into equally sized cells. Each cell is a probabilistic variable describing if the occupation of this cell and is independent from its neighbors.

In this approach the map is used in the same way with the difference that the origin of the map is fixed to car at the coordinates \( x_k \) and \( y_k \) of the object and it is rotated by \( \Psi_k \).

Assuming \( M_k \) is the grid map at time \( k \) and \( m_i \) denotes a grid cell with index \( i \) \( (i = 1, .., N) \), and the grid cells are independent to one another, the occupancy grid map can be modeled as a posterior probability:

\[
P(M_k | z_{1:k}, X_{1:k}) = \prod_i P(m_i | z_{1:k}, X_{1:k})
\]

(15)
where \( P(m_i|Z_{1:k}, X_{1:k}) \) is the inverse sensor model that describes the probability of occupation given the measurements \( Z_{1:k} \) and the dynamic object state \( X_{1:k} \) at all time steps until time \( k \). Each measurement consists of \( n \) reflexion points \( Z_i = \{ z_{1,i}, z_{2,i}, \ldots, z_{n,i} \} \). As stated at the beginning of section IV it is assumed that the measurements \( Z_{1:k} \) induced by the object are correctly segmented and associated.

The occupancy values of each cell are calculated by a binary Bayes filter. In practice, the log odds ratio is used to integrate new measurements efficiently. The log odds occupancy grid map is formalized as:

\[
L_k(m_i) = \log \frac{P(m_i|Z_{1:k}, X_{1:k})}{1 - P(m_i|Z_{1:k}, X_{1:k})}
\]

The recursive formulation of map update in log odds ratio is given by [11]:

\[
L_k(m_i) = L_{k-1}(m_i) + \log \frac{P(m_i|Z_{1:k}, X_{1:k})}{1 - P(m_i|Z_{1:k}, X_{1:k})} - L_0(m_i)
\]

where \( L_{k-1}(m_i) \) and \( L_0(m_i) \) are the previous and prior log odds values of grid cell \( i \).

Instead of performing multiplications, the usage of the log odds ratio simplifies the calculation to additions and avoids instabilities of calculating probabilities near zero or one. Assuming that no prior knowledge is available, the prior probability of unknown cells is set to \( P_0(m_i) = 0.5 \), the above equation produces the prior log odds ratio \( L_0 = 0 \).

To obtain the correspondiung probability from the log odds representation again, Eq. (16) can be inverted.

2) Offset: As stated in IV-A, \((x_k, y_k)\) describes a fixed anchor point on the tracked object that does not change over time. While this is desirable in order to avoid apparent motion, the direct use of this point as the dynamic reference point is not a good choice. The best reference point for arbitrary objects is the center of gravity.

So, to apply the motion dynamics, \( c_{x,k} \) and \( c_{y,k} \) describe offsets between that fixed point and the estimated center of gravity of the object (see also Fig. 3) in object coordinates, with the \( x \)-axis in the movement direction and the \( y \)-axis to the left.

C. Measurement Model

This section describes how we obtain the measurement likelihood \( p(Z_k|X_k, S_k) \) given the object state \( X_k \), the object’s shape \( S_k \) and the current laser data \( Z_k \) that was associated with the object. Although possible, we currently do not consider the rays from the scanner to the laser reflexion centers and mark the space in between the laser scanner and the reflexion as free space, as it is commonly done. This is mainly due to the heavy computational load the tracing of each ray would imply.

Also as seen in Sec. VI-B this assumption is not always valid. However it is planned to evaluate possible benefits in the future.

The measurement likelihood is computed by factoring it under the assumption of independent laser measurements.
and was introduced in [13].

Binary problem. This approach is called Rao-Blackwellisation. Is split up into independent cells, where each cell is now a simplification of the very high-dimensional $S$. To use the multi-modal properties of the particle filters. To take these conditions into account a particle filter for the Bayesian estimation is used and methods like the EKF (Extended Kalman Filter) or UKF (Unscented Kalman Filter) are not suitable.

For particle filtering the condensation algorithm proposed in [12] is used. Overall, we have to estimate $X_k = [x_k \ y_k \ \Psi_k \ v_k \ \omega_k]^T$, and $S_k = (M_k, C_k)$, which means several hundred states have to be estimated for comparatively big objects and a appropriate cell size of the map.

The number of particles needed grows exponentially with the number of dimensions of the estimation problem, while the dimensionality of the presented problem is very high. Because of that the estimation problem is split.

The dynamic states $x, y, \psi, v, \omega$ are estimated via sampling, to use the multi-modal properties of the particle filters. To simplify the estimation of the very high-dimensional $S_k$, it is split up into independent cells, where each cell is now a binary problem. This approach is called Rao-Blackwellisation and was introduced in [13].

So each particle $q_k^j$ consists of a dynamic state and a shape:

$$ q_k^j = (X_k^j, S_k^j) $$

(22)

The weights $w_k^j$ of the particle are drawn according to

$$ w_k^j = w_{k-1}^j P(Z_k|X_k^j, S_k^j) $$

(23)

Currently the offset is modeled as a deterministic variable, i.e. it is not estimated in a probabilistic manner.

It is difficult to detect the completeness of the shape seen so far and it is impossible to extract the weight distribution of the real car from the laser measurement. For this reason there is no valid foundation to assume a measurement uncertainty to use for example a Kalman filter to estimate the parameter.

Instead the best guess is to extract a mean point of the occupied cells. The occupied cells in $M_k$ provide information about the center of gravity to update the offset vector $C_k$. We assume that the mean value of the occupied cells is always the best guess for the center of gravity for a free-form object. So in each time step a bounding box is calculated around occupied cells whose occupation lies beyond a certain threshold. The mean point of this bounding box is considered as a measurement for the current offset to the center of gravity $C_k$. To smooth this output position a moving average calculation over the last $L$ measurements is performed by applying

$$ C_k = \frac{1}{L} \sum_{i=1}^{L} \hat{C}_{k-L+i} $$

(24)

V. OBJECT TRACKING

The environment perception framework presented in [7] integrates the object tracking algorithm presented in this paper. Modules for the detection of moving objects and the association of new segmented measurements are available. A localization module provides the velocity and yaw-rate of the ego-vehicle.

The object tracking itself works according to the following scheme.

After the detection module provides data of a new object, it is initialized via a two step-initialization to obtain an initial velocity $v_0$ and an initial orientation $\Psi_0$. The initial states of the particles are sampled from a Gaussian prior with the mean value of $v_0$ and $\Psi_0$. The variances of the priors depend on accuracy of the initialization. They are obtained using an experimental method. The accuracy of the initialization directly influences the number of particles that have to be used. The better the initial velocity and orientation the smaller the variances can be set and the fewer particles are needed at the beginning.

The anchor point is set to the mean value of the associated laser scanner reflexions in $x$- and $y$-direction. With each new scan the object state is predicted via the model described in section IV-A. If new measurements could be associated with the object, the shape grid map $M_k$ is first used to evaluate the weight of the particle. After that, $M_k$ is updated using the measurement model from section IV-C. From the shape, we derive a “measurement” for the estimation of the center point $C_k$, and update it.

At the moment, we do not use any logic to merge two objects, since we found that this scenario was rare in our data, although this fact heavily depends on the segmentation algorithm that is used.
The main advantage of the proposed algorithm can also be a drawback in the context of applications that are using the tracking output.

The object shape represented by a grid map describes the real shape on a raw level. However most follow-up modules depend on abstract shape descriptions such as boxes or an other polygonal line. To make the algorithm compatible to those requirements an abstract shape has to be extracted from the grid map. For that purpose the convex hull around cells whose occlusion lie above a certain threshold. This approach restricts the representable shape to convex shapes. The threshold depends on the inverse sensor model and the intended level of a safety margin. The lower the threshold, the larger the area the object covers.

VI. EXPERIMENTAL RESULTS

Two Sick LD-MRS400001 laser scanners, mounted at the front of our test car provide the input data for the developed algorithm. A Velodyne HDL-64e laser scanner is mounted at the roof and is used for evaluation purposes only. Data was recorded at an open area with one target car and static obstacles as well as in urban scenarios.

In the scenarios an oncoming vehicle is chosen exemplary as the main tracked object since its considerable extended shape, that differs from a box mainly at the front.

A. Scenario 1

Fig. 6 shows the local grid map of an oncoming Mercedes-Benz C-Class Estate turning to its rights in the field of view. White cells denote unknown areas, while black color denotes occlusion. The right side (which is the upper side in Fig. 6) of the car is detected for a very short period of time at the beginning of the scenario. The back of the car is seen for a short time at the end of the scene. This scenario is intended to show the general performance of the algorithm in estimating the shape of an object.

The cell size is set to an edge length of 5 cm, which is very fine grained compared to standard settings, where 10-20 cm are set. While this is computationally very demanding. It shows well the performance of the tracking and shape estimation algorithm, since almost every area of the target cars surface is measured at some frames.

The estimated size that can be extracted from the shape grid map depends on the threshold and the desired safety margin. For this scenario 100 particles are used.

B. Scenario 2

In the second scenario an oncoming vehicle is tracked. This scenario is more realistic, since oncoming traffic is very common in the real world. The proposed tracking algorithm is compared to a standard Kalman filter approach, where the mean point of the associated measurements is used as a reference point. For both algorithms the borders of the field of view are not modeled in the sensor model. So only associated raw data is used as the algorithms’ input.

This scenario is intended to evaluate, beside the estimation of the shape of the object with a larger grid cell size, mainly

![Fig. 6. Local grid map of an oncoming Mercedes-Benz C-Class Estate turning to its rights in the field of view. White cells denote unknown areas, while black color denotes occlusion. The right side of the car is detected for a very short period of time at the beginning of the scenario.](image-url)
the measurements moves with half the actual velocity in x-direction.

The estimated velocity of the proposed algorithm also decreases slightly. That is because the uncertainty increases.

A second effect that cannot be seen in the figure is that a velocity component in y-direction towards the ego-vehicle is estimated, since the front of the object disappears while the detectability of the side of the object rises. This leads to a movement towards the ego-vehicle and can lead to critical false positives for crash detections.

VII. CONCLUSIONS

We have presented a new concept for tracking dynamic extended objects while simultaneously estimating their shape. Experimental results with real sensor data showed promising results, although there are further investigations needed when there is more data with corresponding ground-truth available. The algorithm and the implementation has to be improved to reach real-time requirements by applying shared maps between particles and parallelization of the particle calculations. Since the initialization is crucial to decrease the particle number a radar-based initialization routine will be implemented. Furthermore radar will be used to improve the tracking results by fusing laser scanner with radar data while especially taking into account the shape estimated by the laser scanners. Probabilistic state-of-the-art association algorithms will help to improve the tracking algorithms in crowded urban traffic scenarios.

ACKNOWLEDGMENT

The methods presented in this paper were developed within the integrated project interactIVe, a European research activity that contributes to accident avoidance by active intervention for Intelligent Vehicles.

The work is part of the research activities of the Daimler AG to develop novel environment perception systems.

REFERENCES