

# Road Intensity Based Mapping Using Radar Measurements With a Probability Hypothesis Density Filter

Christian Lundquist, Lars Hammarstrand, and Fredrik Gustafsson, *Senior Member, IEEE*

**Abstract**—Mapping stationary objects is essential for autonomous vehicles and many autonomous functions in vehicles. In this contribution the probability hypothesis density (PHD) filter framework is applied to automotive imagery sensor data for constructing such a map, where the main advantages are that it avoids the detection, the data association and the track handling problems in conventional multiple-target tracking, and that it gives a parsimonious representation of the map in contrast to grid based methods. Two original contributions address the inherent complexity issues of the algorithm: First, a data clustering algorithm is suggested to group the components of the PHD into different clusters, which structures the description of the prior and considerably improves the measurement update in the PHD filter. Second, a merging step is proposed to simplify the map representation in the PHD filter. The algorithm is applied to multi-sensor radar data collected on public roads, and the resulting map is shown to well describe the environment as a human perceives it.

**Index Terms**—Clustering, Gaussian mixture, mapping, PHD, probability hypothesis density, road edge estimation.

## I. INTRODUCTION

**A**UTONOMOUS vehicles and autonomous functionality in vehicles require situation awareness. Situation awareness is traditionally split into two main tasks: tracking of moving objects and navigation relative to stationary objects represented by a map. The precision in commercial maps and position services today is not sufficient for autonomous functions. For that reason, imagery sensors as radar, vision and laser scanners have been used to build local maps of stationary objects on the fly, which requires sophisticated estimation algorithms. Similarly to the classic target tracking application, going from single-sensor single-object estimation to multi-sensor multi-object estimation increases the complexity of the algorithms substantially. For

that reason, there is a strong need for structured maps rather than point maps, where single objects that together form structures as lines or curves are described as one extended object. This contribution describes an efficient algorithm to create maps that are particularly useful to describe structures along road sides. Such a map is promising for a range of applications, from collision avoidance maneuvers to true autonomy on public roads and off-road.

There exist many algorithms and representations of maps using imagery sensor data. However, so far, methods based on probabilistic theory are most successful; see, e.g., [1] or [2] for a recent overview. Feature based maps have become very popular in the related field simultaneous localization and mapping (SLAM) [3], [4], where a vehicle builds a map of environmental features, while simultaneously using that map to localize itself. Location based maps, such as the occupancy grid map (OGM) [5], were primarily developed to be used with laser scanners to generate consistent maps. They have also been used with radar sensors, where the data comprises a complete signal power profile [6], [7], or a thresholded radar report [8]. The OGM was very popular at the DARPA urban challenge; see the special issues [9]. Generally, line shaped and curved objects, such as roads and coast lines can be identified in topographic maps. Road edge and obstacle detection have been tackled using raw radar images; see, e.g., [10]–[13]. There have been several approaches making use of reflections from the road edges, such as guard rails and reflection posts, to compute information about the free space; see, e.g., [14]–[16] for some examples using laser scanners. Radar reflections were used to track the road edges as extended targets in [17]. This method is promising but a drawback is the large data association problem, which arises since it is hard to create a general model for the various types of objects.

The bin occupancy filter is devised via a quantized state space model, where each cell is denoted bin [18]. In the limit, when the volume of the bins become infinitesimal, the filter equations are identical to the probability hypothesis density (PHD) filter, proposed by Mahler [19]–[21]. The PHD is the first moment density or intensity density of a random set. In Section II, we propose to represent the map by the surface described by the intensity density of point sources of stationary objects. Individual sources are not tracked, but the PHD filter framework makes it possible to estimate the intensity of sources in a given point. To describe the map by a continuous surface distinguishes our method from the other mapping approaches mentioned above. An approach to make use of the PHD filter to solve the related

Manuscript received March 25, 2010; revised July 22, 2010, November 10, 2010; accepted December 07, 2010. Date of publication December 30, 2010; date of current version March 09, 2011. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Wing-Kin Ma. The authors would like to thank the SENSOR Fusion for Safety (SEFS) project within the Intelligent Vehicle Safety Systems (IVSS) program for financial support.

C. Lundquist and F. Gustafsson are with the Department of Electrical Engineering, Division of Automatic Control, Linköping University, SE-581 83 Linköping, Sweden (e-mail: lundquist@isy.liu.se; fredrik@isy.liu.se).

L. Hammarstrand is with Volvo Car Corporation, Active Safety Electronics, SE-405 31 Göteborg, Sweden (e-mail: lhammar5@volvocars.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TSP.2010.2103065

SLAM problem is presented in [22]. In contrast to the intensity based map, the map in [22] is represented by a finite set of map features, which are extracted from the intensity density.

Section III summarizes the Gaussian mixture PHD (GM-PHD) filter recursion. We investigate the possibility to improve the efficiency of the GM-PHD filter by utilizing structure in the map. These structures are first identified and modeled in Section IV, to then be used to improve the prior in the GM-PHD recursion and to create an efficient representation of the intensity in Section V. Road edges and guardrails are typical examples of such map structures. All parts of the road mapping example are put together in Section VI to present the complete map, based on measurements from automotive radar sensors. The map may be used by automotive safety functions, for instance trajectory control, which aims at minimizing the probability to hit objects. Other examples are functions which can perform automatic evasive maneuvers, and need map information for threat assessment [23]. The conclusions are in Section VII.

## II. MAPPING

Sensors that measure range  $r$  and bearing  $\psi$ , e.g., radar and laser, are commonly used for automotive and robotics navigation and mapping. These type of sensors typically do not give the user access to their detection and decision parameters. Automotive radar sensors threshold the amplitude and deliver only those detections with the highest amplitude, whereas a laser sensor delivers the first range detection along a bearing angle. Furthermore, cameras measure two angles and represent them in a pixel matrix. The considered sensors provide a set of noisy measurements (thresholded detections)

$$\mathbf{Z}_k = \{\mathbf{z}_k^{(1)}, \mathbf{z}_k^{(2)}, \dots, \mathbf{z}_k^{(N_{z,k})}\} \quad (1)$$

at each discrete time instant  $k = 1, \dots, K$ .

Landmark detections from the noisy sensor data are used to build a probabilistic map, represented by a set of  $N_m$  objects in the environment

$$\mathbf{M}_k = \{\mathbf{m}_k^{(1)}, \mathbf{m}_k^{(2)}, \dots, \mathbf{m}_k^{(N_m)}\}. \quad (2)$$

There exists primarily two types of indexing for probabilistic maps [1]. In a *feature based map* each  $\mathbf{m}^{(n)}$  specifies the properties and location of one object [24], whereas in a *location based map* the index  $n$  corresponds to a location and  $\mathbf{m}^{(n)}$  is the property of that specific coordinate. Note that the indexing is analogous to the representation of sensor data, the range and bearing measurements from a radar or laser are feature based, whereas the pixels of a camera are location based measurements. The *occupancy grid map* is a classical location based representation of a map, where each cell of the grid is assigned a binary occupancy value that specifies if the location  $n$  is occupied ( $\mathbf{m}^{(n)} = 1$ ) or not ( $\mathbf{m}^{(n)} = 0$ ) [5], [25]. The aim of all stochastic mapping algorithms, independent of indexing, is to estimate the posterior density of the map

$$p(\mathbf{M}_k | \mathbf{Z}_{1:k}) \quad (3)$$

given all the measurements from time 1 to  $k$ .

The bin-occupancy filter, which is described in [18], aims to estimate the probability of a target being in a given point. The approach is derived via a discretized state-space model of the surveillance region, where each grid cell (denoted bin in this approach) can or may not contain a target. One of the important assumptions in [18] is that the bins are sufficiently small so that each bin is occupied by maximum one target. In the limiting case, when the volume of the bins  $|\nu|$  goes to zero, it is possible to define the bin-occupancy density

$$D_{k|k} \triangleq \lim_{|\nu| \rightarrow 0} \frac{\Pr(\mathbf{m}_k^{(n)} = 1 | \mathbf{Z}_{1:k})}{|\nu|} \quad (4)$$

where  $\Pr(\mathbf{m}_k^{(n)} = 1 | \mathbf{Z}_{1:k})$  is the probability that bin  $n$  is occupied by one target. The continuous form of the bin-occupancy filter prediction and update equations are the same as the PHD filter equations [18]. Furthermore, the PHD is the first moment density or *intensity density* in point process theory; see, e.g., [19], [21], and a physical interpretation is given in [26] as the probability that one target is located in the infinitesimal region  $(\mathbf{x}, \mathbf{x} + d\mathbf{x})$  of the state space, divided by  $d\mathbf{x}$ . The continuous form of the physical bin model leads us to a continuous location based map which we denote *intensity based map*, and we intend to estimate it with the PHD filter.

The bin occupancy filter and the PHD filter were developed for target tracking of point sources; however, the aim of the present contribution is to create a probabilistic location based map of the surroundings of a moving vehicle. One of the main differences between standard target tracking problems and the building of a location based map, is that many objects such as, guardrails or walls, are typically not point targets, but extended targets [21], [27], [28]. Furthermore, there is no interest in keeping track of the identity of specific objects. Nevertheless, the bin-occupancy filter attempts to answer the important question: “Is there an object (target) at a given point?” Erdinc *et al.* [18] pose the following assumptions for the bin occupancy filter:

- 1) the bins are sufficiently small so that each bin is occupied by at most one target;
- 2) one target gives rise to only one measurement;
- 3) each target generates measurements independently;
- 4) false alarms are independent of target originated measurements;
- 5) false alarms are Poisson distributed.

Here, only point 2 needs some extra treatment if the aim of the algorithm is mapping and not target tracking. It can be argued that the measurements of the point sources belong to extended objects and that the aim is to create a map of those point sources. Also for mapping cases the assumption that there will not be two measurements from the same point at the same time is justified. The described relation is modeled by a likelihood function  $p(\mathbf{Z}_k | \mathbf{M}_{k|k})$ , which maps the Cartesian map to polar point measurements.

So far, in this section, the discussion has been quite general and the PHD or the intensity has only been considered as a surface over the surveillance region. The first practical algorithms to realize the PHD filter prediction and measurement update equations were based on the particle filter; see, e.g., [29]–[31],

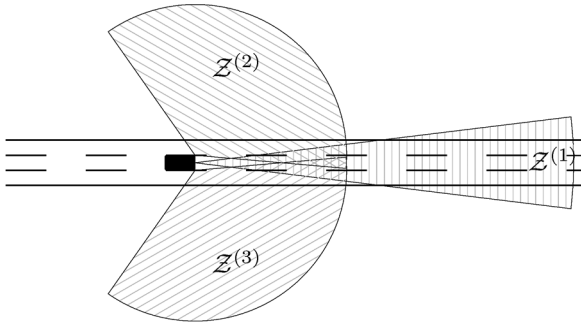


Fig. 1. The observation space of the forward looking radar is denoted  $\mathcal{Z}^{(1)}$  and the observation spaces of the side-looking radars are  $\mathcal{Z}^{(2)}$  and  $\mathcal{Z}^{(3)}$ .

where the PHD is approximated by a large set of random samples (particles). A Gaussian mixture approximation of the PHD (GM-PHD) was proposed by Vo and Ma [32]. The mixture is represented by a sum of weighted Gaussian components and in particular the mean and covariance of the components are propagated by the Kalman filter. In this work the intensity is represented by a Gaussian mixture, since the parametrization and derivation is simpler than for a particle filter based solution. The modeling of the intensity through a number of Gaussian components makes it also simpler to account for structures in the map. We will return to these structures in the next two sections.

The GM-PHD filter estimates the posterior intensity, denoted  $D_{k|k}$ , as a mixture of Gaussian densities as

$$D_{k|k} = \sum_{i=1}^{J_{k|k}} w_{k|k}^{(i)} \mathcal{N}(m_{k|k}^{(i)}, P_{k|k}^{(i)}) \quad (5)$$

where  $J_{k|k}$  is the number of Gaussian components and  $w_{k|k}^{(i)}$  is the expected number of point sources covered by the density  $\mathcal{N}(m_{k|k}^{(i)}, P_{k|k}^{(i)})$ . In Section III it is shown how the intensity is estimated with the GM-PHD filter. The Gaussian components are parametrized by a mean  $m_{k|k}^{(i)}$  and a covariance value  $P_{k|k}^{(i)}$ , which are expressed in a planar Cartesian coordinate frame, according to

$$m_k^{(i)} = \begin{bmatrix} x_k^{(i)} & y_k^{(i)} \end{bmatrix}^T. \quad (6)$$

The aim of the mapping algorithm is to estimate the posterior density (3). The considered intensity based map is continuous over the surveillance region, thus, the volume of the bins becomes infinitesimal, and the number of elements in (2) tends to infinity ( $N_m \rightarrow \infty$ ). Furthermore, the intensity is a summary statistic of the map according to

$$p(\mathbf{M}_k | \mathbf{Z}_{1:k}) \sim p(\mathbf{M}_k; D_{k|k}) \quad (7)$$

see, e.g., [20], and the estimated intensity  $D_{k|k}$  is parametrized by

$$\boldsymbol{\mu}_k^{(i)} \triangleq \left\{ w_k^{(i)} \quad m_k^{(i)} \quad P_k^{(i)} \right\}^T \quad (8)$$

of the Gaussian sum (5). The intensity based map is a multi-modal surface with peaks around areas with many sensor reflections or point sources. It is worth observing that the map  $\mathbf{M}$

is described by a location based function (5), with feature based parametrization (8). The intensity is a distribution and the graphical representation provides an intuitive description of the map.

An automotive mapping example will be studied to exemplify the methods described in each section. In this example, an intensity based map is constructed of the infrastructure surrounding an ego vehicle equipped with three radar sensors. The aim is to represent stationary radar reflectors, i.e., reflection points on road side objects, such as, guard rails or lampposts, using these radar measurements. Each sensor ( $s = 1, 2, 3$ ) collects a set of individual observations  $m = 1, \dots, N_{z,k}^{(s)}$  of range  $r^{(m)}$ , range rate  $\dot{r}^{(m)}$  and bearing angle  $\psi^{(m)}$  to strong radar reflectors according to

$$\left\{ \mathbf{z}_k^{(m)} = \begin{bmatrix} r^{(m)} & \dot{r}^{(m)} & \psi^{(m)} \end{bmatrix}^T \right\}_{m=1}^{N_{z,k}^{(s)}}. \quad (9)$$

The three sensors are one forward looking 77 GHz mechanically scanning FMCW radar, with measurement space  $\mathcal{Z}^{(1)}$ , and two medium distance 24 GHz radars, with measurement spaces  $\mathcal{Z}^{(2)}$  and  $\mathcal{Z}^{(3)}$ , which are mounted on the vehicle as illustrated in Fig. 1. The radars constitute a multi-sensor system, measuring at different ranges and in different directions, but with an overlap of observation spaces. One of the most significant advantages with the PHD filter is that this framework scales very well for multi-sensor multi-target problem, and the classically hard sensor fusion task is solved more or less automatically. A coordinate frame  $E$  is attached to the moving ego vehicle; see Fig. 3. The trajectory  $\mathbf{x}_{1:k} = \mathbf{x}_1, \dots, \mathbf{x}_k$  of the ego vehicle is assumed to be *a priori* known in this work, and it is defined through the sequence of all its positions, orientations and velocities relative to the world coordinate frame  $W$  up to time  $k$ .

Note, that the map is parametrized in a very compact format (8), and that the parameters easily can be sent over the CAN-bus to decision systems in a car.

### III. GM-PHD FILTER

The PHD filter recursion is described in, e.g., [20], the bin occupancy filter is described in [18] and the GM-PHD filter recursion is described in [32]. This section intends to summarize the main algorithmic concept of those contributions and apply the filter to the mapping application. The mentioned filters are primarily developed for multiple-target tracking, where, at least for the PHD filter, the targets are defined as a random finite set, and where the extraction of the most likely target states at the end of each recursion cycle is an important output. However, the output for the proposed mapping is the intensity  $D_{k|k}$ , which describes the density of sensor reflections or point sources. The filter recursion is given by the time update in Section III-A and the measurement update in Section III-B below.

#### A. Time Evolution

Consider the bin occupancy model, a bin can be occupied at time  $k + 1$ , if it already was occupied at time  $k$ , or if a new point source appears in the map at time  $k + 1$ . A new source can appear either by spontaneous birth, as a completely new object, or by spawning from an existing object.

In the mapping application the following definitions are made: If a bin turns occupied independent of other bins' occupancy it is denoted *birth*, whereas if a bin turns occupied dependent on the occupancy of other bins it is denoted *spawn*. The latter is for example the case if radar reflections stem from an extended object, such as a guardrail, and new point sources appear on other positions on the same object. The prediction of the intensities is given by the sum of the existing intensity  $D_{s,k+1|k}$ , the spawn intensity  $D_{\beta,k+1|k}$  and the birth intensity  $D_{b,k+1}$  according to

$$D_{k+1|k} = D_{s,k+1|k} + D_{\beta,k+1|k} + D_{b,k+1}. \quad (10)$$

The prediction for the existing intensity is given by

$$D_{s,k+1|k} = p_{S,k+1} \sum_{i=1}^{J_{k|k}} w_{k|k}^{(i)} \mathcal{N}(m_{k+1|k}^{(i)}, P_{k+1|k}^{(i)}) \quad (11)$$

where the Gaussian components  $\mathcal{N}(m_{k+1|k}^{(i)}, P_{k+1|k}^{(i)})$  are derived using the time update step of, e.g., the Kalman filter. The probability of survival is denoted  $p_S$ . The spawn intensity is a Gaussian mixture of the form

$$D_{\beta,k+1|k} = \sum_{j=1}^{J_{\beta,k+1}} w_{\beta,k+1|k}^{(j)} \mathcal{N}(m_{\beta,k+1|k}^{(j)}, P_{\beta,k+1|k}^{(j)}) \quad (12)$$

where  $w_{\beta,k+1|k}^{(j)}$ ,  $m_{\beta,k+1|k}^{(j)}$  and  $P_{\beta,k+1|k}^{(j)}$  for  $j = 1, \dots, J_{\beta,k+1}$ , are model parameters that determine the shape of the spawn intensity. In the original Gaussian mixture PHD filter [32] the components  $\mathcal{N}(m_{\beta,k+1|k}^{(j)}, P_{\beta,k+1|k}^{(j)})$  are in the proximity of its parents. However, in the mapping application the existing stationary point sources are likely to belong to extended object, which describe some structure in the map, from which new point sources are likely to be spawned. Typical examples of such structures are guard rails in automotive examples or coast lines in terrain mapping. An approach is proposed where the components are derived using a model-based process, which takes the existing components  $\mathcal{N}(m_{k|k}^{(i)}, P_{k|k}^{(i)})$  at time step  $k$  as input. A method, which clusters the components and estimates the model parameters simultaneously is described in Section IV. The Gaussian mixture components  $\mathcal{N}(m_{\beta,k+1|k}^{(j)}, P_{\beta,k+1|k}^{(j)})$  are then sampled from the estimated models.

The birth intensity is assumed to be a Gaussian mixture of the form

$$D_{b,k+1} = \sum_{j=1}^{J_{b,k+1}} w_{b,k+1}^{(j)} \mathcal{N}(m_{b,k+1}^{(j)}, P_{b,k+1}^{(j)}) \quad (13)$$

where  $w_{b,k+1}^{(j)}$ ,  $m_{b,k+1}^{(j)}$  and  $P_{b,k+1}^{(j)}$  for  $j = 1, \dots, J_{b,k+1}$ , are model parameters that determine the shape of the birth intensity. The components  $\mathcal{N}(m_{b,k+1}^{(j)}, P_{b,k+1}^{(j)})$  are uniformly distributed over the state space.

## B. Measurement Update

The measurement update is given by a sum of intensities according to

$$D_{k|k} = (1 - p_{D,k})D_{k|k-1} + \sum_{\mathbf{z} \in \mathbf{Z}_k} D_{d,k|k} \quad (14)$$

where  $p_D$  is the probability of detection. The updated intensity  $D_{d,k|k}$  is given by

$$D_{d,k|k} = \sum_{i=1}^{J_{k|k-1}} w_k^{(i)}(\mathbf{z}) \mathcal{N}(m_{k|k}^{(i)}(\mathbf{z}), P_{k|k}^{(i)}(\mathbf{z})) \quad (15)$$

where the number of Gaussian components is  $J_{k|k-1} = J_{k-1|k-1} + J_{\beta,k|k-1} + J_{b,k}$ . The Gaussian components  $\mathcal{N}(m_{k|k}^{(i)}(\mathbf{z}), P_{k|k}^{(i)}(\mathbf{z}))$  are calculated by using the measurement update step of, e.g., the Kalman filter. Furthermore, the weights are updated according to

$$w_k^{(i)}(\mathbf{z}) = \frac{p_{D,k} w_{k|k-1}^{(i)} q_k^{(i)}(\mathbf{z})}{\kappa_k + p_{D,k} \sum_{\ell=1}^{J_{k|k-1}} w_{k|k-1}^{(\ell)} q_k^{(\ell)}(\mathbf{z})} \quad (16a)$$

$$q_k^{(i)}(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \eta_{k|k-1}^{(i)}, S_{k|k-1}^{(i)}) \quad (16b)$$

where  $\eta_{k|k-1}^{(i)}$  is the predicted measurement and  $S_{k|k-1}^{(i)}$  is the innovation covariance from the  $i^{\text{th}}$  component in the predicted intensity. The clutter intensity at time  $k$  is denoted  $\kappa_k$ , and the clutter measurements are assumed to be Poisson distributed.

## IV. JOINT CLUSTERING AND ESTIMATION

The spawning process was briefly introduced in Section III-A. Provided that there exists a certain structure in between the single point objects in the map, then this structure can be used to improve the spawning process. This section describes how the map structure can be found, modeled, and estimated. We propose a regression clustering algorithm, which alternates between clustering the Gaussian components and estimating the model parameters, using a simple weighted least squares method. The standard  $K$ -means cluster algorithm is first described in Section IV-A followed by the joint clustering and estimation algorithm in Section IV-B. Road edges are typical structures in a map, and it is shown in Section IV-C how to apply the joint clustering and estimation algorithm to estimate these.

### A. $K$ -Means Clustering

Given a set of  $N_y$  observations  $\mathbf{y}$ , then the clustering algorithm aims to find a partition  $\mathbf{Y} = \{\mathbf{Y}^{(1)}, \dots, \mathbf{Y}^{(K)}\}$  of the observations, where  $K < N_y$ . Note that, in this section,  $\mathbf{y}$  represents any arbitrary observation, which should not be confused

with the sensor measurements  $\mathbf{z}$ . The original  $K$ -means clustering algorithm, introduced by Lloyd [33], aims at minimizing the within-cluster sum of squares of the distances according to

$$\min_{\mathbf{Y}^{(1)}, \dots, \mathbf{Y}^{(K)} \atop \mathbf{Y}^{(1)} \sqcup \dots \sqcup \mathbf{Y}^{(K)} = \mathbf{Y}} \sum_{k=1}^K \sum_{\mathbf{y}^{(i)} \in \mathbf{Y}^{(k)}} \left\| \mathbf{y}^{(i)} - \mu^{(k)} \right\|^2 \quad (17)$$

where  $\mu^{(k)}$  is the mean value of the set  $\mathbf{Y}^{(k)}$ . A solution is found by alternating between the assignment step, which assigns each observation to the cluster with the closest mean, and the update step, which calculate the new means, until convergence; see, e.g., the textbooks [34]–[36]. A study of the convergence properties of the  $K$ -means algorithm is given in [37].

### B. Regression Clustering

Now, suppose that the observations  $\mathbf{y}^{(i)}$  of the set  $\mathbf{Y}^{(k)}$  are produced by a regression model, with the regressors  $\varphi^{(i)}$ , according to

$$\mathbf{y}^{(i)} = \left( \varphi^{(i)} \right)^T \boldsymbol{\theta}^{(k)} + \mathbf{e}^{(i)}, \quad \mathbf{y}^{(i)} \in \mathbf{Y}^{(k)}, \quad \mathbf{e}^{(i)} \sim \mathcal{N} \left( 0, R^{(i)} \right) \quad (18)$$

and that the objective is to identify the parameters  $\boldsymbol{\theta}^{(k)}$  for every expected cluster  $k = 1, \dots, K$ . The observations  $\mathbf{y}^{(i)}$  within one set  $\mathbf{Y}^{(k)}$  are chosen to fit the common parameter  $\boldsymbol{\theta}^{(k)}$ , i.e., the aim is to minimize the sum of squared residuals within each cluster according to

$$\min_{\mathbf{Y}^{(1)}, \dots, \mathbf{Y}^{(K)} \atop \mathbf{Y}^{(1)} \sqcup \dots \sqcup \mathbf{Y}^{(K)} = \mathbf{Y}} \sum_{k=1}^K \min_{\boldsymbol{\theta}^{(k)}} \sum_{\mathbf{y}^{(i)} \in \mathbf{Y}^{(k)}} \left\| \mathbf{y}^{(i)} - \left( \varphi^{(i)} \right)^T \boldsymbol{\theta}^{(k)} \right\|_{R^{(i)}}^2 \quad (19)$$

using the norm  $\|x\|_R^2 = x^T R^{-1} x$ . This can be done by first clustering all observations using  $K$ -means (17) and thereafter estimating the parameters  $\boldsymbol{\theta}^{(k)}$  for each cluster. However, we propose a joint clustering and estimation algorithm, which instead of clustering the observations around mean values  $\mu^{(k)}$ , clusters around models  $(\varphi^{(i)})^T \boldsymbol{\theta}^{(k)}$ . Similar methods are used to solve for piecewise affine systems; see, e.g., [38], [39], and for switching dynamic systems, see, e.g., [40], where the observation regions and submodels are identified iteratively. A solution is found by alternating between an assignment step and an estimation step until convergence. These steps are described in detail below.

1) *Assignment Step*: Given an initial guess of  $K$  parameters  $\hat{\boldsymbol{\theta}}_0^{(k)}$  and covariances  $P_0^{(k)}$ , for  $k = 1, \dots, K$ ; each observation  $\mathbf{y}^{(i)}$  is first assigned to the cluster with the smallest estimation error according to

$$\mathbf{Y}_\tau^{(k)} = \left\{ \mathbf{y}^{(i)} : \left\| \mathbf{y}^{(i)} - \left( \varphi^{(i)} \right)^T \hat{\boldsymbol{\theta}}_\tau^{(k)} \right\|_{S_\tau^{(k,i)}} \leq \left\| \mathbf{y}^{(i)} - \left( \varphi^{(i)} \right)^T \hat{\boldsymbol{\theta}}_\tau^{(j)} \right\|_{S_\tau^{(k,i)}}, \forall j \right\} \quad (20a)$$

using the norm  $\|x\|_S^2 = x^T S^{-1} x$ , where

$$S_\tau^{(k,i)} = \left( \varphi^{(i)} \right)^T P_\tau^{(k)} \varphi^{(i)} + R^{(i)}. \quad (20b)$$

The covariance of the parameter estimate, denoted  $P_\tau^{(k)} = \text{Cov} \hat{\boldsymbol{\theta}}_\tau^{(k)}$ , is calculated in the estimation step. Note, that the discrete iteration index is denoted  $\tau$  here.

2) *Estimation Step*: The second step is the update step, where new parameters  $\hat{\boldsymbol{\theta}}^{(k)}$  are estimated for each cluster. Any estimation algorithm may be used, e.g., the weighted least squares according to

$$\hat{\boldsymbol{\theta}}_{\tau+1}^{(k)} = \left( \sum_{\mathbf{y}^{(i)} \in \mathbf{Y}_\tau^{(k)}} \varphi^{(i)} \left( R^{(i)} \right)^{-1} \left( \varphi^{(i)} \right)^T \right)^{-1} \cdot \sum_{\mathbf{y}^{(i)} \in \mathbf{Y}_\tau^{(k)}} \varphi^{(i)} \left( R^{(i)} \right)^{-1} \left( \varphi^{(i)} \right)^T \mathbf{y}^{(i)} \quad (21a)$$

$$P_{\tau+1}^{(k)} = \left( \sum_{\mathbf{y}^{(i)} \in \mathbf{Y}_\tau^{(k)}} \varphi^{(i)} \left( R^{(i)} \right)^{-1} \left( \varphi^{(i)} \right)^T \right)^{-1}. \quad (21b)$$

The assignment step (20) and estimation step (21) are iterated until the assignments no longer change.

### C. Road Edge Estimation

To illustrate the joint clustering and estimation algorithm proposed above, we return to the example discussed in Section II. In this example, it is likely that new stationary objects are spawned from road-side objects, such as guard rails. Consequently, we propose to use a simplified version of the road edge model in [41] in the joint clustering and estimation algorithm to find the road edge in the current map. From this estimate of the road edge, the spawning intensity can be described.

The road edge is described by a polynomial model, linear in its parameters

$$y = a_0^{(k)} + a_1 x + a_2 x^2 + a_3 x^3 \quad (22)$$

given the  $x$  and  $y$ -coordinates of the Gaussian components, and expressed in the vehicle's coordinate frame  $E$ . Furthermore, assume that the left and right edge of the road are approximately parallel and that they can be modeled using at most  $K$  polynomials (22). The polynomials only differ by the lateral distances  $a_0^{(k)}$  for the  $k = 1, \dots, K$  clusters.

Given the  $J_{k|k}$  Gaussian components, the parameters

$$\boldsymbol{\theta}^{(k)} = \begin{bmatrix} a_0^{(k)} & a_1 & a_2 & a_3 \end{bmatrix}^T \quad (23)$$

are estimated by rewriting the linear predictor (22) according to

$$\mathbf{y}^{(i)} = \left( \varphi^{(i)} \right)^T \boldsymbol{\theta}^{(k)} \quad (24)$$

where the regressors are given by

$$\varphi^{(i)} = \begin{bmatrix} 1 & x^{(i)} & (x^{(i)})^2 & (x^{(i)})^3 \end{bmatrix}^T \quad (25)$$

for the mean values of the components (8) in cluster  $k$ . The values of the parameters  $\theta^{(k)}$  are found by minimizing the estimation error within each cluster according to (19), i.e.,

$$\min_{\mathbf{Y}^{(1)}, \dots, \mathbf{Y}^{(K)}} \sum_{k=1}^K \min_{\theta^{(k)}} \sum_{\mathbf{y}^{(i)} \in \mathbf{Y}^{(k)}} \left\| \mathbf{y}^{(i)} - (\varphi^{(i)})^T \theta^{(k)} \right\|_{R^{(i)}}^2. \quad (26)$$

In this case the observations  $\mathbf{y}^{(i)}$  are the mean values  $m^{(i)} = [x^{(i)} \ y^{(i)}]^T$  of the Gaussian components and the noise covariances  $R^{(i)}$  are the estimated weighted covariances  $P^{(i)}/w^{(i)}$ . The parameters  $\theta^{(k)}$  are estimated iteratively using the assignment step (20) and the estimation step (21). Since radar measurements also are obtained by objects on the opposite side of the freeway, the states are clustered into four sets. A fair initial guess for  $K = 4$  clusters could be  $a_0^{(1)} = 10$ ,  $a_0^{(2)} = -10$ ,  $a_0^{(3)} = 30$ ,  $a_0^{(4)} = -30$  and  $a_1 = a_2 = a_3 = 0$ .

Fig. 2 shows the four resulting clusters in two traffic situations. The stars are the mean values of the Gaussian components, which are used as input to the joint clustering and estimation algorithm, and the lines are the resulting road edge models  $\varphi^T \theta^{(k)}$ . The past and future path of the ego vehicle is shown by black dots in the figure and can be used as a good reference of the road edges being correctly estimated. The number of components in each cluster indicates how many objects that should be spawned from the cluster. The spawning along the road edges is described in Section VI-B.

## V. MERGING

The Gaussian mixture formulation of the PHD filter tends to suffer from a high computational complexity as the number of mixture components,  $J_{k|k}$ , in (5) grows rapidly. To keep the complexity under control, methods are typically used to prune mixture components with too low weight  $w_{k|k}^{(i)}$ , which makes small contribution to the posterior density, and to merge components that are similar. The general merging process is discussed in Section V-A. However, while keeping the complexity in check, it is important not to lose too much information in the process. The errors introduced in pruning and merging are analyzed in [42]. In Section V-B, we propose a general algorithm, which can be used for merging Gaussian components if more information about the system is available. The state space is often chosen since it is a convenient space to perform tracking in, i.e., to predict and update tracks in. However, as described in Section V-B, Gaussian components may be better merged in another space, in which the system requirements are easier to describe.

Typically there exists some structure in a map. By exploiting this structure in the merging of the components, it is possible to maintain the information in the map, while reducing the number of components needed to describe it. The algorithm proposed in Section V-B is applied on the road map example in Section V-C.

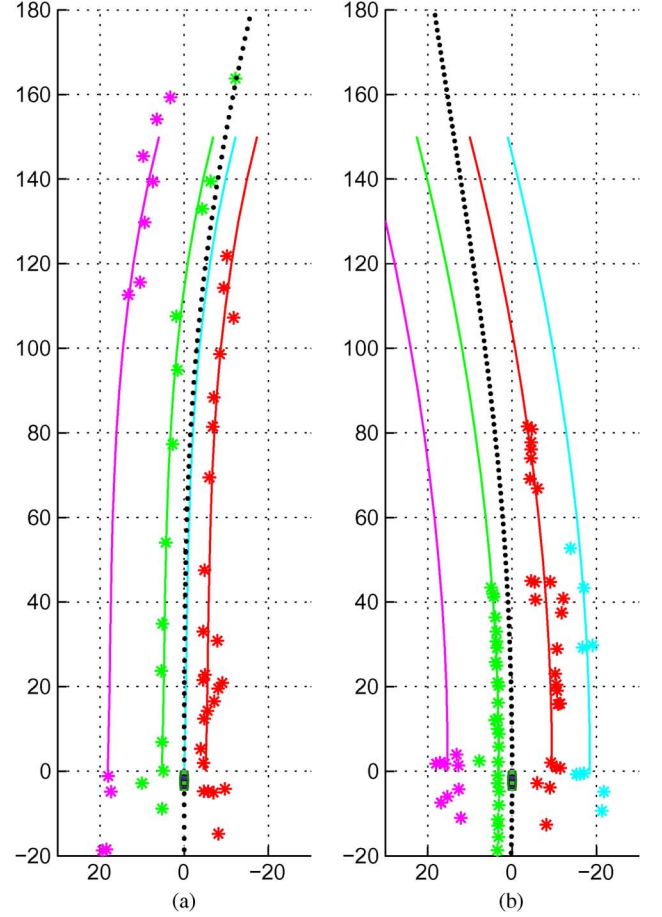


Fig. 2. The four clusters, describing the road edges, are illustrated with different colors, and can be compared with the vehicles driven path indicated with black dots. The lines are a “snapshot” estimate, based on the Gaussian components at one time step  $k$ . The drivers view is shown in Fig. 5(a) and (d) for the two traffic scenarios in (a) and (b), respectively.

The model-based merging method exploits the maps structure to find a more efficient representation of the posterior intensity in (5).

### A. Background

The Gaussian mixture posterior intensity is given in (14). The aim of the merging is to compute a reduced Gaussian mixture, with  $\ell < J_{k|k}$  components, that is close to the original intensity according to some distance measure  $d$ .

Some different methods for Gaussian mixture reduction by means of merging components exists, and have mainly been proposed for Gaussian sum filters. Alspach proposed a method to merge two Gaussian components with the same covariance [43]. The joining algorithm, proposed by Salmond [44], evaluates the distances between every pair of components and successively pairs those components with the smallest distance. To avoid scaling problems, which may occur depending on the size of the elements in the state vector, Mahalanobis distance measure is often used; see, e.g., [45].

The clustering algorithm, also presented in [44], merges Gaussian components in groups rather than in pairs. A cluster center (centroid) is chosen as the component with the largest

TABLE I  
MERGING GAUSSIAN MIXTURE BY MEANS OF THE CLUSTER ALGORITHM

**Require:** the Gaussian components  $\{\tilde{\mu}_k^{(i)}\}_{i=1}^{J_k|k}$  and the threshold  $\delta_m$   
**1: initiate:** by setting  $\ell = 0$ ,  $I = \{i = 1, \dots, J_k|k\}$  and then  
**2: repeat**  
**3:**  $\ell := \ell + 1$   
**4:**  $j := \arg \max_{i \in I} w_k^{(i)}$   
**5:**  $L := \left\{ i \in I \mid d_k^{(i,j)} \leq \delta_m \right\}$ , where  $d_k^{(i,j)}$  is given in (27)  
**6:**  $\tilde{w}_k^{(\ell)} = \sum_{i \in L} w_k^{(i)}$   
**7:**  $\tilde{m}_k^{(\ell)} = \frac{1}{\tilde{w}_k^{(\ell)}} \sum_{i \in L} w_k^{(i)} m_k^{(i)}$   
**8:**  $\tilde{P}_k^{(\ell)} = \frac{1}{\tilde{w}_k^{(\ell)}} \sum_{i \in L} w_k^{(i)} (P_k^{(i)} + (\tilde{m}_k^{(\ell)} - m_k^{(i)})(\tilde{m}_k^{(\ell)} - m_k^{(i)})^T)$   
**9:**  $I := I \setminus L$   
**10: until**  $I = \emptyset$   
**11: return** the Gaussian components  $\{\tilde{\mu}_k^{(i)}\}_{i=1}^\ell$

weight and the algorithm then merges all surrounding components to the centroid. This process is repeated with the remaining components until all have been clustered. The distance measure which is used to represent the closeness of a components  $i$  to the centroid  $j$  is given by

$$\left(d_k^{(i,j)}\right)^2 = \frac{w_k^{(j)} w_k^{(i)}}{w_k^{(j)} + w_k^{(i)}} \left(m_k^{(j)} - m_k^{(i)}\right)^T \left(P_k^{(j)}\right)^{-1} \times \left(m_k^{(j)} - m_k^{(i)}\right) \quad (27)$$

where the distance is normalized to the covariance of the centroid. The factor  $w_k^{(j)} w_k^{(i)} / (w_k^{(j)} + w_k^{(i)})$  makes it easier to cluster small components while large components retain as they are. The clustering algorithm is used in the original GM-PHD filter [32].

The joining and clustering algorithms are local methods in the sense that closely lying individual components are merged without considering the total density. During the last years some global methods, which take the whole Gaussian mixture into account, were published. A cost-function based approach was presented in [46], a global cluster approach in [47] and a bottom-up approach which starts with one component and successively builds up components to approximate the original mixture in [48]. The global methods are typically more accurate at the expense of higher computational effort, and are therefore not a good choice for the GM-PHD filter.

### B. Algorithm

Let us introduce an invertible, possibly nonlinear, mapping function  $\mathcal{T}(\cdot)$ , which transforms the Gaussian components  $\mu$ , defined in (8), from the state space to the merging space. The proposed algorithm consists of the following steps:

- 1) transform  $\tilde{\mu}_k^{(i)} := \mathcal{T}(\mu_k^{(i)})$ ;
- 2) merge  $\tilde{\mu}_k^{(i)}$  using the cluster algorithm in Table I, resulting in  $\tilde{\mu}_k^{(i)}$ ;
- 3) inverse transform  $\tilde{\mu}_k^{(i)} := \mathcal{T}^{-1}(\tilde{\mu}_k^{(i)})$ .

The merging algorithm and the transformation is exemplified on the road mapping example in the subsequent Section V-C.

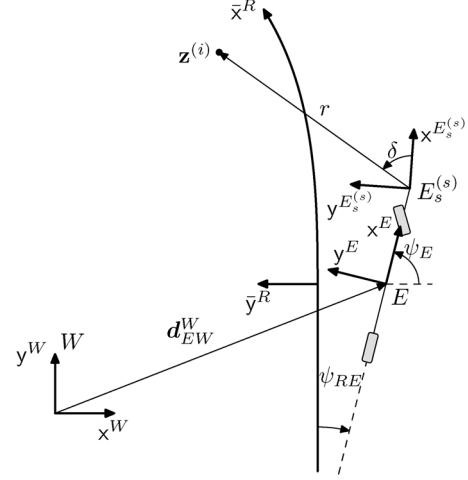


Fig. 3. The world coordinate frame is denoted  $W$ , the ego vehicle's coordinate frame is denoted  $E$  and the curved road coordinate frame is denoted  $R$ .

### C. Road Mapping

The state space is spanned by the Cartesian world coordinate frame. However, for road mapping it is advantageous to merge Gaussian components lying along the road. Hence, the merging space is spanned by a road-aligned coordinate frame. The shape of the road can be described by a polynomial according to (cf. (22))

$$y^E = \psi_{RE} x^E + \frac{c_0}{2} (x^E)^2 + \frac{c_1}{6} (x^E)^3 \quad (28)$$

where  $x^E$  and  $y^E$  are expressed in the ego vehicle's coordinate frame  $E$ . This model was introduced by Dickmanns [49] and a neat feature of the model is that the parameters can be interpreted as physical sizes. The angle between the longitudinal axis of the vehicle and the road lane is  $\psi_{RE}$ ; see Fig. 3. It is assumed that this angle is small and hence the approximation  $\sin \psi_{RE} \approx \psi_{RE}$  is used. The curvature parameter is denoted  $c_0$ . The parameters can be estimated using computer vision and a lane tracker, or taken from the joint clustering and estimation method described in the previous section.

The non-linear transformation  $\mathcal{T}$  from the world coordinate frame to the road fixed coordinate frame is calculated in two steps. First the components  $[x^W \ y^W]^T$ , represented in the world coordinate frame  $W$  are transformed to  $[x^E \ y^E]^T$ , represented in the ego vehicle coordinate frame  $E$ , using

$$\begin{bmatrix} x^E \\ y^E \end{bmatrix} = (R^{WE})^T \left( \begin{bmatrix} x^W \\ y^W \end{bmatrix} - d_{EW}^W \right) \quad (29)$$

where the planar rotation matrix is given by

$$R^{WE} = \begin{bmatrix} \cos \psi_E & -\sin \psi_E \\ \sin \psi_E & \cos \psi_E \end{bmatrix} \quad (30)$$

and where  $\psi_E$  is the angle of rotation from  $W$  to  $E$ . This angle is referred to as the yaw angle of the vehicle. In this work it is assumed that the position  $d_{EW}^W$  and orientation  $\psi_E$  of the ego vehicle is known. In the second step the components are



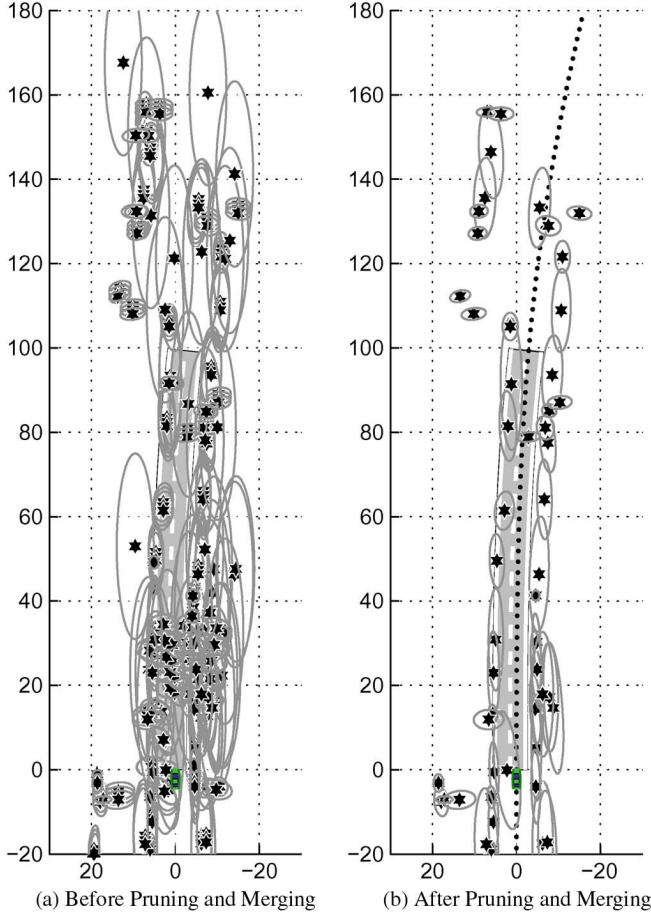


Fig. 4. (a) The Gaussian mixture before pruning and merging. (b) The reduced mixture. The illustrated road is estimated from the camera image, and can be used as a reference here. This is the same example as in Fig. 5(a) and (b), where also a photo of the driver's view is given.

transformed into the road aligned coordinate frame according to

$$\begin{bmatrix} \bar{x}^R \\ \bar{y}^R \end{bmatrix} = \begin{bmatrix} x^E \\ y^E + \psi_{RE}x^E + \frac{c_0}{2}(x^E)^2 + \frac{c_1}{6}(x^E)^3 \end{bmatrix}. \quad (31)$$

Some other approximations of the transformation  $\mathcal{T}$  are described in [50]. Since the transformation is nonlinear, the unscented transform, see, e.g., [51], is used to propagate the Gaussian components between the two spaces.

The covariance  $P_k^{(j)}$  in the distance measure (27) is artificially formed to make it more likely for components lying along the road to be merged. A diagonal covariance matrix  $\Sigma \in \mathbb{R}^{2 \times 2}$  is created, with its two elements well separated, resulting in a pin-shaped covariance ellipse, and added to the state covariance according to

$$\bar{P}_k^{(j)} = P_k^{(j)} + \Sigma. \quad (32)$$

The merging is performed according to Table I, and the components are transformed back to the world coordinate frame using the inverse transform  $\mathcal{T}^{-1}$ , cf. Section V-B. An example showing the components before and after the merging is illustrated in Fig. 4. Compared with the camera view in Fig. 5(a), it

is clearly shown that the resulting components are following the road edge.

## VI. ROAD MAPPING EXAMPLE

In this section all parts of the road mapping example, which was introduced in Section II and configured in the preceding sections, are put together. Especially the spawning of new target is exemplified in Section VI-B. The application dependent parts of GM-PHD filter recursion are described in Section VI-C. The capability of the posterior intensity to map the surroundings of the vehicle is shown in Section VI-D. However, we start with an overview and definition of the system in Section VI-A.

### A. System Model

In this work, variables are defined from different perspectives. The measurements are obtained by the sensor on the ego vehicle and are therefore expressed in the ego vehicles coordinate frame, which is denoted  $E$ . The states, i.e., the position of the stationary objects, are expressed in a world fixed reference frame  $W$ . The radars are mounted at the front bumper at the positions  $E_s^{(s)}$ , for the sensors  $s = 1, 2, 3$ .

The linear process model

$$m_{k+1} = m_k + \mathbf{w}_k, \quad \mathbf{w}_k \sim \mathcal{N}(0, Q) \quad (33a)$$

of the Gaussian components allows for some motion, even though the objects are stationary. The reason is that one Gaussian component covers many stationary point sources and the mean value will change if sources are added or removed. The nonlinear measurement equation

$$\mathbf{z}_k = h(m_k) + \mathbf{e}_k, \quad \mathbf{e}_k \sim \mathcal{N}(0, R) \quad (33b)$$

describes how a measurement (9) i.e., the range  $r$ , the range rate  $\dot{r}$  and the bearing angle  $\psi$  relates to the Gaussian components  $m_k$ . The nonlinear sensor model is given by

$$h(\mathbf{x}_k^{(i)}) = \begin{bmatrix} \sqrt{\left(x^{(i)} - x_{E_s^{(s)}W}^W\right)^2 + \left(y^{(i)} - y_{E_s^{(s)}W}^W\right)^2} \\ -v \cos\left(\arctan\frac{y^{(i)} - y_{E_s^{(s)}W}^W}{x^{(i)} - x_{E_s^{(s)}W}^W} - \psi_{E_s^{(s)}}\right) \\ \arctan\frac{y^{(i)} - y_{E_s^{(s)}W}^W}{x^{(i)} - x_{E_s^{(s)}W}^W} - \psi_{E_s^{(s)}} \end{bmatrix} \quad (34)$$

where  $[x_{E_s^{(s)}W}^W \ y_{E_s^{(s)}W}^W]^T$  is the position and  $\psi_{E_s^{(s)}}$  the orientation of the radar sensor  $s = 1, 2, 3$  in  $W$ ; see Fig. 3. The velocity of the ego vehicle is denoted  $v$ . The range rate is the relative velocity between the source and the moving vehicle. It is primarily included in the measurement vector to be used to sort stationary sources from moving target in the gate process; cf. (39).

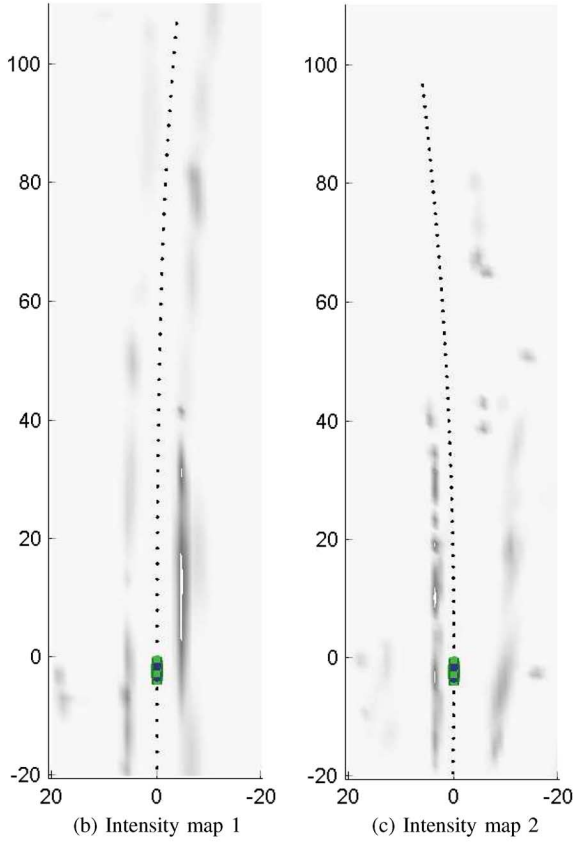
### B. Spawn Process

The left and right road edges are modeled with the polynomial (22) and with the parameter vectors  $\theta^{(1)}$  and  $\theta^{(2)}$  which were estimated in Section IV-C. The parameters  $\theta^{(3)}$  and  $\theta^{(4)}$  are assumed to belong to lines further away, e.g., the other side of the road, and are therefore not considered here. The models





(a) Camera view 1



(b) Intensity map 1

(c) Intensity map 2



(d) Camera view 2

Fig. 5. The image in (a) shows the driver's view of the intensity map in (b), and the image in (d) is the driver's view of the intensity map in (c). The darker the areas in the intensity map, the higher the concentration of objects. The driver's path is illustrated with black dots and may be used as a reference.

are used to create a prior intensity, which is a Gaussian mixture of the form (12) along the road edges. The road model (22) is expressed in vehicle coordinates and after having derived the Gaussian mixture, the components are transformed into the world frame  $W$ .

A number  $J_{\beta,k+1}/2$  of  $x^E$ -coordinates, denoted  $\{x^{E,(j)}\}_{j=1}^{J_{\beta,k+1}/2}$ , are chosen in the range from 0 to the maximum range of the radar sensor. The corresponding  $y^E$ -coordinates  $\{y^{E,(j)}\}_{j=1}^{J_{\beta,k+1}/2}$  and  $\{y^{E,(j)}\}_{j=J_{\beta,k+1}/2+1}^{J_{\beta,k+1}}$  are derived by using the road border model (22) and the parameters  $\hat{\theta}^{(1)}$  and  $\hat{\theta}^{(2)}$ , respectively. The coordinates form the mean values of the position of the Gaussian components on the road edges according to

$$m_{\beta,k+1}^{E,(j)} = \begin{bmatrix} x^{E,(j)} & y^{E,(j)} \end{bmatrix}^T. \quad (35a)$$

The covariance of the Gaussian components is given by the diagonal matrix

$$P_{\beta,k+1}^{E,(j)} = \begin{bmatrix} (\sigma_x^E)^2 & 0 \\ 0 & (\sigma_y^E(x^{E,(j)}))^2 \end{bmatrix} \quad (35b)$$

where it is assumed that the variance of the  $y$ -coordinate increases with the  $x$ -distance, i.e.,  $\sigma_y^E(x^{E,(j)})$  depends on  $x^{E,(j)}$ , to model the increased uncertainty in the shape of the road at long distances.

So far the derivations are accomplished in the ego vehicle's coordinate frame  $E$ , but since the map is expressed in the world frame  $W$ , the components of the Gaussian mixture are transformed into the world coordinate frame according to

$$m_{\beta,k+1}^{(j)} = R^{WE} m_{\beta,k+1}^{E,(j)} + d_{EW}^W \quad (36a)$$

$$P_{\beta,k+1}^{(j)} = R^{WE} P_{\beta,k+1}^{E,(j)} (R^{WE})^T \quad (36b)$$

to be used in (12). The weight  $w_{\beta,k+1}^{(j)}$  represents the expected number of new targets originating from  $m_{\beta,k+1}^{(j)}$ .

### C. GM-PHD Filter Recursion

The general GM-PHD filter recursion was described in Section III. Additional details for the road mapping example are given in this section. The merging of Gaussian components, described in Section V-C, is performed in each iteration.

1) *Prediction*: The process model (33a) is linear and the Gaussian components in (11) are derived using the Kalman filter prediction step according to

$$m_{k+1|k}^{(i)} = m_{k|k}^{(i)} \quad (37a)$$

$$P_{k+1|k}^{(i)} = P_{k|k}^{(i)} + Q_{k-1}. \quad (37b)$$

The probability of survival is  $p_S = 0.99$ .

2) *Measurement Update*: The measurement equation (34) is nonlinear and the unscented transform (UT) is used to propagate the state variables through the measurement equation; see, e.g., [51]. A set of  $L$  sigma points and weights, denoted by  $\{\chi_k^{(\ell)}, u_k^{(\ell)}\}_{\ell=0}^L$  are generated from each Gaussian component  $\mathcal{N}(x_{k|k-1}^{(i)}, P_{k|k-1}^{(i)})$  using the method described in [51]. The sigma points are transformed to the measurement space using (34) to obtain the propagated sigma point  $\zeta_{k|k-1}^{(\ell)}$  and the first

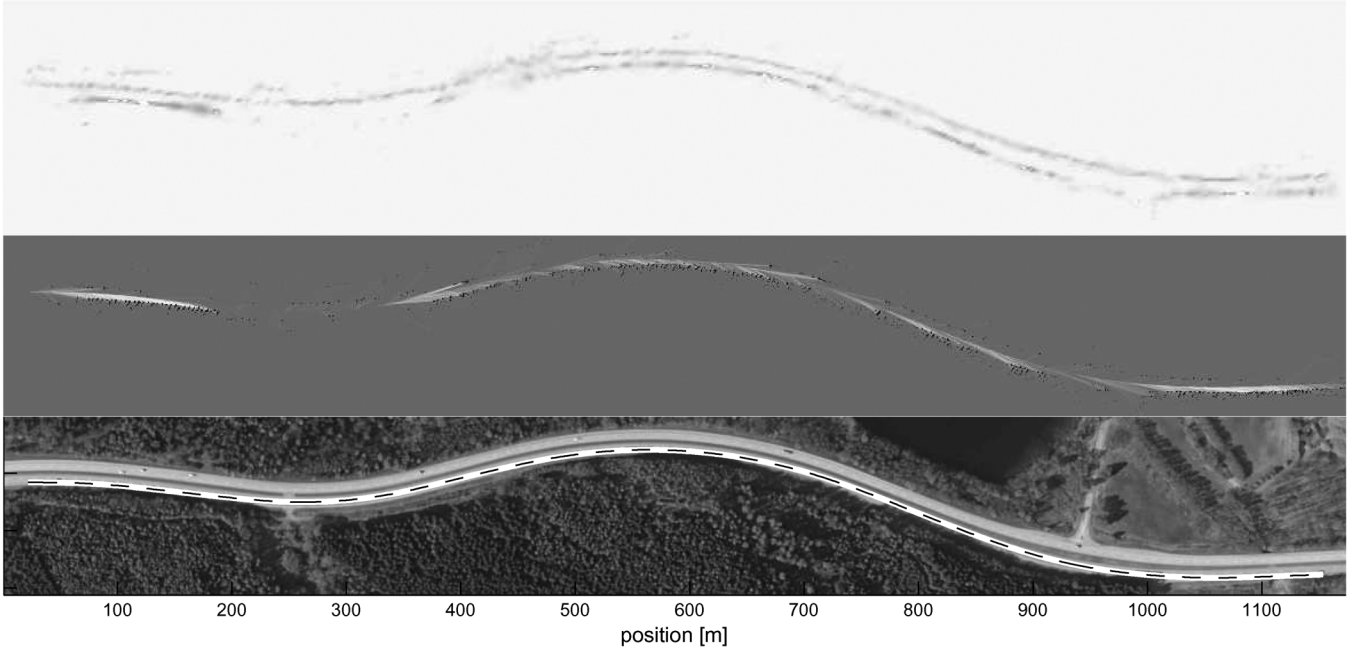


Fig. 6. The top figure shows the intensity based map obtained from radar measurements collected on a highway (drivers view in Fig. 5(a)). The OGM in the middle figure serves as a comparison of an existing algorithm. The bottom figure is a flight photo used as ground truth, where the driven trajectory is illustrated with a dashed line ©Lantmäteriet Gävle 2010. Medgivande I 2010/1540. Reprinted with permission.

and second order moments of the measurement density are approximated as

$$\eta_{k|k-1}^{(i)} = \sum_{\ell=0}^L u^{(\ell)} \zeta_{k|k-1}^{(\ell)} \quad (38a)$$

$$S_{k|k-1}^{(i)} = \sum_{\ell=0}^L u^{(\ell)} \left( \zeta_{k|k-1}^{(\ell)} - \eta_{k|k-1}^{(i)} \right) \times \left( \zeta_{k|k-1}^{(\ell)} - \eta_{k|k-1}^{(i)} \right)^T + R_k \quad (38b)$$

$$G_{k|k-1}^{(i)} = \sum_{\ell=0}^L u^{(\ell)} \left( \chi_{k|k-1}^{(\ell)} - m_{k|k-1}^{(i)} \right) \times \left( \zeta_{k|k-1}^{(\ell)} - \eta_{k|k-1}^{(i)} \right)^T. \quad (38c)$$

Note that these variables also are used to derive the updated weights in (16). The Gaussian components are updated given the information from each new measurement  $\mathbf{z}_k$ , which is in the gate

$$\mathcal{G}_i = \left\{ \mathbf{z}_k \mid \left( \mathbf{z}_k - \eta_{k|k-1}^{(i)} \right)^T \left( S_{k|k-1}^{(i)} \right)^{-1} \left( \mathbf{z}_k - \eta_{k|k-1}^{(i)} \right) < \delta_{\mathcal{G}} \right\} \quad (39)$$

for the gate threshold  $\delta_{\mathcal{G}} = 11.3$ , according to

$$m_{k|k}^{(i)}(\mathbf{z}) = m_{k|k-1}^{(i)} + K_k^{(i)} \left( \mathbf{z}_k - \eta_{k|k-1}^{(i)} \right) \quad (40a)$$

$$P_{k|k}^{(i)} = P_{k|k-1}^{(i)} - K_k^{(i)} S_{k|k-1}^{(i)} \left( K_k^{(i)} \right)^T \quad (40b)$$

$$K_k^{(i)} = G_{k|k-1}^{(i)} \left( S_{k|k-1}^{(i)} \right)^{-1}. \quad (40c)$$

The updated components are together forming the updated intensity in (15).

The posterior intensity at time step  $k$  was given in (14). The probability of detection is adjusted according to

$$p_{D,k}(x) = \begin{cases} p_{D,k}, & \eta_{k|k-1}^{(i)} \in \mathcal{Z} \\ 0, & \eta_{k|k-1}^{(i)} \notin \mathcal{Z} \end{cases} \quad (41)$$

and the value is  $p_D = 10^{-3}$ . Gaussian components which are outside field of view of the sensor, i.e., the measurement space  $\mathcal{Z}^{(s)}$ , are hence not updated. The clutter intensity is  $\kappa = 10^{-8}$ .

The measurements from the three radar sensors are fused by considering their measurement times, synchronizing the data, and updating the filter accordingly.

#### D. Experiments and Results

The experiments were conducted with a prototype passenger car, equipped with the radar sensors configured as shown in Fig. 1. No reference data of the road borders exist, but the vehicle's position was recorded and may be used to illustrate that the resulting map is reasonable. One example of the estimated intensity at a freeway traffic scenario is shown as a bird's eye view in Fig. 5(b). Darker regions illustrate higher concentrations of point sources, which in this figure stem from the guardrails to the left and the right of the road. As expected, the path of the ego vehicle, indicated by the black dots, is in between the two regions of higher object concentration. The driver's view is shown in Fig. 5(a).

A second example is shown in Fig. 5(c) and (d). Here, the freeway exit is clearly visible in the intensity map, which shows that the proposed method to create maps is very conformable.

The Gaussian components are generally removed from the filter when the vehicle passed those parts of the map. However, to give a more comprehensive overview, these components are stored and the resulting intensity based map is shown together

with an occupancy grid map (OGM) and a flight photo in Fig. 6. The top figure is the map produced as described in this contribution. The OGM is based on the same data set and used as a comparison of an existing and well established algorithm; see, e.g., [8]. The gray-level of the OGM indicates the probability of occupancy, the darker the grid cell the more likely it is to be occupied. As seen in the figure the road edges are not modeled as distinct with the OGM. The OGM representation of the map is not very efficient, since huge parts of the map are gray indicating that nothing is known about these areas. An OGM matrix with often more than 10000 elements must be updated and communicated to other safety functions of a car at each time step. The compact representation is an advantage of the intensity based map. Each Gaussian components is parametrized with 7 scalar values according to (8). Since most maps are modeled with 10–30 components it summarizes to around 70–210 scalar values, which easily can be send on the vehicles CAN bus to other safety functions. Finally, the bottom photo is a very accurate flight photo (obtained from the Swedish mapping, cadastral and land registration authority), which can be used as ground truth to visualize the quality of the intensity based map. Note for example the junction at 1000 m.

## VII. CONCLUSION

In this work it is shown how the GM-PHD filter can be used to create a map of the environment using noisy point measurements. The map is represented by the intensity, which describes the concentration of point sources. Methods to identify and utilize typical map structures and to create an efficient prior in the GM-PHD-filter have been presented. The structures are identified by a clustering algorithm which alternates between assigning existing map features to clusters, and estimating structures within each cluster, until convergence. Furthermore, these structures are also used to find an efficient representation of the resulting map.

The usefulness of the proposed mapping approach is demonstrated using automotive radar measurements collected on Swedish freeways. The resulting intensity map of stationary objects, such as guardrails and lampposts, can be viewed as a map over the concentration of radar reflectors. The parametrization is compact and the map is easily sent over the CAN bus to active safety functions, which can use it to derive drivable trajectories with low probability of occupancy.

The intensity is a distribution and its graphical representation provides an intuitive description of the map, which is useful to decision making and path planning algorithms. However, the intensity is an approximation of a true map. It remains for future work to find an expression for the real map and also to develop a metric to quantify the accuracy of the intensity map.

## REFERENCES

- [1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA: MIT Press, 2005.
- [2] M. Adams, W. Wijesoma, and A. Shacklock, "Autonomous navigation: Achievements in complex environments," *IEEE Instrum. Meas. Mag.*, vol. 10, no. 3, pp. 15–21, Jun. 2007.
- [3] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robot Autom. Mag.*, vol. 13, no. 2, pp. 99–110, Jun. 2006.
- [4] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part II," *IEEE Robot Autom. Mag.*, vol. 13, no. 3, pp. 108–117, Sep. 2006.
- [5] A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE J. Robot Autom.*, vol. 3, no. 3, pp. 249–265, Jun. 1987.
- [6] J. Mullane, M. D. Adams, and W. S. Wijesoma, "Robotic mapping using measurement likelihood filtering," *Int. J. Robot Res.*, vol. 28, no. 2, pp. 172–190, 2009.
- [7] A. Foessel-Bunting, J. Bares, and W. L. Whittaker, "Three-dimensional map building with mmw radar," in *Proc. Int. Conf. Field Service Robotics*, R. C. A. Halme and E. Prassler, Eds., Helsinki, Finland, Jun. 2001.
- [8] C. Lundquist, T. B. Schön, and U. Örguner, "Estimation of the free space in front of a moving vehicle," *Electr. Eng. Dept., Linköping Univ., Linköping, Sweden, Tech. Rep. LiTH-ISY-R-2892*, Apr. 2009.
- [9] M. Buehler, K. Iagnemma, and S. Singh, Eds., *J. Field Robot. (Special Issue on the 2007 DARPA Urban Challenge, Part I-III)*, vol. 25, no. 8–10, 2008.
- [10] K. Kaliyaperumal, S. Lakshmanan, and K. Kluge, "An algorithm for detecting roads and obstacles in radar images," *IEEE Trans. Veh. Technol.*, vol. 50, no. 1, pp. 170–182, Jan. 2001.
- [11] S. Lakshmanan, K. Kaliyaperumal, and K. Kluge, "Lexluther: An algorithm for detecting roads and obstacles in radar images," in *Proc. IEEE Conf. Intelligent Transp. Syst.*, Boston, MA, Nov. 1997, pp. 415–420.
- [12] M. Nikolova and A. Hero, "Segmentation of a road from a vehicle-mounted radar and accuracy of the estimation," in *Proc. Intelligent Vehicles Symp.*, Dearborn, MI, Oct. 2000, pp. 284–289.
- [13] B. Ma, S. Lakshmanan, and A. Hero, "Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 3, pp. 135–147, Sep. 2000.
- [14] A. Kirchner and T. Heinrich, "Model based detection of road boundaries with a laser scanner," in *Proc. IEEE Intelligent Vehicles Symp.*, Stuttgart, Germany, Oct. 1998, pp. 93–98.
- [15] A. Kirchner and C. Ameling, "Integrated obstacle and road tracking using a laser scanner," in *Proc. Intelligent Vehicles Symp.*, Dearborn, MI, Oct. 2000, pp. 675–681.
- [16] J. Sparbert, K. Dietmayer, and D. Streller, "Lane detection and street type classification using laser range images," in *Proc. IEEE Conf. Intelligent Transp. Syst.*, Oakland, CA, Aug. 2001, pp. 454–459.
- [17] C. Lundquist, U. Örguner, and F. Gustafsson, "Extended target tracking using polynomials with applications to road-map estimation," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 1–12, Jan. 2011.
- [18] O. Erdinc, P. Willett, and Y. Bar-Shalom, "The bin-occupancy filter and its connection to the PHD filters," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4232–4246, Nov. 2009.
- [19] R. Mahler, "A theoretical foundation for the stein-winter probability hypothesis density (PHD) multitarget tracking approach," presented at the Nat. Symp. Sensor Data Fusion, San Antonio, TX, Jun. 2000.
- [20] R. Mahler, "Multi-target bayes filtering via first-order multi-target moments," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1152–1178, 2003.
- [21] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Boston, MA: Artech House, 2007.
- [22] J. Mullane, B.-N. Vo, M. D. Adams, and W. S. Wijesoma, "A random set formulation for bayesian SLAM," in *Proc. IEEE Int. Conf. Intelligent Robots Syst.*, Nice, France, Sep. 2008, pp. 1043–1049.
- [23] A. Eidehall, J. Pohl, F. Gustafsson, and J. Eklund, "Toward autonomous collision avoidance by steering," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 1, pp. 84–94, Mar. 2007.
- [24] R. Smith, M. Self, and P. Cheeseman, "A stochastic map for uncertain spatial relationships," in *Proc. Int. Symp. Robotics Res.*, Cambridge, MA, 1988, pp. 467–474.
- [25] H. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Mag.*, vol. 9, no. 2, pp. 61–74, 1988.
- [26] D. J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes. Vol. 1. Elementary Theory and Method*, 2nd ed. New York: Springer, 2003.
- [27] K. Gilholm, S. Godsill, S. Maskell, and D. Salmond, "Poisson models for extended target and group tracking," in *Proc. Signal Data Process. Small Targets*, San Diego, CA, Aug. 2005, vol. 5913, pp. 230–241.
- [28] K. Gilholm and D. Salmond, "Spatial distribution model for tracking extended objects," *Proc. Inst. Electr. Eng.—Radar, Sonar, Navig.*, vol. 152, no. 5, pp. 364–371, Oct. 2005.
- [29] B.-N. Vo, S. Singh, and A. Doucet, "Random finite sets and sequential monte carlo methods in multi-target tracking," in *Proc. Int. Radar Conf.*, Adelaide, Australia, Sep. 2003, pp. 486–491.

- [30] T. Zajic and R. Mahler, "Particle-systems implementation of the PHD multitarget-tracking filter," in *Proc. Signal Process., Sensor Fusion, Target Recog. XII*, Orlando, FL, Apr. 2003, vol. 5096, pp. 291–299.
- [31] H. Sidenbladh, "Multi-target particle filtering for the probability hypothesis density," in *Proc. 6th Int. Conf. Inf. Fusion*, Cairns, Queensland, Australia, Mar. 2003, vol. 2, pp. 800–806.
- [32] B.-N. Vo and W.-K. Ma, "The gaussian mixture probability hypothesis density filter," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4091–4104, Nov. 2006.
- [33] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. IT-28, no. 2, pp. 129–137, Mar. 1982.
- [34] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006.
- [35] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York: Springer, 2009.
- [36] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley-Interscience, 2001.
- [37] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Stat. Probability*, Berkeley, CA, Jul. 1967, vol. 1, pp. 281–297.
- [38] J. Roll, "Local and piecewise affine approaches to system identification," Ph.D. dissertation No. 802, Linköping Studies in Sci. and Technol., Linköping, Sweden, Apr. 2003.
- [39] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari, "A clustering technique for the identification of piecewise affine systems," *Automatica J. IFAC*, vol. 39, no. 2, pp. 205–217, Feb. 2003.
- [40] V. Petridis and A. Kehagias, "Identification of switching dynamical systems using multiple models," in *Proc. IEEE Conf. Decision Control*, Tampa, FL, Dec. 1998, vol. 1, pp. 199–204.
- [41] C. Lundquist and T. B. Schön, "Estimation of the free space in front of a moving vehicle," in *Proc. SAE World Congr.*, Detroit, MI, Apr. 2009, ser. SAE paper 2009-01-1288.
- [42] D. Clark and B.-N. Vo, "Convergence analysis of the Gaussian mixture PHD filter," *IEEE Trans. Signal Process.*, vol. 55, no. 4, pp. 1204–1212, Apr. 2007.
- [43] H. W. Sorenson and D. L. Alspach, "Recursive Bayesian estimation using Gaussian sum," *Automatica J. IFAC*, vol. 7, pp. 465–479, Jul. 1971.
- [44] D. J. Salmond, "Mixture reduction algorithms for target tracking in clutter," *Proc. Signal Data Process. Small Targets*, vol. 1305, no. 1, pp. 434–445, Jan. 1990.
- [45] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: Wiley, 2001.
- [46] J. Williams and P. Maybeck, "Cost-function-based gaussian mixture reduction for target tracking," in *Proc. Int. Conf. Inf. Fusion*, Cairns, Australia, Jul. 2003, vol. 2, pp. 1047–1054.
- [47] D. Schieferdecker and M. Huber, "Gaussian mixture reduction via clustering," in *Proc. Int. Conf. Inf. Fusion*, Seattle, WA, Jul. 2009, pp. 1536–1543.
- [48] M. Huber and U. Hanebeck, "Progressive gaussian mixture reduction," in *Proc. Int. Conf. Inf. Fusion*, Cologne, Germany, Jul. 2008, pp. 1–8.
- [49] E. D. Dickmanns and A. Zapp, "A curvature-based scheme for improving road vehicle guidance by computer vision," in *Proc. SPIE Conf. Mobile Robots*, Cambridge, MA, 1986, vol. 727, pp. 161–198.
- [50] A. Eidehall, J. Pohl, and F. Gustafsson, "Joint road geometry estimation and vehicle tracking," *Control Eng. Pract.*, vol. 15, no. 12, pp. 1484–1494, Dec. 2007.

- [51] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.



**Christian Lundquist** received the M.Sc. degree in automation and mechatronics engineering from Chalmers University of Technology, Gothenburg, Sweden, in 2003. He is currently working towards the Ph.D. degree at the Division of Automatic Control in the Department of Electrical Engineering of Linköping University, Linköping, Sweden.

Between 2004 and 2007, he worked with active steering systems at ZF Lenksysteme GmbH, Germany. His research interests include sensor fusion and target tracking for automotive applications.



**Lars Hammarstrand** received the M.Sc. and Ph.D. degree in electrical engineering from Chalmers University of Technology, Göteborg, Sweden, in 2004 and 2010, respectively.

He is currently with the Active Safety and Chassis Department, Volvo Car Corporation, Göteborg, working as a sensor data fusion specialist for automotive active safety systems. His main research interests are in the fields of target tracking and radar sensor modeling.



**Fredrik Gustafsson** (S'91–M'93–SM'05) received the M.Sc. degree in electrical engineering and the Ph.D. degree in automatic control, both from Linköping University, Linköping, Sweden, in 1988 and 1992, respectively.

From 1992 to 1999, he held various positions in automatic control, and from 1999 to 2005 he was a Professor in communication systems at Linköping University, where he has been a Professor in sensor informatics at the Department of Electrical Engineering since 2005. His research interests are in stochastic

signal processing, adaptive filtering, and change detection, with applications to communication, vehicular, airborne, and audio systems. His work in the sensor fusion area involves design and implementation of nonlinear filtering algorithms for localization, navigation, and tracking of all kind of platforms, including cars, aircraft, spacecraft, UAVs, surface and underwater vessels, cell phones, and film cameras for augmented reality. He is a co-founder of the companies NIRA Dynamics and Softube, developing signal processing software solutions for automotive and music industry, respectively.

Dr. Gustafsson was an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2000 to 2006 and is currently Associate Editor for the EURASIP Journal on Applied Signal Processing and International Journal of Navigation and Observation. In 2004, he was awarded the Arnberg prize by the Royal Swedish Academy of Science (KVA), and in 2007 he was elected member of the Royal Academy of Engineering Sciences (IVA).