Bayesian road estimation using on-board sensors

Ángel F. García-Fernández, Lars Hammarstrand, Maryam Fatemi, Lennart Svensson

Abstract—This paper describes an algorithm for estimating the road ahead of a host vehicle based on the measurements from several on-board sensors: a camera, a radar, wheel speed sensors and an inertial measurement unit. We propose a novel road model that is able to describe the road ahead with higher accuracy than the usual polynomial model. We also develop a Bayesian fusion system that uses the following information from the surroundings: lane marking measurements obtained by the camera, leading vehicles and stationary objects measurements obtained by a radar-camera fusion system. The performance of our fusion algorithm is evaluated in several drive tests. As expected, the more information we use, the better the performance is.

Index Terms—Information fusion, camera, radar, unscented Kalman filter, road geometry.

I. INTRODUCTION

Active safety systems on vehicles are becoming more and more advanced and nowadays they are able to assist the driver in complicated scenarios [1]. These systems use noisy observations from on-board sensors, such as radar and camera, to perceive the current traffic situation. Based on this description, the system detects dangerous situations and makes decisions on how to assist by means of warnings or autonomous interventions. While some older systems mainly focus on keeping track of other vehicles [2], knowing the geometry of the road has become important to handle more complex situations, e.g., higher speeds and earlier interventions [3], [4].

Systems for estimating the geometry of the lane markings in relation to the host vehicle using a camera sensor have been around for a while [5]–[7]. However, extracting road geometry information from a camera sensor, typically mounted on the windscreen of the vehicle, will always suffer from poor effective resolution of the lane markings at far distances due to the projection of the roughly horizontal road onto the vertical camera sensor. Accurate estimation of the road at long ranges is of importance in highways due to the possible high speed of the vehicles. Therefore, in order to be able to attain a high enough accuracy at these distances, information coming from a different kind of sensor must be used.

Other systems estimate the road using a radar or lidar sensor. Radars and lidars can estimate object positions located at close

and long ranges with high precision. This information can be used to estimate the shape of the road. For example, in [8]–[10], the road is estimated using stationary object detections from a radar and in [11], [12] from a lidar. In road safety systems, radars are more commonly used than lidars due to their longer range and better performance in bad weather [11]. However, in clear weather conditions, the camera can obtain quite detailed information for close range that radars and lidars cannot obtain, e.g., lane markings.

To overcome the drawbacks of camera and radar sensors, camera-radar fusion systems have been developed [13]–[15]. In [13], [14], measurements from the leading vehicles are taken into account to estimate the road. However, the described systems are not complete as they do not use radar measurements from the stationary objects such as barriers. In [15], the system uses lane marking measurements and measurements from the barriers but measurements from leading vehicles are not taken into account.

In this paper, we present an algorithm for estimating the road ahead of a host vehicle using the observations of the lane markings, leading vehicles and barriers that the on-board camera and radar sensors provide. To the authors' knowledge, such an integrated approach to estimate the road has not been proposed in the existing literature. The algorithm uses the Bayesian framework, in which the road is modelled as a random variable, to fuse the information from all these measurements. In this approach, the road estimate is obtained using an approximation to the posterior probability density function (PDF), i.e., the PDF of the road conditioned on all past measurements up to the current time.

Other novelties of our paper are in the road model. When fusing information from the leading vehicles, the road is usually described by third-degree polynomial or constant curvature models [13], [14]. These models, which are based on a clothoid approximation, cannot estimate the road accurately at far distance in important cases, e.g., if a straight road is followed by a sharp curve at a relatively far range [14]. Therefore, the model itself prevents these systems from performing properly in such cases. The model we propose in this paper is based on sampling a continuous curve and describes the road using the curvature at the sampled points. This model is capable of describing any kind of road for a sufficiently small sampling distance. In addition, the Bayesian framework requires some prior knowledge about the road. In our case, we use the fact that in most roads curvature does not change abruptly [16]. This can be easily incorporated in our road model. Another contribution of this paper in road modelling is that we use an extended road model in which the barriers are also considered in the posterior PDF. Apart from the intrinsic interest of active safety systems in knowing the barriers, the

Table I: Radar specifications

Parameter	Value
Frequency	76.5 GHz
Accuracy in range	0.5 m
Accuracy in range rate	0.12 m/s
Accuracy in angle	0.5°

Table II: Camera specifications

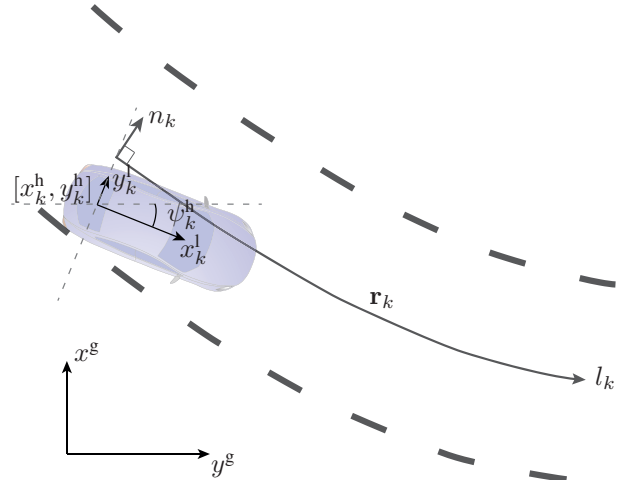| Parameter	Value
Resolution	640 x 480 (VGA)
Frame rate	10 Hz

Figure 1: Illustration of the coordinate systems and the geometry of the road. The figure shows two Cartesian coordinate systems, the fixed global one given as (x_g, y_g) and the moving local one attached to the host vehicle denoted as (x_k^l, y_k^l). The so-called curved road coordinate system (l_k, n_k) is also represented.

extended road model makes it easier for the fusion system to use the information from the stationary object measurements to estimate the road.

In short, fusing the information coming from the lane markings, stationary objects and leading vehicles as well as using a road model that can represent any kind of road allows the system we present in this paper to improve the performance of other systems previously reported in the literature, especially, at far distance.

The rest of the paper is organised as follows. We describe our system and pose the road estimation problem in Section II. The road model and the theoretical solution to the problem are provided in Section III. The process and measurement models we use are given in Section IV. Section V explains how the theoretical solution to the problem is approximated. We provide some experimental results showing the performance of our algorithm in Section VI. Finally, conclusions are drawn in Section VII.

II. PROBLEM FORMULATION AND SYSTEM DESCRIPTION

This paper is concerned with the problem of estimating the geometry of the road ahead of a host vehicle equipped with a set of sensors. In our case, the host vehicle is fitted with a radar sensor in the grill and a camera sensor behind the windscreen. Some specifications of these sensors provided by the supplier are given in Tables I and II. The radar is able to detect objects up to approximately 200 m and the camera information is typically able to describe the lane markings up to 60 m. In addition, data from internal sensors, such as wheel speed sensors and inertial measurement unit, are available. The camera sensor is capable of detecting the left and right lane markings and gives a description of their shape while the radar sensor detects and tracks other vehicles as well as stationary objects on the side of the road, e.g., guard rails and barriers.

The objective is to accurately describe the geometry of the road at the current time up to 200 m in front of the host vehicle using all the relevant observations from the on-board sensor system. The focus is put on describing highway type roads having relatively sharp curves. The work is limited to only one road, i.e., exits or forks in the road are not included in the models. Slopes are not explicitly considered either as is usually the case [10], [13], [17], [18] so the road is represented as a plane curve. The reason why slopes are not taken into account despite their importance in some scenarios is that the on-board sensors of our vehicle do not provide information about slopes. If we wanted to include slope information, we

would need sensors that provide this kind of information, e.g., a stereo vision camera [19]. Furthermore, it is assumed that there exists a basic fusion system like the one described in [20] to handle asynchronous sensor data and estimate the host vehicle state.

We define needed notation in Section II-A. The observations from the aforementioned sensors are described in more detail in Section II-B. In Section II-C, we conclude by mathematically defining the estimation problem that is to be solved.

A. Road geometry definition and coordinate systems

In order to describe the geometry of the road we must first define what we mean by road geometry. In this paper we use the following definition:

The geometry of the road is the shape of the middle of the host vehicle lane.

As such, we assume that there is a parameterisation of the geometry of the road at time k given by road state vector $\mathbf{r}_k$, which will be defined in Section III. Vector $\mathbf{r}_k$ describes the shape (geometry) of the middle of the host vehicle lane in the local Cartesian coordinate system (x_k^l, y_k^l). The local coordinate system is attached to the middle of the rear axel of the host vehicle with one axis parallel to this axel as shown in Figure 1. Note that this local coordinate frame moves with the host vehicle. We denote the position and orientation of the host vehicle in a fixed global coordinate system (x_g, y_g) as $\left[x_k^h, y_k^h\right]^T$ and ψ_k^h, respectively, where T stands for transpose. In this paper, we also use the so-called curved road coordinate system (l_k, n_k) [17] in which a point is represented by its longitude l_k on the road and its normal distance n_k to the road. Time index k will be used throughout this paper to refer to a time instance t_k for which the k-th measurement in total (from any of the sensors) was made. As the sensors are asynchronous, the time between two measurements $T_k = t_k - t_{k-1}$ is not constant but known.
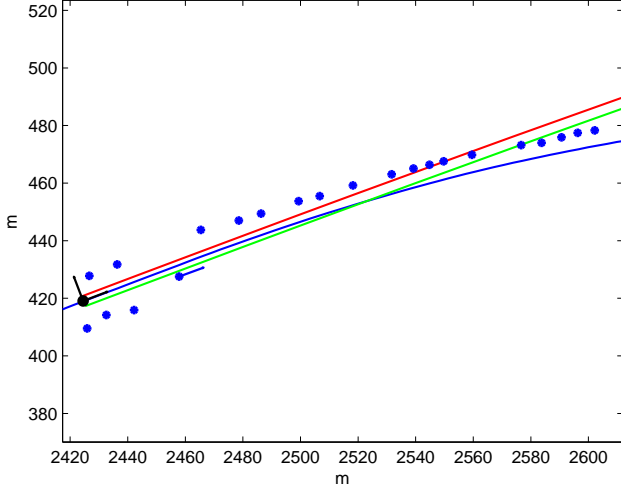
Figure 2: Measurements from the sensors: The host vehicle and its local coordinate system are represented in black. The red and green lines represent the lane marking (polynomial) measurements. The blue dots without an arrow represent the stationary objects detected by the camera/radar. The blue dot with an arrow represents a moving object with its heading detected by the camera/radar. The blue line represents the true road. The lane markings are unable to estimate the curve accurately but we can use the information of the stationary and moving objects to do it.

B. Observations

The observations of the road geometry come from two different types of sensors: a lane tracking camera sensor detecting the lane markings on the road and a radar-camera fusion system observing vehicles and stationary objects (guard rails and barriers). Figure 2 shows a typical set of measurements from the radar and camera sensors. In the following sections we present these observations in more detail.

1) Lane marking measurement: The lane tracking camera delivers lane marking measurements roughly every 100 ms. After some pre-processing, our system receives the coefficients $\mathbf{z}_k^l = \left[l_k^0, l_k^1, l_k^2, l_k^3\right]^T$ and $\mathbf{z}_k^r = \left[r_k^0, r_k^1, r_k^2, r_k^3\right]^T$ of two third order polynomials describing the shape of the lane markings to the left and right of the host vehicle, respectively. The polynomials are given in the local coordinate system $\left(x_k^l, y_k^l\right)$.

2) Object measurements: The host vehicle is equipped with a radar-camera fusion system that detects and tracks other vehicles as well as stationary road side objects that reflect radar energy, e.g., guard rails. The observations are delivered every 25 ms in two vectors, one for vehicles and one for stationary unclassified objects with a total maximum of 64 items. We denote these vectors $\mathbf{z}_k^o$ and $\mathbf{z}_k^s$, respectively. This segmentation is performed based on the speed over ground and camera classification.

For each observed vehicle i, a measurement $\mathbf{z}_k^{o,i} = \left[x_k^i, y_k^i, \phi_k^i, v_k^i\right]^T$ is reported by the sensor where $\left[x_k^i, y_k^i\right]^T$ is the position in the local coordinate frame and ϕ_k^i is the heading angle relative to the heading of the host and v_k^i is the speed in that direction. Similarly, for the stationary unclassified

objects, the sensor gives a measurement $\mathbf{z}_k^{s,i} = \left[x_k^i, y_k^i\right]^T$ that is the position of the stationary radar detection. It should be noted that these observations are already filtered by the radar-camera system and we therefore do not include vehicle states in our state vector.

C. Estimation problem

The main objective of this paper is to estimate $\mathbf{r}_k$ at time step k given the measurements up to time step k. We pursue this aim using the Bayesian approach in which the variables of interest are modelled as random variables. In this approach, inference is done by recursively approximating the posterior PDF $p\left(\mathbf{r}_k \left| \mathbf{z}_{1:k}\right.\right)$ of the road state $\mathbf{r}_k$ given the sequence of measurements $\mathbf{z}_{1:k}$, which denotes all the measurements from the lane tracker and the radar-camera fusion system up to the current time k.

III. STATE PARAMETERISATION AND THEORETICAL SOLUTION

In Section III-A, we introduce the extended road model, which includes the road geometry and the barriers. In Section III-B, we indicate how the estimation problem indicated in Section II-C is solved theoretically. Some assumptions about our model and a brief discussion is given in Section III-C.

A. Road model

Our model of the road ahead of the vehicle is illustrated in Figure 3. It consists of M samples of a continuous 2-D curve that has been sampled with sampling distance Δ. The sampled curve is parameterised in the local coordinate system and is described at time step k by vector $\mathbf{r}_k = \left[y_k^1, \varphi_k, c_k^2, c_k^3, ..., c_k^{M-1}\right]^T$ where y_k^1 is the lateral offset between the host vehicle and the road, φ_k is its initial heading and c_k^i is the sampled curvature, which is a concept explained in the Appendix, at the ith point. Given $\mathbf{r}_k$, we can obtain the position vector of the road $\mathbf{p}_k = \left[\left(\mathbf{p}_k^1\right)^T, \left(\mathbf{p}_k^2\right)^T, ..., \left(\mathbf{p}_k^M\right)^T\right]^T$ where $\mathbf{p}_k^i = \left[x_k^i, y_k^i\right]^T$ is the position of the ith sampled point of the road at time k. It should be noted that $\mathbf{p}_k^1 = \left[0, y_k^1\right]^T$, $\mathbf{p}_k^2 = \left[\Delta \cos \varphi_k, y_k^1 + \Delta \sin \varphi_k\right]^T$ and

$$\mathbf{p}_k^i = \mathbf{g}_p\left(\mathbf{p}_k^{i-2}, \mathbf{p}_k^{i-1}, c_k^{i-1}\right) \text{ for } i \geq 3 \qquad (1)$$

where $\mathbf{g}_p\left(\cdot\right)$ is given by (25) in the Appendix. The function that relates $\mathbf{p}_k$ and $\mathbf{r}_k$ is denoted as $\mathbf{p}_k = \mathbf{g}\left(\mathbf{r}_k\right)$. It should be noted that we can obtain $\mathbf{r}_k$ given $\mathbf{p}_k$ and that the length of the road is $(M-1)\Delta$. We introduce this model based on curvature because of the importance of curvature in road design. We will use the prior knowledge we have about curvature in roads to model the prediction step in Section IV-A.

In order to utilise stationary road side detections from the radar to estimate $\mathbf{r}_k$, we propose to include a description of the road side barriers, which is also illustrated in Figure 3. In this paper we assume that, if a barrier exists, it is parallel to the road. As a result, the barrier state at time k is parameterised by vector $\mathbf{b}_k = \left[e_k^l, b_k^l, e_k^r, b_k^r\right]^T$ where $e_k^l = 1$ if the left barrier
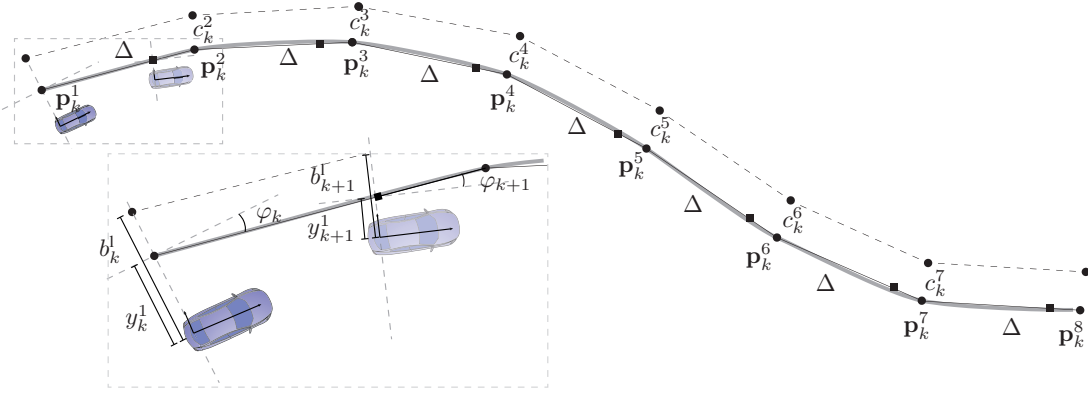
Figure 3: Road model with left barrier and prediction step: The road is a curve sampled with distance Δ. The road is characterised by the initial offset y_k^1, initial heading φ_k and the curvatures at the next points. It is represented as a grey line. The left barrier is assumed to be parallel to the road and characterised by the initial offset b_k^l. It is represented as a dashed grey line. The augmented road position vector $\mathbf{p}_k^a$ is represented by circles. At time step $k+1$, the local coordinate system moves. In order to obtain the parametrisation of the road required in $\mathbf{r}_{k+1}$, we have to do a change of coordinates and interpolate the road. The road position vector $\mathbf{p}_{k+1}$ is represented by squares. Note that we have one point less in $\mathbf{p}_{k+1}$ than in $\mathbf{p}_k^a$ as required.

exists and zero otherwise, b_k^l is the lateral offset (in the local coordinate system) from the center of the host vehicle lane to the left barrier, and equivalently for e_k^r and b_k^r for the right barrier.

B. Posterior PDF

Instead of approximating $p\left(\mathbf{r}_k \,|\mathbf{z}_{1:k}\right)$ directly, we approximate the joint posterior PDF $p\left(\mathbf{r}_k, \mathbf{b}_k \,|\mathbf{z}_{1:k}\right)$ of the extended road model that accounts for the barriers. This makes it easier to take into account the radar measurements from the barriers.

Every time we receive a measurement, the posterior PDF is calculated in two steps: prediction and update [21]. The prediction step uses the transition density $p\left(\mathbf{r}_{k+1}, \mathbf{b}_{k+1} \,|\mathbf{r}_k, \mathbf{b}_k\right)$ and the Chapman–Kolmogorov equation to calculate the predicted PDF at the next time step

$$p\left(\mathbf{r}_{k+1}, \mathbf{b}_{k+1} \,|\mathbf{z}_{1:k}\right) = \int p\left(\mathbf{r}_{k+1}, \mathbf{b}_{k+1} \,|\mathbf{r}_k, \mathbf{b}_k\right)$$
$$p\left(\mathbf{r}_k, \mathbf{b}_k \,|\mathbf{z}_{1:k}\right) d\mathbf{r}_k d\mathbf{b}_k \qquad (2)$$

The update step uses Bayes' rule, the predicted PDF and the likelihood $p\left(\mathbf{z}_{k+1} \,|\mathbf{r}_{k+1}, \mathbf{b}_{k+1}\right)$ to calculate the posterior

$$p\left(\mathbf{r}_{k+1}, \mathbf{b}_{k+1} \,|\mathbf{z}_{1:k+1}\right) \propto p\left(\mathbf{z}_{k+1} \,|\mathbf{r}_{k+1}, \mathbf{b}_{k+1}\right)$$
$$p\left(\mathbf{r}_{k+1}, \mathbf{b}_{k+1} \,|\mathbf{z}_{1:k}\right) \qquad (3)$$

where $\propto$ means "is proportional to". Therefore, in order to solve the problem, we need to model the transition density and the likelihood for each kind of measurement. This is equivalent to providing the process and measurement equations, which are given in Section IV. Besides, we also need to approximate (2) and (3) as they do not admit a closed-form expression in general. This is explained in Section V.

C. Assumptions and discussion

The first important assumption we make in this paper is that we know the difference $\mathbf{p}_k^{hl}$ between the host vehicle pose (position and heading) at time $k+1$ and the host vehicle pose at time k in the local coordinate system at time k as in

[10]. This is equivalent to saying that, given the host pose at time k and the measurements of the internal sensors (wheel speed sensors and inertial measurement unit), the uncertainty of the host pose at time $k+1$ is negligible in the sense that it does not make a big difference in road estimation to consider $\mathbf{p}_k^{hl}$ a random variable or a known parameter. This uncertainty is actually quite low as the internal sensor measurements of the velocity, yaw rate and lateral acceleration produce a very accurate estimate of $\mathbf{p}_k^{hl}$ [22]. Therefore, this assumption is reasonable.

It should be noted that lower errors could be achieved by modelling $\mathbf{p}_k^h$ as a random variable and including it in the state vector. However, the computational burden of this approach is considerably higher than the method explained in this paper. This option is rather similar to simultaneous localisation and mapping (SLAM) [23]. However, SLAM aims to estimate the global map (the global road for a static observer) which is not our objective.

The second important assumption we make is that we consider the inputs to our systems, i.e., lane marking observations ($\mathbf{z}_k^l$, $\mathbf{z}_k^r$), moving object observations $\mathbf{z}_k^o$ and stationary object observations $\mathbf{z}_k^s$ as independent measurements of a sequence of underlying vectors $\mathbf{r}_k$ $k = 0, ..., \infty$. That is, we do not take into account that these observations are already the output of the filtering algorithms carried out in the radar-camera sensor. As we do not have access to the details of these filtering algorithms (as this was done by the sensor provider), we treat them as independent measurements. Despite the invalidity of this assumption, road estimation is greatly improved with respect to the lane marking measurements due to the fusion of all these kinds of information. It should be noted however that higher performance is expected if we design robust fusion schemes that account for unknown correlations of the observations along time, for example, following the ideas put forward in [24]. Nevertheless, this is beyond the scope of this paper.

IV. PROCESS AND MEASUREMENT MODELS

In this section, we introduce the process and measurement equations used in (2) and (3).

A. Process equation

Most roads are built such that there are not abrupt changes in their curvature [16], [25]. Therefore, given the curvature at a certain point, the curvature at a nearby point is expected to be close. How similar it is depends on the road type, i.e., roads with higher speed limit have smoother changes in curvature. We use this prior knowledge in our Bayesian framework as it can be easily modelled as a Markov random process characterised by

$$c_k^{i+1} = c_k^i + w^{c,i+1} \qquad (4)$$

where $w^{c,i+1}$ is a sequence of independent zero-mean Gaussian noises with variance σ_c^2. It should be noted that σ_c^2 depends on the type of road so it should be chosen accordingly. The model is valid to represent roads with sharp bends by making σ_c^2 high enough.

The process equation, which is a characterisation of the transition density in (2), includes a transformation of the road state vector $\mathbf{r}_k$ and the barrier state vector $\mathbf{b}_k$ due to the change of the local coordinate system. We assume that the host vehicle moves a distance lower than Δ from time step k to $k+1$. First, we add a component to the state vector $\mathbf{r}_k$ with the curvature of the next point (located at a distance Δ from the last point). The augmented state vector is represented as $\mathbf{r}_k^a = \left[\mathbf{r}_k^T, c_k^M\right]^T = \left[y_k^1, \varphi_k, c_k^2, c_k^3, ..., c_k^{M-1}, c_k^M\right]^T$. Note that the prior knowledge we have about c_k^M given $\mathbf{r}_k$ is modelled by (4).

In order to account for modelling errors, it is convenient to add process noise to the transformation. Moreover, due to the fact that different measurements are taken asynchronously, the process noise covariance matrix is proportional to the time difference between measurements. Thus, the process equation for the road is modelled by

$$\mathbf{r}_{k+1} = \mathbf{f}_{k+1}\left(\mathbf{r}_k^a\right) + \mathbf{w}_{k+1}^r \qquad (5)$$

where $\mathbf{f}_{k+1}\left(\cdot\right)$ is a nonlinear transformation to be described in the rest of the section, $\mathbf{w}_{k+1}^r$ is a sequence of independent zero-mean Gaussian process noise with covariance matrix $\mathbf{Q}_{k+1}^r = T_{k+1}\tilde{\mathbf{Q}}^r$ where T_{k+1} is the time difference between the measurement at time $k+1$ and the measurement at time k and $\tilde{\mathbf{Q}}^r$ is a reference covariance matrix.

We proceed to describe $\mathbf{f}_{k+1}\left(\cdot\right)$, which consists of the composition of several functions. The procedure is illustrated in Figure 3. The road position vector of the augmented road is $\mathbf{p}_k^a = \mathbf{g}^a\left(\mathbf{r}_k^a\right) = \left[\left(\mathbf{p}_k^1\right)^T, \left(\mathbf{p}_k^2\right)^T, ..., \left(\mathbf{p}_k^M\right)^T, \left(\mathbf{p}_k^{M+1}\right)^T\right]^T$. We denote the augmented road position vector in the local coordinate system at time $k+1$ as $\mathbf{p}_{k+1|k}^a = \tilde{\mathbf{f}}_{k+1}\left(\mathbf{p}_k^a\right)$, where function $\tilde{\mathbf{f}}_{k+1}\left(\cdot\right)$ represents the change of local coordinate systems from time k to $k+1$. Thus, function $\tilde{\mathbf{f}}_{k+1}\left(\cdot\right)$ is a

translation followed by a rotation applied to each point of the road:

$$\begin{aligned}
\mathbf{p}_{k+1|k}^a &= \tilde{\mathbf{f}}_{k+1}\left(\mathbf{p}_k^a\right) \\
&= \left[\breve{\mathbf{f}}_{k+1}^T\left(\mathbf{p}_k^1\right), ..., \breve{\mathbf{f}}_{k+1}^T\left(\mathbf{p}_k^{M+1}\right)\right]^T
\end{aligned} \qquad (6)$$

where

$$\begin{aligned}
\breve{\mathbf{f}}_{k+1}\left(\mathbf{p}_k^i\right) = &\,\Theta\left(-\psi_{k+1}^h + \psi_k^h\right) \\
&\left[\mathbf{p}_k^i - \left(\left[x_{k+1}^h, y_{k+1}^h\right]^T - \left[x_k^h, y_k^h\right]^T\right)\right]
\end{aligned} \qquad (7)$$

where $\Theta\left(\alpha\right)$ is a rotation matrix with angle α and $\left[x_k^h, y_k^h, \psi_k^h\right]^T$ is the host state at time k.

To obtain the road position vector $\mathbf{p}_{k+1}$ we need to interpolate $\mathbf{p}_{k+1|k}^a$ because the first point in $\mathbf{p}_{k+1}$ must have a coordinate of the form $[0, y]^T$ as required by our road model, which is described in Section III-A. We use linear interpolation and the function is denoted as $\mathbf{p}_{k+1} = \mathbf{i}\left(\mathbf{p}_{k+1|k}^a\right)$, which can be transformed to the road state $\mathbf{r}_{k+1}$ using $\mathbf{g}^{-1}\left(\cdot\right)$. In short, the prediction step for the road state is illustrated in Figure 3 and can be written as

$$\mathbf{r}_{k+1} = \mathbf{g}^{-1}\left(\mathbf{i}\left(\tilde{\mathbf{f}}_{k+1}\left(\mathbf{g}^a\left(\mathbf{r}_k^a\right)\right)\right)\right) + \mathbf{w}_{k+1}^r \qquad (8)$$

If the left barrier exists, the process equation for the barrier is

$$b_{k+1}^l = b_k^l \cos\left(\psi_{k+1}^h - \psi_k^h\right) + w_{k+1}^b \qquad (9)$$

where w_{k+1}^b is a sequence of independent zero-mean Gaussian process noise with variance $Q_{k+1}^b = T_{k+1}\tilde{Q}^b$ where $\tilde{Q}^b$ is a reference variance. The right barrier has an analogous process equation.

B. Measurement equations

In this section, we introduce the measurement equation for the different types of observations: lane markings, moving objects and stationary objects.

1) Lane markings: The road state represents the middle of the host vehicle lane. The middle of the lane at a given distance is the average of the left and right lane markings at a given distance. Therefore, the average of the measured polynomials (left and right lane markings), whose coefficients are given by $\left(\mathbf{z}_{k+1}^l + \mathbf{z}_{k+1}^r\right)/2$, is a direct measurement of the road, see Figure 2.

We recall that the road state $\mathbf{r}_{k+1}$ can be transformed into a road position vector $\mathbf{p}_{k+1}$, where the distance between consecutive points is Δ. There is only one third order polynomial that passes through four points. Therefore, if we sample four points from the averaged lane marking polynomials, we keep the information of the measurement. If these four points are selected such that the distance between consecutive points is Δ, represented in vector $\mathbf{p}_{k+1}^l$, then, we are observing the first four positions of $\mathbf{p}_{k+1}$ plus measurement noise. Therefore, we assume that

$$\mathbf{p}_{k+1}^l = \mathbf{g}^4\left(\mathbf{r}_{k+1}\right) + \boldsymbol{\eta}_{k+1}^l \qquad (10)$$

where $\mathbf{g}^4\left(\cdot\right)$ represents the function that takes the first four points of function $\mathbf{g}\left(\cdot\right)$ and $\boldsymbol{\eta}_{k+1}^l$ is a zero-mean Gaussian measurement noise with covariance matrix $\mathbf{R}^l$.

2) Moving objects: The assumption we make to get information about the road from other vehicles' measurements is the following:

- M1 If a vehicle is not changing lane, its heading is roughly parallel to the road.

This assumption was also made in [13]. Thus, if a moving object measurement comes from a leading vehicle that is not changing lane with position $\left[x_{k+1}^i, y_{k+1}^i\right]^T$ and heading ϕ_{k+1}^i, the measurement equation is modelled as

$$\phi_{k+1}^i = \phi_{k+1}^r \left(x_{k+1}^i, y_{k+1}^i, \mathbf{r}_{k+1}\right) + \eta_{k+1}^m \qquad (11)$$

where η_{k+1}^m is the measurement noise which is zero-mean Gaussian with variance R^m and $\phi_{k+1}^r\left(x_{k+1}^i, y_{k+1}^i, \mathbf{r}_{k+1}\right)$ is the heading of the road in the road interval that is the closest to the vehicle position.

We explain how $\phi_{k+1}^r(\cdot)$ is calculated in the following. We recall that the road can be represented as the position vector $\mathbf{p}_{k+1}$. We calculate the two consecutive points of this vector that are the closest to $\left[x_{k+1}^i, y_{k+1}^i\right]^T$. Let $\left[x_{k+1}^j, y_{k+1}^j\right]^T$ and $\left[x_{k+1}^{j+1}, y_{k+1}^{j+1}\right]^T$ denote these points, then

$$\phi_{k+1}^r\left(x_{k+1}^i, y_{k+1}^i, \mathbf{r}_{k+1}\right) = \arctan\left(\frac{y_{k+1}^{j+1} - y_{k+1}^j}{x_{k+1}^{j+1} - x_{k+1}^j}\right) \qquad (12)$$

In practice, we have to determine whether a measurement comes from a vehicle that meets M1, i.e., we have to detect if a vehicle is likely to follow the road or not. The measurements of the vehicles that are not deemed to follow the road are discarded and hence not used in the estimation of the road geometry. This is addressed in Section V-C.

3) Stationary objects: If the ith stationary measurement $\mathbf{z}_{k+1}^{s,i}$ comes from the left barrier (the argument for the right barrier is analogous), $\mathbf{z}_{k+1}^{s,i}$ is an observation of the barrier at the longitude l_{k+1}^i on the road of this observation, which is assumed to be known, plus measurement noise[1]. As the barrier is modelled as parallel to the road, i.e., it is described in the so-called road coordinate system, we think it is more convenient to use a measurement noise that is described in this coordinate system rather than the local coordinate system. Therefore, the measurement equation is modelled as

$$\mathbf{z}_{k+1}^{s,i} = \mathbf{p}_{k+1}^{lb}\left(l_{k+1}^i, \mathbf{r}_{k+1}, \mathbf{b}_{k+1}\right) + \mathbf{B}\left(l_{k+1}^i, \mathbf{r}_{k+1}\right)\boldsymbol{\eta}_{k+1}^b \qquad (13)$$

where $\mathbf{p}_{k+1}^{lb}\left(l_{k+1}^i, \mathbf{r}_{k+1}, \mathbf{b}_{k+1}\right)$ is the position of the left barrier at the longitude l_{k+1}^i along the road of the given measurement, $\mathbf{B}\left(l_{k+1}^i, \mathbf{r}_{k+1}\right)$ is a rotation matrix that performs a rotation with an angle that is the heading of the road at longitude l_{k+1}^i and $\boldsymbol{\eta}_{k+1}^b$ is the measurement noise which has a zero-mean Gaussian PDF with covariance $\mathbf{R}^s$. It should be noted that $\mathbf{B}\left(l_{k+1}^i, \mathbf{r}_{k+1}\right)$ is used as $\boldsymbol{\eta}_{k+1}^b$ is the measurement noise in the road coordinate system and the observation is given in the local coordinate system. The transformation $\mathbf{p}_{k+1}^{lb}\left(l_{k+1}^i, \mathbf{r}_{k+1}, \mathbf{b}_{k+1}\right)$ can be easily performed using linear

---

[1]It should be noted that l_{k+1}^i depends on $\mathbf{r}_{k+1}$ and $\mathbf{z}_{k+1}^{s,i}$ but we assume that once it is calculated, it is a fixed parameter in the measurement equation (13) for simplicity.

interpolation in the road intervals and the angle of rotation of $\mathbf{B}\left(l_{k+1}^i, \mathbf{r}_{k+1}\right)$ is found with a similar equation as (12).

In practice, we have to determine whether a measurement belongs to the right barrier, left barrier or is an outlier. This practical issue among others are addressed in Section V-D.

V. CALCULATION OF THE POSTERIOR PDF APPROXIMATION

Once we have introduced the road model and the process and measurement equations, we proceed to approximating the posterior PDF, which is given by (2) and (3), in a recursive fashion.

The posterior approximation of the extended road we propose is a Gaussian hierarchical model of the form

$$\begin{aligned} p\left(\mathbf{r}_k, \mathbf{b}_k \,|\, \mathbf{z}_{1:k}\right) \approx &\, \mathcal{N}\left(\mathbf{r}_k; \bar{\mathbf{r}}_{k|k}, \boldsymbol{\Sigma}_{k|k}\right) \\ &\left[\delta\left(e_k^l - 1\right) p_k^l \mathcal{N}\left(b_k^l; \bar{b}_{k|k}^l, \left(\sigma_{k|k}^l\right)^2\right)\right. \\ &\left. + \delta\left(e_k^l - 0\right)\left(1 - p_k^l\right)\right] \\ &\left[\delta\left(e_k^r - 1\right) p_k^r \mathcal{N}\left(b_k^r; \bar{b}_{k|k}^r, \left(\sigma_{k|k}^r\right)^2\right)\right. \\ &\left. + \delta\left(e_k^r - 0\right)\left(1 - p_k^r\right)\right] \qquad (14) \end{aligned}$$

where $\bar{\mathbf{r}}_{k|k}$ and $\boldsymbol{\Sigma}_{k|k}$ denote the mean and covariance matrix of the road at time k conditioned on the current and past measurements, $\delta(\cdot)$ is the Kronecker delta, p_k^l and p_k^r are the probabilities that the left and right barriers exist, respectively, $\bar{b}_{k|k}^l$, $\sigma_{k|k}^l$, $\bar{b}_{k|k}^r$, $\sigma_{k|k}^r$ are the mean and standard deviation of the PDF of the left and right barriers conditioned on the current and past measurements. For simplicity, we assume that if a barrier exists, its state is independent of the road state and the other barrier state. Also, in the implementation, we assume that p_k^l and p_k^r can only take values 0 and 1.

In the rest of the section, we assume that $p\left(\mathbf{r}_k, \mathbf{b}_k \,|\, \mathbf{z}_{1:k}\right)$ is given by (14) and provide an approximation to $p\left(\mathbf{r}_{k+1}, \mathbf{b}_{k+1} \,|\, \mathbf{z}_{1:k+1}\right)$ with the same Gaussian hierarchical model. The prediction step, in which (2) is approximated, is explained in Section V-A. The update step, in which (3) is approximated, is explained in Section V-B for the lane markings, in Section V-C for the moving object measurements and in Section V-D for the stationary object measurements.

A. Prediction step

The posterior of $\mathbf{r}_k^a$, which is needed to perform the prediction step as indicated by (8), can be obtained using the posterior of $\mathbf{r}_k$, see (14), and the road transition model, which is given by (4):

$$p\left(\mathbf{r}_k^a \,|\, \mathbf{z}_{1:k}\right) = \mathcal{N}\left(\mathbf{r}_k^a; \left[\bar{\mathbf{r}}_{k|k}^T, \bar{c}_k^{M-1}\right]^T, \begin{bmatrix} \boldsymbol{\Sigma}_{k|k} & \boldsymbol{\xi}_{k|k} \\ \boldsymbol{\xi}_{k|k}^T & \xi_{k|k}^l + \sigma_c^2 \end{bmatrix}\right) \qquad (15)$$

where $\bar{c}_k^{M-1}$ represents the last component of $\bar{\mathbf{r}}_{k|k}^T$, $\boldsymbol{\xi}_{k|k}$ is the last column of $\boldsymbol{\Sigma}_{k|k}$ and $\xi_{k|k}^l$ is the last component of $\boldsymbol{\xi}_{k|k}$.

In the implementation, we calculate a Gaussian approximation to the predicted density

$$p\left(\mathbf{r}_{k+1} \,|\, \mathbf{z}_{1:k}\right) \approx \mathcal{N}\left(\mathbf{r}_{k+1}; \bar{\mathbf{r}}_{k+1|k}, \boldsymbol{\Sigma}_{k+1|k}\right) \qquad (16)$$

where $\bar{\mathbf{r}}_{k+1|k}$ and $\boldsymbol{\Sigma}_{k+1|k}$ are obtained applying the unscented transformation to the PDF (15) and the nonlinear function (8) [26]. The prediction step for the barriers can be done analytically using (9).

B. Update step: lane markings

Before addressing the update step itself, first, we indicate how we detect lane changes of the host vehicle based on lane marking measurements and how the filter takes them into account.

1) Lane change detection: Lane marking measurements allow us to detect lane changes of the host vehicle easily. In this subsection, we indicate how we detect lane changes and how this affects our filter. Every time we receive a lane marking measurement, before doing the update step, we follow these steps.

First, we estimate the width W of the lane by calculating the difference between the zero order coefficients of the polynomials that represent the left and right lane markings, see Section II-B1. We recall that the zero order coefficient is an accurate measurement of the lateral distance to the left and right lane markings, respectively. Second, we estimate the y component of the middle of the current lane at time $k + 1$, which is the average of the zero order coefficients of the left and right polynomials and is denoted as $y_{k+1}^{l,1}$. If the host vehicle does not change lanes, the y component $y_{k+1|k}^1$ of the middle of the predicted lane, which is the first component of $\bar{\mathbf{r}}_{k+1|k}$, is expected to be close to $y_{k+1}^{l,1}$. If the host vehicle changes lanes, the difference between $y_{k+1}^{l,1}$ and $y_{k+1|k}^1$ is roughly the width of the lane. Therefore, we use the following detection rule

$$y_{k+1}^{l,1} - y_{k+1|k}^1 > 0.8W \rightarrow \text{LC to the left}$$
$$y_{k+1}^{l,1} - y_{k+1|k}^1 < -0.8W \rightarrow \text{LC to the right}$$

where LC stands for lane change and the threshold is set to 0.8 times the width of the lane because it provides excellent results in the test drives.

If we change lanes, we account for the lane change in the predicted density by using $\bar{\mathbf{r}}_{k+1|k} \pm [W, 0, 0, ..., 0]^T$ instead of $\bar{\mathbf{r}}_{k+1|k}$ in (16) and $\bar{b}_{k|k}^l \mp W$ instead of $\bar{b}_{k+1|k}^l$ in the predicted density of the left barrier where $\bar{b}_{k+1|k}^l$ is the predicted left barrier. The upper sign in $\pm$ and $\mp$ is used if we change to the left and the lower sign if we do it to the right. The same procedure is performed for the right barrier.

2) Update: We use the unscented Kalman filter (UKF) [26] to approximate the posterior using the measurement equation (10).

It can also happen that we only get one lane marking measurement. In this case, we use the last estimate of the lane width W, which was obtained the last time we got both lane marking measurements, to create a measurement equation of the form (10). Now, the nonlinear measurement function must account for a displacement of $\pm W/2$, where the sign depends on the lane marking we get, in the normal direction of road.

C. Update step: moving object measurements

The moving object measurements mainly come from other vehicles. However, we only consider those measurements that come from vehicles that meet Assumption M1. Therefore, for every moving object measurement, we include an outlier detection step so that we only consider the useful measurements. We recall from Section II-B2 that moving object measurements have position, heading and speed components.

First, we only consider the measurements whose speed is higher than a threshold (we use 5 m/s). Otherwise, they are likely to be outliers as we assume the host vehicle is on a highway. Second, we perform gating [27]. Gating is a widely used technique to detect outliers and works as follows. If the Mahalanobis distance between the measurement and the predicted measurement using the variance of the predicted measurement, which are calculated using the UT and (11), is higher than a threshold (we use 1.5), then the measurement is deemed an outlier and is not considered in the update step. Finally, we build a measurement equation stacking measurement equation (11) for each measurement that meets Assumption M1. The update is then carried out using a UKF.

D. Update step: stationary object measurements

Each stationary object measurement is a position vector $\mathbf{z}_{k+1}^{s,i}$ that contains the coordinates of the ith stationary object in the local coordinate system, see Section II-B2. In practice, a large number of stationary object measurements come from the barriers. Therefore, these measurements can be used to update the extended road state, which includes the barrier states, using the measurement equation (13). However, we should decide which ones of these measurements were originated in the barriers and which ones are outliers. Barrier measurements have the following characteristics that are taken into account in our filter:

- S1: They tend to appear around a line that is parallel to the road (middle of our lane). This is due to the fact that barriers are usually parallel to the road.
- S2: They do not appear isolated. That is, there are usually several barrier measurements in close proximity. This stems from the fact that the barrier is an extended object so several detections are expected.

Prior to explaining how we initialise and update the barrier state, we perform the following steps to determine what measurements are to the left of the road (candidates for updating the left barrier) and which ones to the right. Every time we receive stationary object measurements, we describe the position $\mathbf{z}_{k+1}^{s,i}$ in the road coordinate system, i.e., using the longitude l_{k+1} on the road and distance n_{k+1} to the road [17]. This is represented by the transformation

$$\mathbf{z}_{k+1}^{sr,i} = \mathbf{m}\left(\mathbf{z}_{k+1}^{s,i}, \mathbf{r}_{k+1}\right) \qquad (17)$$

where $\mathbf{z}_{k+1}^{sr,i} = \left[l_{k+1}^i, n_{k+1}^i\right]^T$ and $\mathbf{m}(\cdot)$ is the function that represents the transformation[2]. It should be noted that $\mathbf{r}_{k+1}$ is

---

[2]The mapping of $\mathbf{m}(\cdot)$ can be easily done using linear interpolation between the points that represent the road.

a random variable with PDF given by (16). Therefore, $\mathbf{z}_{k+1}^{sr,i}$ is another random variable whose PDF is approximated as a Gaussian using the UT drawing sigma points from (16).

If $n_{k+1}^i > 0$, the stationary object is located to the left of the road, otherwise, it is located to the right. Therefore, using the Gaussian PDF of $\mathbf{z}_{k+1}^{sr,i}$, we can easily calculate the probability that the stationary object is to the left or the right of the road. If one of these probabilities is higher than a threshold (we use 0.7 in the filter), then, we consider that this point is a candidate to initiate or update the right or left barrier. Otherwise, we discard it as it is not clearly on any side of the road.

1) Barrier initialisation: We recall from (14) that the barrier state is the lateral offset from the middle of our lane to the barrier and that the barrier is assumed to be parallel to the road. The posterior PDF of the barrier state is approximated as Gaussian with a certain mean and variance. If a barrier has not been initiated (existence probability equal to zero), we check the following conditions to initiate it.

1) We require at least four measurements on a given side of the road. This is necessary to meet S2.
2) The variance of the component n_{k+1}^i of a given measurement must be lower than a threshold (we use $0.3^2\,\mathrm{m}^2$). This is necessary as we need to be sure about the distance at which the measurements are located to meet S1. If a measurement is not accurately located in the road coordinate system, it is better not to consider it for the initialisation.
3) We remove the isolated measurements, i.e., those which do not have another measurement at a distance lower than Δ. We do this to meet S2.
4) We apply an outlier detection step to select the valid measurements for initialisation. That is, all the valid measurements must be located within a distance around the average distance to the road of all the valid measurements. To do this, we take all the measurements and calculate the mean distance to the road (using the mean of n_{k+1}^i). Then, we calculate the square error between the mean distance $\bar{d}$ and the distances for every measurement. If the maximum of these errors is higher than a threshold (we use $0.3^2\,\mathrm{m}^2$), we remove the corresponding measurement. This procedure is repeated until the maximum is not removed. This step is necessary to ensure that the valid measurements meet S1.
5) Finally, if there are at least four valid measurements after Step 4, we initiate the barrier. The barrier is initiated with mean $\bar{d}$ and variance $\sigma_{ini}^2 = 0.1^2\,\mathrm{m}^2$.

2) Update: Among all the stationary object measurements, we check if there are outliers, i.e., measurements that do not come from one of the barriers. To this end, we use a two-step procedure. First, we use gating [27] with measurement equation (13). That is, if the Mahalanobis distance between the measurement and the predicted measurement using the variance of the predicted measurement, which are calculated using the UT and (13), is higher than a threshold (we use 3), then, the measurement is an outlier. Second, among the measurements that have passed the gating test, we discard those which appear isolated to take S2 into account. More specifically, if there is a measurement whose distance to its closest measurement is higher than Δ, it is discarded. The remaining measurements are the valid barrier measurements.

Finally, we build the measurement equation stacking measurement equation (13) for each valid barrier measurement. The update is then carried out using a UKF.

3) Barrier removal: There are two reasons why we delete a barrier, i.e., we set its existence probability to zero. The first reason and most usual one is that the barrier does not exist in the current stretch of road ahead of the host vehicle. What we do in this case is to delete the barrier if the barrier has not been updated for a certain time. In our implementation, we use 0.5 s, which corresponds to 20 stationary measurement updates, see Section II-B2.

Another motive to delete a barrier is that there is a possibility that the filter might not be tracking the barrier but another extended object approximately parallel to the road. For example, there are cases where a barrier has been initiated because some stationary object measurements meet S1 and S2 but these measurements do not correspond to a barrier but something similar, e.g., a fence outside the road. A way to detect that the filter is not working properly is to monitor the stationary objects on the left and right hand side of the road, i.e., the objects between the middle of our lane and the left barrier and the right barrier. We track the longitude on the road and normal distance to the road of these objects. If the number of stationary objects that have appeared in the current stretch of road at roughly the same normal distance is higher than a threshold (we use 3), it is an indication that the true barrier is not what the barrier filter is indicating and we should delete the right or left barrier depending on the side of the road this has happened. At the next time step, the filter is expected to initiate the barrier properly.

E. Filter initialisation

The filter is initialised when we get a lane marking measurement with left and right lane markings for the first time. Then, we use a Gaussian prior with mean $\bar{\mathbf{r}}_{0|0} = \mathbf{0}$ and covariance matrix $\mathbf{\Sigma}_{0|0}$ and perform the update as explained in Section V-B2.

We also want to mention that in practice we should monitor the performance of the filter to ensure that it does not diverge. This can be done following the ideas in [28, Chap. 7]. Finally, a summary of the filter steps is given in Table III.

VI. Experimental results

The objective of this section is to assess the performance of our filter at estimating the road ahead at different distances from the host vehicle in real driving conditions. That is, we want to know the estimation error of the road position at different distances ahead of our car: 20 m, 40 m, 60 m... up to 200 m. We also want to show how the error is affected by using only the lane marking measurements, or considering stationary and moving objects as well. It will be demonstrated that lane marking measurements provide accurate estimates for close range. However, for long range and especially for windy roads, lane marking measurements do not provide accurate estimates

Table III: Filter steps when we receive a measurement

- If the filter has been initialised
  - Prediction step, see Section V-A
  - Update step
    * For lane marking measurements, see Section V-B
      · Lane change detection
      · Update with the lane markings
    * For moving object measurements, see Section V-C
      · Outlier detection step
      · Update with valid measurements
    * For stationary object measurements, see Section V-D
      · If a barrier has not been initialised: Barrier initialisation step
      · If a barrier has been initialised: Outlier detection step, update with valid measurements and barrier removal step
- Else
  - If we get a lane marking measurement, initialise the filter, see Section V-E.

and performance can be greatly improved by accounting for stationary and moving object measurements.

The main problem to calculate the road position error is that the true road is not available, i.e., the coordinates of the middle of our lane for the whole test drive. This is solved by the ground truth analysis explained in Section VI-A.

A. Ground truth analysis

In this section, we explain the ground truth analysis (GTA) tool we use. The GTA tool estimates the road the host vehicle has driven on. Once the test drive has finished, per each time step, we can calculate the error between our estimate of the road ahead of the vehicle and the quite accurate estimate of the GTA, which is regarded as the true road [28].

For a test drive, the GTA tool estimates the trajectory of the host vehicle pose (host path) based on the internal sensor measurements in the global coordinate system. This estimation technique is known as dead-reckoning. Then, it estimates the road position offset y_k^1 at distance 0 at time k based on the lane marking measurements, see Section IV-B1. This estimate of the road position is given in the local coordinate system at time k. Using the host path estimate, the road position can be estimated for the local coordinate system of any time step[3].

The main drawback of this technique is that the error of the dead-reckoning estimate of the host path increases unboundedly with time. As a result, the estimated host path of the whole test drive is not accurate and neither is the road estimate. Nevertheless, the GTA estimate can be regarded as the true road to estimate 200 m of road ahead of the host vehicle under the following assumptions:

- G1: The road position offset estimate y_k^1 based on the lane marking measurements is considered as the true road position in the local coordinate system at time k.
- G2: The estimate of the host vehicle pose using dead-reckoning is accurate if the host vehicle has moved less than 200 m.

Assumption G1 is reasonable as measurement tests indicate that the estimation error in the road position offset y_k^1 is around

[3]Equation (7) indicates how to perform a change of coordinate systems between time steps k and $k+1$.

5 cm, which can be considered negligible for our application. Assumption G2 is met for modern internal sensors as the dead-reckoning error is negligible up to 200 m for normal driving conditions [22].

B. Results

We analyse and compare the performances of four different versions of our filter. The first version only takes into account the lane markings; the second, the lane markings and the barriers; the third, the lane markings and other vehicles and; the fourth, lane markings, barriers and other vehicles altogether. It is clear that if more information from the sensors is used, performance is expected to improve. In Section VI-B1, we analyse one interesting drive test thoroughly while in Section VI-B2 we analyse the averaged results over several drive tests.

It should be noted that the widely used polynomial model, which is based on the approximation of a clothoid, is reasonably accurate up to 60 m for usual road curves [14]. However, it cannot model reasonably sharp curves for distances longer than 60 m or situations in which the road is straight but followed by a sharp bend. In addition, as indicated in [29], it is a poor parameter space due to large sensitivity to coefficient errors and, consequently, it is not adequate for multisensor fusion. As the aim of our paper is to estimate the road ahead of the host vehicle up to 200 m using a multisensor fusion algorithm, the polynomial model is not suitable for our purpose and we have not implemented it.

Nevertheless, we want to recall that neither of the state-of-the-art road geometry algorithms mentioned in the introduction fuse information from the lane markings, barriers and leading vehicles. The way we use the lane markings and leading vehicles is based on the same ideas as the algorithm in [13], i.e., lane markings measurements are given by a third degree polynomial and the heading of the vehicles is roughly parallel to the road. Therefore, the version of our algorithm that does not consider the barriers can be seen as an adaptation of [13] to our fusion system, which works under the assumptions mentioned in Section III-C, and road model based on sampled curvatures. It is demonstrated in the rest of this section that using the barrier information can lower the estimation error remarkably. This certainly underscores the importance of our work.

We use the following filter parameters:

- The model parameters are: $\Delta = 20$ m, $M = 11$.
- The process equation parameters are: $\sigma_c^2 = 4 \cdot 10^{-8} \Delta$ m^{-2}, $\tilde{\mathbf{Q}}^r$ is a diagonal matrix with variance 0.1^2 m^2/s for the offset element, $(0.5\pi/180)^2$ rad^2/s for the heading element and 10^{-5} m^{-2}/s for the curvature elements.
- The measurement equation parameters are: $R^m = (3\pi/180)^2$ rad^2, $\mathbf{R}^s = \text{diag}\left(\left[(1.2)^2, (1.2)^2\right]^T\right)$ (m^2) and $\mathbf{R}^l = \text{diag}\left(\left[\sigma_x^2, \sigma_y^2, \sigma_x^2, 2\sigma_y^2, \sigma_x^2, 4\sigma_y^2, \sigma_x^2, 8\sigma_y^2\right]^T\right)$ where $\sigma_x^2 = 10^{-6}$ m^2, $\sigma_y^2 = 0.05^2$ m^2 and $\text{diag}(\mathbf{a})$ represents a diagonal matrix with diagonal entries given by $\mathbf{a}$.
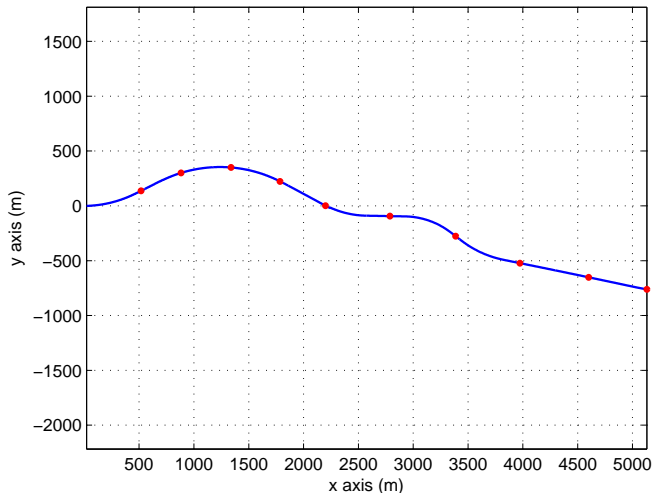
Figure 4: True road in the global coordinate system as estimated by the GTA tool. The red dots denote the position on the road of the host vehicle every 20 seconds.

- Initialisation parameters: We use the prior mean $\bar{\mathbf{r}}_{0|0} = \mathbf{0}$. The covariance matrix $\mathbf{\Sigma}_{0|0}$ is obtained using a variance $0.1^2 \ \mathrm{m}^2$ for the offset element, $(0.5\pi/180)^2 \ \mathrm{rad}^2$ for the angle element and $10^{-8} \ \mathrm{m}^{-2}$ for the first curvature element. The rest of the components of $\mathbf{\Sigma}_{0|0}$, which correspond to curvature elements, are obtained using (4). The offset and angle elements are independent of the curvature elements.

The process equation parameters have been adjusted such that they approximately describe the kind of roads we consider in the drive tests. This can be done with the GTA tool by analysing how the curvature changes with distance. The lane marking measurement covariance matrix has been selected such that it corresponds to the error estimated by the GTA tool in several drive tests. Parameters Δ and M have been selected to cover 200 m with a reasonable computational complexity. The lower Δ is selected, the higher M should be to cover the same distance. This increases the model accuracy and the computational burden. The rest of the parameters have been tuned to obtain a reasonably good performance.

1) Single drive test: The drive test we use to show the performance of our filter is 200 s long. The road is a highway with relatively sharp bends and its GTA estimate is shown in Figure 4.

First, we show the error (Eucledian norm) of the road position at a distance of 200 m along time for the first 100 seconds of test in Figure 5 (a). In this figure, when the error is zero, it means that the host vehicle performed a lane change in the next 200 m and, therefore, there is no ground truth at 200 m for that time instant and the error cannot be calculated. We also show the road curvature and host vehicle speed in Figures 5 (b) and (c). In general, the algorithm that accounts for lane markings, barriers and vehicles has the lowest error and the one which only uses information from the lane markings has the highest error. We can divide the road into two time

intervals according to Figure 4 and 5 (b): Curvy interval from time 0 s to time 80 s and straight interval from time 80 s to 100 s. In the curvy interval, accounting for barriers and vehicles implies a significant improvement in performance with respect to only using the lane markings. This is due to the fact that lane markings can only estimate the road accurately in close range if the road is turning, especially, because they use polynomial approximations and the difficulty of projecting horizontal shapes (lane markings) on a vertical plane (the camera). In the straight interval, there is not a substantial difference among the filters. According to Figure 5 (c), the host vehicle slows down as it enters the first curve and after then, it speeds up to reach roughly 80 km/h.

We analyse more thoroughly the effect of taking into account the barriers and leading vehicles on performance. The output of the filter and the camera view at around time 40 s is shown in Figure 6. The lane marking measurements fail to provide an accurate estimate of the road. Nevertheless, if we use the measurements from the barriers and leading vehicles as explained in this paper, the filter is able to estimate the road accurately in this difficult situation.

Now, we study the performance of the filters averaged over time. To this end, we show the root mean square error of the road position against distance averaged over time for the different information fusion strategies in Figure 7. Lowest error is achieved if we use the lane markings, barriers and leading vehicles. The algorithm that only uses lane markings roughly provides the same results as the others up to distances of 80 m. However, at far distances (from 100 m to 200 m), the improvement of using other types of information is meaningful as the averaged error of our filter is much lower than the error of the filter that only uses the lane markings.

Our implementation, in which the code is not optimised for maximum speed, has been done in Matlab. We used a laptop with an Intel Core i5 processor at 2.67 GHz to obtain the computation times. The average running times of the different parts of the fusion algorithm are broken down as follows: prediction (7 ms), update with lane markings measurements (4 ms), update with stationary object measurements (60 ms), update with moving object measurements (7 ms). It is clearly the update with stationary objects the part of the algorithm with highest computational burden.

Finally, in Table IV, we show the change of the position error for different filter parameters. As can be seen in the first column of Table IV, we increase and decrease 10% the process and measurement noise parameters. This implies a change in the position error, which is indicated by an interval. The error can decrease and this is indicated by a minus sign. The changes in the short range are basically negligible as they are always lower than 2%. The changes in the long range are more important, especially if we change the stationary object measurement covariance matrix (14%). On the whole, the filter behaves quite well when we change its parameters in a reasonable range.
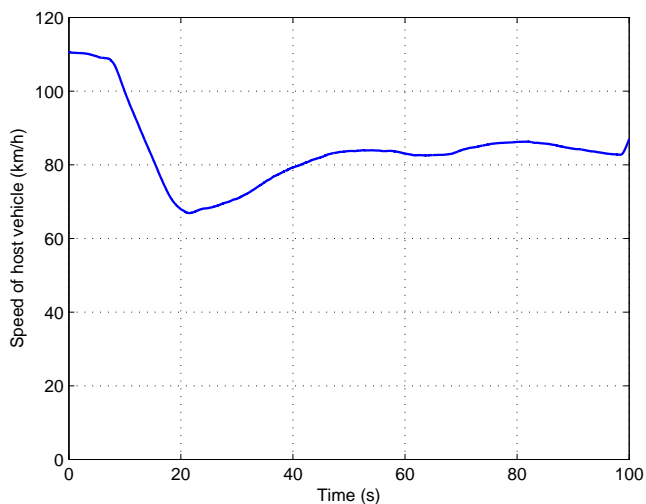
2) Average results on multiple drive tests: In this section, we show the performance of our algorithm averaged over time in different drive tests. The drive tests are sorted into two groups: 1) roads without sharp curves and 2) roads with sharp
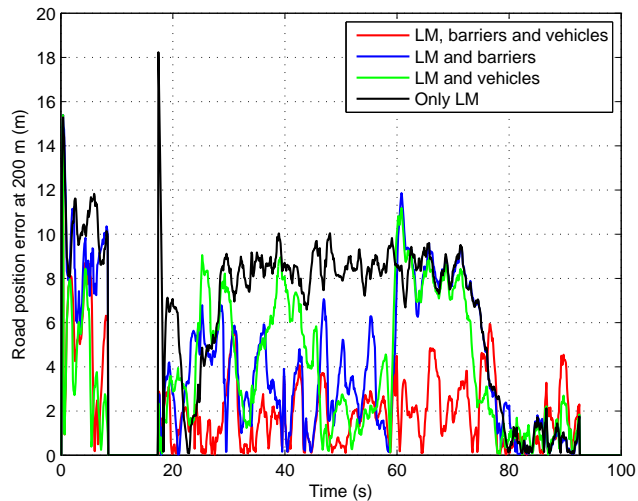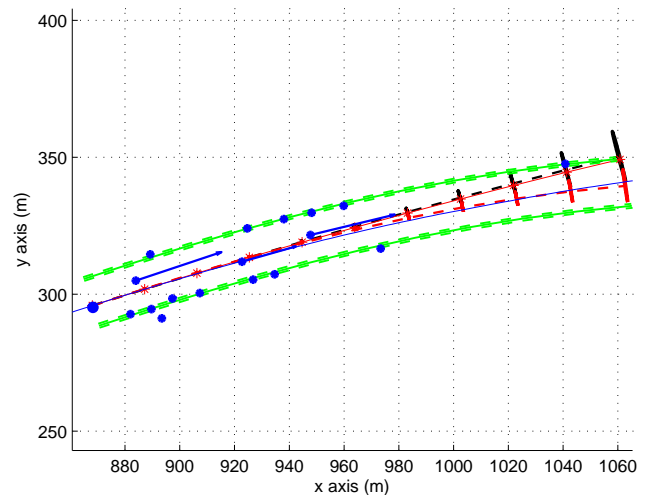
(a)
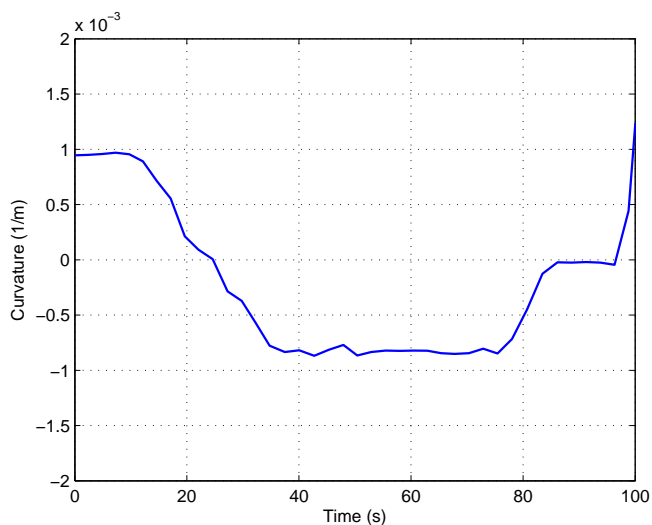
(b)

(c)

Figure 5: (a) Error of the road position at 200 m, (b) curvature of the road and (c) speed against time for the first 100 s of the test drive. At the time instants when the error is zero is means that there is no ground truth available and the error cannot be computed.

(a)

(b)

Figure 6: Example of the filtering process (a) Representation of the filter output at around time 40 s (b) camera view at that time. The blue solid line represents the true road. The big blue dot represents the host vehicle position. The blue dots represent the stationary object measurements. The blue dots with an arrow indicate the leading vehicles positions and headings. The red dashed line represents the estimate of the filter accounting for lane markings, barriers and leading vehicles. The red ellipses, which could be confused by lines, are the $9 - \sigma$ ellipses [30] of the covariance matrix of the road position elements (which are located every $\Delta = 20$ m). The green lines represents the state of the barriers. The red solid line represents the lane marking measurement (average of left and right polynomial) taken at that time. The black dashed line and black ellipses represent the estimate of the filter only accounting for lane markings. Accounting for barriers and leading vehicles allows us to estimate the road accurately if the host vehicle is entering a sharp bend.

Table IV: Change in percentage of the position error averaged over time for different filter parameters

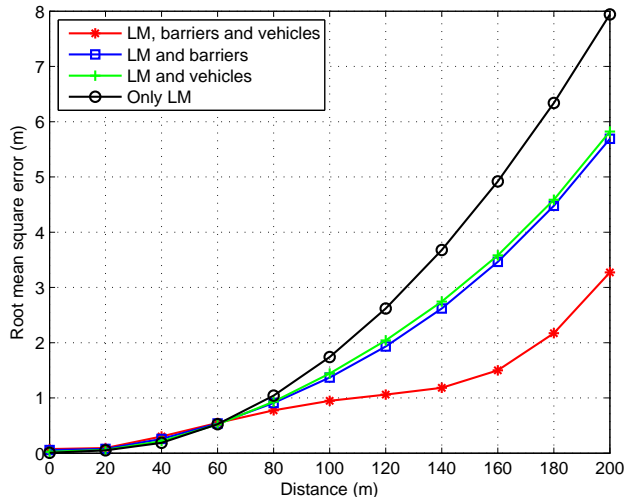|	Short range (0-80 m)	Long range (100-200 m)
$(1 \pm 0.1)\,\mathbf{R}^l$	$(-0.1, 0.4)$	$(-2.2, 6.4)$
$(1 \pm 0.1)\,\mathbf{R}^s$	$(0, 2)$	$(0.1, 14)$
$(1 \pm 0.1)\,R^m$	$(-0.1, 0.1)$	$(0.2, 0.3)$
$(1 \pm 0.1)\,\sigma_c^2$	$(0.36, 0.4)$	$(4.3, 6.2)$
$(1 \pm 0.1)\,\tilde{\mathbf{Q}}^r$	$(-0.1, 0.4)$	$(-0.1, 4.8)$

Figure 7: Root mean square error of the road position against the distance averaged over time for the drive test in Figure 4. Accounting for lane markings, barriers and leading vehicles leads to a high improvement in performance at far distances.

Figure 8: Root mean square error of the road position against the distance averaged over time for roads of type 1 (without sharp bends). Taking into account lane markings, barrier and vehicles slightly increases the performance with respect to the lane markings.

curves. In our tests, roads of type 1 have curves with curvatures lower than $9 \cdot 10^{-4}\,\mathrm{m}^{-1}$ while roads of type 2 have curves with curvatures in the range $\left(1.25 \cdot 10^{-3}\,\mathrm{m}^{-1}, 1.5 \cdot 10^{-3}\,\mathrm{m}^{-1}\right)$. As indicated in this paper, our algorithm is expected to significantly improve the estimation of the lane markings especially if the road has sharps curves. Nevertheless, we want to show that performance also increases for roads that do not have these kinds of curves. The results for roads of type 1 are obtained in 6 drive tests in which we have removed parts of the roads where there were exits as our filter does not take them into account. Each of the drive tests lasts around 200 s and the average speed of the host vehicle is 111 km/h. The root mean square error of the road position against distance averaged over time is shown in Figure 8. The more information we take into account, the lower the error is. However, the improvement in performance is minor.

The results for roads of type 2 are obtained in 3 drive tests. Each of them lasts around 200 s and the average speed of the host vehicle is 107 km/h. Even though roads of type 2 have sharper bends, the average speed of the host vehicle is just slightly lower than on roads of type 1. The root mean square error of the road position against distance averaged over time is shown in Figure 9. In this case, taking into account lane markings, barriers and vehicles implies an important decrease in the error at far distances. This shows the benefits of our fusion algorithm for this kind of road.

VII. Conclusions

We have developed a road model that is able to describe any kind of road and is therefore not limited as the polynomial model. Our model also captures the prior knowledge we have about roads in a simple fashion. Based on this model, we have developed a Bayesian filter that is used to fuse the information coming from different objects (lane markings, leading vehicles and barriers) to estimate the road ahead of
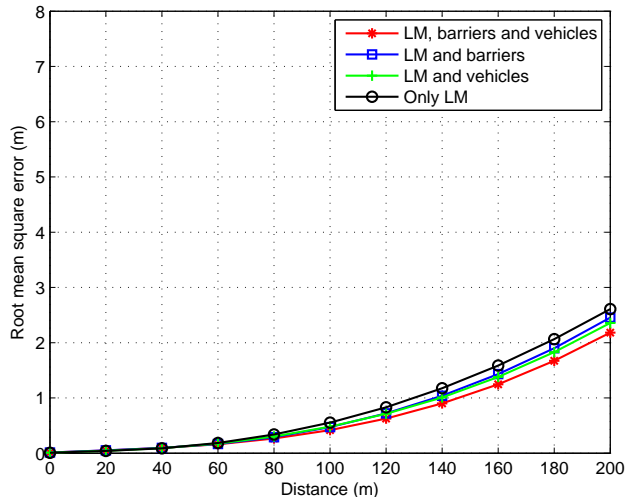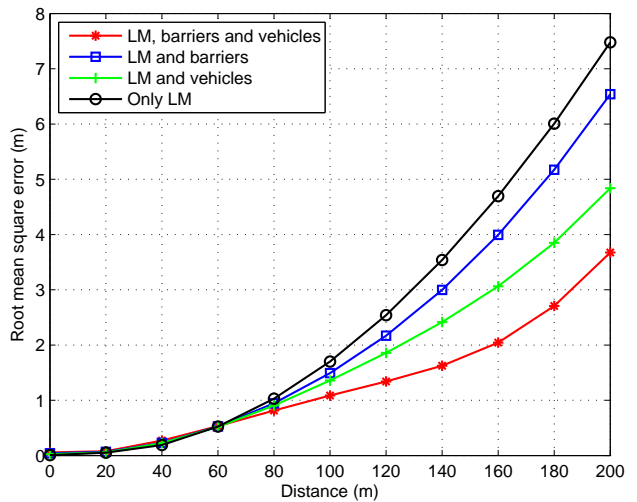

Figure 9: Root mean square error of the road position against the distance averaged over time for roads of type 2 (with sharp bends). Taking into account lane markings, barrier and vehicles considerably increases the performance with respect to the lane markings for far distance.

the vehicle accurately. We are able to estimate the road ahead of the vehicle more precisely than traditional methods based on lane marking estimation using a camera. This is evident at far distance in highways with relatively sharp bends.

The future lines of work are manifold. It is of interest to develop a model that accounts for exits and other kinds of roads. If GPS measurements and maps are available, we would like to make use of this knowledge to improve the estimate of the road.

Another important improvement can be made if our system used the camera to obtain information from the barriers as in

[31]. This would help us detect barriers with higher accuracy. As a result, the overall performance of the filter could be highly improved especially in cases where barrier detection from radar measurements is difficult.

We also aim to develop a filtering algorithm that takes into account that the sensor outputs are already filtered as discussed in Section III-C.

APPENDIX

In this appendix, we provide some notions about curvature and sampled curves. Let us assume a curve is parameterised by $\mathcal{C} := [x(t), y(t)]^T \ t > 0$ where t is the arc length. The signed curvature at a given point is defined as [32]

$$c(t) = \frac{x'(t)y''(t) - y'(t)x''(t)}{\left((x'(t))^2 + (y'(t))^2 \right)^{3/2}} \tag{18}$$

where $x'(t)$ is the derivative of $x(t)$ and $x''(t)$ is the second derivative of $x(t)$.

If we sample this curve using a sampling distance Δ, the sampled curve is $\mathcal{C}_\Delta := [x_i, y_i]^T \ i \in \mathbb{N} \cup \{0\}$ where

$$x_i = x(i\Delta)$$
$$y_i = y(i\Delta)$$

The curvature of the sampled curve can be approximated using finite differences to approximate the derivatives. The approximation of the derivatives are

$$h'_{x,i} = \frac{x_i - x_{i-1}}{\Delta} \tag{19}$$

$$h'_{y,i} = \frac{y_i - y_{i-1}}{\Delta} \tag{20}$$

$$h''_{x,i} = \frac{x_{i-1} - 2x_i + x_{i+1}}{\Delta^2} \tag{21}$$

$$h''_{y,i} = \frac{y_{i-1} - 2y_i + y_{i+1}}{\Delta^2} \tag{22}$$

where $h'_{x,i} \approx x'(i\Delta)$, $h'_{y,i} \approx y'(i\Delta)$, $h''_{x,i} \approx x''(i\Delta)$ and $h''_{y,i} \approx y''(i\Delta)$. Therefore, the sampled curvature is

$$c_i = \frac{h'_{x,i}h''_{y,i} - h'_{y,i}h''_{x,i}}{\left(\left(h'_{x,i}\right)^2 + \left(h'_{y,i}\right)^2 \right)^{3/2}} \tag{23}$$

It should be noted that curve $\mathcal{C}_\Delta$ can also be defined by the first two points $[x_0, y_0]^T \ [x_1, y_1]^T$ and the sampled curvatures $c_i \ i = 1, 2, 3, 4, ...$ at the following points. Given the points $[x_{i-1}, y_{i-1}]^T$, $[x_i, y_i]^T$ and the curvature c_i, $[x_{i+1}, y_{i+1}]^T$ can be calculated using (19)-(23) and the fact that the distance between consecutive points is Δ

$$(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 = \Delta^2 \tag{24}$$

Then

$$[x_{i+1}, y_{i+1}]^T = \mathbf{g}_p \left([x_{i-1}, y_{i-1}]^T, [x_i, y_i]^T, c_i \right) \tag{25}$$

where function $\mathbf{g}_p(\cdot)$ is defined by

$$x_{i+1} = x_i + \frac{-a_i h'_{y,i} \pm \sqrt{a_i^2 \left(h'_{y,i}\right)^2 - b_i \left(a_i^2 - \left(h'_{x,i}\right)^2 \Delta^2\right)}}{b_i} \tag{26}$$

$$y_{i+1} = y_i + \frac{h'_{y,i}(x_{i+1} - x_i) + a_i}{h'_{x,i}} \tag{27}$$
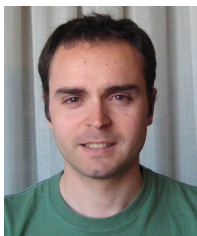
$$a_i = \Delta^2 c_i b_i^{3/2} \tag{28}$$

$$b_i = \left(h'_{x,i}\right)^2 + \left(h'_{y,i}\right)^2 \tag{29}$$

There are two solutions for $[x_{i+1}, y_{i+1}]^T$ using (26)-(27). The right solution is the one that makes the scalar product of $[x_{i+1} - x_i, y_{i+1} - y_i]^T$ and $[x_i - x_{i-1}, y_i - y_{i-1}]^T$ positive.

REFERENCES

[1] A. Amditis, E. Bertolazzi, M. Bimpas, F. Biral, P. Bosetti, M. Da Lio, L. Danielsson, A. Gallione, H. Lind, A. Saroldi, and A. Sjögren, "A holistic approach to the integration of safety applications: The INSAFES subproject within the european framework programme 6 integrating project PReVENT," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 554–566, Sept. 2010.

[2] H. Rohling and E. Lissel, "77 GHz radar sensor for car application," in *Record of the IEEE 1995 International Radar Conference*, May 1995, pp. 373–379.

[3] A. Eidehall, J. Pohl, F. Gustafsson, and J. Ekmark, "Toward autonomous collision avoidance by steering," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 1, pp. 84–94, March 2007.

[4] J. McCall and M. Trivedi, "Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 1, pp. 20–37, March 2006.

[5] R. Chapuis, R. Aufrere, and F. Chausse, "Accurate road following and reconstruction by computer vision," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, no. 4, pp. 261–270, Dec. 2002.

[6] Y. Wang, L. Bai, and M. Fairhurst, "Robust road modeling and tracking using condensation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 4, pp. 570–579, Dec. 2008.

[7] Y. Wang, E. K. Teoh, and D. Shen, "Lane detection and tracking using B-snake," *Image and Vision Computing*, vol. 22, no. 4, pp. 269–280, 2004.

[8] K. Kaliyaperumal, S. Lakshmanan, and K. Kluge, "An algorithm for detecting roads and obstacles in radar images," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 1, pp. 170–182, Jan. 2001.

[9] Y. Yamaguchi, M. Sengoku, and S. Motooka, "Using a van-mounted FM-CW radar to detect corner-reflector road-boundary markers," *IEEE Transactions on Instrumentation and Measurement*, vol. 45, no. 4, pp. 793–799, Aug. 1996.

[10] C. Lundquist, L. Hammarstrand, and F. Gustafsson, "Road intensity based mapping using radar measurements with a probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1397–1408, April 2011.

[11] W. Wijesoma, K. R. S. Kodagoda, and A. Balasuriya, "Road-boundary detection and tracking using ladar sensing," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 456–464, 2004.

[12] K. Peterson, J. Ziglar, and P. Rybski, "Fast feature detection and stochastic parameter estimation of road shape using multiple LIDAR," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 612–619.

[13] C. Lundquist and T. B. Schön, "Joint ego-motion and road geometry estimation," *Information Fusion*, vol. 12, no. 4, pp. 253–263, 2011.

[14] A. Eidehall, J. Pohl, and F. Gustafsson, "Joint road geometry estimation and vehicle tracking," *Control Engineering Practice*, vol. 15, no. 12, pp. 1484–1494, 2007.

[15] B. Ma, S. Lakshmanan, and A. Hero III, "Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 3, pp. 135–147, Sep. 2000.

[16] A. Gern, U. Franke, and P. Levi, "Robust vehicle tracking fusing radar and vision," in *International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2001, pp. 323–328.

[17] J. Sörstedt, L. Svensson, F. Sandblom, and L. Hammarstrand, "A new vehicle motion model for improved predictions and situation assessment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1209–1219, Dec. 2011.

[18] M. Brännström, F. Sandblom, and L. Hammarstrand, "A probabilistic framework for decision-making in collision avoidance systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 637–648, June 2013.

[19] R. Labayrade, D. Aubert, and J.-P. Tarel, "Real time obstacle detection in stereovision on non flat road geometry through "v-disparity" representation," in *IEEE Intelligent Vehicle Symposium*, vol. 2, 2002, pp. 646–651.

[20] F. Bengtsson and L. Danielsson, "A design architecture for sensor data fusion systems with application to automotive safety," in *15th world congress on Intelligent Transport Systems*, 2008.

[21] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[22] J. Borenstein and L. Ojeda, "Heuristic reduction of gyro drift in gyro-based vehicle tracking," in *Proceedings of SPIE Defense, Security + Sensing, Conference 7305: Sensors, and Command, Control, Communications, and Intelligent (C3I) Technologies for Homeland Security and Homeland Defense VIII*, 2009.

[23] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics Automation Magazine*, vol. 13, no. 2, pp. 99–110, June 2006.

[24] S. J. Julier and J. K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *Proceedings of the American Control Conference*, vol. 4, Jun. 1997, pp. 2369–2373.

[25] M. Rogers, *Highway Engineering*. Blackwell Publishing, 2003.

[26] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.

[27] Y. Bar-Shalom and E. Tse, "Tracking in a cluttered environment with probabilistic data association," *Automatica*, vol. 11, no. 5, pp. 451–460, 1975.

[28] F. Gustafsson, *Statistical Sensor Fusion*. Studentlitteratur AB, 2010.

[29] D. Schwartz, "Clothoid road geometry unsuitable for sensor fusion clothoid parameter sloshing," in *Proceedings of IEEE Intelligent Vehicles Symposium*, 2003, pp. 484–488.

[30] Y. Bar-Shalom, T. Kirubarajan, and X. R. Li, *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., 2001.

[31] G. Alessandretti, A. Broggi, and P. Cerri, "Vehicle and guard rail detection using radar and vision data fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 1, pp. 95–105, March 2007.

[32] T. M. Apostol, *Calculus. Volume I.* John Wiley & Sons, 1967.

Lars Hammarstrand was born in Landvetter, Sweden in 1979. He received his M.Sc. and Ph.D. degree in electrical engineering from Chalmers University of Technology, Gothenburg, Sweden, in 2004 and 2010, respectively.

From 2004 to 2011, he was with the Active Safety and Chassis Department at Volvo Car Corporation, Gothenburg, conducting research on tracking and sensor data fusion methods for active safety systems. Currently, Lars is a Postdoctoral Research Fellow at the Signal Processing group at Chalmers University of Technology where his main research interests are in the fields of estimation, sensor fusion and radar sensor modeling, especially with application to active safety systems.

Maryam Fatemi received the B.Sc. degree in electrical engineering from K.N. Toosi University of Technology, Tehran, Iran in 2005, the M.Sc. degree in communication systems engineering form AmirKabir University of Technology, Tehran, Iran in 2008. She is currently working toward her Ph.D. degree at Chalmers University of Technology, Gothenburg, Sweden. Her research interests include statistical signal processing with applications to sensor data fusion and active safety systems.

Ángel F. García-Fernández received the telecommunication engineering degree (with honours) and the Ph.D. degree from Universidad Politécnica de Madrid, Madrid, Spain, in 2007 and 2011, respectively.

He is currently a Research Associate in the Department of Electrical and Computer Engineering at Curtin University, Perth, Australia. His main research activities and interests are in the area of Bayesian nonlinear filtering and radar imaging.

Lennart Svensson was born in Älvängen, Sweden in 1976. He received the M.S. degree in electrical engineering in 1999 and the Ph.D. degree in 2004, both from Chalmers University of Technology, Gothenburg, Sweden.

He is currently Associate Professor at the Signal Processing group, again at Chalmers University of Technology. His research interests include Bayesian inference in general, and nonlinear filtering and tracking in particular.