

Article

Vehicle Detection Based on Probability Hypothesis Density Filter

Feihu Zhang * and Alois Knoll

Robotics and Embedded Systems, Technische Universität München, 80333 München, Germany; knoll@in.tum.de

* Correspondence: feihu.zhang@tum.de; Tel.: +49-89-3603-52239; Fax: +49-89-3603-52250

Academic Editor: Felipe Jimenez

Received: 7 January 2016; Accepted: 31 March 2016; Published: 9 April 2016

Abstract: In the past decade, the developments of vehicle detection have been significantly improved. By utilizing cameras, vehicles can be detected in the Regions of Interest (ROI) in complex environments. However, vision techniques often suffer from false positives and limited field of view. In this paper, a LiDAR based vehicle detection approach is proposed by using the Probability Hypothesis Density (PHD) filter. The proposed approach consists of two phases: the hypothesis generation phase to detect potential objects and the hypothesis verification phase to classify objects. The performance of the proposed approach is evaluated in complex scenarios, compared with the state-of-the-art.

Keywords: LiDAR; vehicle detection

1. Introduction

Traffic accidents are a major cause of death worldwide. A study by the World Health Organization (WHO) reports that an estimated 1.2 million people die in traffic accidents every year, and up to 50 million people are injured [1]. Autonomous driving thus becomes significantly important in order to prevent accidents in traffic scenarios. However, it is still quite challenging for autonomous driving in all scenarios. For automotive manufacturers, the technology behind autonomous driving has been continually refined as a long-term goal, whereas the Advanced Driver Assistance System (ADAS) has been proposed as a short-term development to gradually improve road safety. Numerous ADAS functions have been developed to help drivers avoid accidents, improve driving efficiency, and reduce driver fatigue, in which vehicle detection plays an important role.

Most approaches rely on vision techniques to first detect Regions Of Interest (ROI) and then classify vehicles [2]. Khammari *et al.* use the Adaboost classification to detect vehicles [3]. Miller *et al.* and Paragios *et al.* have also utilized filtering techniques to detect vehicles [4,5]. Meanwhile, vehicle profile symmetry and the corresponding shadows are used in Reference [6]. However, vision techniques suffer from light intensities.

LiDAR is also widely used in vehicle detection. In contrast to vision sensors, LiDAR is robust against light intensities and offers a range of information [7–11]. Teichman *et al.* use the log odds estimators to recognize objects, where the performance is demonstrated in a large scale environment. Dominguez *et al.* demonstrate a data fusion platform for tracking vehicles [12]. Compared with vision sensors, LiDAR measurement often suffers from data association issues.

This paper extends our previous work to detect vehicles by using information from LiDAR, where objects are represented by the position and shape parameters (more details would be explained later). In Reference [13], we used the Difference of Normal (DoN) operator and the Random Hypersurface Model (RHM) to cluster the points cloud data and estimate the shape parameters [14,15]. To avoid the data association issue, the Probability Hypothesis Density (PHD) filter is proposed to detect vehicles based on Random Finite Set statistics (RFSs) [16]. In RFSs,

several approaches are developed to avoid the data association issue, including the PHD filter, the Cardinalized PHD (CPHD) filter [17] and the Bernoulli filter [18]. The PHD filter propagates the probability hypothesis density function over the single target state space, whereas the CPHD filter also propagates the distribution of the target numbers (cardinality). By using the CPHD filter, the system requires more complex implementations and achieves more reliability in cardinality estimation. As the main goal of this paper is to estimate the states in a speed-critical environment, the PHD filter is thus considered. Unlike the PHD or CPHD filter, the multi-Bernoulli filter propagates the posterior target density. Although it has the same complexity as the PHD filter, the performance is better in highly nonlinear environments (it does not require the additional clustering step for state estimation) [18]. As the proposed RHM could be linearly implemented, the PHD filter is considered as the cheapest solution. The estimated states are then classified by the Support Vector Machine (SVM) to eliminate the non-vehicle objects.

The contributions are summarized as: first and foremost, the proposed solution achieves high performance in the presence of unknown data association environments. Furthermore, the shape parameter is first proposed to classify objects.

This paper is structured as follows: Section 2 describes the random hypersurface model, as well as the probability hypothesis density filter for hypothesis generation. Section 3 introduces the support vector machine for hypothesis verification. Section 4 demonstrates the performance of the experiments in urban environments. Finally, the paper is concluded in Section 5.

2. Hypothesis Generation

In the hypothesis generation phase, LiDAR measurements are filtered based on the Random Hypersurface Model (RHM). To further extend RHM in the presence of unknown data association scenarios, the Gaussian Mixture Probability Hypothesis Density (GMPHD) filter is proposed. Notice that objects are tracked and estimated in the 2D Cartesian coordinate system, whereas the depth information is unnecessarily required.

The result of the generation phase is then utilized for the verification phase to eliminate non-vehicles.

2.1. Random Hypersurface Model (RHM)

As illustrated in Figure 1, in a 2D Cartesian coordinate system, a point is considered as a scaled point with the factor within the range $[0, 1]$ drawn on the surface. Thus, the RHM is defined as:

$S(\bar{b}_k)$ denotes the surface which consists of both the shape parameter \bar{b}_k and the center c_k . Each point that lies on the surface is represented as a scaled boundary when s is drawn from $[0, 1]$:

$$c_k + s \cdot (S(\bar{b}_k) - c_k) \quad (1)$$

In Figure 2, $r(\phi)$ denotes the distance function calculated from the center to boundary on angle ϕ in the polar coordinate system.

Assuming $r(\phi)$ consists of the shape parameter \bar{b}_k and the center c_k , the surface is represented as

$$S(\bar{b}_k) = \{s \cdot r(\bar{b}_k, \phi) \cdot e(\phi) + c_k | \phi \in [0, 2\pi], s \in [0, 1]\} \quad (2)$$

where $e(\phi) = \begin{bmatrix} \cos \phi \\ \sin \phi \end{bmatrix}$ and $r(\bar{b}_k, \phi)$ denote the unit vector and the radial function in the form of the Fourier series, respectively. Due to its periodic properties, the Fourier series expansion of degree N_F becomes

$$r(\bar{b}_k, \phi) = a_k^0 + \sum_{j=1}^{N_F} \{a_k^j \cos(j\phi) + b_k^j \sin(j\phi)\} \quad (3)$$

where \bar{b}_k is given by

$$\bar{b}_k = [a_k^0, a_k^1, b_k^1, \dots, a_k^{N_F}, b_k^{N_F}]^T \quad (4)$$

If ϕ is fixed, Equation (3) is represented as

$$r(\bar{b}_k, \phi) = R(\phi) \cdot \bar{b}_k \tag{5}$$

where

$$R(\phi) = [1, \cos(\phi), \sin(\phi), \dots, \cos(N_F\phi), \sin(N_F\phi)] \tag{6}$$

Notice that a low number of Fourier coefficients encode rough information of the surface, whereas a larger number of coefficients give more details.

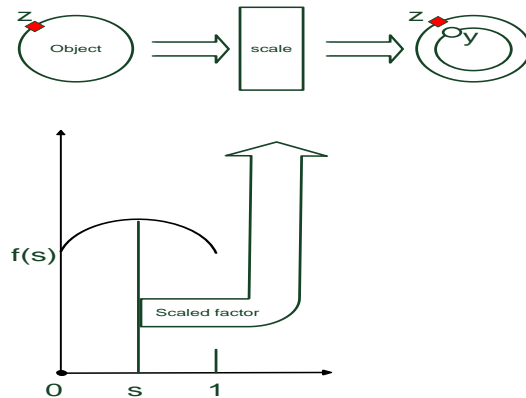


Figure 1. Random hypersurface model.

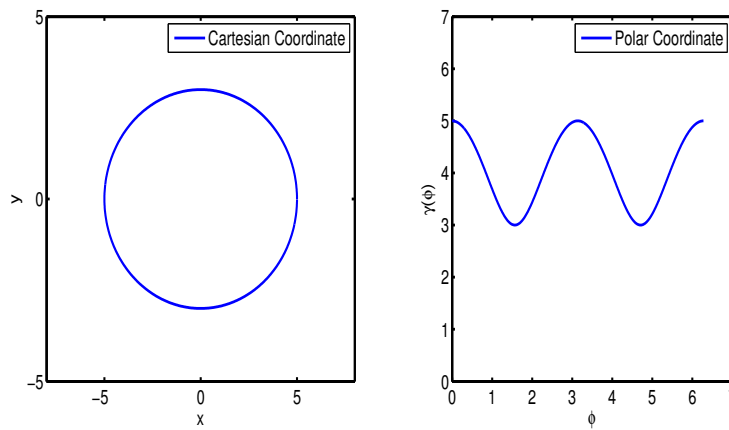


Figure 2. Example of star-convex object.

Bayes Filter

The RHM represents the shape information by using the Fourier coefficients, and the Bayes filter is utilized to calculate the corresponding parameters.

- Process model

Assuming the state \mathbf{x}_k denotes the Fourier descriptors \bar{b}_k and does not drift against time, the process model is described as

$$\mathbf{x}_k = A_k \mathbf{x}_{k-1} + w_k \tag{7}$$

where A_k and w_k denote the identity matrix and the process noise, respectively.

- Measurement model

As illustrated in Figure 1, a single measurement y_k from LiDAR is originated from a surface boundary point z_k with scaled factor s ,

$$y_k = f(z_k, s) + v_k, f(z_k, s) \in S(\bar{b}_k) \quad (8)$$

where v_k denotes the measurement noise.

Using Equation (2), Equation (8) becomes

$$\begin{aligned} y_k &= f(z_k, s) + v_k \\ &= s \cdot r(\bar{b}_k, \phi) \cdot e(\phi) + c_k + v_k \\ &:= h(\mathbf{x}_k, v_k) \end{aligned} \quad (9)$$

which maps the relationship between the state \bar{b}_k and the measurement y_k .

Based on Equation (5), the measurement model is represented as

$$y_k = s \cdot R(\phi) \cdot \bar{b}_k \cdot e(\phi) + c_k + v_k \quad (10)$$

with algebraic manipulations on Equation (10), we get

$$\|y_k - c_k\|^2 = s^2 \cdot \|R(\phi) \cdot \bar{b}_k\|^2 + 2sR(\phi)\bar{b}_k e(\phi)^T v_k + \|v_k\|^2 \quad (11)$$

The measurement model is thus acquired as:

$$\mathbf{0} = s^2 \cdot \|R(\phi) \cdot \bar{b}_k\|^2 + 2sR(\phi)\bar{b}_k e(\phi)^T v_k + \|v_k\|^2 - \|y_k - c_k\|^2 \quad (12)$$

where a pseudo measurement $\mathbf{0}$ is used to model the relationship between the state, the scaled factor, the measurement and its noise. The state \mathbf{x}_k converges to the true Fourier descriptor \bar{b}_k by updating with a large number of measurements.

Notice that the proposed measurement model is implemented in the 2D Cartesian coordinate system and is only effective on the backside of the target. In addition, the depth information from LiDAR measurement is unrequested.

Figure 3 exhibits the estimated result with a cross target, in which the state \mathbf{x} and measurement y are represented by 13 Fourier coefficients and 2D Cartesian coordinates, respectively.

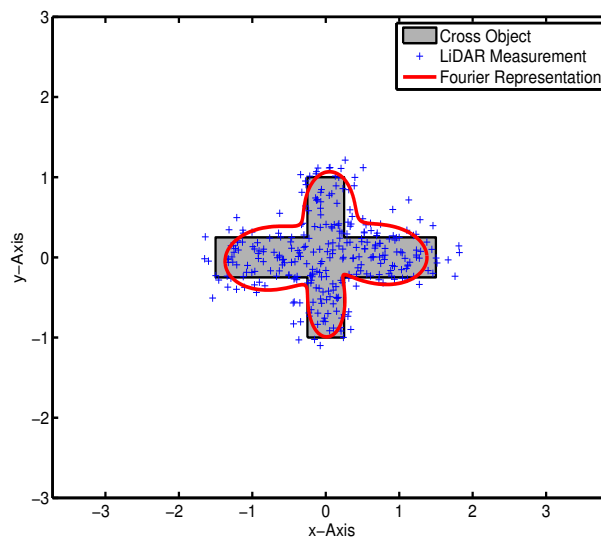


Figure 3. Bayesian estimation in a data association known environment.

2.2. Probability Hypothesis Density (PHD) Filter

As illustrated in Figure 3, the RHM estimates the shape information by using the Bayes filter. The key challenge for practice implementation is data association. A traditional estimator operates in measurement-to-target known scenarios, where all assignments are confirmed. However, LiDAR provides a large number of measurements without any association information. In previous work, the Difference of Normal (DoN) operator is utilized as a preprocessing procedure to cluster measurements. The miss detection is quite high since the clustering process may also eliminate objects. Hence, the PHD filter is proposed to track objects in presence of unknown data association scenarios.

2.2.1. Overview

The PHD filter is represented with the set-valued state and observation for multiple-object Bayesian filtering. All targets and observations are collected and represented in the set space, whereas the data association problem in traditional filtering domain is avoided. Figure 4 is a basic introduction of the RFS statistic. Compared to the single target filtering, the PHD filter relies on the random finite set statistics to process data in the set space level.

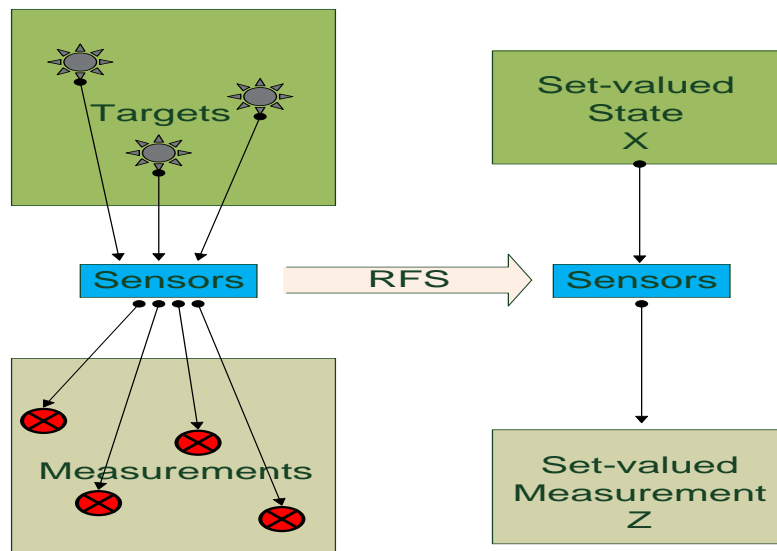


Figure 4. Set-valued states and set-valued observations.

2.2.2. Mathematic Background

Considering the survived targets $S_{k|k-1}$, the spontaneous targets σ_k and the spawned targets $B_{k|k-1}$, the set-valued state is described as:

$$\mathbf{X}_k = [\bigcup_{\zeta \in \mathbf{X}_{k-1}} S_{k|k-1}(\zeta)] \cup [\bigcup_{\zeta \in \mathbf{X}_{k-1}} B_{k|k-1}(\zeta)] \cup \sigma_k \tag{13}$$

In addition, the set observation \mathbf{Z}_k consists of the reflections from both the targets $\theta_k(\mathbf{x})$ and the clutters κ_k .

$$\mathbf{Z}_k = [\bigcup_{\mathbf{x} \in \mathbf{X}_k} \theta_k(\mathbf{x})] \cup \kappa_k \tag{14}$$

Similar to Bayesian estimator, the PHD filter is also divided into prediction and update processes. Notice that D and $f_{k|k-1}(\mathbf{x}|\zeta)$ denote the posterior density and the transition function, respectively. ζ is the previous state.

Thus, the prediction is represented as:

$$D_{k|k-1}(\mathbf{x}) = \left[\int [P_S(\zeta) f_{k|k-1}(\mathbf{x}|\zeta) + \beta(\mathbf{x}|\zeta)] D_{k-1}(\zeta) d\zeta \right] + \gamma_k \quad (15)$$

The intensity function is updated based on the measurement set \mathbf{Z}_k :

$$D_k(\mathbf{x}) = (1 - P_D) D_{k|k-1}(\mathbf{x}) + \sum_{z \in \mathbf{Z}_k} \frac{P_D g_k(\mathbf{z}_i|\mathbf{x}) D_{k|k-1}(\mathbf{x})}{\kappa(\mathbf{z}_i) + \int P_D g_k(\mathbf{z}_i|\zeta) D_{k|k-1}(\zeta) d\zeta} \quad (16)$$

where $g_k(\mathbf{z}_i|\mathbf{x})$ and P_D denote the likelihood function and the detection probability, respectively.

Notice that the predict function in Equation (15) are affected by targets, which enter the scene (γ_k), and survive from the previous time step P_S and the spawn targets ($\beta(\mathbf{x}|\zeta)$).

The update function in Equation (16) corrects the prediction by using innovations from the observations. Notice that the clutter rate is also considered in the update function.

$$N(k) = \int_{\Psi} D_{k|k}(\mathbf{x}) d\mathbf{x} \quad (17)$$

Equation (17) exhibits that the integration of the intensity function represents the number of targets. Meanwhile, the intensity is not a probability density and thus unnecessarily sums up to 1 [19].

The PHD recursions have multiple integrals with no closed form representation. Thus, the most common approach is to use Gaussian Mixture (GM)-PHD approximations [20]:

$$f_{k|k-1}(\mathbf{x}|\zeta) = \mathcal{N}(\mathbf{x}; F_{k-1}\zeta, Q_{k-1}) \quad (18)$$

$$g_k(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; H_k\mathbf{x}, R_k) \quad (19)$$

where \mathbf{z} and \mathbf{x} denote the current measurement and state, whereas ζ denotes the previous state. A Gaussian distribution is represented as $\mathcal{N}(\cdot; \mathbf{m}, P)$ with mean \mathbf{m} and covariance P . F_{k-1} and Q_{k-1} denote the transition matrix and process covariance, respectively. H_k and R_k denote the observation matrix and observation noise covariance. Notice that the detection and survival probabilities are constant values:

$$P_{S,k}(\mathbf{x}) = P_S, P_{D,k}(\mathbf{x}) = P_D \quad (20)$$

Birth targets γ_k are modeled as:

$$\gamma_k(\mathbf{x}) = \sum_{i=1}^{J_{\gamma,k}} \omega_{\gamma,k}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)}) \quad (21)$$

where $\omega_{\gamma,k}^{(i)}$, $P_{\gamma,k}^{(i)}$, $\mathbf{m}_{\gamma,k}^{(i)}$ and $J_{\gamma,k}$ denote the weight, covariance, mean and amount of the Gaussians.

Assuming the posterior intensity at time $k-1$ is a Gaussian mixture:

$$D_{k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k-1}} \omega_{k-1}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{k-1}^{(i)}, P_{k-1}^{(i)}) \quad (22)$$

Equation (15) to time k is also a Gaussian mixture

$$D_{k|k-1}(\mathbf{x}) = P_S \sum_{i=1}^{J_{k-1}} \omega_{k-1}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{S,k|k-1}^{(i)}, P_{S,k|k-1}^{(i)}) + \gamma_k(\mathbf{x})$$

$$\mathbf{m}_{S,k|k-1}^{(i)} = F_{k-1} \mathbf{m}_{k-1}^{(i)}, P_{S,k|k-1}^{(i)} = Q_{k-1} + F_{k-1} P_{k-1}^{(i)} F_{k-1}^T$$

and the Equation (16) at time k is calculated as

$$D_k(\mathbf{x}) = (1 - P_D)D_{k|k-1}(\mathbf{x}) + \sum_{z \in Z_k} D_{D,k}(\mathbf{x}; \mathbf{z}) \quad (23)$$

where

$$D_{D,k}(\mathbf{x}; \mathbf{z}) = \sum_{j=1}^{J_{k|k-1}} \omega_k^{(j)}(\mathbf{z}) \mathcal{N}(\mathbf{x}; \mathbf{m}_{k|k}^{(j)}(\mathbf{z}), P_{k|k}^{(j)})$$

$$\omega_k^{(j)}(\mathbf{z}) = \frac{P_D w_{k|k-1}^{(j)} q_k^{(j)}(\mathbf{z})}{\kappa_k(\mathbf{z}) + P_D \sum_{l=1}^{J_{k|k-1}} w_{k|k-1}^{(l)} q_k^{(l)}(\mathbf{z})}$$

$$q_k^{(j)}(\mathbf{z}) = \mathcal{N}(\mathbf{z}; H_k \mathbf{m}_{k|k-1}^{(j)}, H_k P_{k|k-1}^{(j)} H_k^T + R_k)$$

$$\mathbf{m}_{k|k}^{(j)}(\mathbf{z}) = \mathbf{m}_{k|k-1}^{(j)} + K_k^{(j)}(\mathbf{z} - H_k \mathbf{m}_{k|k-1}^{(j)})$$

$$P_{k|k}^{(j)} = [I - K_k^{(j)} H_k] P_{k|k-1}^{(j)}$$

$$K_k^{(j)} = P_{k|k-1}^{(j)} H_k^T [H_k P_{k|k-1}^{(j)} H_k^T + R_k]^{-1}$$

The GMPHD filter addresses the data association challenge in contrast to the standard Bayes filter. The process model Equation (18) and measurement model Equation (19) are equal to the RHM Bayes filter in Equations (7) and (8), whereas the implementation process is different.

2.3. RHM–GM–PHD Filter

In Section 2.2, the GMPHD filter is introduced for dealing with unknown data association issues. Notice that the standard GMPHD filter operates in condition that one reflection is received for each target per frame, called “Point Target (PT) tracking”. In the vehicle detection scenario, a large number of measurements would be collected from the surface of a single object, called “Extended Target (ET) tracking”. Therefore, the GMPHD filter should be redesigned for dealing with extended targets solely relying on LiDAR measurement.

The process model and measurement model are similar to the RHM Bayes filter in Equations (7) and (8). The prediction equation of the ET–GM–PHD filter are also the same as the standard GMPHD filter. The measurement update formulas for the ET–GM–PHD filter is introduced as:

$$D_{k|k}(\mathbf{x}) = L_{Z_k}(\mathbf{x}) D_{k|k-1}(\mathbf{x} | Z) \quad (24)$$

and the pseudo-likelihood function $L_{Z_k}(\mathbf{x})$ is defined as

$$L_{Z_k}(\mathbf{x}) = 1 - (1 - e^{-\gamma(\mathbf{x})}) P_D(\mathbf{x}) + e^{-\gamma(\mathbf{x})} P_D(\mathbf{x}) \sum_{p \in Z_k} w_p \sum_{W \in p} \frac{\gamma(\mathbf{x})^{|W|}}{d_W} \prod_{z \in W} \frac{\phi_z(\mathbf{x})}{\lambda_k c_k(\mathbf{z})} \quad (25)$$

where $\lambda_k c_k(\mathbf{z})$ is the mean number of clutter measurements, $c_k(\mathbf{z})$ is the spatial distribution of the clutter, notation $p \in Z_k$ denotes that p partitions the measurement set Z_k into non-empty cells W , notation $W \in p$ denotes that W is a cell in the partition p , w_p and d_W denote the non-negative coefficients for each partition and cell, and $\phi(\mathbf{x})$ denotes the same likelihood function for a single measurement in Equation (12).

Here,

$$w_p = \frac{\prod_{W \in p} d_W}{\sum_{p' \in Z_k} \prod_{W' \in p'} d_{W'}} \quad (26)$$

and

$$d_W = \delta_{|W|,1} + D_{k|k-1}[e^{-\gamma} \gamma^{|W|} P_D \prod_{z \in W} \frac{\phi_z}{\lambda_k c_k(z)}] \tag{27}$$

where $\delta_{i,j}$ is the Kronecker delta function and $|W|$ is the number of measurements in cell W . More details of the implementation process could be found in [21,22].

Hence, the ET-GM-PHD filter tracks the potential objects by solely relying on LiDAR without any association or cluster process. The estimated states represent the potential objects and would be filtered again by the support vector machine to eliminate non-vehicle objects.

3. Hypothesis Verification

To eliminate the outliers, the support vector machine is utilized to classify the vehicle and non-vehicle Fourier coefficients.

3.1. Support Vector Machine (SVM)

As exhibited in Figure 5, the SVM is proposed to obtain classifiers with good generalization [23]. The mathematical background is introduced as follows:

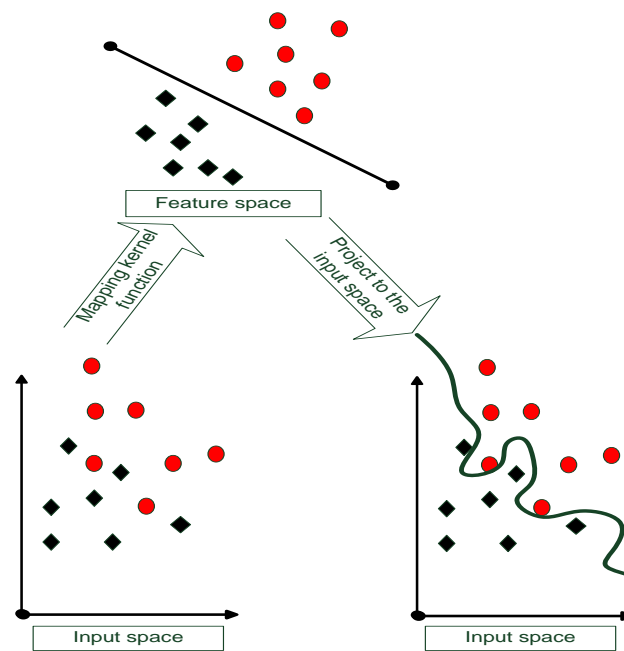


Figure 5. The concept of support vector machine (SVM)

For $\mathbf{x}_i \in \mathbb{R}^n$ with respect to the classification $y_i \in \{-1,+1\}, i = 1, \dots, k$, the hyperplane is defined as:

$$f(\mathbf{x}) = w^T \mathbf{x} + b \tag{28}$$

to linearly separate each data. Notice that \mathbf{x}, w and b denote the input vector, weight vector and the bias, respectively. Hence a maximum margin is found to separate positive class from negative class based on

$$f(\mathbf{x}) = \text{sgn}(w^T \mathbf{x} + b) \tag{29}$$

The calculation of the hyperplane is subjected to the following constraints:

$$\text{Minimize} \quad \frac{1}{2} \|w\|^2$$

$$\text{Subject to} \quad y_i(w_i x_i + b) \geq 0$$

and the classification performance relies on the optimization as:

$$\text{Maximize} \quad \sum_{i=1}^k \alpha_i - \frac{1}{2} \sum_{i,j=0}^k \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

$$\text{Subject to} \quad \sum_{i=1}^k y_i \alpha_i = 0, \quad \alpha_i \geq 0 \quad \text{for } i = 1, \dots, k$$

where $K(x_i, x_j)$ and α_i denote the Kernel function and Lagrange multiplier, respectively. Notice that if the data can not be separated linearly, the kernel function changes according to

$$K(x_i, x_j) = K(x_i, x_j) + \frac{1}{C} \delta_{ij} \quad (30)$$

where C denotes the penalize parameter.

3.2. Implementation Detail

During the implementation, there are still issues in both the generation and verification phases.

- ET-GM-PHD Implementation

To track objects in multiple frames, the alignment issue should also be considered. Thus the state to be tracked in the ET-GM-PHD filter is $\mathbf{x}_k = [v_x, v_y, x, y, (\bar{b}_k)^T]^T$, where $[v_x, v_y, x, y]^T$ is the normal state of the object centroid in the 2D Cartesian coordinate system and (\bar{b}_k) is the shape parameter by using 13 Fourier descriptors. Although a higher number of Fourier descriptors returns more details of the surface, the computation performance and robustness is unsatisfied. In this paper, 13 Fourier descriptors are selected based on the experience during the experiment.

Each object follows the liner Gaussian dynamic Equation (18) with the following configurations:

$$F_k = \text{diag}(A_k, \mathbf{I}_{13}), \quad A_k = \begin{bmatrix} \mathbf{I}_2 & \mathbf{0}_2 \\ \mathbf{I}_2 & \mathbf{I}_2 \end{bmatrix} \quad (31)$$

$$Q_k = \text{diag}(C_k, 0.03\mathbf{I}_{13}), \quad C_k = \begin{bmatrix} \mathbf{I}_2 & \frac{0.2}{2} \\ \frac{0.2}{2} & \frac{1}{4} \end{bmatrix} \quad (32)$$

where \mathbf{I}_n and $\mathbf{0}_n$ denote the $n \times n$ identity and zero matrix, respectively. The measurement covariance is given using parameters $\text{diag}[0.5, \dots, 0.5]$.

Regarding the measurement model, it follows the non-linear Gaussian dynamic Equation (19) and is described as:

$$\mathbf{0} = s^2 \cdot \|R(\phi) \cdot \bar{b}_k\|^2 + 2sR(\phi)\bar{b}_k e(\phi)^T v_k + \|v_k\|^2 - \|y_k - m_k\|^2$$

where s is a Gaussian function with mean 0.5 and variance 0.03. The birth intensity of the PHD filter is given according to the geometry information. To reduce the computation complexity, measurements which fall into the road regions are utilized to initialize the vehicles. The birth intensity is thus given by

$$\gamma_k(\mathbf{x}) = 0.1\mathcal{N}(\mathbf{x}; \mathbf{m}_{\gamma,k}^1, P_{\gamma,k}^1) + 0.1\mathcal{N}(\mathbf{x}; \mathbf{m}_{\gamma,k}^2, P_{\gamma,k}^2) + 0.1\mathcal{N}(\mathbf{x}; \mathbf{m}_{\gamma,k}^3, P_{\gamma,k}^3) \quad (33)$$

where $\mathbf{m}_{\gamma,k}^1 = [0, 0, 0, 1, 2, 0, \dots, 0]$, $\mathbf{m}_{\gamma,k}^2 = [0, 0, -5, 1, 2, 0, \dots, 0]$ and $\mathbf{m}_{\gamma,k}^3 = [0, 0, 5, 1, 2, 0, \dots, 0]$ represents the vehicles in the middle (0,1), left (-5,1) and right (5,1), respectively. $P_{\gamma,k}^1 = P_{\gamma,k}^2 = P_{\gamma,k}^3 = \text{diag}[2, 2, 2, 2, 0.3, \dots, 0.3]$.

- SVM Implementation

The KITTI dataset is utilized for training the classifier, which provides a set of 5000 training frames with 1893 manually labeled objects (car, van, tram, misc, pedestrian, cyclist, trunk and so on). For each object, the original measurements (all objects are labeled with a 2D box, where measurement-to-target association is confirmed) are projected to the 2D Cartesian coordinate system around its original point. The 13 Fourier coefficients are then calculated by using the standard nonlinear Kalman filter [15]. Instead of multiple categories, objects are only considered as vehicles and non-vehicles (actually it is divided by cars and non-cars). Then, the SVM is trained by collecting Fourier coefficients from all calculated objects. In further evaluation, another 2000 frames of test data is utilized to guarantee the performance both quantitatively and qualitatively. The SVM is implemented in both the training and test phases without cross-validation.

- Key Parameters and Open Issues

During the ET-GM-PHD process, potential objects are collected in the set-valued state. The Fourier coefficients describe the shape information, and the position vector represents the location. The estimated objects are also shown by using bounding boxes, where the mean point uses the position and the width/height are calculated based on the Fourier coefficients (the Fourier coefficients represent the rough shapes in polar coordinate systems, in which the width and height of the boundary box are calculated by setting the ϕ equals to 0 and $\frac{\pi}{2}$). Since the PHD filter addresses the data association issue, the points cloud data is directly utilized to estimate the set-valued states. No further cluster process is required. Meanwhile, for each single measurement, the probabilities of both detection and survival are constant and no more than 1. In addition, the PHD filter may estimate close objects as single objects mainly due to the scale factor s . For the RHM model, measurements are considered as random draws from boundaries with different scaling factors in the range $[0, 1]$. When objects are close to each other, it is quite challenging to distinguish them.

During the SVM process, the calculated coefficients from objects are utilized to eliminate the outliers. Since most cars have a similar width/length rate, the car-labeled objects are treated as vehicles and the rest are non-vehicles. Furthermore, the 13 Fourier coefficients have been found to be quite challenging for linearly separating vehicles and non-vehicles. To better train the SVM, the radial based kernel function is utilized as:

$$K(x_i, x_j) = \exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (34)$$

where σ affects distributing complexity in the feature space.

4. Experiment Evaluation

To evaluate the approach quantitatively and qualitatively, the KIT dataset is utilized and compared with the state-of-the-art [24]. During the experiment, the proposed approach is implemented in Matlab with 2 Cores@3 GHz, and the average time is 5 s per frame.

Figures 6–8 demonstrate the detection performance based on the proposed approach in one scenario. In Figure 6, a bicycle is fully observed in the middle of the road. On the left side, a parking car is observed with partial occlusion. On the right side, both car and pedestrian are fully observed. As illustrated in Figure 7, potential objects are detected based on the RHM-ET-PHD filter. It is observed that the proposed approach extracts the potential objects based on the geometry of the road, where the birth model plays an important role. The extracted objects are drawn by boundary boxes calculated by the set-valued state (the center point is based on the position, and the width/height are

based on the Fourier coefficients). Afterwards, the SVM is utilized to eliminate the non-vehicle objects. Figure 8 shows the results of the verification phase. Due to the occluded issues, it is observed that the left vehicle is also eliminated.

Table 1 demonstrates the overall performance, in contrast to the state-of-the-art, in all scenarios from the KITTI dataset. Although there are also approaches using cameras, the evaluation focuses on the algorithms which only use LiDAR measurements.

As illustrated in Table 1, moderate, easy and hard denote the occlusion level of vehicles, with respect to partly occluded, fully visible and difficult to see. Among the references, the proposed approach achieves a high performance for the easy category and poor performance for both moderate and hard categories. In easy scenarios, all vehicles are fully observed and the corresponding reflections on surfaces are uniformly distributed. Hence, the proposed approach can track and estimate the states successfully, and the final classification process has high performance. In moderate and hard scenarios, the corresponding performance drops significantly in both the generation and verification processes. For PHD filter, although it detects the potential objects, the calculated Fourier coefficients are strongly influenced by the invisible measurements. For SVM classification, the training process mainly relies on the visible measurements to calculate Fourier coefficients.



Figure 6. Original data from image view.

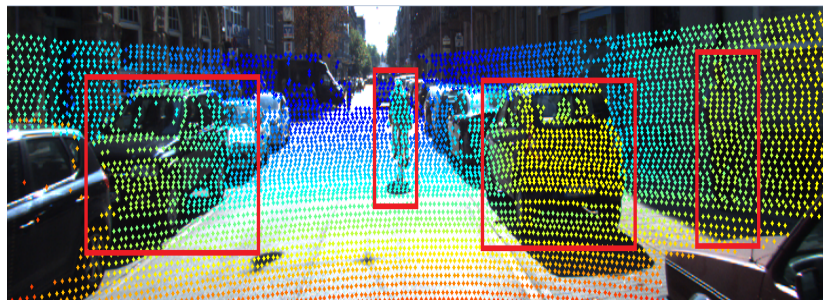


Figure 7. Result on hypothesis generation phase.

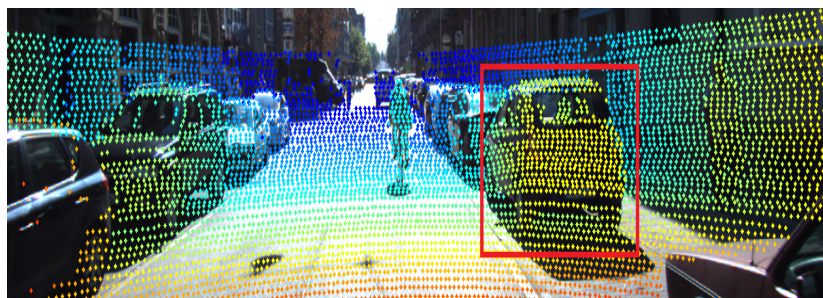


Figure 8. Result on hypothesis verification phase.

Table 1. Performance of the vehicle detection approach compared with the state-of-the-art.

Method	Ours	Vote3D [25]	CSoR [26]	mBoW [27]
Easy	75%	57%	35%	36%
Moderate	15%	48%	26%	24%
Hard	3%	43%	23%	18%
Average time	5 s	0.5 s	3.5 s	10 s

Nevertheless, compared with the overall performance in easy scenarios, the proposed approach still improves almost 20%–40%.

As a summary, the contributions are concluded as follows: first and foremost, the proposed framework solely relies on LiDAR measurements for vehicle detection in the presence of unknown data association environments. Furthermore, the Fourier coefficient is first proposed for object classification and concluded with high performance for fully visible vehicles.

5. Conclusions

Vehicle detection is important for developing driver assistance systems. To address the data association problem that suffers from points cloud data, the Probability Hypothesis Density (PHD) filter is proposed in this paper. The proposed scheme utilizes contour information for classification. The evaluation results illustrate a high performance in contrast to the state-of-the-art techniques.

Future work focuses on the improvement of detecting occluded vehicles.

Acknowledgments: This work was supported by the German Research Foundation (DFG) and the Technische Universität München within the funding program Open Access Publishing.

Author Contributions: All authors have contributed to the manuscript. Feihu Zhang performed the experiments and analyzed the data. Alois Knoll provided helpful feedback on the experiments design.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Global Status Report on Road Safety 2013. Available online: http://www.who.int/violence_injury_prevention/road_safety_status/2013/en/ (accessed on 1 October 2013).
- Leon, L.; Hirata, R. Vehicle detection using mixture of deformable parts models: Static and dynamic camera. In Proceedings of the 25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Ouro Preto, Brazil, 22–25 August 2012; pp. 237–244.
- Khammari, A.; Nashashibi, F.; Abramson, Y.; Laurgeau, C. Vehicle detection combining gradient analysis and AdaBoost classification. In Proceedings of the 2005 IEEE Intelligent Transportation Systems, Vienna, Austria, 13–15 September 2005; pp. 66–71.
- Sun, Z.; Bebis, G.; Miller, R. On-road vehicle detection using evolutionary Gabor filter optimization. *IEEE Trans. Intell. Transp. Sys.* **2005**, *6*, 125–137.
- Paragios, N.; Deriche, R. Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 266–280.
- Sun, Z.; Bebis, G.; Miller, R. Monocular precrash vehicle detection: Features and classifiers. *IEEE Trans. Image Process.* **2006**, *15*, 2019–2034.
- Huang, L.; Barth, M. Tightly-coupled LIDAR and computer vision integration for vehicle detection. In Proceedings of the 2009 IEEE Intelligent Vehicles Symposium, Xi'an, China, 3–5 June 2009; pp. 604–609.
- Behley, J.; Kersting, K.; Schulz, D.; Steinhage, V.; Cremers, A. Learning to hash logistic regression for fast 3D scan point classification. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Taipei, Taiwan, 18–22 October 2010; pp. 5960–5965.

9. Li, Y.; Ruichek, Y.; Cappelle, C. Extrinsic calibration between a stereoscopic system and a LIDAR with sensor noise models. In Proceedings of the 2012 IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Hamburg, Germany, 13–15 September 2012; pp. 484–489.
10. Xiong, X.; Munoz, D.; Bagnell, J.; Hebert, M. 3-D scene analysis via sequenced predictions over points and regions. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 2609–2616.
11. Li, Y.; Ruichek, Y.; Cappelle, C. Optimal Extrinsic Calibration Between a Stereoscopic System and a LIDAR. *IEEE Trans. Instrum. Meas.* **2013**, *62*, 2258–2269.
12. Dominguez, R.; Onieva, E.; Alonso, J.; Villagra, J.; Gonzalez, C. LIDAR based perception solution for autonomous vehicles. In Proceedings of the 11th International Conference on Intelligent Systems Design and Applications (ISDA), Cordoba, Spain, 22–24 November 2011; pp. 790–795.
13. Zhang, F.; Clarke, D.; Knoll, A. LiDAR based vehicle detection in urban environment. In Proceedings of the 2014 International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI), Beijing, China, 28–29 September 2014; pp. 1–5.
14. Ioannou, Y.; Taati, B.; Harrap, R.; Greenspan, M. Difference of normals as a multi-scale operator in unorganized point clouds. In Proceedings of the 2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), Zurich, Switzerland, 13–15 October 2012; pp. 501–508.
15. Baum, M.; Hanebeck, U. Shape tracking of extended objects and group targets with star-convex RHMs. In Proceedings of the 14th International Conference on Information Fusion (FUSION), Chicago, IL, USA, 5–8 July 2011; pp. 1–8.
16. Mahler, R. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1152–1178.
17. Mahler, R. Approximate multisensor CPHD and PHD filters. In Proceedings of the 13th Conference on Information Fusion (FUSION), Edinburgh, UK, 26–29 July 2010; pp. 1–8.
18. Vo, B.T.; Vo, B.N.; Hoseinnezhad, R.; Mahler, R. Robust multi-Bernoulli filtering. *IEEE J. Sel. Top. Signal Process.* **2013**, *7*, 399–409.
19. Mahler, R. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Trans. Aerosp. Electron. Syst.* **2003**, *39*, 1152–1178.
20. Vo, B.N.; Ma, W.K. The Gaussian Mixture Probability Hypothesis Density Filter. *IEEE Trans. Signal Process.* **2006**, *54*, 4091–4104.
21. Mahler, R. PHD filters for nonstandard targets, I: Extended targets. In Proceedings of the 12th International Conference on Information Fusion, FUSION '09, Seattle, WA, USA, 6–9 July 2009; pp. 915–921.
22. Han, Y.; Zhu, H.; Han, C. A Gaussian-mixture PHD filter based on random hypersurface model for multiple extended targets. In Proceedings of the 16th International Conference on Information Fusion (FUSION), Istanbul, Turkey, 9–12 July 2013; pp. 1752–1759.
23. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297.
24. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
25. Wang, D.Z.; Posner, I. Voting for voting in online point cloud object detection. In Proceedings of the Robotics: Science and Systems, Rome, Italy, 13–17 July 2015.
26. Plotkin, L. PyDriver: Entwicklung Eines Frameworks für Räumliche Detektion und Klassifikation von Objekten in Fahrzeugumgebung. Bachelor's Thesis, Karlsruhe Institute of Technology, Karlsruhe, Germany, March 2015.
27. Behley, J.; Steinhage, V.; Cremers, A. Laser-based segment classification using a mixture of bag-of-words. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan, 3–7 November 2013; pp. 4195–4200.

