2012 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 7-12, 2012. Vilamoura, Algarve, Portugal

Single Camera Visual Odometry Based on Random Finite Set Statistics

Feihu Zhang, Hauke Stähle, Andre Gaschler, Christian Buckl, Alois Knoll

Abstract— **This paper presents a novel approach based on Random Finite Set (RFS) Statistics for estimating a vehicle's trajectory in complex urban environments by using a fixed single camera. For this, we extend our earlier works which used Probability Hypothesis Density (PHD) filtering under sensor fusion framework and are among the first to apply this technique to visual odometry in real traffic scenes. We consider features acquired from the camera as a group targets, use the PHD filter to update the overall group state and then estimate the ego-motion vector of the camera. Compared to other approaches, our approach presents a recursive filtering algorithm that provides dynamic estimation of multiple-targets states in the presence of clutter and avoids the association problem. Experimental results show that this method provides good robustness under real traffic scenarios.**

I. Introduction

Using cameras for vehicle navigation is the current trend in the field of intelligent vehicles. Visual odometry is gaining importance for the estimation of the vehicle's trajectory. The main idea is using cameras to find corresponding features and calculate the displacement between them in successive frames. However, challenges still need to be considered in real traffic scenes as discussed in [1]:

- Features that are used to estimate the ego-motion vector may contain some false associated pairs. Robust matching techniques are needed to avoid false matching.
- Unevenly distributed features that are aggregated in a small region may influence the performance of estimation since they are not uniformly distributed throughout the whole space. Effective extracting techniques are needed to overcome this challenge.
- The algorithms for ego-motion are typically based on features of stationary objects. However, typical road scenes may contain a large amount of features stemming from moving objects. All these artifacts reduce the performance of ego-motion estimation. There should be an approach under the Bayesian filtering framework to reduce the influences of features from non-stationary objects.

In this paper, we propose a method for ego-motion computation based on the Probability Hypothesis Density (PHD) filter [2] under Random Finite Set (RFS) statistics. PHD filter works on sets of features, called set-valued states, instead of single features. The observations associated with the features are treated as set-valued observations. Modeling set-valued states and set-valued observations as Random Finite Sets allows to solve the problem of dynamically estimating multiple-targets in the presence of clutter and association uncertainty in a Bayesian filtering framework [3] [4].

In our earlier work [5], we presented a visual odometry (VO) system based on stereo cameras and a gyroscope to provide localization information in urban environments by using a PHD filter. In that case, the stereo cameras provide accurate 3D information of the feature points in the vehicle coordinates and the gyroscope helps the PHD filter to detect the falsely associated features and moving targets. The experiments show that the PHD filter provides good robustness under the sensor fusion framework. This paper enhances our previous work by only using a single camera to calculate the ego-motion vectors. The ego-motion vectors are estimated by using the coordinates of the feature points on the 2D space, which may contain huge inaccurate information since the single camera cannot provide the whole 3D information of the features. In addition, the rotation is estimated based on the non-linear mapping from the feature points' location by using the PHD filter. We extend our previous work to show the feasibility and the reliability of the PHD filter in the VO domain compared with other methods by different kinds of sensors. Combining with the previous work, we came to the conclusion that this method keeps good robustness under real traffic scenarios, either the sensors provide quite accurate 3D information and orientation or rough 2D information.

The general idea of our approach is as follows: The location of features on the image plane (in Sec. III) can be mapped to the ground plane (vehicle coordinates in Sec. III) via its homography. Considering feature points' locations on this ground plane as the targets, a PHD filter is applied to estimate the camera motion (velocity and rotation angle in vehicle coordinates) from the group targets' state set at each frame.

Our method consists of two phases: a preprocessing phase and a tracking phase. The preprocessing phase starts by extracting features using SIFT (Scale Invariant Feature Transform) from consecutive frames and matching them as feature pairs from consecutive frames. Then, the coordinates of the features are transformed to 2D vehicle coordinates. Finally, we record features positions as measurements for the tracking phase.

The tracking phase is performed in spatial dimension. In this phase the algorithm tracks features in vehicle coordinates by using the PHD filter to estimate the ego-motion vector (velocity and rotation angle) at each frame.

We used an off-the-shelf platform (iPhone4) to recorded data under real traffic scenarios to evaluate the approach. The

Feihu Zhang, Hauke Stähle and Alois Knoll are with the Technische Universität München, Garching bei München, Germany, e-mail:feihu.zhang@tum.de, {staehle,knoll}@in.tum.de.

Andre Gaschler and Christian Buckl are with the fortiss GmbH, München, e-mail: {gaschler,buckl}@fortiss.org.

978-1-4673-1735-1/12/S31.00 ©2012 IEEE

platform offers data from GPS, camera and gyroscope sensors. The GPS and gyroscope data is only used to compare the performance of the PHD filter used in this paper (GPS provides the velocity while the gyro provides the rotation angle, using these two information we can calculate the real trajectory on the ground plane). The data from the camera is used to calculate the velocity and the rotation angle of the camera on the ground plane at each frame. Finally a dead-reckoning method is used to calculate the whole trajectory of the vehicle. The experimental results indicate that the proposed method yields precise estimation.

The benefits of our approach are threefold: First, it eliminates false matched features since the PHD filter does not need to focus on the data association problem. Second, with the pruning and merging approach in the PHD filter, the aggregated targets can be treated as a single target. It avoids that a small region contains many features which may influence the estimation's precision. Third, the PHD filter can utilize the dynamic motion model under the Bayesian filtering framework to reduce the influences of non-stationary objects.

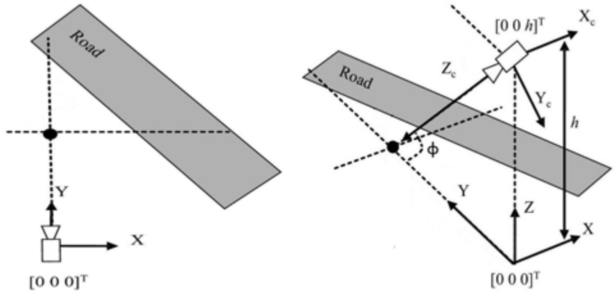
The remainder of this paper is structured as follows: Sec. II describes the related work in visual odometry. Sec. III introduces details about the preprocessing phase. Sec. IV describes the PHD filter and its implementation. Sec. V presents experimental results under real traffic scenes. Finally, the paper is concluded in Sec. VI.

II. Related Work

Much work has been done in visual odometry using e.g. a single camera [6] [7] [8], stereo cameras [9] [10], or an omnidirectional camera [11]. One approach to visual odometry uses the Structure-from-Motion (SfM) technique. The idea is to find good quality features in one frame and the corresponding features in the next frame, estimating displacements from these features and translating them to the motion of the camera [12]. Compared with single cameras, stereo cameras and omnidirectional cameras provide good performance of the 3D construction to the features, which is often used to calculate the motion of the camera in SfM technique. Using RANSAC approach [1] [10] enables the method to overcome a large number of outliers as encountered in real traffic scenes.

Optical flow is a different approach which focuses on the change in the brightness of the image, where this change in brightness results from the apparent motion in the image [13] [14]. This method is much simpler and computationally cheaper than the extraction and tracking of features. However, the precision is not very good. Corke et al.[15] compared these two approaches and got the conclusion that SfM methods allow higher precision at the cost of higher computational needs.

In this paper, we apply the SfM technique to estimate the displacements of the vehicle within the RFS framework. The contribution of our approach is that it avoids the data association issue and the influence of the unevenly distributed features in SfM technique. Although there are other recent visual odometry techniques which do not require data association, do not suffer from uneven distribution and are based alternatively on robust motion estimation [16], PHD filter still plays an important role in VO domain.



(a) Top view of the scene (b) Same scene viewed from the left side

Fig. 1. Coordinate systems

III. Preprocessing Phase

A. Interesting Points Extraction

In most of the previous work on visual odometry using SfM technique, features are used for establishing correspondences between consecutive frames in a video sequence. In this paper, we use the SIFT features to estimate the ego-motion [17].

In our system, the features are extracted from two consecutive frames and matched as feature pairs. We transform these features in 2D vehicle coordinates and use them as measurements for the PHD filter and the compared RANSAC method. Our approach uses the SIFT extraction technique to calculate the position of features on the image plane. It does not use the feature descriptors of SIFT as a matching is not performed. However, the descriptors are used for comparison with the RANSAC approach, which relier on a feature matching method.

B. Transformation from Image Coordinates to Vehicle Coordinates

We determine the mapping between the image coordinates (u,v) of a tracked feature and its corresponding point (x,y) on the ground plane (vehicle coordinates). Fig.1 shows that the camera is placed at height h above the road with down/tilt angle ϕ. X, Y, Z are the axes of the vehicle's coordinate system while X_c, Y_c, Z_c, are the axes of the camera's coordinates system. The camera was calibrated by Zhang's approach [18]. Suppose the camera has zero skew, we transform the feature coordinates from image coordinates to vehicle coordinates as follows [19]:

A point $\mathbf{X} = [x, y, z, 1]^T$ in the vehicle coordinates is related to its image coordinates $\mathbf{x} = [uw, vw, w]^T$,

$$\begin{bmatrix} uw \\ vw \\ w \end{bmatrix} = KR[I_{3\times3}| - T] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{1}$$

$$K = \begin{bmatrix} f & 0 & u_c \\ 0 & f & v_c \\ 0 & 0 & 1 \end{bmatrix}, R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -sin\phi & -cos\phi \\ 0 & cos\phi & -sin\phi \end{bmatrix}$$

where K is the camera calibration matrix, R is the rotation matrix corresponding to a rotation of ϕ around the X-axis, I is the identity matrix and $T = [0, 0, h]^T$ is the translation of the camera from the origin of the vehicle coordinate system. $[I_{3\times3}| - T]$ is the concatenation of I and T. (u_c, v_c) are the principal point coordinates. The actual pixel coordinates (u, v) are defined with respect to the origin in the top left corner of the image plane and w is a scale factor.

Using equation (1) we can express the relationship between the vehicle coordinates (x_k, y_k, z_k) of a point on the road $(z_k = 0)$ to its image coordinates (u_k, v_k) at frame k as follows:

$$u_k = \frac{wu_k}{w} = \frac{fx_k}{y_k cos\phi + hsin\phi} + u_c \tag{2}$$

$$v_k = \frac{wv_k}{w} = \frac{fhcos\phi - fy_k sin\phi}{y_k cos\phi + hsin\phi} + v_c \tag{3}$$

Rearranging above equations, we get

$$x_k = \frac{(u_k - u_c)(hfcos\phi + fhtan\phi sin\phi) + 2hu_c sin\phi(v_c - v_k)}{f(v_k + ftan\phi - v_c)} \tag{4}$$

$$y_k = \frac{h(f - v_k tan\phi + v_c tan\phi)}{v_k + ftan\phi - v_c} \tag{5}$$

IV. Tracking Phase

A. Overview on RFS Statistics

The Random Finite Set (RFS) is a hidden markov chain model with set-valued states and set-valued observations while the PHD filter is a predict and correct framework for recursive Bayesian filtering in such a RFS formulation. The RFS approach to multiple-target tracking is an emerging and promising alternative to the traditional association-based methods [20] [2]. A comparison of the RFS approach and traditional multiple-target tracking methods has been given in [20]. The focus of this paper is the PHD filter, a recursion that propagates the first-order statistical moment, or intensity, of the RFS of states in time [2] [21]. In the PHD filter, the collection of individual targets is treated as set-valued states, and the collection of individual observations is treated as set-valued observations. Fig. 2 is a basic introduction of the PHD filter which shows that the observations and their estimated states are treated as a single valued measurement and its corresponding estimation at each frame [22]. The PHD filter operates on the single-target state space and avoids the combinatorial problem that arises from data association.

The Gaussian Mixture Probability Hypothesis Density (GM-PHD) filter is a closed form implementation of the PHD filter, which is based on the Bayesian estimation framework utilizing random finite sets as the mathematical backbone [21].

B. Kalyan [23] and John. M [24] implemented the PHD filter in the field of simultaneous localization and mapping (SLAM) problem. Results show that proposed PHD filter is an effective solution to the SLAM problem.

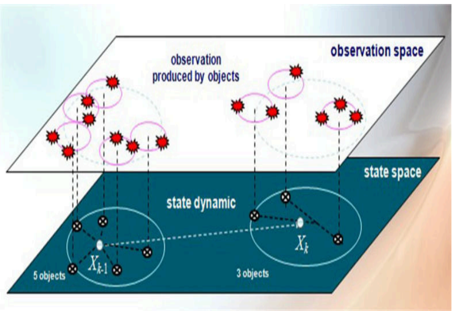


Fig. 2. Set-valued states and Set-valued observations

In this paper, we use the PHD filter to estimate the motion of the target set instead of tracking individual targets. Assuming the motion of the whole targets to be equal, we calculate the average of each target's state to acquire the motion of the whole set. From the physical model we consider the whole set to have the same motion vector as the vehicle. According to this model we estimate the vehicle's ego-motion vector at each frame. The differences between this paper and the PHD-SLAM approaches [24] [23] are as follows: This paper presents the PHD filter to estimate the trajectory of the vehicle from complex scenes, which contain many moving targets and clutter. Our approach focuses on the implementation of the PHD filter for visual odometry under real traffic scenes and a motion model to not only predict the ego-motion vector but also distinguish real targets from clutter. Compared to earlier approaches, our work is among the first to apply PHD filtering to visual odometry in real traffic scenes.

B. Mathematic Background on the PHD Filter

The RFS is a hidden markov chain model with set-valued states and set-valued observations while the PHD filter is a predict and correct framework for recursive Bayesian filtering in such a RFS formulation. It is an approximation to alleviate the computational intractability of the optimal multi-target Bayes filter, proposed by Mahler [2].

The targets in a multi-target scenario at time k is represented as a finite set of vectors $\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,N(k)}$ which takes values from the state space $\mathcal{X} \in \mathbb{R}^{n_\mathbf{x}}$. Similarly the observations are represented as a finite set of vectors $\mathbf{z}_{k,1}, \ldots, \mathbf{z}_{k,M(k)}$ which takes values from the observation space $\mathcal{Z} \in \mathbb{R}^{n_\mathbf{z}}$. $N(k)$ and $M(k)$ represent the number of targets and observations at time k respectively. These finite sets are known as the multi-target state and observation:

$$X_k = \{\mathbf{x}_{k,1}, \ldots, \mathbf{x}_{k,N(k)}\} \in \mathcal{F}(\mathcal{X}) \tag{6}$$

$$Z_k = \{\mathbf{z}_{k,1}, \ldots, \mathbf{z}_{k,M(k)}\} \in \mathcal{F}(\mathcal{Z}) \tag{7}$$

where $\mathcal{F}(\mathcal{X})$ and $\mathcal{F}(\mathcal{Z})$ denote the sets of all finite subsets of $\mathcal{X}$ and $\mathcal{Z}$, respectively.

The model must encapsulate the time varying numbers of targets in a multi-target scenario. Also the model must consider sensor imperfections such as missed detections and false alarms. The multi-target state is modeled as the union of different random finite sets:

$$X_k = [\bigcup_{\zeta \in X_{k-1}} S_{k|k-1}(\zeta)] \cup \Gamma_k \tag{8}$$

$S_{k|k-1}$ represents the targets that have survived from the previous time increment $k-1$. It is modeled as a Bernoulli RFS which means it can either survive with probability $P_{S,k}(\mathbf{x}_{k-1})$ and take on the new value $\{\mathbf{x}_k\}$ with probability density $f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$ or die and take on the empty set $\varnothing$ with probability $1 - P_{S,k}(\mathbf{x}_{k-1})$. Γ_k represents targets which are spontaneously born at the current time k. It is modeled as a Poisson RFS which is specified by a mean birth rate and spatial birth density, or equivalently by its PHD or intensity $\gamma_k(\cdot)$ where the mean birth rate is $\int \gamma_k(\mathbf{x})d\mathbf{x}$ and the spatial birth density is $\gamma_k(\cdot)/\int \gamma_k(\mathbf{x})d\mathbf{x}$.

Similarly the set observation Z_k can be seen as the union of two random finite sets:

$$Z_k = [\bigcup_{\mathbf{x} \in X_k} \Theta_k(\mathbf{x})] \cup K_k \tag{9}$$

Θ_k represents the measurements that originate from the targets and is modeled as a Bernoulli RFS which generates a detection with probability $P_{D,k}(\mathbf{x}_k)$ and yields the measurement $\{\mathbf{z}_k\}$ with probability density $g_k(\mathbf{z}_k|\mathbf{x}_k)$ or results in a missed detection yielding an empty measurement set $\varnothing$ with $1 - P_{D,k}(\mathbf{x}_k)$.

K_k represents the set of false alarms or clutter and is modeled as a Poisson RFS, specified by its intensity $\kappa_k(\cdot)$ where the mean clutter rate is $\int \kappa_k(\mathbf{z})d\mathbf{z}$ and the spatial clutter density is $\kappa_k(\cdot)/\int \kappa_k(\mathbf{z})d\mathbf{z}$.

Using these random finite set models it is possible to construct multi-target dynamical and observation models analogous to the single-target case. Randomness in X_k and Z_k can be encapsulated into a multi-target transition density and multi-target observation likelihood.

Under the above models, the multi-target Bayes filter propagates the posterior multi-target density $\pi_k(\cdot|Z_{1:k})$ recursively in time. However, due its combinatorial nature, it is intractable in most applications. To alleviate this, the PHD filter propagates the first moment or PHD $D_k(\cdot)$ of multi-target posterior density $\pi_k(\cdot)$.

The PHD recursion is given by :

$$\begin{gathered} D_{k|k-1}(\mathbf{x}_k) = \\ \int P_{S,k}(\mathbf{x}_{k-1}) f_{k|k-1}(\mathbf{x}_k|\mathbf{x}_{k-1}) D_{k-1}(\mathbf{x}_{k-1}) d\mathbf{x}_{k-1} \\ +\gamma_k(\mathbf{x}_k) \end{gathered} \tag{10}$$

$$\begin{gathered} D_k(\mathbf{x}_k) = (1 - P_{D,k}(\mathbf{x}_k)) D_{k|k-1}(\mathbf{x}_k) \\ + \textstyle\sum_{z \in Z_k} \frac{P_{D,k}(\mathbf{x}_k) g_k(\mathbf{z_i}|\mathbf{x}_k) D_{k|k-1}(\mathbf{x}_k)}{\kappa_k(\mathbf{z_i}) + \int P_{D,k} g_k(\mathbf{z_i}|\zeta) D_{k|k-1}(\zeta) d\zeta} \end{gathered} \tag{11}$$

Equation (12) illustrates that the integral of the PHD over a certain domain Ψ yields the estimated number of targets $N(k)$ in that domain at time k. The PHD is not a probability density and does not necessarily sum to 1[2].

$$N(k) = \int_{\Psi} D_k(\mathbf{x}_k) d\mathbf{x}_k \tag{12}$$

It is to be noted that the PHD recursion involving equations (11) and (11) have multiple integrals that have no closed form solutions in general. One of the common approaches to mitigate this problem is to use GM-PHD approximations. The GM-PHD filter [21] is a specialized version of the PHD filter. It assumes that the target's motion and observation process can be modeled as:

$$f_{k|k-1}(\mathbf{x}|\zeta) = \mathcal{N}(\mathbf{x}; F_{k-1}\zeta, Q_{k-1}) \tag{13}$$

$$g_k(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; H_k\mathbf{x}, R_k) \tag{14}$$

where $\mathbf{x}$ refers to the current state, $\mathbf{z}$ to the current measurement, ζ to the previous state, $\mathcal{N}(\cdot; \mathbf{m}, P)$ denotes a Gaussian distribution with mean $\mathbf{m}$ and covariance P, F_{k-1} is the state transition matrix, Q_{k-1} is the process noise covariance, H_k is the observation matrix, and R_k is the observation noise covariance. Survival and detection probability are supposed constant on the entire observed area:

$$P_{S,k}(\mathbf{x}) = P_S, P_{D,k}(\mathbf{x}) = P_D \tag{15}$$

Birth targets γ_k are modeled by a RFS written as a Gaussian mixture:

$$\gamma_k(\mathbf{x}) = \sum_{i=1}^{J_{\gamma,k}} \omega_{\gamma,k}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)}) \tag{16}$$

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

If the posterior PHD at time $k-1$ is a Gaussian mixture:

$$D_{k-1}(\mathbf{x}) = \sum_{i=1}^{J_{k-1}} \omega_{k-1}^{(i)} \mathcal{N}(\mathbf{x}; \mathbf{m}_{k-1}^{(i)}, P_{k-1}^{(i)}) \tag{17}$$

then the predicted PHD (11) to time k is 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 update PHD equation (11) at time k is also a Gaussian mixture and is given by

$$D_k(\mathbf{x}) = (1 - P_D) D_{k|k-1}(\mathbf{x}) + \sum_{z \in Z_k} D_{D,k}(\mathbf{x}; \mathbf{z}) \tag{18}$$

where

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

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

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

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

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

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

A table of codes for the GM-PHD filter is in [21]. Corresponding to the approximate multi-sensor update for the PHD recursion, the approximate multi-sensor update for the GM-PHD filter is obtained by consecutive application of the GM update step given directly above.

Although the original GM-PHD filter applies to linear Gaussian multi-target models, the formulation can accommodate non-linear dynamics and measurement models by linearization or unscented transforms. An Extended Kalman (EK) and Unscented Kalman (UK) version of the GM-PHD filter has also been proposed in [21].

C. Implementation Details

Our algorithm is implemented in 2D vehicle coordinates. In this paper, we treat the positions of the features as the measurements on each frame. Suppose at frame k, the coordinates of one feature point is (x_k, y_k). According to Euler's rotation theorem the relationship between the feature points in consecutive frames is as follows:

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} cos\Delta\beta_k & -sin\Delta\beta_k \\ sin\Delta\beta_k & cos\Delta\beta_k \end{bmatrix} \begin{bmatrix} x_k - \Delta x_k \\ y_k - \Delta y_k \end{bmatrix} \tag{19}$$

where $(\Delta x_k, \Delta y_k, \Delta\beta_k)$ is the vehicle's ego-motion vector. In this paper, $(\Delta x_k, \Delta y_k)$ is the vehicle's movement in (x, y) direction and $\Delta\beta_k$ is the change of the vehicle's rotation angle.

We consider the features as the targets and clutter while the targets satisfy the above equations and the clutter does not. In this paper, the targets are associated features from the ground plane and the clutter are features from falsely matched pairs, moving objects or associated features which are not on the ground plane (cannot provide the accurate location information). The goal of this paper is to utilize the PHD filter to estimate the targets' states at each frame, and then calculate the vehicle's ego-motion vector. The benefit of using PHD filter is that it models the set-valued states and set-valued observations as RFS and allows to solve the problem of dynamically estimating multi-targets in the presence of clutter and association uncertainty in a Bayes filtering framework.

- From the physical model we assume the targets have the same motion process. In the same manner like equation (19), we assume that at frame k the targets move independently and the motion process parameters are:

$$\mathbf{x}_k = [x_k, y_k, \beta_k, \dot{x}_k, \dot{y}_k]^T \tag{20}$$

$$\mathbf{F} = \begin{bmatrix} cos\beta_k & -sin\beta_k & 0 & -cos\beta_k & sin\beta_k \\ sin\beta_k & cos\beta_k & 0 & -sin\beta_k & -cos\beta_k \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{21}$$

The process noise is defined by:

$$Q_k = diag([\sigma_{x1}^2, \sigma_{y1}^2, \sigma_{\beta 1}^2, \sigma_{\dot{x}1}^2, \sigma_{\dot{y}1}^2]) \tag{22}$$

- The measurement vector is as follows:

$$\mathbf{z} = [x, y]^T \tag{23}$$

Where (x, y) is acquired according to the coordinates transformation process (transformed from the image coordinates to the vehicle coordinates). To map the state vectors to the observation space, the observation matrix is:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{24}$$

The observation noise is described as:

$$R_k = diag([\sigma_x^2, \sigma_y^2]) \tag{25}$$

- Aggregated targets falling below a given threshold are pruned and the remaining targets are reweighted accordingly (more details about weight can be found in [21]). If the distance of the targets defined by the state matrix and covariance matrix falls within a merging threshold τ, then the targets are merged. More specifically, starting with the targets with the weights ω_k^j, we merge targets in set M_k^j as follows:

$$M_k^j := \{i : (\mathbf{m}_k^i - \mathbf{m}_k^j)^T (P_k^i)^{-1} (\mathbf{m}_k^i - \mathbf{m}_k^j) \leq \tau\} \tag{26}$$

According to the pruned and merged technology, the PHD filter can effectively process the features when they are unevenly distributed on the whole space or even aggregated in a small region, which may influence the performance of the estimation.

From the physical model we consider the whole group set to have the same motion vector. According to this motion model we calculate the average state of the targets as the ego-motion vector at frame k.

$$\mu_{\mathbf{k}} = [\Delta x_k, \Delta y_k, \Delta\beta_k]^T \tag{27}$$

where $(\Delta x_k, \Delta y_k, \Delta\beta_k)$ is the mean of the state set.

Since the PHD filter estimates the whole set's motion vector within the RFS framework, $\mu_{\mathbf{k}}$ is not only used to calculate the trajectory of the vehicle at frame k, but also used to initialize the parameters of the birth models at frame $k+1$.

In this paper, the PHD filter is used as follows: Suppose at frame k there are a total number of n associated feature pairs between two continues frames, the ego-motion vector is calculated as $\mu_{\mathbf{k}}$ from the previous estimations, the proposed method considers previous frame's estimated targets' set as the existed targets with corresponding weights, the features in frame k are considered as the birth targets (the position parameters are initialized by the coordinates of the points, the other parameters are initialized by the previous estimated motion vector $\mu_{\mathbf{k}}$), the features in frame $k+1$ are considered as the measurements to the PHD filter. Finally, the PHD filter calculates the targets state and discards the clutter under the RFS framework. This is how the PHD filter works. However, in our previous work [5], the PHD filter recorded each frames estimated targets in the system as the existed targets and used the pruned and merged technology to distinguish the effective targets. Since a single camera cannot provide depth information of the features, the PHD filter in this paper uses data from three consecutive frames. Otherwise, the inaccurate measurements may influence the performance of the estimation.

The differences between our approach and visual odometry using Kalman filter (tracking individual features) are as follows:

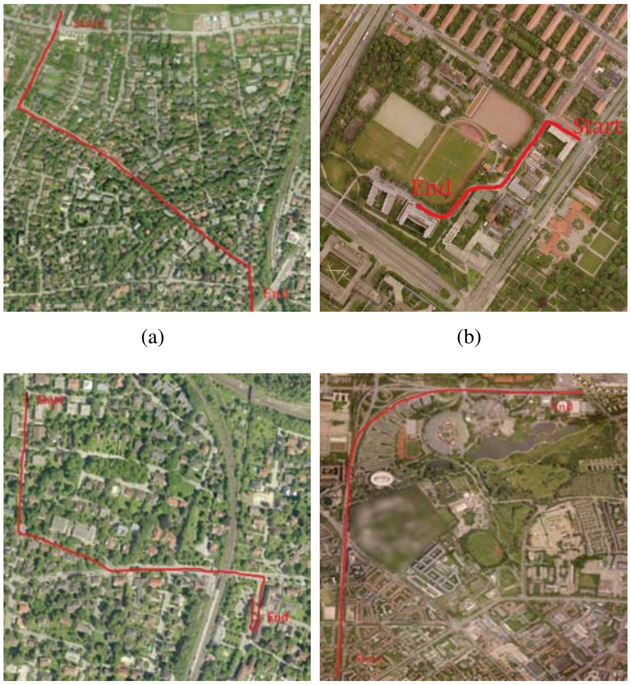
First, the Kalman filter is only used to track individual targets. A matching process is required before using the Kalman filter. Even though the features which have been matched may avoid the data association problems, the false features pairs still influence the estimation while the PHD filter can distinguish the false features as clutters according to the random set statistics. Second, the feature's life cycle in the tracking process is short (some features may only be matched once in two consecutive frames and disappear later). The Kalman filter cannot calculate the optimal estimate in such conditions. In our approach, we calculate the optimal estimate by updating the PHD filter according to the birth models and the survival models in consecutive frames within the whole process, which continues to provide the optimal estimation at each frame. Third, the Kalman filter is neither able to aggregate measurements nor to separate one measurement to multiple targets. It needs a one-to-one relation between real measurements and expected measurements, solved by data association. The PHD filter overcomes this, as it is an n-to-m mapping, which is a robust way for tracking group targets in cluttered environments. It calculates the average of each target's state to acquire the whole set's motion vector within the RFS framework.

Since equation (21) is nonlinear, we use the GM-EK-PHD filter to estimate the state which is very similar to the GM-PHD. More details about EK-PHD can be found in [21].

V. Experimental Implementation and Evaluation

The visual odometry algorithm described in this paper has been implemented on a Core **2** Duo 3.0Ghz computer. An iPhone4 platform is used to record data at 30 frames/s which included GPS, gyro and images with a resolution of 480 × 640. All sequences correspond to real traffic conditions in urban environments with pedestrians and other cars. In the experiments, the vehicle was driven with an average velocity of 50km/h.

As can be observed, we also use the RANSAC (based on the least squares method) toolbox to estimate the ego-motion vector [25]. The RANSAC toolbox is public software in Matlab, it combines the non-linear least square method and RANSAC together to estimate the rotation angle and translation vector in 2D Cartesian space. More details can be found in [25]. In this paper, we utilize a standard Kalman filter to estimate the RANSAC results in order to remove noise. The inputs for the RANSAC are the matched feature pairs while the inputs for the PHD filter are the same but without association. Assuming the whole features contain 30% outliers to fit the RANSAC model, a dead reckoning method [26] is used to calculate the trajectory according to the ego-motion vector provided by the RANSAC algorithm (after filtering techniques) and the PHD filter.



(a) (b)
(c) (d)

Fig. 3. Aerial view of the path

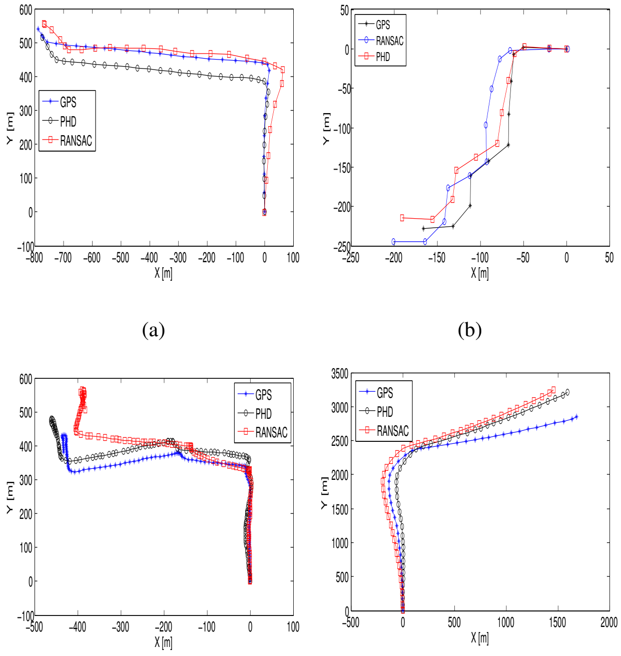
Fig. 3 and Fig. 4 show the result of our approach. Fig. 3 shows an aerial view of the area where the experiment was conducted [27]. Fig. 4 illustrates the 2D trajectory estimated by the visual odometry algorithm presented in this paper compared with RANSAC algorithm. From Fig. 4 we can see that our approach provides good robustness under real traffic scenarios.

Since the feature pairs' locations in vehicle coordinates may not be correct (our transformation is only active for those features on the road while others may fail, the reason is that we use a single camera which is fixed at a certain height with a predefined orientation to the ground, it needs certain conditions to reconstruct the location), the estimated results on Fig. 4 may have certain bias from its real status during the visual odometry process, which can be seen from the image. The other reason why the bias exists on Fig. 4 might be from the rotation estimation. Compared with our previous work, the rotation angle is estimated according to the non-

TABLE I

PERFORMANCE OF THE ALGORITHM

Index	Size	Distance	Frames	RANSAC Distance error	PHD Distance error
a	480 × 640	1281m	3510	24m(1.8%)	28m(2.1%)
b	480 × 640	413m	1500	38m(9.2%)	27m(6.5%)
c	480 × 640	950m	1800	135m(14.2%)	47m(4.9%)
d	480 × 640	4078m	6600	458m(11.2%)	376m(9.2%)



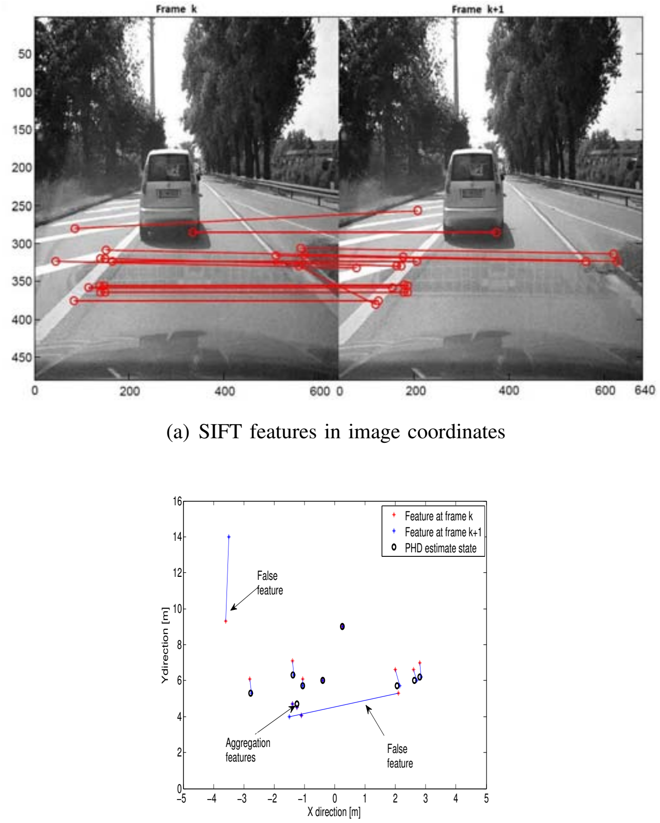
(a)

(b)

(c)

(d)

Fig. 4. Visual odometry result



(a) SIFT features in image coordinates

(b) Results in vehicle coordinates

Fig. 5. SIFT feature pairs and the PHD estimation results

linear mapping from the PHD filter's motion equation and the corresponding measurement equation, the precision of the rotation estimation is lower than the rotation angle which was directly acquired from the gyroscope. However, the PHD filter still leads to a good robustness compared with the RANSAC approach.

Table I summarizes the results for the data set. The index illustrates the experiments on Fig. 4 and the distance is the length of the experiments conducted. In this paper, the distance error means the distance between the estimated location and the GPS location in the end.

These results show that better accuracy can be obtained with the PHD filter compared with the RANSAC method.

Fig. 5 illustrates three benefits compared with RANSAC in visual odometry:

1) RANSAC has been established as the standard method for motion estimation in the presence of outliers (false features). RANSAC achieves its goal by iteratively selecting a random subset of the original data. However, the PHD filter treats the false features as clutters within the RFS framework by using the dynamics model (physical laws of motion).
2) From Fig. 5(b) we can see that there exist feature (or target) aggregation, which means that a small region contains lots of features (targets). This phenomenon may influence the estimation results since these features may not be effectively chosen according to RANSAC. However, with the pruning and merging approach in PHD filter, the estimated targets are uniformly distributed in the whole space and contribute with the same importance to the result.
3) It is difficult for RANSAC to remove features that were originating from moving objects. These features can influence the precision of the results. However, within the RFS framework, the PHD filter propagates the posterior intensity, a first-order statistical moment of the posterior multiple-target state at each frame. According to the dynamic system model (equation

(13)), it can utilize the estimated state to reduce the influences from those features.

4) There must be prematched feature pairs in RANSAC. As a matter of fact, most visual odometry methods need prematching, such as using Harris corner features [28]. The system should have the same number of feature pairs on two consecutive frames and then calculate the motion vector. However, our PHD visual odometry system, which can easily be applied to these visual odometry systems (focus on features, not the optical flow) since the PHD filter avoids the data association problem and also due with different number of features at each frame without concerning the rough feature preprocessing approaches. This insures that it is theoretically possible that there is no loss of information from the original measurement set compared to other methods which focus on features.

The results of our experiments indicate that the algorithm performs robustly in the presence of pedestrians, vehicles and shadows on the road.

VI. Conclusion

Visual odometry using a monocular camera in urban scenes is challenging due to a large amount of outliers. The clutter from falsely matched features and moving objects cause the results to deviate from the real status. In this paper, an approach of PHD filtering under RFS framework is presented. In comparison to the earlier works, this contribution is among the first to apply a PHD filter to visual odometry in real traffic scenes. With this, visual features are considered as a group target – we then utilize the average of each target's state to approximate the ego-motion vector since all targets should have the same motion vector in the whole group set within the RFS framework. Compared to other approaches, our approach presents a recursive filtering algorithm that provides dynamic estimation of multiple-target states in the presence of clutter and high association uncertainty. The evaluation results show that the algorithm achieves robustness under different scenes.

Future work should include other sensors such as laser, stereo cameras and wheel sensors to improve the estimation precision and robustness.

References

[1] R. Garcia, M. Sotelo, I. Parra, D. Fernandez, and M. Gavilan, "2d visual odometry method for global positioning measurement," in *IEEE International Symposium on Intelligent Signal Processing, WISP 2007*, Oct. 2007, pp. 1 –6.

[2] R. Mahler, "Multitarget bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic Systems*, no. 4, pp. 1152 – 1178, Vol. 39, Oct. 2003.

[3] R.Mahler, "A theoretical foundation for the stei-winter probability hypothesis density (phd) multi-target tracking approach," in *Proc. 2002 MSS Nat . Symp. Sensor Data Fusion*, Vol. 2, 2002.

[4] B.-N. Vo, S. Singh, and A. Doucet, "Sequential monte carlo implementation of the phd filter for multi-target tracking," in *Proceedings of the Sixth International Conference on Information Fusion*, 2003, pp. 792 – 799, Vol. 2.

[5] F. Zhang, G. Chen, H. Stahle, C. Buckl, and A. Knoll, "Visual odometry based on random finite set statistics in urban environment," in *Intelligent Vehicles Symposium (IV), 2012 IEEE*, june 2012, pp. 69 –74.

[6] A. Davison, "Real-time simultaneous localization and mapping with a single camera," in *International Conference on Computer Vision*, 2003, pp. 1403–1410.

[7] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004*, July 2004, pp. I–652 – I–659, Vol. 1.

[8] D. Burschka and G. Hager, "V-gps(slam): vision-based inertial system for mobile robots," in *2004 IEEE International Conference on Robotics and Automation*, May 2004, pp. 409 – 415, Vol.1.

[9] A. Milella and R. Siegwart, "Stereo-based ego-motion estimation using pixel tracking and iterative closest point," in *IEEE International Conference on Computer Vision Systems, 2006 ICVS '06*, Jan. 2006, p. 21.

[10] B. Kitt, A. Geiger, and H. Lategahn, "Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme," in *Intelligent Vehicles Symposium (IV), 2010 IEEE*, June 2010, pp. 486 –492.

[11] D. Scaramuzza and R. Siegwart, "Appearance-guided monocular omnidirectional visual odometry for outdoor ground vehicles," *IEEE Transactions on Robotics*, no. 5, pp. 1015 –1026, Vol. 24, Oct. 2008.

[12] D. Scaramuzza, F. Fraundorfer, M. Pollefeys, and R. Siegwart, "Absolute scale in structure from motion from a single vehicle mounted camera by exploiting nonholonomic constraints," in *2009 IEEE 12th International Conference on Computer Vision*, Oct. 2009, pp. 1413 –1419.

[13] C. McCarthy and N. Barnes, "Performance of optical flow techniques for indoor navigation with a mobile robot," in *IEEE International Conference on Robotics and Automation, ICRA '04*, May 2004, pp. 5093 – 5098, Vol. 5.

[14] J. Campbell, R. Sukthankar, and I. Nourbakhsh, "Techniques for evaluating optical flow for visual odometry in extreme terrain," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2004*, Sept. 2004, pp. 3704 – 3711, Vol. 4.

[15] P. Corke, D. Strelow, and S. Singh, "Omnidirectional visual odometry for a planetary rover," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2004*, Oct. 2004, pp. 4007 – 4012, Vol. 4.

[16] A. Comport, E. Malis, and P. Rives, "Accurate quadrifocal tracking for robust 3d visual odometry," in *Robotics and Automation, 2007 IEEE International Conference on*, april 2007, pp. 40 –45.

[17] D. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999, pp. 1150 –1157, Vol. 2.

[18] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *The Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999, pp. 666 –673, Vol. 1.

[19] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[20] I. Goodman, R. Mahler, and H. Nguyen, *Mathematics of Data Fusion*. Norwell, MA: Kluwer Academic, 1997.

[21] B.-N. Vo and W.-K. Ma, "The gaussian mixture probability hypothesis density filter," *IEEE Transactions on Signal Processing*, no. 11, pp. 4091 –4104, Vol. 54, Nov. 2006.

[22] B.-N. Vo, "Random finite sets in stochastic filtering," EEE Department University of Melbourne Australia, Tech. Rep., 2009. [Online]. Available: http://www.ee.unimelb.edu.au/staff/bv/

[23] B. Kalyan, W. S. Wijesoma, and K. W. Lee, "Fisst-slam: Finite set statistical approach to simultaneous localization and mapping," *International Journal of Robotics Research*, no. 2, Sept. 2010.

[24] J. Mullane, B.-N. Vo, M. Adams, and B.-T. Vo, "A random-finite-set approach to bayesian slam," *IEEE Transactions on Robotics*, no. 2, pp. 268 –282, Vol 27, April 2011.

[25] M. Zuliani, "Ransac toolbox for matlab," [web page] http://www.mathworks.com/matlabcentral/fileexchange/18555, Nov. 2008.

[26] N. Houshangi and F. Azizi, "Mobile robot position determination using data integration of odometry and gyroscope," in *Automation Congress, 2006. WAC '06. World*, july 2006, pp. 1 –8.

[27] "Openstreetmap database," [web page] http://www.openstreetmap.org/.

[28] C. Harris and M. Stephens, "A combined corner and edge detector," in *Alvey Vision Conference*, 1988, pp. 147–152.