2009 IEEE International Conference on Robotics and Automation
Kobe International Conference Center
Kobe, Japan, May 12-17, 2009

Laser-based Detection and Tracking Moving Objects using Data-Driven Markov Chain Monte Carlo

Trung-Dung Vu and Olivier Aycard
INRIA Rhône Alpes, Grenoble, France
firstname.lastname@irialpes.fr

Abstract— **We present a method of simultaneous detection and tracking moving objects from a moving vehicle equipped with a single layer laser scanner. A model-based approach is introduced to interpret the laser measurement sequence by hypotheses of moving object trajectories over a sliding window of time. Knowledge of various aspects including object model, measurement model, motion model are integrated in one theoretically sound Bayesian framework. The data-driven Markov chain Monte Carlo (DDMCMC) technique is used to sample the solution space effectively to find the optimal solution. Experiments and results on real-life data of urban traffic show promising results.**

I. INTRODUCTION

Recent years have seen many research works employing laser scanners to detect and track moving objects [11], [9], [5], [2]. Existing methods usually separate the detection and tracking as two independent procedures. Since detection at one time instant usually results in ambiguities that make the data association become more difficult with missing detections and false alarms. We propose a probabilistic framework for simultaneous detection and tracking of objects which combines the detection and tracking together. This allows object detection to make use of temporal information and facilitates robust tracking of the objects.

On the other hand, state-of-the-art laser-based approaches to detect moving objects usually involves performing segmentation of laser points and representing objects by bounding-boxes of laser segments. These conventional approaches pose several problems. Firstly, since a laser scanner only sees parts of the object currently facing the scanner, as the object moves we get different contours (Fig. 1(a)) that leads to a significant degradation of tracking results (Fig. 1(b)). Secondly, due to occlusions or laser-absorbed surfaces (ex: glasses, black surfaces), an object can be divided into several segments (Fig. 1(d),(e)). This makes object detection and tracking much harder when dealing with object merging and track grouping. Here we take a model-based approach and will discuss how using object models of several predefined classes to interpret the laser measurements can overcome these problems (Fig. 1(f)).

We formulate the detection and tracking problem as finding the most likely trajectories of moving objects given measurements over a sliding window of time (Fig. 2). A trajectory (track) is regarded as a sequence of object shapes produced over time by an object satisfying the constraint of an underlying measurement model and the smoothness in motion from frame to frame. In this way, our approach can be seen as a batch method searching for the global optimum solution in the spatio-temporal space. Due to the high computational complexity of such a scheme, we employ a data driven Markov chain Monte Carlo (DDMCMC) technique [14] that enables traversing efficiently in the solution space. The key contribution of this paper is designing a general framework to perform object detection and tracking at the same time with an explicit integration of various aspects including prior information, object model, measurement model, motion model in a theoretically sound formulation. We test the algorithm on real-life data of urban traffic and the results demonstrate the effectiveness of our approach.

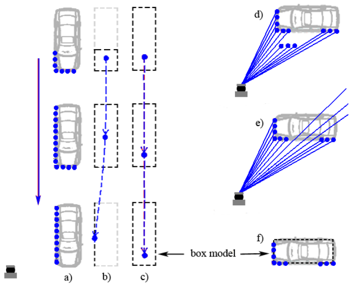

Fig. 1. Known problems with laser-based detection and tracking. (a) an example of a car approaching the laser scanner; (b) car tracking using centers of laser impact bounding-boxes leads to incorrect result; (c) correct tracking using car model; (d)(e) objects can be divided into several segments making tracking harder that requires object merging, track grouping; (f) using car model can overcome these problems.

The paper is organized as follows. The related work is reviewed in Section II. We formulate the general multiple object detection and tracking problem in Section III. In Section IV we present our algorithm to find the optimal trajectories of moving objects using a spatio-temporal MCMC sampling method. We discuss experiments and provide some initial results on real-life traffic datasets in Section V, followed by conclusions and future works.

II. RELATED WORKS

Multiple object tracking (MOT) has been an active research topic over decades. When object observations are known, object tracking becomes a data association problem. Among popular data association methods, multiple hypothesis tracking (MHT) [1] is widely used. MHT is a multi-frame tracking method that is capable of handling ambiguities in data association by propagating hypotheses until they can be solved when enough observations are collected. The main

978-1-4244-2789-5/09/$25.00 ©2009 IEEE

disadvantage of MHT is its computational complexity since the number of hypotheses grows exponentially over time. The joint probabilistic data association (JPDA) filter [6] is more efficient but prone to make erroneous decision since only single frame is considered and the association made in the past is not reversible. Other sequential approaches using particle filters [7] share the same weakness that they cannot reverse time back when ambiguities exist. Recently, batch MOT methods [4], [12] attracted research attentions. Based on a powerful MCMC technique [14], the entire solution space can be explored in an efficient way to find the global optimum. Oh *et al.* [4] used MCMC to organize the temporal association between objects and punctual observations over time. This method is extended in [12] to solve the data association in a spatio-temporal space.

Taking object detection result as input for tracking, trackers are different by detection methods. In [6], Schulz introduced a simple method only to detect people using the local minimal caused by legs. Wang [11], followed by Vu [9] proposed a more general approach which build an online occupancy grid representing the local map and moving parts of objects can be detected when they enter object-free regions. However, ambiguities occur when objects appear in unexplored regions, it is difficult to say it is moving or static. Moreover, using laser impact contours to represent objects, this approach encounters problems of tracking degradation and object splitting as mentioned in the introduction section.

Our approach here emphasizes on the use of object models like that of [5] to overcome known problems of laser-based tracking. We propose to solve at the same time both the detection and tracking using a MCMC technique to search for the optimum solution in the spatio-temporal space. In this way, temporal information will help to reduce ambiguities and false alarms for detection at one time instant and facilitate a more robust tracking. The moving object detection result in our previous work [9] is employed to generate object hypotheses that helps to drive the MCMC search more efficiently. In addition, we introduce a new way to model the temporal prior and the likelihood that takes the whole considered measurement sequence into account.

III. MULTIPLE OBJECT TRACKING FORMULATION

We consider tracking in a sliding window of time comprised of $T \in N^+$ last frames. Let Z be the set of all measurements within the time interval $[1,T]$ and Z_t denote the laser measurement at time t. The current time corresponds to $t = T$. Assume that within $[1,T]$ there are K unknown number of moving objects appear in the vehicle sensing range.

Not like [5] only considering vehicles, we distinguish four classes of moving objects: bus, car, bike, pedestrian (motorcycles and bicycles belong to the class bike). We use the box model of fixed sizes to represent bus, car, bike and the point model to represent pedestrian. For the box model, object is parameterized by $M = \{c, x, y, \theta, w_c, l_c\}$ which are object class, object center position, orientation, width and length respectively. Herein w_c and l_c are constants with respect to each object class c. For the point model, object is parameterized by $M = \{c, x, y\}$ which are object class and object center.

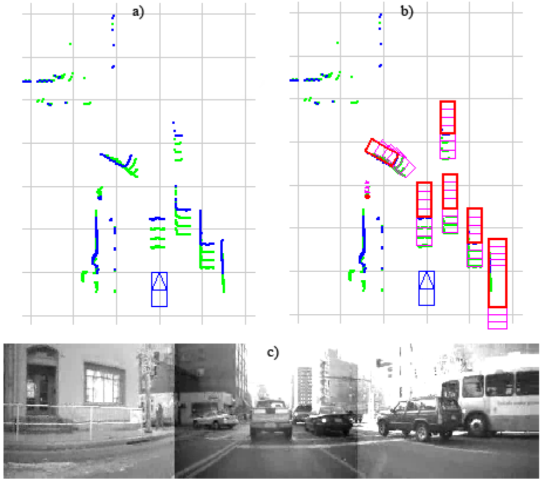

Fig. 2. Example of an interpretation of moving objects from laser data. (a) four scans consecutive: in blue is current scan and in green are scans in the past; (b) one possible solution including seven tracks of four cars and one bus represented by red boxes and one pedestrian represented by red dots which are imposed on the range data; (c) situation reference.

We cast the detection and tracking problem as interpreting the given observation set Z by an unknown number of K tracks $\omega = \{\tau_1, \tau_2, ..., \tau_K\}$. Each track τ_k in ω is defined as a sequence of moving objects of the same class $\tau_k = \{\tau_k(t_1), ..., \tau_k(t_{|\tau_k|})\}$ where $t_i \in [1,T]$, $\tau_k(t)$ is a moving object detected at time t and is either of form $\{c, x, y, \theta, w, l\}$ or $\{c, x, y\}$. Since missing detection may happen $1 \leq (t_{i+1} - t_i) \leq t_{max}$. We introduce the notation $\omega_t = \bigcup_{k=1}^{K} \tau_k(t)$ that is the set of moving objects detected at time t.

Fig. 2 shows an example of four laser scans and one possible interpretation of moving objects which is comprised of seven tracks of five cars, one bus and one pedestrian.

The moving object detection and tracking problem is formulated as maximizing a posterior (MAP) of an interpretation of tracks ω, given the set of observations Z over T frames:

$$\omega^* = \underset{\omega}{\operatorname{argmax}} P(\omega|Z) \tag{1}$$

Following the Bayes rule, the posterior probability is decomposed into a prior term and a likelihood term:

$$P(\omega|Z) \propto P(\omega)P(Z|\omega) \tag{2}$$

In the following sections, we discuss the prior and the likelihood model used in our framework.

A. Prior Distribution

Assuming that the occurrence of an object is independent of the others, we define the prior of a solution ω is the product of the probabilities of individual tracks:

$$P(\omega) = \prod_{k=1}^{K} P(\tau_k) \tag{3}$$

The probability $P(\tau_k)$ encodes the probability of the appearance of objects independent over time $P_O(.)$ and the temporal consistency of objects within track $P_T(.)$. The probability $P_T(.)$ controls the inner-smoothness of each track independently (Fig. 3). However, without an *a priori* knowledge of the number of targets, the inner-smoothness constraint will favor shorter paths, and therefore will split a trajectory into a large number of sub-tracks. To overcome this overfitting problem, we add the prior term $P_L(.)$ which encodes the preference of long track. We can write:

$$P(\tau_k) = P_L(\tau_k)P_O(\tau_k)P_T(\tau_k) \tag{4}$$

1) We adopt an exponential model of the length of each track:

$$P_L(\tau_k) \propto \exp(\lambda_L|\tau_k|) \tag{5}$$

2) The probability of appearance of objects in track independent of time:

$$P_O(\tau_k) = \prod_{i=1}^{|\tau_k|} P(\tau_k(t_i)) = P(c_k)\prod_{i=1}^{|\tau_k|} P(x_i,y_i)P(\theta_i) \tag{6}$$

where $P(c_k)$, $P(x_i,y_i)$ and $P(\theta_i)$ are the prior probabilities over the object class, object position and object orientation respectively. In our current implementation, we set these prior probabilities as uniform distributions. However, knowledge of road type and road border, if available can be added for better performance.

3) There remains how to model the temporal consistency within each track $P_T(\tau_k)$. This term is measured by the smoothness of object motion according to its underlying dynamics model. Since in urban scenarios objects are high maneuvers, we opt for multiple model approach [13] to model object dynamics. For classes of bus, car, bike, four modes of dynamics are used: constant velocity, constant acceleration, turning and stationary mode. For pedestrians, we force the acceleration to zero and only one constant velocity model is used. Box-model dynamic states are $(x,y,\theta,\dot{\theta},v,a)$ with the velocity v and acceleration a are always in the direction θ of the longer edge l. Dynamic states for point-model objects are (x,y,v_x,v_y).

We apply a IMM filter similar to [13] to estimate states of objects in the track sequentially: $\hat{\tau}_k(t_1),...,\hat{\tau}_k(t_{|\tau_k|})$ and their corresponding covariances: $cov(\hat{\tau}_k(t_1)),...,cov(\hat{\tau}_k(t_{|\tau_k|}))$. Set $cov(\tau_k) = cov(\hat{\tau}_k(t_{|\tau_k|}))$. A smaller value of $cov(\tau_k)$ implies that the track is more consistent to the dynamics model. The temporal consistency of a track is then calculated by:

$$P_T(\tau_k) \propto \exp(-\lambda_T cov(\tau_k)) \tag{7}$$

B. Likelihood Probability

The likelihood $P(Z|\omega)$ reflects the probability we observe the measurement Z given ω which contains the states of all moving objects over the time interval $[1,T]$.

To model this likelihood, first we identify dynamic measurements which are caused by moving objects in ω. Fig. 4 shows a box-model object (solid rectangle) and its dilated bounding box (dotted rectangles). The laser ray Az is considered as being caused by the object if it has the impact lying on BD that corresponds to the measurement on visible sides. We notice that if the impact lies on the segment AB, the object is occluded. If the impact lies on the ray Dz and it is not a maximum range reading, the visibility constraint is violated (rays seeing through glasses or black surfaces are not the case)For the point-model object, it is dilated by a circle of a fixed diameter and all measurements fall inside the circle are considered as dynamic measurements.

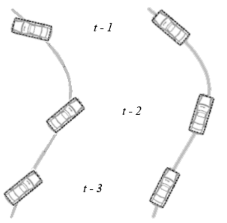

Fig. 3. Temporal consistency of a track. Obviously the arrangement of cars on the right is more relevant to a correct motion than that on the left.

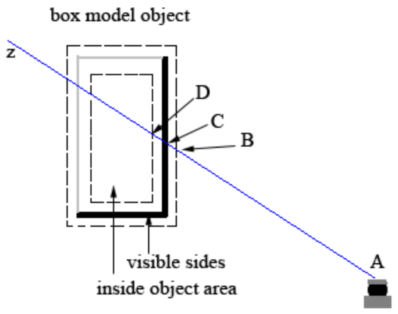

Fig. 4. Object measurement likelihood computation.

Let $Z^{(d)}$ denote all dynamic measurements of Z. We have $Z^{(s)} = Z - Z^{(d)}$ the remained measurements that are supposed to be caused by static objects. The notations $Z_t^{(d)}$ and $Z_t^{(s)}$ are dynamic and static measurements at time t respectively. In this way, besides information about dynamic objects, ω partitions Z into dynamic and static measurements. From frame to frame, these measurements should be coherent with each others. We can therefore decompose the likelihood into the product of two terms:

$$p(Z|\omega) = \prod_{i,j=1}^{T} P(Z_i|\omega_j) \prod_{i,j=1}^{T} P(Z_i|Z_j^{(s)}) \tag{8}$$

where the first term encodes the likelihood of measurements at each time step given observations of dynamic objects and the second term encodes the consistency of the measurements with the static parts of the environment. The first term is then further decomposed:

$$\begin{aligned}\prod_{i,j=1}^{T} P(Z_i|\omega_j) &= \prod_i P(Z_i|\omega_i) \prod_{i\neq j} P(Z_i|\omega_j) \\ &= \prod_i P_{M_1}(Z_i^{(d)}|\omega_i) P_{M_2}(Z_i^{(s)}|\omega_i) \prod_{i\neq j} P_{M_3}(Z_i^{(s)}|\omega_j) \end{aligned} \tag{9}$$

The second term in (8) is rewritten as:

$$\prod_{i,j=1}^{T} P(Z_i|Z_j^{(s)}) = \prod_{i\neq j} P_{M_4}(Z_i|Z_j^{(s)}) \tag{10}$$

The meaning of each probability component is as follows. P_{M_1} scores the fitness of dynamic measurements to the moving objects. P_{M_2}, P_{M_3} and P_{M_4} penalizes the violation of laser visibility constraint. In particularly, P_{M_2} penalizes situations that laser can see through dynamic objects. P_{M_3} penalizes the situations where moving objects are detected at position that has seen static objects. P_{M_4} penalizes the situations where laser can see through static objects. Fig. 5, in order from left to right, illustrates the meanings of probabilities $P_{M_1}, P_{M_2}, P_{M_3}$ and P_{M_4} respectively. Note that here we do not consider maximum range laser readings so that rays caused by glasses or black surfaces will not be penalized.

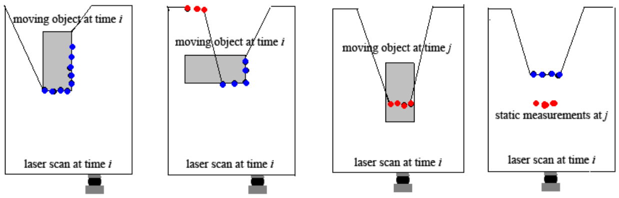
moving object at time *i*
laser scan at time *i*
moving object at time *i*
laser scan at time *i*
moving object at time *j*
laser scan at time *i*
static measurements at *j*
laser scan at time *i*

Fig. 5. Four types of constraint used to compute the likelihood. The red dots are measurements that violate the laser visibility constraint.

In the following, we discuss how to model these probabilities:

1) Assuming that $Z_i^{(d)}$ is comprised of N dynamic measurements $Z_i^{(d)} = \{z_i^1, ..., z_i^N\}$. We consider measurements obtained along each laser ray independent of each other. The measurement likelihood P_{M_1} factors as:

$$P_{M_1}(Z_i^{(d)}|\omega_i) = \prod_{n=1}^{N} P(z_i^n|\omega_i) \tag{11}$$

Each measurement z_i^n corresponds to a laser beam that hits an object in ω_i where the laser impact lies on the segment BD (Fig. 4). We model each ray's likelihood as a zero-mean Gaussian with respect to the distance d_n from the impact to the ideal measurement point C:

$$P(z_i^n|\omega_i) \propto exp(-\lambda_1 d_n) \tag{12}$$

2) For each object in ω_i, we count the number of non-maximal measurements $Z_i^{(s)}$ that fall behind the object model, call c_2 is the total number.

$$P_{M_2}(Z_i^{(s)}|\omega_i) \propto \exp(-\lambda_2 c_2) \tag{13}$$

P_{M_2} supports the remark following: If there exists such a violation of visibility constraint, there is unlikely a moving object appeared at that position.

3) Similarly, we count c_3 the number of measurements $Z_i^{(s)}$ that fall inside occupied areas of the objects in ω_j.

$$P_{M_3}(Z_i^{(s)}|\omega_j) \propto \exp(-\lambda_3 c_3) \tag{14}$$

P_{M_3} is in favor of the fact that if there is a moving object appeared at one time step then all measurements backward or afterward falling inside the object area should be dynamic measurements.

4) In the same way, let c_4 be the total number of violated measurements:

$$P_{M_4}(Z_i|Z_j^{(s)}) \propto \exp(-\lambda_4 c_4) \tag{15}$$

P_{M_4} holds the observation that if there are measurements in one frame observed passing through other measurements in another frame then the later measurements are likely dynamic measurements.

C. Posterior Probability

By combining the above likelihood and the prior probability, we get the posterior probability function as:

$$P(\omega|Z) \propto \exp\{\, \lambda_L S_{len} - \lambda_T S_{mot} - \lambda_1 S_{ms1} - \lambda_2 S_{ms2} - \lambda_3 S_{ms3} - \lambda_4 S_{ms4}\} \tag{16}$$

where $\lambda_L, \lambda_T, \lambda_1, \lambda_2, \lambda_3, \lambda_4$ are positive real constants and:

$$\begin{aligned} S_{len} &= \sum_{k=1}^{K} |\tau_k| & S_{mot} &= \sum_{k=1}^{K} cov(\tau_k) \\ S_{ms1} &= \sum_{i=1}^{T}\sum_{n=1}^{N} d_n^i & S_{ms2} &= \sum_{i=1}^{T} c_2^i \\ S_{ms3} &= \sum_{i\neq j} c_3^{ij} & S_{ms4} &= \sum_{i\neq j} c_4^{ij} \end{aligned} \tag{17}$$

IV. EFFICIENT POSTERIOR PROBABILITY COMPUTATION

Although the problem is restricted within a sliding window, searching in the solution space for Eq. (1) is still challenging. We employ a MCMC method to solve this problem. The basic idea of MCMC is as follows. A Markov chain can be designed to sample a probability distribution $\pi(\omega)$ (in our case $\pi(\omega) = P(\omega|Z)$). At each iteration, we sample a new state ω' from the current state ω following the proposal distribution $q(\omega'|\omega)$. The new candidate state ω' is accepted with the following probability $A(\omega, \omega')$ where

$$A(\omega, \omega') = min(1, \frac{\pi(\omega')q(\omega|\omega')}{\pi(\omega)q(\omega'|\omega)}) \tag{18}$$

otherwise the sampler stay at ω. The overview of MCMC algorithm is shown in Algorithm 1.

Algorithm 1 MCMC Sampler

```
Input: $Z, n_{mc}, \omega^* = \omega_0$ Output: $\omega^*$
for $n = 1$ to $n_{mc}$ do
    Propose $\omega'$ according to $q(\omega'|\omega_{n-1})$
    Sample $U$ from $Uniform[0,1]$
    if $U < A(\omega, \omega')$ then
        $\omega_n = \omega'$
        if $P(\omega_n|Z) > P(\omega^*|Z)$ then $\omega^* = \omega_n$
    else
        $\omega_n = \omega_{n-1}$
end for
```

This is the well-known Metropolis-Hasting algorithm. It is proved that the Markov chain constructed this way has its stationary distribution equal to $\pi()$, independent of the choice of the proposal probability $q()$ and the initial state ω_0 [8]. However, the choice of the proposal probability $q()$ can affect the efficiency of the MCMC significantly. A random proposal probability will lead to very slow convergence rate while a proposal probability designed with domain knowledge [14] will make the Markov chain traverse the solution space more efficiently. If the proposal probability is informative enough so that each sample can be thought of as a *hypothesis*, then the MCMC approach can be though of as a stochastic version of the *hypothesize and test* approach that is also named the data-driven MCMC method (DDMCMC).

A. *Moving object hypothesis proposals*

To make the proposals more informative, we take advantage of the detection module in our previous work [9] which can help to identify moving parts of dynamic objects. Combined with suitable object models, all possible object hypotheses are generated at location of these detected motion evidences. The principle is that we want to keep the detection rate high and accept false alarms to cover as many potential moving objects as possible. These rough hypotheses provide initial proposals for the MCMC sampler (Algorithm 1) that performs a finer search over the spatio-temporal space to find the most likely trajectories of moving objects with a maximum of posterior probability. We detail the moving object hypotheses generation process in the following.

Firstly, as presented in [9], we incrementally constructs an online occupancy grid representing a local map of the vehicle environment based on good vehicle localization obtained by a fast scan matching technique. Moving parts of dynamic objects are then detected when objects enter object-free regions. In this paper, the detection condition is loosened to include the situations when objects are observed in unexplored regions that could be static or new moving objects. Taking all these as dynamic evidences increases the detection rate and false alarms as well. Result of this step is a set of dynamic segments corresponding to potential moving objects. Note that objects can be divided into several parts so that several segments might be related to the same object.

Fig. 6 illustrates our detection process. In the figure, the bottom image describes a situation when the host vehicle moving along the street seeing two cars moving ahead, another car coming out of the left turn and two pedestrians walking on the left pavement. The image on top left shows the local grid map constructed around the host vehicle (blue box). In red color is the current laser scan. Laser impacts that fall into free or unexplored regions are detected as dynamic measurements and are displayed in the top right image. Dynamic measurements are then grouped into segments represented in green boxes corresponding to moving objects. Note that the car coming from the left turn is divided into two segments.Two false alarms are also displayed.

Secondly, starting from identified dynamic segments, we generate hypotheses by fitting predefined object models to each segment. The objective is to generate all possible hypotheses corresponding to potential moving objects. The model fitting is carried out as follows. For each segment, a minimum bounding box is computed and corresponding sides of the segment are extracted. We remark that at one time instant, maximum two sides of a segment can be seen by the laser scanner. Providing that the size of a segment bounding box is larger than a threshold, the segment is classified as a L-shape if it has two visible sides, as an I-shape if only one side is visible. Otherwise it is classified as a ”mass point”-shape. Depending on the shape and size of segments, object hypotheses are generated using suitable models. L-shape segments will generate bus, car hypotheses, I-shape segments create bus, car, bike hypotheses and ”mass-point” segments will generate pedestrian hypotheses.

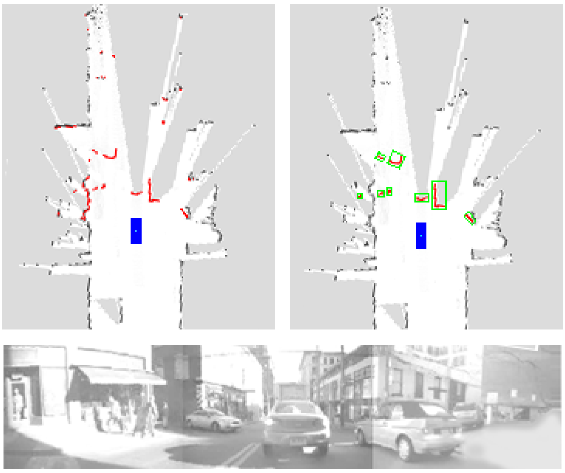

Fig. 6. Object detection based on occupancy grid.

Fig. 7 shows possible hypotheses of an object as a box model given L-shape and I-shape segments of different sizes. Note that by fitting object models to segments in this way, models can cover segments nearby so that naturally overcome object splitting problem caused by laser measurement discontinuities.

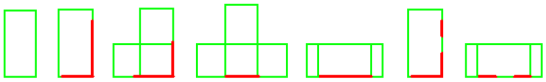

Fig. 7. Illustration of fitting object box model (green) to L-shape and I-shape segments (red). The last two shapes show that using box model helps connecting discontinued segments.

In subsequent sections we discuss how using the MCMC sampler to search for the solution from the space of moving object hypotheses. To make the sampling easier, we introduce a neighborhood graph structure as described next.

B. *Neighborhood graph of hypotheses*

We use a graph representation $\langle V,E\rangle$ of all moving object hypotheses within the time interval $[1,T]$. Let h_t^i denote the i-th hypothesis generated at time t. Each hypothesis h_t^i is represented by a node in V. We define the neighborhood between two nodes in the graph by edges of two types: sibling edge and parent-child edge. Sibling edges are defined by: $E_{sb} = \{(h^i_t, h^j_t)\}$ with the condition that h^i_t and h^j_t have spatial overlap. Parent-child edges are defined by: $E_{pc} = \{(h^i_{t_1}, h^j_{t_2})\}$ with the condition that $h^i_{t_1}$ and $h^j_{t_2}$ are of the same object class and $\|h^i_{t_1}(x,y) - h^j_{t_2}(x,y)\| < |t_1 - t_2| v_{max}$, where $\|.\|$ is the Euclidean distance, $1 \le |t_1 - t_2| \le t_{max}$ and v_{max} is the maximum speed of the object class. Sibling edges represent exclusion relationship between hypotheses that are generated from the same moving evidence so that if one is selected then the other are excluded. Parent-child edges reflect possible temporal association between hypotheses.

We use $N(.)$ to denote the neighbor set of a hypothesis in the graph structure, i.e. $N(h^i_{t_1}) = \{h^j_{t_2} | (h^i_{t_1}, h^j_{t_2}) \in E\}$. Hypothesis $h^j_{t_2} \in N(h^i_{t_1})$ belongs to the parent set $N^p(h^i_{t_1})$, child set $N^c(h^i_{t_1})$, sibling set $N^s(h^i_{t_1})$, when $t_2 < t_1$, $t_2 > t_1$ and $t_2 = t_1$ respectively. Using the neighborhood graph helps the MCMC sampler to traverse the solution space effectively.

C. Markov chain dynamics

The Markov chain dynamics correspond to sampling the proposal distribution $q(\omega'|\omega_{n-1})$ described in Algorithm 1 (line 3). We assume that in the $(n-1)$-th iteration we have a sample $\omega_{n-1} = \{\tau_1, ..., \tau_K\}$ and now propose a candidate ω' for the n-th iteration. Let V^* denote the set of all unselected hypotheses and do not share any sibling edge with nodes in ω_{n-1}. In order for the Markov chain to traverse the solution space, we design the following reversible dynamics.

Track Extension/Reduction: The purpose of the extension/reduction move is to extend or shorten the estimated trajectories. For a forward extension, we select uniformly at random (*u.a.r*) a track τ_k. Let $\tau_k(end)$ denote the last node in the track τ_k. Then select *u.a.r* node $h \in \{N^c(\tau_k(end)) \cap V^*\}$ and append the new hypothesis h to τ_k. Similarly, for a backward extension, we take a node $h \in \{N^p(\tau_k(start)) \cap V^*\}$. We keep on extending the track τ_k with a probability $\gamma \in [0,1]$.

The reduction move consists of randomly shortening a track τ_k by selecting a cutting index r *u.a.r* from $\{2, ..., |\tau_k| - 1\}$. In the case of a forward reduction the track τ_k is shortened to $\{\tau_k(t_1), ..., \tau_k(t_r)\}$, while in a backward reduction we take the sub-track $\{\tau_k(t_r), ..., \tau_k(t_{|\tau_k|})\}$.

Track Birth/Death: This move controls the creation of new track or termination of an existing trajectory. In a birth move, we select *u.a.r* a node $h \in V^*$, associate it to a new track and increase the number of tracks $K' = K + 1$. The birth move is always followed by an extension move. From the node h we select the extension direction forward or backward *u.a.r* to extend the track $\tau_{K'}$. If $|\tau_{K'}| < 2$ the move is rejected.

For a death move, we simply choose *u.a.r* a track τ_k and delete it.

Track Split/Merge: For a split move, we *u.a.r* select a track τ_k with $|\tau_k| \ge 4$ and a split point $s \in \{2, ..., |\tau_k| - 2\}$. And we split τ_k into two new tracks $\tau_{s_1} = \{\tau_k(t_1), ..., \tau_k(t_s)\}$ and $\tau_{s_2} = \{\tau_k(t_{s+1}), ..., \tau_k(t_{|\tau_k|})\}$ and increase the number of tracks $K' = K + 1$.

Often, due to missing detection or erroneous detection, trajectories of objects are fragmented. The merge move provides the ability to link these fragmented sub-tracks. If a tracks's (τ_{k_1}) end node is in the parent set of another track's (τ_{k_2}) start node, this pair of two tracks is candidate for a merge move. We select *u.a.r* a pair of tracks from candidates and merge the two tracks into a new track $\tau_k = \{\tau_{k_1}\} \cup \{\tau_{k_2}\}$ and reduce the number of tracks $K' = K - 1$.

Track Switch: If there exist two break points p, q in two tracks τ_{k_1}, τ_{k_2} such that $\tau_{k_1}(t_p) \in N^p(\tau_{k_2}(t_{q+1}))$ and $\tau_{k_2}(t_q) \in N^p(\tau_{k_1}(t_{p+1}))$ as well, this pair of nodes is one candidate for a switch move. We *u.a.r* select a candidate and define two new tracks as:

$\tau'_{k_1} = \{\tau_{k_1}(t_1), ..., \tau_{k_1}(t_p), \tau_{k_2}(t_{q+1}), ..., \tau_{k_2}(t_{|t_{k_2}|})\}$ and
$\tau'_{k_2} = \{\tau_{k_2}(t_1), ..., \tau_{k_2}(t_q), \tau_{k_1}(t_{p+1}), ..., \tau_{k_1}(t_{|t_{k_1}|})\}$.

Track Diffusion: Randomly select a track τ_k and an index d from $\{1, ..., |\tau_k|\}$ and update the position and orientation of the object $\tau_k(t_d)$ under some random noise.

Four first types of moves are temporal dynamics and the last one is a spatial dynamics. The temporal moves help to form tracks (data association) and the spatial move helps to improve the detection results. In each iteration, one of the above dynamics is chosen randomly. It is guaranteed that the Markov chain designed this way is *ergodic* (*i.e.*, any state is reachable from any other state within finite number of iterations) and *aperiodic* (*i.e.*, the Markov chain does not oscillate in a fixed pattern) since all of moves are stochastic.

D. Incremental computation

For each MCMC move, we need to compute the ratio $\frac{\pi(\omega')}{\pi(\omega)} = \frac{P(\omega'|Z)}{P(\omega|Z)}$ in (18). In one iteration, our algorithm only changes maximum two tracks. Thus the new posterior probability can be computed more efficiently by incrementally computing it only within the related terms in (17). This is in contrast to the particle filters where the evaluation of each particle (joint state) needs the computation of the full joint likelihood. One more interesting property of the MCMC approach is that, it only needs to keep one hypothesis in the memory at one time instant compared with all hypotheses have to be maintained in case of tracking with MHT. Moreover, the execution time can be controlled by the number of sampling iterations n_{mc}.

V. EXPERIMENTAL RESULTS

We test our approach described above using the Navlab datasets [3]. The datasets were collected using a SICK laser scanner mounted on a moving vehicle. The vehicle was driven in urban streets with real-life traffics. The data was collected at 37.5 Hz with 0.5 degree resolution. The maximum measurement range of the scanner is 80 m.

We have implemented the described algorithm as an online process within a sliding window which contains $T = 10$ frames and only moving object hypotheses generated within this sliding window are stored in the graph structure. Actually, our tracking process is performed for every four "original" scans so that the data cycle time from frame to frame in our program is about 100 ms. For each time step when new

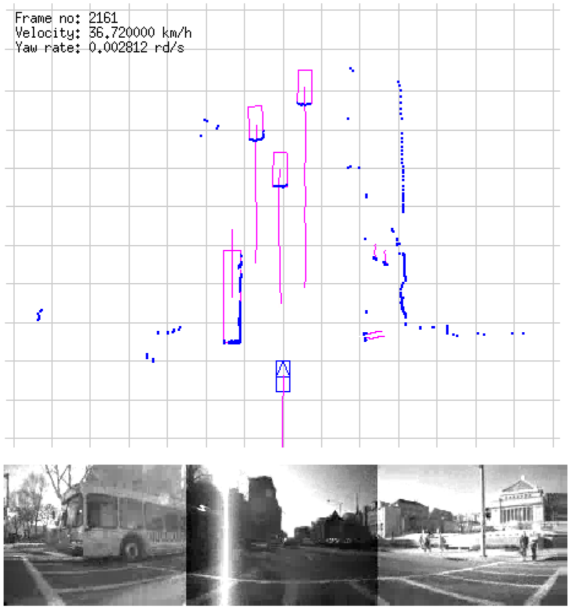

Fig. 8. Moving object detection and tracking in action.

measurements arrive, the solution obtained in the previous step is used to initialize a new MCMC search. Tracks are confirmed if their length are greater than three, otherwise they are considered false alarms. The MCMC sampler is run for a total of 250 iterations. The average computational time for the total detection and tracking process is about 60 ms on P4 3.0 GHz PC with unoptimized codes so that it can fulfill the real time requirement.

Fig. 8 shows an example of our detection and tracking algorithm in action. In the ego-vehicle's view, the detected moving objects and their trajectories are shown in pink color with current laser scan is in blue color. Moving objects in the situation include a bus moving in the opposite direction on the left, three cars moving ahead, two pedestrians walking on the left pavement and the other two pedestrians passing the intersection. The bus is divided into several segments and pedestrians are easy to be confused with noises and other small objects that make the detection and tracking challenging. However, thanks to our approach, the bus model helps connecting the discontinued segments and temporal information helps to distinguish pedestrians with noises and reduce ambiguities. All moving objects are identified and tracked successfully.

With initial evaluations, the MCMC detection and tracking outperforms the detection and tracking using MHT in our previous work [10] in terms of a higher detection rate and less false alarms. Particularly pedestrians are tracked successfully. In addition, with the use of object models, segmented objects caused by laser discontinuities are no longer a problem and tracking results are more accurate. Note that in our model-based approach moving objects are naturally classified. The readers can refer to: http://emotion.inrialpes.fr/˜tdvu/videos/ for more resulting videos.

VI. CONCLUSIONS AND FUTURE WORKS

We have presented a general framework for simultaneous detection, classification and tracking moving objects in real time using a laser scanner. The success of our approach mainly lies in the combination of a top-down analysis process and a bottom-up detection method that takes advantages of the computational efficiency of the detection and retains the optimality and robustness of the Bayesian formation from the global view. A model-based approach is introduced to help overcoming existing problems of detection and tracking using laser scanners. Experiments and initial evaluations on challenging real-life data show promising results.

Future works include a quantitative evaluation of our algorithm. We intend to integrate a road detection procedure in order to provide prior information on moving objects that certainly improve the effectiveness of the detection and tracking process. An optimization of code is ongoing.

VII. ACKNOWLEDGMENTS

We thank Chieh-Chih Wang and members at Navlab group, the Robotics Institute, CMU, Pittsburgh, PA, U.S.A for providing the public dataset [3] which is used in this research for testing purpose. The work is supported by the European project PReVENT-ProFusion [1].

REFERENCES

[1] S. S. Blackman. Multiple hypothesis tracking for multiple target tracking. *Aerospace and Electronic Systems Magazine, IEEE*, 2004.

[2] M. Darms, P. Rybski, and C. Urmson. Classification and tracking of dynamic objects with multiple sensors for autonomous driving in urban environments. In *IV*, Eindhoven, The Netherlands, June 2008.

[3] C.C. Wang et al. Navlab slammot datasets. http://www.cs.cmu.edu/˜bobwang/datasets.html, May 2004. CMU.

[4] S. Oh, S. Russell, and S. Sastry. Markov chain monte carlo data association for general multiple-target tracking problems. In *IEEE Conference on Decision and Control*, 2004.

[5] A. Petrovskaya and S. Thrun. Model based vehicle tracking for autonomous driving in urban environments. In *Proceedings of Robotics: Science and Systems IV*, Zurich, Switzerland, June 2008.

[6] D. Schulz, W. Burgard, D. Fox, and A.B. Cremers. Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In *ICRA*, 2001.

[7] K. Smith, D. Gatica-Perez, and J.M. Odobez. Using particles to track varying numbers of objects. In *CVPR*, 2005.

[8] L. Tierney. Markov chain concepts related to sampling algorithms. *Markov Chain Monte Carlo in Practice*, pages 59–74, 1996.

[9] T.D. Vu, O. Aycard, and N. Appenrodt. Online localization and mapping with moving object tracking in dynamic outdoor environment. In *IEEE Intelligent Vehicle Symposium*, Istanbul, Turkey, June 2007.

[10] T.D Vu, J. Burlet, and O. Aycard. Grid-based localization and local mapping with moving objects detection and tracking. *Accepted for publication in the Elsevier Journal Information Fusion*, 2009.

[11] C.C. Wang. *Simultaneous Localization, Mapping and Moving Object Tracking*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, April 2004.

[12] Q. Yu, G. Medioni, and I. Cohen. Multiple target tracking using spatio-temporal markov chain monte carlo data association. In *CVPR*, 2007.

[13] L. Zhao and T. Chuck. Qualitative and quantitative car tracking from a range image sequence. In *CVPR*, 1998.

[14] S.C. Zhu, R. Zhang, and Z. Tu. Integrating Top-down/Bottom-up for object recognition by data driven markov chain monte carlo. In *CVPR*, 2000.

[1] http://www.prevent-ip.org/profusion