A Laser-Scanner-Based Approach Toward Driving Safety and Traffic Data Collection

Huijing Zhao, *Member, IEEE*, Masaki Chiba, Ryosuke Shibasaki, *Member, IEEE*, Xiaowei Shao, Jinshi Cui, *Member, IEEE*, and Hongbin Zha, *Member, IEEE*

***Abstract*—This work is motivated by the following two potential applications: 1) enhancing driving safety and 2) collecting traffic data in a large dynamic urban environment. A laser-scanner-based approach is proposed. The problem is formulated as a simultaneous localization and mapping (SLAM) with object tracking and classification, where the focus is on managing a mixture of data from both dynamic and static objects in a highly dynamic environment. A trajectory-oriented closure is also proposed using the sporadically available global positioning system (GPS) measurements in urban areas to assist for global accuracy, particularly when the vehicle makes a noncyclical measurement in a large outdoor environment. Experiments are conducted using the data that were collected along a course near 4.5 km in a highly dynamic environment. Possibilities of the approaches toward the two potential applications are demonstrated, and avenues for future works are discussed.**

***Index Terms*—Detection, intelligent vehicle, laser scanner, moving object, SLAM, tracking.**

I. Introduction

OUR goal is to use a vehicle-borne sensor to perceive a large dynamic urban environment, such as an intersection or a crowded road in a downtown area. We are motivated by two potential applications. One is enhancing driving safety, where it is important to understand the state of both the host vehicle itself and the objects in its local surroundings. The other application involves collecting detailed traffic data, such as the motion trajectories of cars, bicycles, and pedestrians for control and traffic analysis. In this latter application, it is important to associate the perceptions of local surroundings with a global coordinate system, and the traffic data are required to achieve a certain level of global accuracy. In other words, if a perception of local surroundings could be registered to a global coordinate system, other data sources, such as a computer-aided design (CAD) map, could be used to assign higher level attributes to the perceived data. Both tasks could be assembled together as a perception in a large dynamic environment with both local and global accuracy. Here, managing the data of a large dynamic environment and achieving both local and global accuracy are major concerns.

To assist cars for driving safety, research effort has shown the possibility of detecting and tracking objects in front of the car using a stereo [1], [2] or monocular video camera [3], a laser scanner [4]–[6], or through sensor fusion [7], [8]. This is reasonable and efficient when a car drives on a straight path. However, when facing a complicated environment such as crowded roads and intersections in a downtown area, where the host vehicle might be close to other moving objects and where a continuous and reliable global positioning system (GPS) measurement might not be available, a highly accurate perception to the state of both the host vehicle and the objects nearby is required for an efficient warning system. Vu *et al.* [9] and Weiss *et al.* [10] performed online calculation of an occupancy map to detect objects that entered an object-free zone. This idea can be traced to the pioneering work by Wang [11].

There has been further research effort to collect traffic data using probe vehicles. Most of it uses GPS to find the speed and trajectory of the probe vehicle and assumes that these parameters somehow reflect the current traffic conditions of the road. Some probe vehicles have environmental sensors to monitor the surroundings, such as video cameras, laser scanners, and radars. Subsequent data processing is still a great difficulty. Gandhi and Trivedi [12] developed a system platform to detect, classify, and log the surrounding vehicles using a video camera. Gao and Coifman [13] proposed a method using a laser scanner to identify surrounding vehicles and correct GPS errors.

Perceiving a large dynamic environment while achieving both local and global accuracy is a particularly challenging problem. The following three approaches are suggested: 1) We can use positioning sensors like GPS/inertial navigation system (INS) to estimate vehicle pose, and with these as input, sensors like laser, radar, and camera conduct environmental perception. This is the most popular one and has been widely accepted in existing intelligent vehicle (e.g., [14] and [15]) and mobile mapping systems (e.g., [16]). As localization and environmental perception are conducted individually using different sensing technologies, the system architecture is straightforward. However, a disadvantage of such approaches is that environmental perception is heavily dependent on the output of localization module. For example, erroneous localization output might yield displacements between the environmental

Manuscript received August 31, 2008; revised January 19, 2009 and April 20, 2009. First published July 14, 2009; current version published September 1, 2009. This work was supported in part by the Japan Society for the Promotion of Science, by the Grants-in-Aid for Scientific Research under Contract 19300067, and by the 985 II Project of Peking University. The Associate Editor for this paper were B. De Schutter and S. Shladover.

H. Zhao, J. Cui, and H. Zha are with the Key Laboratory of Machine Perception (MOE), and the School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: zhaohj@cis.pku.edu.cn).

M. Chiba is with the Technical Research Center, Mazda Motor Corporation, Yokohama 221-0022, Japan.

R. Shibasaki and X. Shao are with the Center for Spatial Information Science, University of Tokyo, Chiba 277-8568, Japan.

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

Digital Object Identifier 10.1109/TITS.2009.2026450

sensing data to the same static objects. Moreover, even with an expensive GPS/inertial-measurement-unit-based navigation unit, the motion of slow objects, such as pedestrians, could not be reliably detected due to localization error. 2) If we suppose a map of the environment is available, for example, through previous exploration, then the map could be used to improve the accuracy and efficiency of localization by map matching [17]. Data on moving objects could then be obtained by subtraction from the map [18], [19] or by region masking [20]. However, this relies on the fact that a map of the dynamic environment with high-enough accuracy has already been generated. 3) We can combine the inputs from both positioning and environmental sensors and formulate the localization, mapping, moving object detection, and tracking simultaneously as an optimization problem [9], [11], [21]. This is the most difficult but necessary approach when exploring an unknown environment without an expensive positioning system or prior knowledge.

In this paper, we adopt the third approach by integrating a one-layer horizontally profiling laser scanner with GPS and yaw rate sensors. As we consider the applications in a downtown area, where GPS signals might be blocked by buildings, trees, bridges, etc., and where its accuracy might be degraded due to multipath problem, it is not reasonable to assume that a continuous and reliable GPS measurement be always available. On the other hand, a laser-based simultaneous localization and mapping (SLAM) can provide much continuous vehicle pose estimation while achieving more consistency among the environmental measurements, i.e., high local accuracy. However, it suffers from an error-accumulation problem, i.e., low global accuracy. Based on the above consideration, we use SLAM to obtain a continuous vehicle pose estimation and use GPS/INS to assist for global accuracy and robustness.

We propose a laser-based SLAM in a dynamic environment. Contrasting with other pioneering works such as those of Hahnel *et al.* [21], Wang [11], Vu *et al.* [9], and Weiss *et al.* [10], we formulate the problem as a SLAM with object tracking and classification, where the focus is on managing a mixture of data from both dynamic and static objects in a highly cluttered environment. For example, people and cars might get very close to each other, and their motion patterns have much variability and are always unpredictable. Thus, it is risky to discriminate moving or static objects by buffering an area using the data from a previous measurement. Furthermore, it is risky to judge based only on an instance measurement, as many objects might have a similar data appearance due to limited spatial resolution, range error, partial observation, and occlusion. The general idea behind our system is that the detected objects should be discriminated in a spatial–temporal domain. This way, after an object is detected, it is tracked until the system can classify the object into either a static or a moving object with certainty.

We also propose a trajectory-oriented closure algorithm using GPS signals. To achieve a localization of global accuracy, particularly when the vehicle makes a noncyclical measurement in a large outdoor environment, the sporadically available GPS measurements in urban areas are used to diagnose errors in vehicle pose estimation, and vehicle trajectory is then adjusted to close the gap between the estimated vehicle pose and the GPS measurement.

Here, we need to explicitly state that the algorithms and experimental results presented in this paper are 2-D ones, with an assumption that the ground surface is almost flat. In the future, we will extend them to 3-D cases by using a multilaser scanner or a composition of laser scanners. This paper is organized as follows. Section II describes the framework of SLAM with simultaneous detection and tracking of moving objects using a laser scanner. The localization and trajectory-oriented closure algorithm using GPS measurements are addressed in Section III. We present experimental results and discuss some major topics for future work in Section IV, followed by conclusions in Section V.

II. SLAM With Object Tracking and Classification

A. Problem Statement

1) SLAM in General: The SLAM problem has been widely studied for decades. In addition to the problem of SLAM been theoretical formulated, much research effort has demonstrated its implementation in a number of different domains, such as indoor, outdoor, underwater, and airborne systems. A good tutorial to SLAM was given by Durrant-Whyte and Bailey [22], [23], which could be a good start to learn SLAM from its history, achievements, key problems, and future. A broad survey can also be found in [24].

The problem of SLAM can be formulated as the following probabilistic form (see [22] for details):

$$p(x_k, m|z_{0:k}, u_{0:k}) \tag{1}$$

where, given a sequence of observation $z_{0:k}$ and a sequence of control inputs $u_{0:k}$, the objective is to generate a map (m) of the surrounding environment and simultaneously locate the vehicle's pose x_k at m.

The SLAM problem could be parsed as follows using Bayes' rule, with the assumption that the system is a Markov:

$$\propto p(z_k|x_k, m) \times \int p(x_k|x_{k-1}, u_k) \cdot p(x_{k-1}, m|z_{0:k-1}, u_{0:k-1}) dx_{k-1}. \tag{2}$$

Here, $p(x_k|x_{k-1}, u_k)$ is the vehicle's motion model and describes the probability for a state transition. $p(z_k|x_k, m)$ is the observation model (also called likelihood function) and describes the probability of making an observation z_k when a vehicle's pose x_k and a map m of the environment is known.

The SLAM problem could be solved practically as incrementally looking for a vehicle pose of the maximal probability [21], i.e.,

$$x_k = \arg\max_{x_k^-} \left\{ p\left(z_k|x_k^-, \hat{m}(x_{1:k-1}, z_{1:k-1})\right) \times p\left(x_k^-|x_{k-1}, u_k\right)\right\} \tag{3}$$

while map $\hat{m}(x_{1:k}, z_{1:k})$ is considered as an integration of observations $z_{1:k}$ along vehicle poses $x_{1:k}$. For discrimination, x_k^- denotes a pose prediction based on the vehicle's motion model, and x_k denotes the posterior of pose at time k.

Map consistency could be achieved through the formulation above as an observation model confines a match between the observation and a map. A limitation exists in that there is no guarantee of global (or absolute) accuracy in recovered vehicle pose or in the map. Distortion could occur due to the featureless environment, error accumulation, and so on. This is crucial when fusing the result with a CAD map or other data resources.

2) Loop Closure: When a vehicle returns to a previously mapped region, a problem occurs in which the newly estimated location of landmarks does not match with previous ones. Loop closure is used to associate the landmarks in a current measurement with those in a map database to correct the vehicle pose and, subsequently, the map. When facing a large pose error (after a large loop or in a cluttered unknown environment), data association becomes much more difficult. An incorrect data association could cause a catastrophic failure of SLAM. Loop closure is also a data-association problem. Much research effort has focused on improving the accuracy of data association, including batch gating [25] and visual appearance [26], [27], or reducing the risk in erroneous associations, such as multiple hypothesis [21].

However, when applying SLAM in a large outdoor environment, the vehicle may traverse complicated road situations. Requirements of cyclic measurements and limited loop size are strong restrictions to real applications. In addition to data-association-based solutions, the vehicle needs a different means to diagnose its pose error and to guarantee an error bound, even though its trajectory does not cross after a long trip.

3) Dynamic Environment: Most of the existing SLAM methods assume that the environment is static. If there is a moving object, and if the data are erroneously associated with a landmark in the map database, many localization algorithms will fail, and the map will be deteriorated by the data of the moving object. If we can discriminate the data of a moving object from those of static ones, the problem could be solved by sending only static object data to SLAM. However, data discrimination is the key and, in fact, is the greatest obstacle for applying SLAM to a dynamic environment.

Without assuming any prior knowledge of the environment, and given that the environment cannot be intentionally controlled (e.g., a downtown area where people and cars always exist), a routine is required to discriminate the data from moving and static objects before sending them to SLAM or moving object's tracking modules. Hahnel *et al.* [21] filtered out moving people by using the local minimum caused by legs and subsequently created a difference map between consecutive scans to remove those static but people-like objects. An implicit assumption here is that dynamic objects move all the time during their measurement. However, this is not reasonable in normal situations because people and cars may stop for a while. Furthermore, a classification based only on data appearance is risky since a person standing still will look similar to a pole in a horizontal laser scan. In the pioneering work of Wang [11] and recent research applications [9], [10], moving objects are detected generally based on the following two rules: 1) If an object entered the object-free zone defined by previous measurements, it is a moving object, and 2) if an object entered the zone previously occupied by moving objects, it is a moving object. Such an approach relies on the following: 1) An object-free zone is reliably defined, and 2) moving objects are reliably extracted. However, these are difficult to be met in some cases. For example, if a laser shoot did not have return, either of the following might be true: 1) There is no object along the beam up to range limit, or 2) there is an object but do not give reflection toward the sensor, e.g., dark objects, mirror reflection, shallow incidence angle, etc. The no-return-beams bring much ambiguity in generating a reliable object-free zone. In addition, as the robot explores an unknown environment, many new objects might be measured in previously undeveloped or occupied zones. It is difficult to reliably decide at the moment whether they are moving or static ones.

The general idea behind our system is that a classification routine in a spatial–temporal domain is required. This way, whenever an object is discovered in a newly explored zone, it is tracked (called "seed" in this paper) until a judge could be given with a certainty (upgraded to either "static object" or "moving object"). Therefore, we formulate this task into a problem of SLAM with object tracking and classification.

B. Proposed Approach

The problem of SLAM in a dynamic environment can be formulated as follows:

$$p(x_k, y_k, s_k, m | z_{0:k}, u_{0:k}) \tag{4}$$

where y denotes the moving object, and s is the object of unknown class (here, we call it the "seed").

It can further be parsed as follows based on the Bayes' rule, with the assumption that the system is Markov:

$$\begin{aligned} &= p(s_k | x_k, m, z_{0:k}) && // \text{ detection and tracking} \\ &\cdot p(y_k | x_k, z_{0:k}) && // \text{ tracking problem} \\ &\cdot p(x_k, m | z, u_{0:k}) && // \text{ a standard SLAM} \end{aligned} \tag{5}$$

where the last item is a standard SLAM, and the first two are the detection and tracking problems. However, to solve the posteriors, two implicit problems, i.e., 1) classification of the measurement data and 2) classification of the detected objects, have to be solved.

A measurement z_k is a mixture of the data from static, moving, seed, and newly detected objects. If the mixture can be classified as follows:

$$z_k = \left\{ z_k^{(m)}, z_k^{(y)}, z_k^{(s)}, z_k^{(n)} \right\} \tag{6}$$

an estimate of (5) can sequentially be achieved as follows:

$$x_k = \arg\max_{x_k^-} \left\{ p\left(z_k^{(m)} | x_k^-, m_{k-1}\right) \cdot p\left(x_k^- | x_{k-1}, u_k\right) \right\} \tag{7}$$

$$m_k = \hat{m}\left(x_{1:k}, z_{1:k}^{(m)}\right) \tag{8}$$

$$y_k = \arg\max_{y_k^-} \left\{ p\left(z_k^{(y)} | x_k, y_k^-\right) \cdot p\left(y_k^- | y_{k-1}\right) \right\} \tag{9}$$

$$s_k^- = \arg\max_{s_k^-} \left\{ p\left(z_k^{(s)} | x_k, s_k^-\right) \cdot p\left(s_k^- | s_{s-1}\right) \right\}. \tag{10}$$

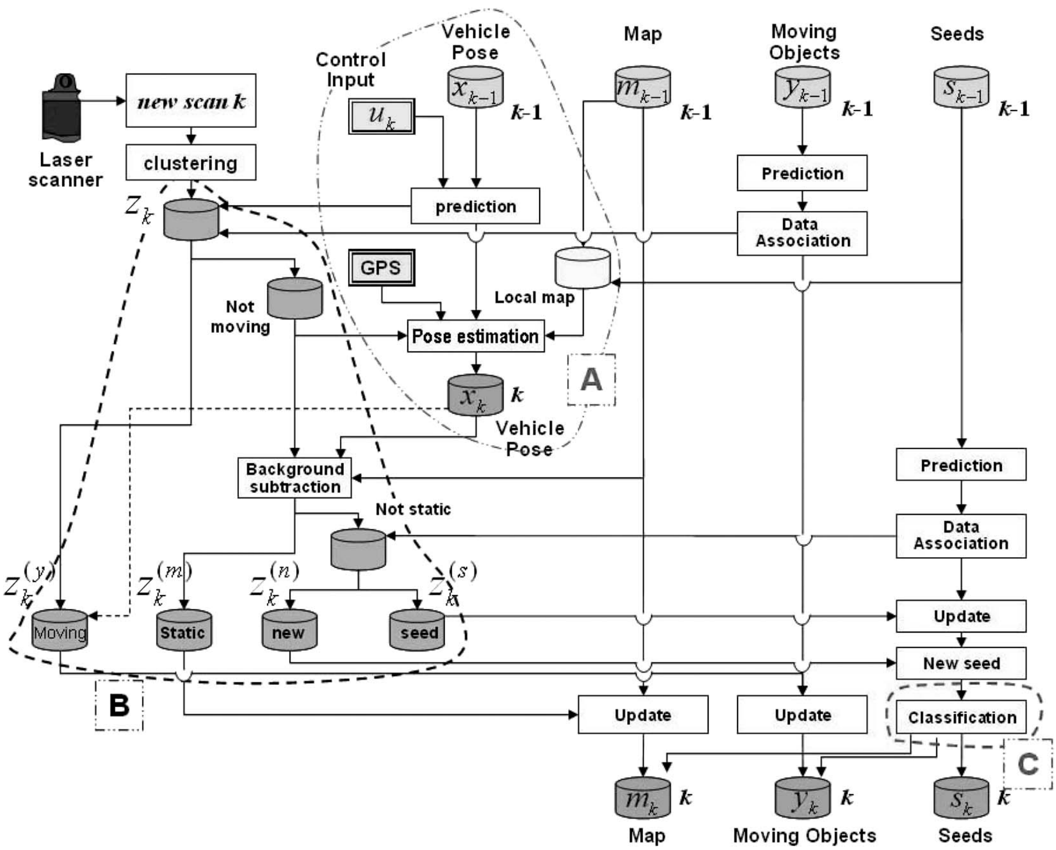


Fig. 1. Implementation of SLAM with object tracking and classification.

For the measurements that could not be associated with an existing map, any of the moving objects or seeds, new seeds are generated for each as the newly detected objects so that

$$s_k^+ = newseed\left(z_k - z_k^{(m)} - z_k^{(y)} - z_k^{(s)}\right) \tag{11}$$

$$s_k = s_k^+ + s_k^-. \tag{12}$$

Correctly classifying z_k into $z_k^{(m)}$, $z_k^{(y)}$, $z_k^{(s)}$, and $z_k^{(n)}$ is critical in the estimation of x_k, m_k, y_k, and s_k. This task is quite difficult because of the limited information in an instance measurement. All the newly detected objects (called "seed" here) are tracked until they can be classified into either static or moving object with certainty. Tracking all the detected objects is time consuming, while an incorrect classification might lead to a system failure. An efficient and accurate classifier is critical to the performance of the whole system. In this paper, we solve both classification problems in systematic ways.

C. Implementation

Fig. 1 shows the framework implemented in our system. The figure looks trivial. However, we believe that technical details are always very important to ensure that a new technique can be applied to real situations. The system uses the following three kinds of sensor inputs: 1) laser scan; 2) GPS; and 3) control inputs from both the inertial sensor and wheel encoder. The system maintains the following four databases: 1) vehicle pose; 2) map; 3) moving objects; and 4) seeds. At each iteration, given a set of current sensor inputs, the purpose is to update the state of each database from the previous time stamp $(k-1)$ to the current time (k). Iteration continues until the measurement completes.

Three modules are highlighted in Fig. 1. As the focus of this paper is not the algorithm of tracking, data association, or clustering, where we implemented popular methods such as the Kalman filter [28] and the nearest-neighbor method, we do not discuss their details.

Module A is the estimation of vehicle pose x_k through scan matching assisted by GPS and control inputs. We apply a grid-based method (e.g., [11] and [29]) for scan matching. A pyramid of grid map is used to represent the state of stationary objects. The value of each grid can be 1 or 0, denoting whether the grid has been occupied or not. Grid sizes are set from large to small in different layers so that a coarse-to-fine matching can be achieved with a limited time cost. Vehicle pose is estimated by projecting laser scan onto the grid maps and searching for a pose of best correlation. Localization guided by GPS and control inputs will be addressed in detail Section III. Modules B and C correspond to the classification of the measurement data and seed objects, respectively.

1) Classification of a Measurement Data: Classification of z_k into $z_k^{(m)}$, $z_k^{(y)}$, $z_k^{(s)}$, and $z_k^{(n)}$ is conducted in a sequential procedure as described in module B. To prevent interference from moving object data in SLAM, $z_k^{(y)}$ are first extracted through data association based on the prediction of existing moving objects y_{k-1}^- and vehicle pose x_{k-1}^-. SLAM is conducted by matching the measurement to motionless objects $\{z_k - z_k^{(y)}\}$ with a local map covering the region near x_{k-1}^-,

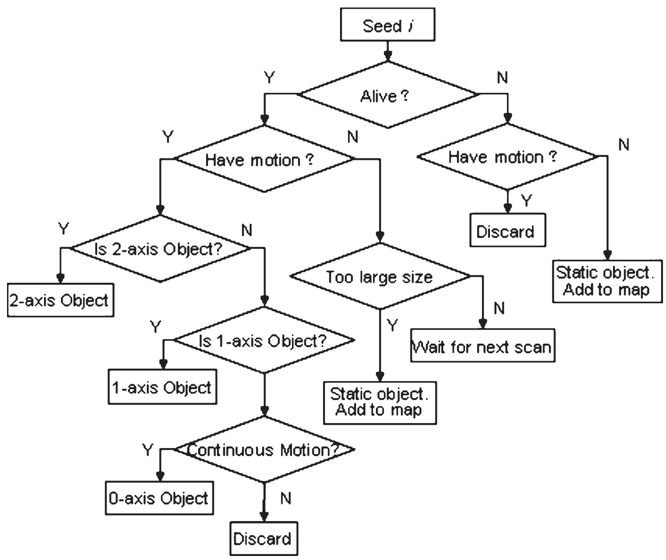


Fig. 2. Implementation of seed object classification.

which is a cutout from $\hat{m}(x_{1:k-1}, z_{1:k-1}^{(m)+(s)})$ containing the data of both static and unknown objects (seeds). Estimation of vehicle pose x_k is implemented as follows:

$$x_k = \arg\max_{x_k^-} \left\{ p\left(z_k - z_k^{(y)} | x_k^-, \hat{m}\left(x_{1:k-1}, z_{1:k-1}^{(m)+(s)}\right)\right) \cdot p\left(x_k^- | x_{k-1}, u_k\right) \right\} \quad (13)$$

where vehicle pose x_k, through subtraction with map m_{k-1}, and $z_k^{(m)}$ are extracted, leaving seeds $z_k^{(s)}$ and the newly measured objects $z_k^{(n)}$ at rest.

2) Classification of a Seed Object: A seed can be either a static or a moving object. Here, we consider the following moving objects in a normal urban scene: a person, a bicycle, and a car. Each moving object might remain static temporarily, e.g., while waiting for a traffic signal. To discriminate such an object from a permanently static object, we define an object to be static if it has an exactly different data appearance (e.g., the size is much bigger) than that of a moving one, and no motion is detected along its trajectory.

Fig. 2 shows our implementation flow. In classifying a seed, it is first examined on features such as motion vector and object size to discriminate it as a static, moving, or still unknown (seed) object. Classification of moving objects is an extension to our previous work that uses a stationary laser scanner to monitor moving objects at an intersection [30]. We consider the moving objects of people, bicycles, and cars in a normal urban scene, and the moving objects are characterized as zero-, one-, and two-axis objects, respectively, according to the maximum number of axes that could be detected from an instantaneous measurement of the object.

To prevent a temporarily motionless object from being misrecognized as static, we carefully consider the case when the seed is still alive (i.e., being measured). If an obvious and continuous motion is detected, the still-alive seed is upgraded to the database of moving objects. If its shape is obviously different from predefined models, e.g., much larger than any possible moving object, it is added to the map if no motion is detected in its history, or it is discarded as irregular data, e.g., reflections from the ground, which always occur when the vehicle makes a turn and its platform slants downward. Many seeds may expire before being recognized as moving or static objects, e.g., by exiting the measurement range of the moving laser scanner. Some of these are static objects that should be integrated into the map. Others are dynamic but without enough evidence to distinguish them. These data cannot be added to the map. Some are wrongly extracted seeds that should be discarded. As our approach do not rely on scan-based discrimination, the moving object is able to be detected as long as it moves during any period of the measurement. However, we agree that if a moving object remains static during the measurements, and if its horizontal contour is similar with the static object like tree, pole, or small square object, it is difficult to detect based on laser scan only. In this paper, a tradeoff is taken that we add these seeds to the map. In the future, we will integrate visual-based methods to solve the aforementioned problem.

D. Experimental Data

Let us take a closer look at the algorithm in operation on real data. The experimental data were collected using the LD1 of testbed 1 (see Fig. 7) in a flat environment where many buildings, trees, poles, and parked cars exist. There is only one moving object, namely, a car driving in front of the host vehicle, throughout the experiment. A movie of the result can be found at the attachment, and Fig. 3 presents snapshots from the video. Each is composed of two screen captures. The image on the right shows the processing of laser data, and the one on the left is a back-projection of the laser-based result onto a video image for better understanding and visualization of the algorithm. Laser scans are processed at a rate of 10 Hz. The number shown at the bottom-right corner denotes the scan (Frm) number.

When measurement began (see Frm 090), the car in front of the testbed vehicle, as well as other nearby static objects, was measured. Two large pieces of wall were recognized quickly as static objects (green points are those in the current scan that have been recognized as static data) because no motion was detected from them, and their size exceeded the model of a normal moving object. Although no motion was detected from the objects, they may be either static or moving objects so they were treated as seeds (water blue). A key to the different colors can be found in Fig. 4.

When the front car began to move, after a few frames (see Frm 98), it was proven to be a moving object (red with orange trajectory). The front car began to make a left turn at about Frm 262 and was soon lost at Frm 289. When the front car reappeared in Frm 321, it was initially treated as a seed and then recognized as a moving object again after a few frames (see Frm 330). Later, the front car made another turn and was again lost from the video image, yet still measured and tracked by the laser scanner (see Frm 417). When the car reentered the video image, it was directly located as a moving object using the laser-based result (see Frm 426).

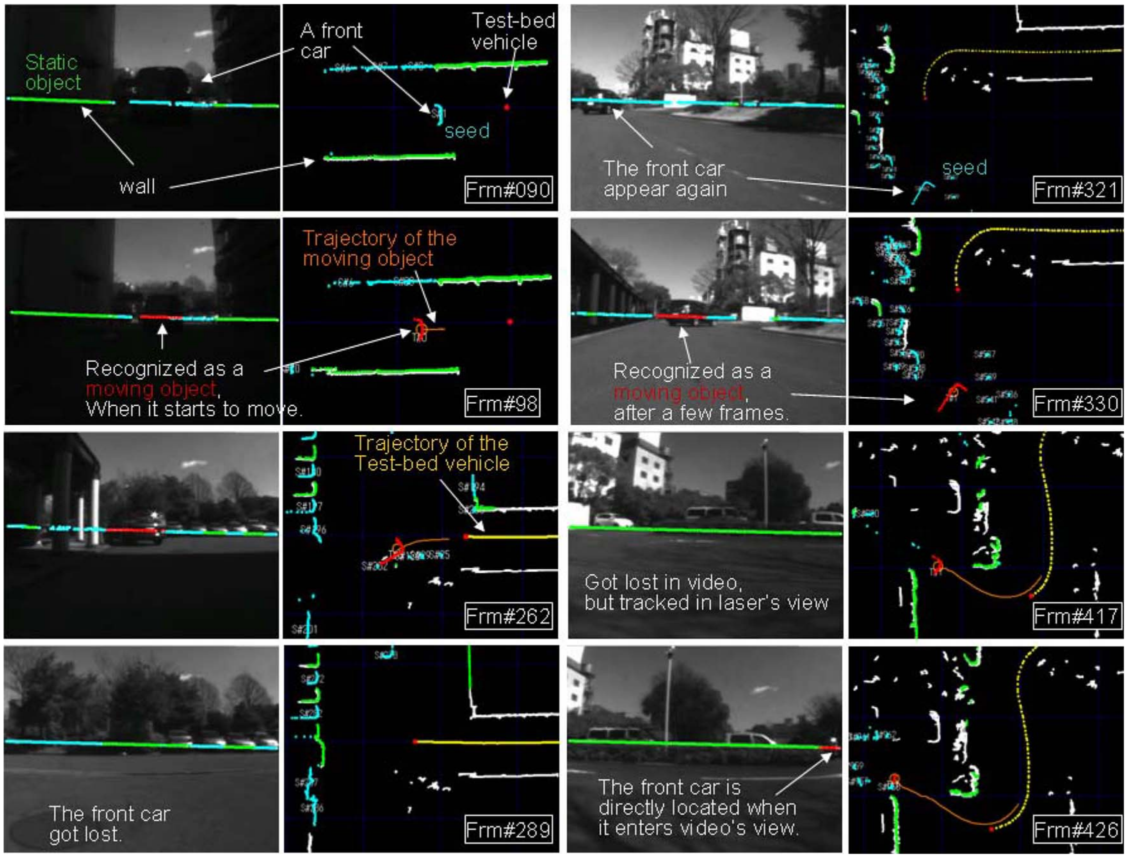


Fig. 3. Screen captures of the laser-based processing and its back-projection onto a video image to demonstrate the workings of the algorithm.

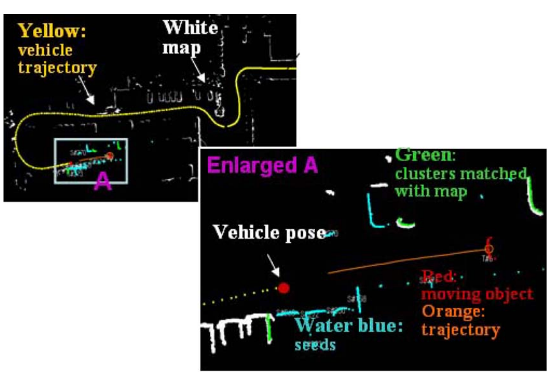


Fig. 4. Definition to the colors.

Fig. 5 presents some of the final results. To have a global view of the experimental site, Fig. 5(a) shows the map $(m(x_{1:k}, z_{1:k}^{(m)})$ and the trajectory $(\{x_{1:k}\})$ of the testbed vehicle that was recovered by the algorithm. The experiment was conducted with many cars parked in the surrounding area. Whether we should leave the data of parked cars in the map is a difficult question.

A parked car is a temporarily static object. Some of them may start to move suddenly, while most of them remain static during the period when they are measured. According to the tradeoff that we discussed previously, if motion could be reliably detected from an object at any time during the measurement, it was considered to be a moving one. Otherwise, we treated it as static and integrated its data into the map, which is why the parked cars are left in the final map.

Fig. 5(b) shows a classification result. Colored points represent the laser points labeled as moving objects. Color represents the ID no of the moving object. As discussed in Fig. 3, the front car was lost when it made turns. When it reappeared, it was detected as a new object and assigned a different ID no. Some false alarms also occurred. These are enlarged in Fig. 5(c). Most of the false alarms occurred only briefly, and their incidence can be reduced through parameter tuning in classification. However, a false alarm is shown in Fig. 5(d), which lasted for a long period of time (trajectory represents the length of the alarm). A major reason for a false alarm is occlusion, where a static object looks like a moving one due to partial observation.

III. Localization Guided By GPS Data and Control Inputs

A. Outline of the Localization Algorithm

The processing in Fig. 6 corresponds to module A in Fig. 1. It describes the flow of estimating vehicle pose x_k, where scan matching is used to achieve local consistency of the map, and control inputs u_k and GPS data are used to detect pose error and to achieve global accuracy. In this paper, control inputs are vehicle speed and yaw rate.

Normally, control inputs are used to predict a vehicle pose x_k^-, which is then updated by maximizing a matching between observation z_k and map m_{k-1}. This is a biased definition. Its efficiency relies heavily on whether the geometric relationship between z_k and m_{k-1} can be uniquely defined. Thus, an

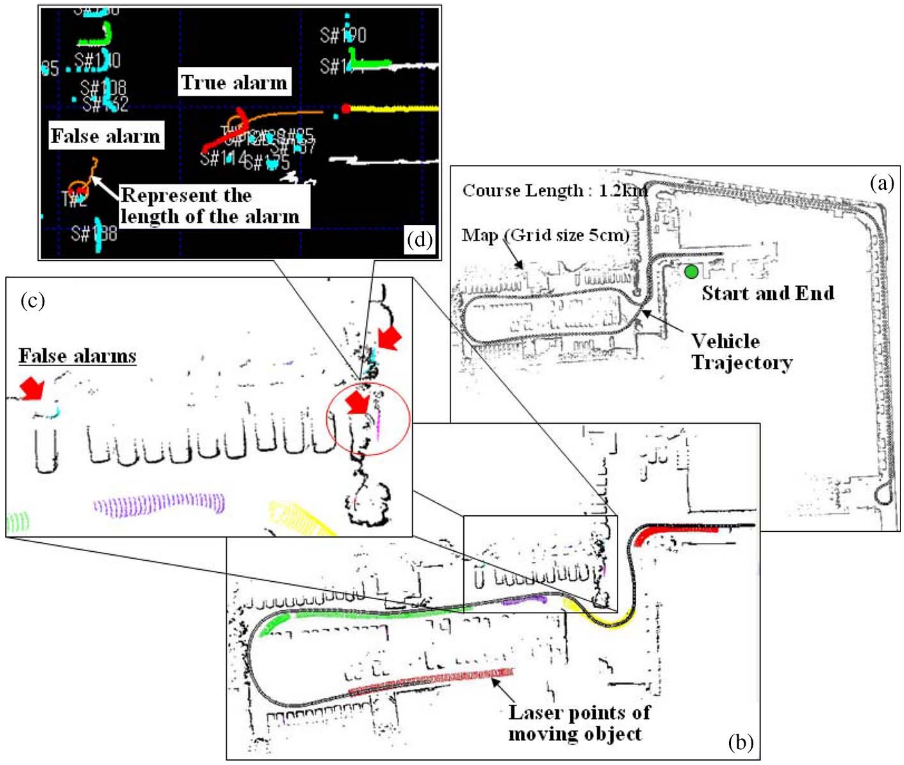


Fig. 5. Results of map, vehicle trajectory, moving objects, and false alarms.

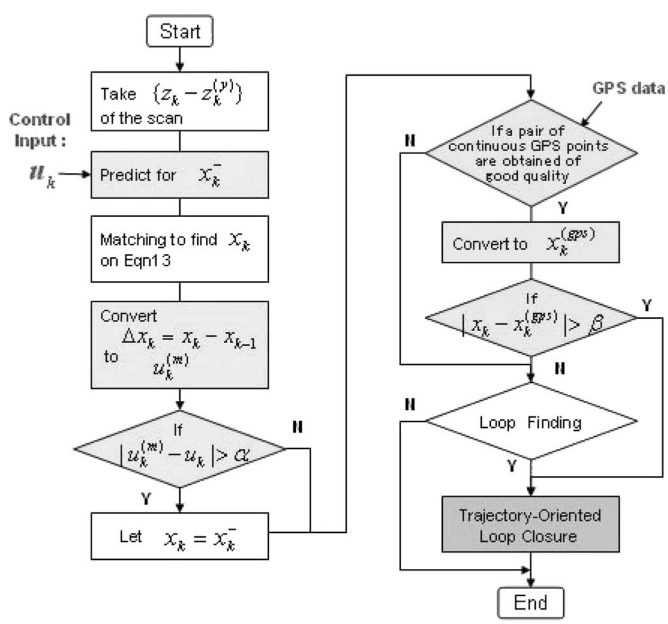


Fig. 6. Flow of localization algorithm, corresponding to module A of Fig. 1.

independent diagnosis method is required to detect erroneous scan matching.

In this paper, we previously trained a threshold vector α, which represents the error bound of the yaw rate sensor and speed encoder. We convert $\Delta x_k = x_k - x_{k-1}$ to speed and yaw rate $(u_k^{(m)})$ and take the difference with u_k. If the difference is beyond α, then x_k is replaced using the predicted state x_k^-.

On the other hand, whenever a pair of continuous GPS coordinates with good signal conditions is received, a vehicle pose x_k^{gps} can be calculated based on the GPS coordinates. If the difference with x_k is larger than a previously defined threshold β, which represents the error bound of GPS measurement, a trajectory-oriented closure is conducted to match the trajectory's end point x_k to the GPS measurement x_k^{gps} while maintaining the local consistency of the trajectory.

The same algorithm is used when a trajectory crosses at a certain point. A x'_k is estimated by matching the current scan with a previously generated map, and the trajectory-oriented closure is conducted to match x_k to x'_k while maintaining the local consistency of the trajectory.

B. Trajectory-Oriented Closure Algorithm

In this section, we discuss vehicle pose x_k using the form of T_k, representing a transformation matrix from the vehicle (or laser scanner's) coordinate system to a global one.

Let T'_k denote the transformation matrix of x_k^{gps} or x'_k, representing a different estimation of vehicle pose, with reduced error compared to T_k. The objective is to modify the vehicle trajectory $T_{s,\ldots,k}$ to reduce the gap $\Delta T_k = T'_k - T_k$.

Let $t_{ij} = T_i^{-1} \cdot T_j$ represent the relative vehicle motion from time stamp i to j. Recalling (13), the current vehicle pose x_k is updated from the previous estimation x_{k-1} by taking the matching of the current scan with an online generated map. The general idea here is to estimate vehicle motion Δx_k (i.e., $t_{k-1,k}$) by scan matching and to align it on x_{k-1} to compose the current vehicle pose x_k (i.e., $T_k = t_{k-1,k} \cdot T_{k-1}$).

Normally, t_{ij} has better local consistency and could reflect the local features of sensor motion in detail. However, it might be erroneous in the case where the vehicle's relative motion

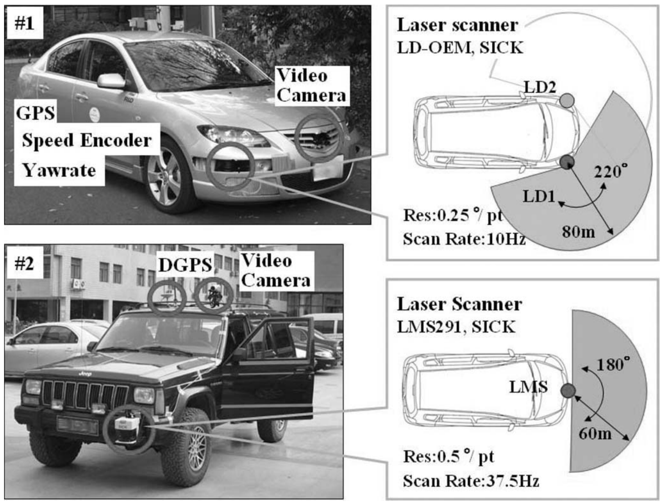


Fig. 7. Testbed vehicles and their sensor configurations.

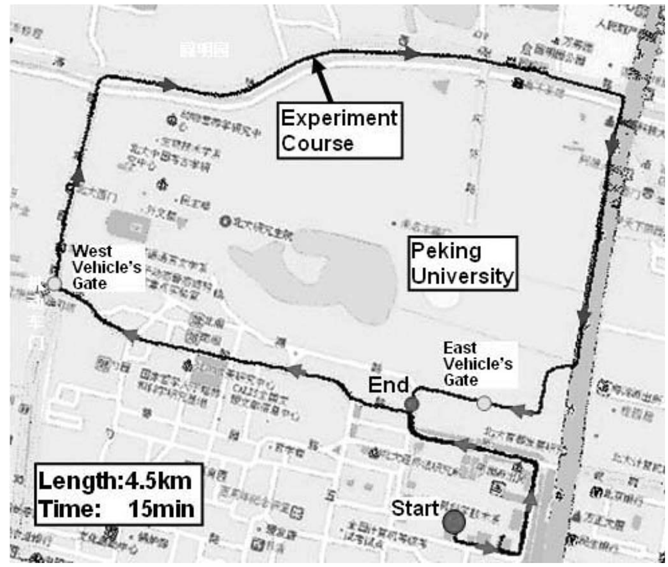


Fig. 8. Experimental course.

could not be determined by scan matching. The absolute drift of T_k has its origin in the error accumulation of t_{ij}s. A brute-force modification of T_k might break the consistency in scans that is achieved by t_{ij}s.

In this paper, instead of directly modifying $T_{s,\ldots,k}$, an adjustment $\Delta t_{i,i+1}$ is estimated for each $t_{i,i+1}$, $s < i < k$ to match T_k with T'_k, i.e.,

$$T'_k = t_{i+1,k} \cdot (\Delta t_{i,i+1} \cdot t_{i,i+1}) \cdot t_{s,i} \cdot T_s \qquad (14)$$

$$\Delta t_{i,i+1} = t^{-1}_{i+1,k} \cdot T'_k \cdot T^{-1}_s \cdot {t_{s,i}}^{-1} \cdot {t_{i,i+1}}^{-1}. \qquad (15)$$

An iterative modification is conducted. In each step, the $\Delta t_{i,i+1}$ that has the smallest norm is selected, representing the best efficiency in matching T_k to T'_k. The $t_{i,i+1}$ and $T_m, \forall m \in (i,k]$ are adjusted as follows:

$$\begin{aligned} \bar{t}_{i,i+1} &= \gamma \cdot \Delta t_{i,i+1} \cdot t_{i,i+1} \\ \bar{T}_m &= T_s \cdot t_{s,i} \cdot \bar{t}_{i,i+1} \cdot t_{i+1,m}, \qquad \forall m \in (i,k] \end{aligned} \qquad (16)$$

where $0 < \gamma < 1$ is a scaling factor. If $\gamma = 1$, then $\bar{T}_k$ will exactly meet on T'_k, yielding the error ΔT_k shift to $t_{i,i+1}$. To distribute the error ΔT_k to a number of $t_{i,i+1}$'s, here, we set $\gamma = 0.1$. The iteration continues until $|\Delta T_k|$ is smaller than a given threshold ϵ or can no longer be reduced.

IV. Experimental Results

Fig. 7 shows the testbed vehicles used in this work. The sensor configurations of the testbeds are slightly different, but their functions are similar, and their data are processed using the same approaches. We present a set of experimental results, focusing on the possibility of improving driving safety and on traffic data collection in a large populated environment. Here, we need to make clear that both experimental results demonstrated below are achieved in an offline mode shortly after the data collection. Furthermore, in each testbed vehicle, a video camera is mounted and calibrated with the laser scanner. In this paper, they are used to examine and visualize the results of laser-based processing. In the future, we will fuse both sensors to achieve higher intelligence and accuracy.

A. Sensor, Data, and Software

The experimental data were collected using testbed 2 in Fig. 7. Here, we describe the sensor configuration and data setting of the experiment. A laser scanner (LMS291 from SICK) is mounted at the front of the testbed vehicle, monitoring a wide angle (180° and 0.5°/point) of the vicinity with a scan rate of about 37.5 Hz. The data are downsized to 10 Hz during processing, considering the performance and computation efficiency. A differential GPS (DGPS) is used in the following two scenarios: 1) *Localization*: When the GPS signals are in good conditions, and its difference with estimated position is larger than a predefined threshold, the DGPS measurement is used to correct the estimated trajectory points. 2) *Accuracy evaluation*: All the DGPS measurements are used to find their difference with the estimated results. As the yaw rate sensor and wheel encoder were not ready in the experiment, we produced the values using the GPS data. To guarantee a certain reliability of the estimation, we choose only continuous GPS values of good signal condition with the vehicle running on a straight path (refer to $\{g_k\}$). The values of yaw angle $\{y_k\}$ and speed $\{s_k\}$ are calculated from $\{g_k\}$ so that the control inputs in this experiment are temporally broken segments. On the other hand, the coupled values $\{x^{\text{gps}}_k\} = \{g_k, y_k\}$ (see the red dots in Fig. 11) are used to adjust the trajectory of the testbed vehicle to achieve global accuracy. All the software used in the experiment are developed in C/C++ under MS Windows. The time stamps for different sensor data are basically the personal computer clock.

B. Experimental Course

The experimental course is shown in Fig. 8, where the testbed vehicle started from the campus of Peking University along the light gray arrows (left campus at the west vehicle's gate), ran

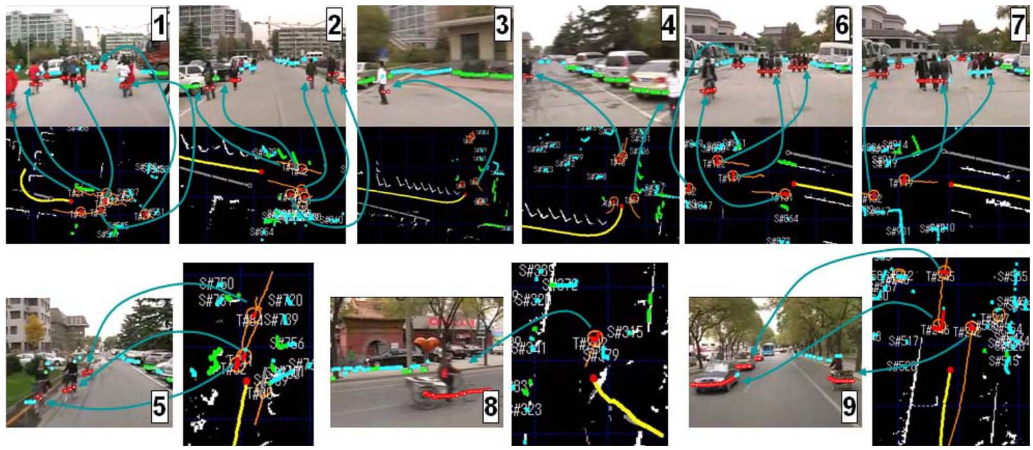
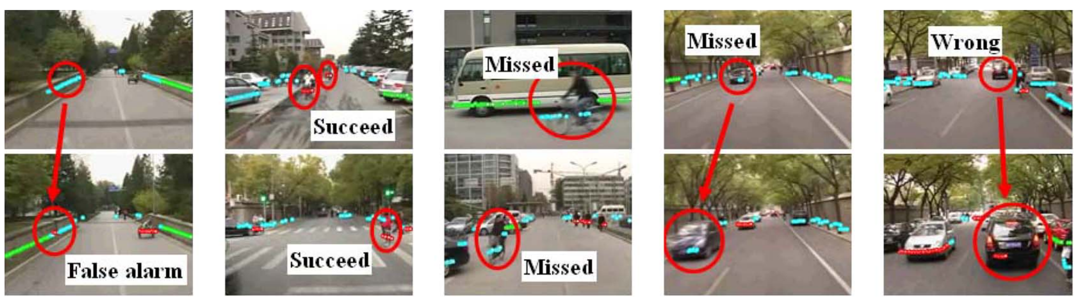


Fig. 9. Some screen captures of the laser-based processing and its back-projection on the video image. An experiment in a highly populated environment.



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

Fig. 10. Some of the failure cases.

on public roads along the black arrows, and entered the campus at the east vehicle's gate. The course lasted for 4.5 km, and the run took 15 min following the normal traffic flow. The course inside the campus is very crowded with pedestrians, bicycles, and parked cars. The course outside the campus is also very dynamic, composed of a number of intersections and crowded roads. The vehicle speed was limited to 20 km/h inside the campus and about 30–40 km/h outside the campus. This is the most challenging of all the data sets that we have collected.

C. SLAM With Object Tracking and Classification

When applied to improving driving safety, if the perception results are not to be associated with other data resources, global accuracy is not a requirement. However, accuracy in perceiving the local surroundings is very much required, particularly in a populated environment where the host vehicle might be close to other objects. Such local accuracy involves detecting and classifying the objects in the local vicinity, along with tracking and estimating their states, such as speed, direction, and size.

Fig. 9 demonstrates some of the experimental results on SLAM with object tracking and classification. Nine pairs of results are presented, each containing a screen capture of the processing program on laser scans, as well as a back-projection of the current laser scan onto the corresponding video image for visualization.

Colors have the same meaning as in Fig. 4. The laser points of current scans are labeled as static, moving, and seed objects, which are colored in green, red, and blue, respectively. They are consistent in both results. To gain better understanding, the green arrow lines in each pair of results are manually drawn, denoting the correspondences between the laser points in different views.

The results are indexed, and their locations are denoted in Fig. 11. Note that most moving objects are successfully detected and tracked (marked in red), particularly those near the testbed vehicle. The laser scanner is very efficient at monitoring a wide angle of the surrounding environment, as demonstrated in result 3. Although we can only find one person in the video image, the laser data captured three in the vicinity. The motion trajectories of moving objects are also clearly grasped. For example, in result 4, two people did not notice the existence of the testbed vehicle initially, and later, they walked away.

In the case of a group of persons, the program shows unstable results. For example, in result 6, the groups were tracked successfully. However, in later frames, a person in the right group walked slightly apart from the others so that one more cluster was detected and many state parameters of the trajectory showed discontinuous change. This iterated a couple of times. Finally, in result 7, the trajectories were rejected as unreliable, and a new seed object was created. Results 8 and 9 are the detections on public roads, which are crowded with cars and bicycles.

The accuracy of moving object detection is examined by back-projecting laser points on the video image and comparing to human assessment. In total, the data of 362 moving objects (trajectories) are back-projected onto the video images, while four of them are false alarms and 11 others are missed in the experiment. Some of the failure cases are demonstrated in Fig. 10.

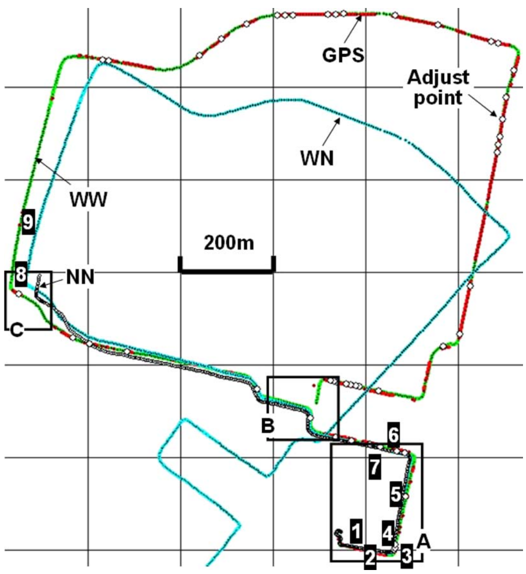


Fig. 11. Comparison of the localization results.

Since alarms beyond the viewing range of video images cannot be confirmed, we do not count any such false alarms that might exist. An example of a false alarm is shown in Fig. 10(a). Generally speaking, the system works well at detecting moving people. Except for people keeping static throughout their measurement, which are judged as static objects according to our tradeoff rule, all moving people (i.e., pedestrians) have been detected.

However, the detection results on bicycles and cars are not as good. Some results on bicycles are shown in Fig. 10(b) and (c). All of them are detected as seed objects, while some failed to be classified as moving objects. To prevent false alarms, our system carefully examined the continuities in motion vectors. In the case that the data appearance changes dramatically, estimation of motion vector might be erroneous so these could be discarded as wrong detections. These also happen in the case of cars. Cars also have another problem, as shown in Fig. 10(d) and (e). For a dark car, few range points are measured so the car may either be missed or detected as a different kind of object.

To reduce the failure cases, it will be necessary to improve the current implementation as well as to fuse with other sensing technologies.

D. Localization Guided by GPS Data and Control Inputs

In the sense of traffic data collection in a large populated environment, a localization of global accuracy is the major premise. Fig. 11 compares some of the localization results. The red dots, which are denoted by "GPS," are the $\{x_k^{\rm gps}\}$ that are picked up for adjustment, while the large white dots, which are denoted by "Adjust point," are a subset of $\{x_k^{\rm gps}\}$ that truly functioned in adjustment. The black dots, which are denoted by "NN," are the localization results that had assistance from neither $\{y_k\}$, $\{s_k\}$, nor $\{x_k^{\rm gps}\}$, except that an initial vehicle pose is assigned by $\{x_0^{\rm gps}\}$. This had a catastrophic failure in scan matching and aborted halfway. The blue dots, which are denoted by "WN," are generated with the assistance of $\{y_k\}$ and $\{s_k\}$ but without trajectory-oriented loop closure using $\{x_k^{\rm gps}\}$. A large global error occurred. Reasons may be found in the erroneous scan matching in a highly dynamic environment and error accumulation during a long and noncyclic measurement.

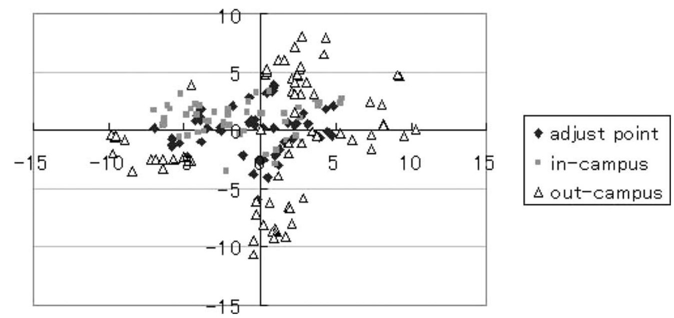


Fig. 12. Difference between the GPS coordinates and estimated trajectory points.

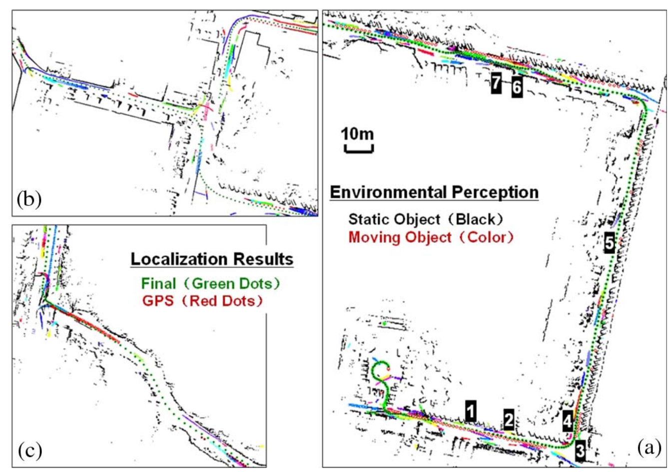


Fig. 13. Final maps of both static and dynamic objects in a highly populated environment.

The green dots, covered by other dots in many places and denoted by "WW," are generated with the assistance of $\{y_k\}$, $\{s_k\}$, and $\{x_k^{\rm gps}\}$. The threshold β in Fig. 6 represents the reliability of GPS measurement. Here, we set it to 10 m, which means that the trajectory-oriented loop closure will be conducted only when the distance from the estimated trajectory point to $\{g_k\}$ is larger than 10 m. The reason for such a setting is that we do not intend to make the system rely on an expensive precision GPS, while a normal navigation GPS has an error of about 10 m. In addition, we set the threshold ϵ in trajectory-oriented closure be $\epsilon = 0.7 * \beta$, as we do not intend to meet the trajectory points exactly on GPS values considering its localization error. Residuals are taken between all the GPS coordinates with the estimated trajectory points of the same time stamp, regardless of whether it belongs to $\{x_k^{\rm gps}\}$ or not. They are shown in Fig. 12. As described before, the experimental course can be divided into in- and out-campus sections, where the host vehicle drives in different traffic conditions with different speed. Gray dots in Fig. 12 denote the residuals for the in-campus section, triangles for the out-campus one, and black for adjust points (in accordance with Fig. 11). It is obvious that residuals of the in-campus section are lower than those of the out-campus one. This might be due to the better GPS

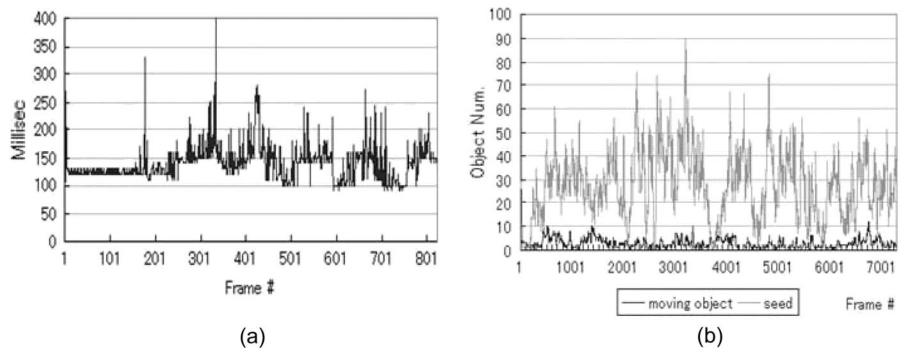


Fig. 14. (a) Computation cost and (b) number of objects tracked in each frame.

condition and less density of dynamic objects in the in-campus section. While almost all the dxs and dys in Fig. 12 are within or near the boundary of 10 m, demonstrating the algorithm has efficiently achieved a certain global accuracy.

E. Final Results

A global map, containing the data of both static and dynamic objects, with a pixel size of 5 cm $\times$ 5 cm, is generated by summarizing the local perceptions to a global coordinate system. Three areas, i.e., A–C, in Fig. 11 are enlarged in Fig. 13. The black pixels on the map denote static objects. Colored pixels denote moving objects, and different colors represent different moving objects. This demonstrates that after a run of the testbed vehicle, a global map of the environment, containing the information of both static and dynamic objects, can be generated. In addition, the speed and trajectory of both the host vehicle itself and the objects (e.g., cars, pedestrians, and bicycles) observed along the street can be recorded simultaneously with respect to a global coordinate system, i.e., many traffic data samples can be obtained through a single run.

F. Time Cost

The experiment was conducted by measuring data first, and processing the data later in an online procedure. After acquiring data from the testbed vehicle, all processing was carried out on a Lenovo ThinkPad X32, which has a 1.8-GHz Pentium R(M) processor and 1.5 GB random access memory. The laser scan covers a range of 180° with an angular resolution of 0.5°. The processing is conducted at a rate of 10 Hz, which means that less than one-third of the laser scans are used in processing, compared with the scanning rate ($\approx$37.5 Hz). The computation cost of each frame is demonstrated in Fig. 14(a) at an average of 140 ms/frame. Compared with the laser scan rate (10 Hz), the current computation cost is high but not far from real time. In addition, the number of moving and seed objects that are simultaneously tracked in each frame is output in Fig. 14(b). At the maximum, 12 moving objects were simultaneously tracked at the same time. However, if we count both moving and seed objects, more than 90 were simultaneously tracked.

G. Discussions and Future Studies

We have demonstrated the possibility of a laser-scanner-based approach for the purpose of both enhancing driving safety and traffic data collection. Despite positive results, there are still many problems with respect to real applications. Some of these could be solved by improving system implementations, while others need further studies to find a solution. Here, we discuss some of the more challenging issues that require more work.

1) Duplicated Courses: In Fig. 8, the start and end of the experiment course are at different locations, while in data acquisition, the testbed vehicle returned back to its start place, and data were measured up until then. The reason that we cut the duplicated course is that we failed to generate a consistent map using the current system and algorithm. As we used only one laser scanner covering the front 180°, matching the back-and-forth laser scans are difficult, particularly in such a dynamic environment. For the application of driving safety, a consistent map and global accuracy are not a must. However, an algorithm to update a previously explored map in a highly dynamic environment is required for traffic-data collection, where omniview laser scanning is required to solve the revisit problem.

2) Object Model and Classification Strategy: A parked car or a person standing still is a temporarily static object. Some of these might start to move suddenly, while others remain static during the period when they are measured. If we keep tracking all the moving-object-like ones, the computation cost will be too high for an online system. A balance between accuracy and time cost is made, and a brute forth tradeoff is taken in this work. However, this is not the image of our final goal. To improve computation efficiency while maintaining a certain accuracy, a comprehensive and intelligent classification strategy is required. Future study will be addressed in modeling the different kinds of objects and making a classification method using either a laser scanner or a fusion of multimodel sensing technologies.

3) Accuracy Measurement: A comprehensive accuracy examination is important to lead a research product to a final application, while it is always a big challenge. In the case of counting false alarms or missed or wrong detections, things are clear as long as we make a certain rule. However, if we want to evaluate the state estimations, such as the speed and direction of the detected moving objects, and analyze their error correlations, things are complicated. Obtaining such a ground truth from the moving objects in a real-world scenario is quite difficult, and a microsimulator that can mimic the true situation is required. Future study is required to develop such a simulator and make an accuracy examination strategy to evaluate the distance from current system(s) toward potential applications.

V. Conclusion

Motivated by two potential applications, namely, 1) enhancing driving safety and 2) collecting traffic data, this paper has proposed a laser-scanner-based approach, where the localization, mapping, moving object detection, and tracking issues are formulated as a SLAM with object tracking and classification problem, and a trajectory-oriented closure algorithm that uses GPS and control inputs to assist for global accuracy and robustness is proposed. The advantages of our approach are listed as follows: 1) The host vehicle is able to know the states (e.g., speed, direction, and trajectory) of both itself and the objects nearby in cluttered situations without relying on continuous and always reliable GPS/INS outputs, and 2) the host vehicle can record the speed and trajectory of both itself and the objects (e.g., cars, pedestrians, and bicycles) that it observed along the street so that many traffic data samples can be collected through a single run. We demonstrated the possibilities of the approach in high dynamic environments and discussed avenues for future work.

References

[1] Q. Yu, H. Araujo, and H. Wang, "A stereo vision method for obstacle detection and tracking in non-flat urban environments," *Auton. Robots*, vol. 19, no. 2, pp. 141–157, Sep. 2005.

[2] A. Broggi, C. Caraffi, P. P. Porta, and P. Zani, "The single frame stereo vision system for reliable obstacle detection used during the 2005 DARPA grand challenge on TerraMax," in *Proc. IEEE Intell. Trans. Syst. Conf.*, 2006, pp. 745–752.

[3] G. Ma, S.-B. Park, A. Ioffe, S. Muller-Schneiders, and A. Kummert, "A real time object detection approach applied to reliable pedestrian detection," in *Proc. IEEE Intell. Veh. Symp.*, 2007, pp. 755–760.

[4] D. Streller and K. Dietmayer, "Object tracking and classification using a multiple hypothesis approach," in *Proc. IEEE Intell. Veh. Symp.*, 2004, pp. 808–812.

[5] A. Mendes and U. Nunes, "Situation-based multi-target detection and tracking with laser scanner in outdoor semi-structured environment," in *Proc. Conf. Intell. Robots Syst.*, 2004, pp. 88–93.

[6] F. Fayad and V. Cherfaoui, "Tracking objects using a laser scanner in driving situation based on modeling target shape," in *Proc. IEEE Intell. Veh. Symp.*, 2007, pp. 44–49.

[7] N. Kaempchen, K. C. Fuerstenberg, A. G. Skibicki, and K. C. J. Dietmayer, "Sensor fusion for multiple automotive active safety and comfort applications," in *Advanced Microsystems for Automotive Applications*. Berlin, Germany: Springer-Verlag, 2004, pp. 137–163.

[8] N. Floudas, A. Polychronopoulos, O. Aycard, J. Burlet, and M. Ahrholdt, "High level sensor data fusion approaches for object recognition in road environment," in *Proc. IEEE Intell. Veh. Symp.*, 2007, pp. 136–141.

[9] T. Vu, O. Aycard, and N. Appenrodt, "Online localization and mapping with moving object tracking in dynamic outdoor environment," in *Proc. IEEE Intell. Veh. Symp.*, 2007, pp. 190–195.

[10] T. Weiss, B. Schiele, and K. Dietmayer, "Robust driving path detection in urban and highway scenarios using a laser scanner and online occupancy grids," in *Proc. IEEE Intell. Veh. Symp.*, 2007, pp. 184–189.

[11] C. C. Wang, "Simultaneous localization, mapping and moving object tracking," Ph.D. dissertation, Carnegie Mellon Univ., Pittsburgh, PA, 2004. CMU-RI-TR-04-23.

[12] T. Gandhi and M. M. Trivedi, "Video based surround vehicle detection, classification and logging from moving platforms: Issues and approaches," in *Proc. IEEE Intell. Veh. Symp.*, 2007, pp. 1067–1071.

[13] B. Gao and B. Coifman, "Vehicle identification and GPS error detection from a LIDAR equipped probe vehicle," in *Proc. IEEE Intell. Trans. Syst. Conf.*, 2006, pp. 1537–1542.

[14] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney, "Stanley: The robot that won the DARPA grand challenge," *J. Field Robot.*, vol. 23, no. 9, pp. 661–692, Sep. 2006.

[15] C. Urmson, C. Ragusa, D. Ray, J. Anhalt, D. Bartz, T. Galatali, A. Gutierrez, J. Johnston, S. Harbaugh, H. Kato, W. C. Messner, N. Miller, K. Peterson, B. Smith, J. Snider, S. Spiker, J. Ziglar, W. Whittaker, M. Clark, P. L. Koon, A. Mosher, and J. Struble, "A robust approach to high-speed navigation for unrehearsed desert terrain," *J. Field Robot.*, vol. 23, no. 8, pp. 467–508, 2006.

[16] H. Zhao and R. Shibasaki, "Reconstructing textured CAD model of urban environment using vehicle-borne laser range scanners and line cameras," *Mach. Vis. Appl.*, vol. 14, no. 1, pp. 35–41, Apr. 2003.

[17] K. W. Lee, S. Wijesoma, and J. I. Guzman, "A constrained SLAM approach to robust and accurate localisation of autonomous ground vehicles," *Robot. Auton. Syst.*, vol. 55, no. 7, pp. 527–540, Jul. 2007.

[18] M. Montemerlo, S. Thrun, and W. Whittaker, "Conditional particle filters for simultaneous mobile robot localization and people-tracking (slap)," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, vol. 1, pp. 695–701.

[19] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "People tracking with a mobile robot using sample-based joint probabilistic data association filters," *Int. J. Robot. Res.*, vol. 22, no. 2, pp. 99–116, 2003.

[20] S. Wender, T. Weiss, K. C. J. Dietmayer, and K. Fürstenberg, "Object classification exploiting high level maps of intersections," in *Proc. Adv. Microsyst. Autom. Appl.*, 2006, pp. 187–203.

[21] D. Hahnel, W. Burgard, D. Fox, and D. Thrun, "An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *Proc. Conf. Intell. Robots Syst.*, 2003, pp. 206–211.

[22] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–108, Jun. 2006.

[23] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping—Part 2," *IEEE Robot. Autom. Mag.*, vol. 13, no. 3, pp. 108–117, Sep. 2006.

[24] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA: MIT Press, 2005, pp. 309–484.

[25] T. Bailey, "Mobile robot localisation and mapping in extensive outdoor environment," Ph.D. dissertation, Australian Control Field Robot., Univ. Sydney, Sydney, Australia, 2002.

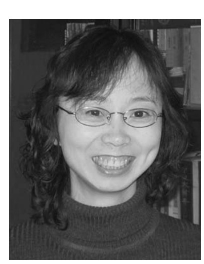
[26] P. Newman, D. Cole, and K. Ho, "Outdoor SLAM using visual appearance and laser ranging," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 1180–1187.

[27] F. T. Ramos, J. Nieto, and H. F. Durrant-Whyte, "Recognising and modelling landmarks to close loops in outdoor SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2007, pp. 2036–2041.

[28] G. Welch and G. Bishop, *An Introduction to the Kalman Filter*. Chapel Hill, NC: Univ. North Carolina Press, Feb. 2001. TR95-041.

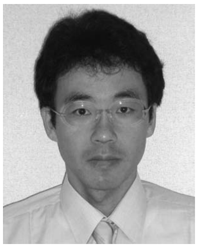
[29] K. Konolige and K. Chou, "Markov localization using correlation," in *Proc. IJCAI*, 1999, pp. 1154–1159.

[30] H. Zhao, X. W. Shao, K. Katabira, and R. Shibasaki, "Joint tracking and classification of moving objects at intersection using a single-row laser range scanner," in *Proc. IEEE Intell. Trans. Syst. Conf.*, 2006, pp. 287–294.



Huijing Zhao (M'07) received B.S. degree in computer science from Peking University, Beijing, China, in 1991 and the M.E. and Ph.D. degrees in civil engineering from the University of Tokyo, Tokyo, Japan, in 1996 and 1999, respectively.

From 1991 to 1994, she was with Peking University, where she was involved with a project developing a graphic information system platform. In 2003, after several years of postdoctoral research with the University of Tokyo, she became a Visiting Associate Professor with the Center for Spatial Information Science. In 2007, she joined Peking University as a Professor with the Key Laboratory of Machine Perception (MOE), and the School of Electronics Engineering and Computer Science. Her research interest covers image sensing, computer vision, and spatial data handling.

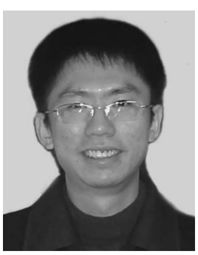


Masaki Chiba received the B.E. degree in electric engineering from Waseda University, Tokyo, Japan, in 1991.

In 1991, he joined Mazda Motor Corporation, Hiroshima, Japan, where he is currently with its Technical Research Center. After several years of research in the areas of the autonomous vehicle and the navigation system, he has been engaged in the development of driver-support systems that use roadside-to-vehicle/vehicle-to-vehicle communications.

Ryosuke Shibasaki (M'08) received the B.S., M.S., and Ph.D. degrees in civil engineering from the University of Tokyo, Tokyo, Japan, in 1980, 1982, and 1987, respectively.

From 1982 to 1988, he was with the Public Works Research Institute, Ministry of Construction. From 1988 to 1991, he was an Associate Professor with the Department of Civil Engineering, University of Tokyo. In 1991, he moved to the Institute of Industrial Science, University of Tokyo. In 1998, he was promoted to Professor with the Center for Spatial Information Science, University of Tokyo. His research interest covers 3-D data acquisition for graphic information system (GIS), conceptual modeling for spatial objects, and agent-based microsimulation in GIS environments.



Xiaowei Shao received the B.E. and Ph.D. degrees in electronic engineering and information science from the University of Science and Technology of China, Hefei, China, in 1999 and 2006, respectively.

From 2004 and 2008, he was a Researcher with the Center for Spatial Information Science, University of Tokyo, Tokyo, Japan, where he has been a Project Assistant Professor since April 2008. His research interests include machine vision, pattern recognition, surveillance, target tracking, and spatial data processing.

Jinshi Cui (M'09) received the B.S. and Ph.D. degrees in computer science from Tsinghua University, Beijing, China, in 1999 and 2004, respectively.

In 2004, she joined Peking University, Beijing, as an Assistant Professor with the Key Laboratory of Machine Perception (MOE), and the School of Electronics Engineering and Computer Science. She was promoted to Associate Professor in 2007. Her research interest covers computer vision and robotics.

Hongbin Zha (M'06) received the B.E. degree in electrical engineering from the Hefei University of Technology, Hefei, China, in 1983 and the M.S. and Ph.D. degrees in electrical engineering from Kyushu University, Fukuoka, Japan, in 1987 and 1990, respectively.

After being a Research Associate with the Kyushu Institute of Technology, he joined Kyushu University in 1991 as an Associate Professor. He was also a Visiting Professor with the Centre for Vision, Speech, and Signal Processing, Surrey University, Surrey, U.K., in 1999. Since 2000, he has been a Professor with the Key Laboratory of Machine Perception (MOE), and the School of Electronics Engineering and Computer Science, Peking University, Beijing, China. He is the author of more than 200 technical papers published in journals, books, and international conference proceedings. His research interests include computer vision, digital geometry processing, and robotics.

Dr. Zha received the Franklin V. Taylor Award from the IEEE Systems, Man, and Cybernetics Society in 1999.