

Vehicle Autonomy Using Cooperative Perception for Mobility-on-Demand Systems

Seong-Woo Kim, Tirthankar Bandyopadhyay, Baoxing Qin, Zhuang Jie Chong, Wei Liu, Xiaotong Shen, Scott Pendleton, James Guo Ming Fu, Marcelo H. Ang Jr., Emilio Frazzoli and Daniela Rus

Abstract A holy grail of research in urban transportation systems is to increase throughput of people while minimizing the requirement of building additional road and rail networks. The promising new paradigm of Mobility-on-Demand (MoD), where shared personal transportation vehicles provide necessary service, is fast becoming a viable and preferred alternative to the traditional framework of having either public fixed route service or privately owned vehicles. This chapter looks at innovative approaches that enable an MoD system to be economical in development, robust in operation, efficient and sustainable in deployment. We develop algorithms to efficiently use the combination of prior information with minimalistic sensing with only a single 2-D LIDAR, utilizing vehicle odometry and prior road information which may not have accurate metric information but is topologically consistent. Additionally, we use the rich capability of cooperative perception, which can far extend perception range without expensive long-range sensors, by exchanging local perception information with other vehicles or infrastructure via wireless communications. The augmented perception capability enables a vehicle to see the oncoming traffic situation ahead even beyond human line-of-sight and field-of-view, which thereby contributes to traffic flow efficiency and safety improvement through long-term perspective driving, e.g., early obstacle detection and avoidance, and early lane changing. This chapter develops these ideas, presents results in demonstration and provides insights of the motion and operation planning for a case study of a fully autonomous vehicle deployment to the general public.

S.-W. Kim
Seoul National University, Seoul, Korea

T. Bandyopadhyay (✉) · J.G.M. Fu
Singapore MIT Alliance for Research and Technology, Singapore, Singapore
e-mail: tirtha.bandy@gmail.com

B. Qin · Z.J. Chong · W. Liu · X. Shen · S. Pendleton · M.H. Ang Jr.
National University of Singapore, Singapore, Singapore

E. Frazzoli · D. Rus
Massachusetts Institute of Technology, Boston, USA

Keywords Capítulo 11.-Mobility-on-demand systems · Cooperative perception · Autonomous vehicles · Driver assistance

1 Introduction

A holy grail of research in urban transportation systems is to increase throughput of people while minimizing the requirement of building additional road and rail networks. The promising new paradigm of Mobility-on-Demand (MoD), where shared personal transportation vehicles provide necessary service, is fast becoming a viable and preferred alternative [1] to the traditional framework of having either public fixed route service or privately owned vehicles. Many such MoD systems are already available commercially where the shared transportation modules are either cars [2, 3], bikes [4] or shared vans [3, 5]. These systems aim to reduce ownership of private vehicles and increase effective resource utilization.

Our research group at Singapore-MIT Alliance for Research and Technology looks at technologies that use vehicle automation as a component of MoD systems. In order for the systems to be viable, they have to be economical in development, robust in operation, efficient and sustainable in deployment. To achieve these challenging goals, we develop algorithms to efficiently use the combination of prior information with minimalistic sensing. In accordance with minimalistic sensing, our vehicles are localized reliably in the campus using only one single 2D LIDAR (Light Detection And Ranging), odometry and prior road information.

In the same context, we use the rich capability of cooperative perception, which can far extend perception range without expensive long-range sensors, by exchanging local perception information with other vehicles or infrastructure via wireless communications. The augmented perception capability enables a vehicle to see the oncoming traffic situation ahead even beyond human line-of-sight and field-of-view, which thereby contributes to traffic flow efficiency and safety improvement through long-term perspective driving, e.g., early obstacle detection and avoidance, and early lane changing [6].

As a first step of system development and deployment, we focus on the crowded campus environment of National University of Singapore, where demands on transportation and the mobility challenges due to pedestrians and other vehicles are significant. Figure 1 shows our personal transporters in operation. We have demonstrated this system successfully with accumulated autonomous run mileage of over 100 km in the campus environment.

While there has been significant progress in autonomous driving in terms of technological capability, a major challenge that remains is development of public policy that stems from a positive public perception towards adopting such a system. To get an insight into such questions and the feedback from public, we had various demonstrations to visiting dignitaries and researchers from around the world. We also had a field deployment of the system on public grounds in the Science Center where we gained valuable feedback and insights with the interaction with members of the public, some of which we present in this paper.



Fig. 1 Our autonomous personal transporters in operation, providing Mobility-on-Demand service in a campus environment of National University of Singapore

The contribution of this chapter can be summarized as the following:

- We introduce our autonomous vehicles for Mobility-on-Demand systems.
- We propose cooperative perception for vehicle automation and planning.
- We present the case for autonomous vehicle deployment to the general public.

This chapter is outlined as follows. Section 2 provides the overview of our systems. Section 3 presents how our autonomous vehicles perceive urban road environment using local range sensors. Section 4 addresses how our systems use remote information from other vehicles or infrastructure via wireless communications in the context of cooperative perception. Section 5 presents driving assistance capability using cooperative perception via MoD systems, which enables a see-through collision warning, overtaking and lane changing assistance. In Sect. 6, we discuss the impact and next steps for autonomous vehicles in MoD systems as highlighted through the experiments and case studies of public demonstrations. We conclude this chapter in Sect. 7.

2 System Overview

We have operated a number of autonomous vehicles for MoD service at NUS campus, which are implemented on mass-produced automotive vehicles as shown in Fig. 1. The overall architecture of vehicles is presented in Fig. 2.

The vehicles start to move toward the next destination by self-driving upon the requests of users or operators. For this purpose, an autonomous driving system is necessary, which consists of several subsystems such as a perception, planning and control system. For the perception, the vehicles are equipped with range sensors like 2D LIDARs, a vision camera, and wireless interface IEEE 802.11g, IEEE 802.11n, 3G HSDPA (High-Speed Downlink Packet Access), and 4G LTE (Long Term Evolution).

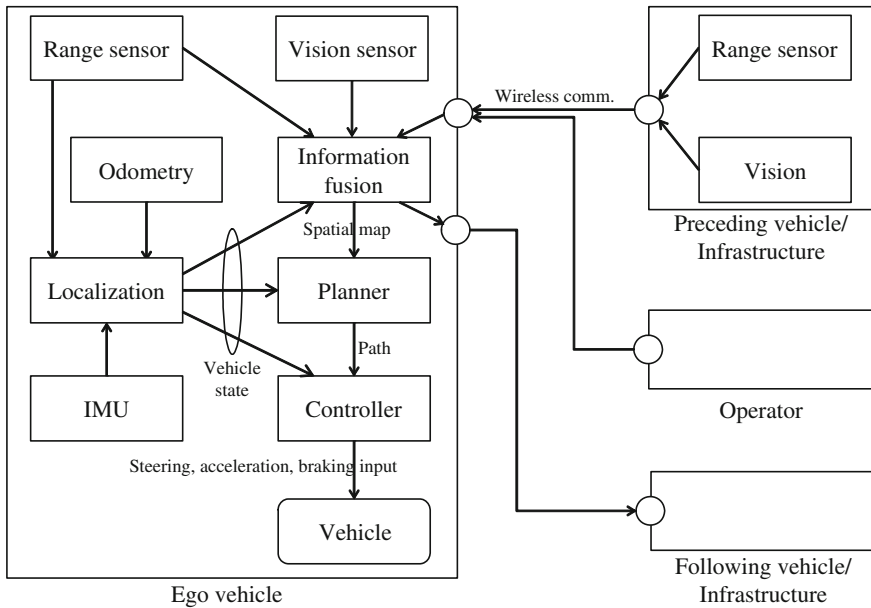


Fig. 2 Overall system architecture

The software architecture of this system was established on Robot Operating System (ROS) suite [7] using only open source libraries. Detail specifications are available in [8].

In Fig. 2, an odometry system is used for motion estimation such as moving direction and speed. The range sensors are used for vehicle detection and tracking. The vision sensors are used for identification and classification of pedestrians and vehicles, and provide driver-friendly visual traffic information.

The vehicles exchange their local perception results with other vehicles or infrastructure via wireless communications where the information is delivered in a form of network messages. Since there exists a trade-off between information quantity and communication performance, the message profile, e.g., transmission period and message size, should be carefully chosen according to driver preferences and application requirements. The detailed message profiles and communication performance investigation will be provided in Sect. 4.

All local and remote sensing information is properly fused at the box of information fusion in Fig. 2. Compared to data fusion of on-board sensing information, data fusion of remote sensing information on the road includes a number of practical challenges [9], among which this chapter focuses on map merging problem and sensor multi-modality.

After the information fusion procedure, the fused information can be delivered to the local planner for self-driving, or vehicles following behind or nearby infrastructure for the purpose of cooperative perception.

From the perspective of local planner, the vehicles can provide autonomous collision avoidance and lane changing through our several intelligent algorithms based on local perception and cooperative perception, which include vehicle detection, pedestrian detection, intention aware planning [10], localization [11], on-road recognition [12] and path planning [13] and cooperative perception [6].

From the perspective of cooperative perception, it can be definitely beneficial to both human drivers and autonomous vehicles for safe and comfortable driving, because the fused information includes oncoming traffic situation ahead. In this context, we provide a method to let a driver know the moment when the driver should be careful, such as hidden obstacle detection or sudden braking of preceding vehicles beyond line-of-sight. The notification is performed by visual and sound alarms for a human driver, which enables a driver to focus on driving until any dangerous situation is detected by our system. Moreover, the subsystem notifies a driver when there are any vehicles coming from behind or at blind spots, which can contribute to safe lane changing or overtaking. This notification can trigger path re-planning earlier for safe and comfort autonomous driving including smooth deceleration/acceleration and early lane changing.

From the next section, we first investigate local perception and localization using local range sensors. Then, we move our focus to cooperative perception, cooperative driving assistance, and cooperative autonomous driving.

3 Synthetic 2D LIDAR for Vehicle Localization in 3D Urban Environment

Vehicle localization is a fundamental requirement for autonomous vehicle, which is the problem of determining the pose of a robot relative to a given map of the environment [14]. This section presents our precise localization algorithm for vehicles in 3D urban environment using one 2D LIDAR and odometry information. A novel idea of synthetic 2D LIDAR is proposed to solve the localization problem on a virtual 2D plane. A Monte Carlo Localization scheme is adopted for vehicle position estimation, based on synthetic LIDAR measurements and odometry information.

3.1 Localization on a Virtual Plane

Vehicle localization on a planar surface has been studied for decades and many algorithms have been proposed. The 2D scan-matching algorithm may be the most popular choice due to its accuracy and robustness [15]. However, it cannot be directly applied for vehicles moving in the 3D world. Since outdoor road has many up-and-downs, laser points from a planar LIDAR may cast on the road surface, rather than the desired vertical objects, as discussed in [16]. Our previous research in [11] uses

a tilted-down LIDAR to extract road boundary features on urban road, and then uses these features for vehicle localization. Actually, there are many other salient features in urban environment that can benefit localization. What features to extract, how to extract them, and how to feed them into the localization scheme are questions to be further addressed.

3D range data is usually desired to extract features for robot navigating in the 3D world [17]. In this paper, we use a tilted-down LIDAR to generate 3D point cloud of the environment in a push-broom configuration.¹

Rather than directly apply 3D scan-matching with the raw data [16], we try to extract features from the 3D point cloud, and use the vertical features for localization. The assumption of our method is that urban environment is rich in vertical surfaces, such as curbs, wall of buildings, and even vertical tree trunks.

The vertical world assumption is actually a popular assumption used in many works in the literature. Harrison and Newman in [18] proposed a method to generate high quality 3D laser range data while the robot is moving. By exploiting the assumption of vertical world, useful information (e.g., roll and pitch angles) can be inferred. Kohlbrecher et al. in [19] achieved 2D SLAM and 6-DOF pose estimation using one single 2D LIDAR and an IMU (Inertial Measurement Unit). Although not explicitly explained, the underlying assumption in the work is that the environment contains many vertical surfaces. Weingarten et al. in [20] used this assumption to realize fast structured environment reconstruction. Rezaei et al. proposed a navigation method in a 3D scenario with 2D LIDAR data in [21].

In this work, since outdoor environments may have more arbitrary-shape objects other than structural environments, a classification step has to be taken before using the vertical assumption. In the classification procedure, laser points cast on the vertical surfaces are extracted based on surface normal estimation. When the tilted-down LIDAR sweeps the environment, some vertical surfaces will be swept from bottom to up in consecutive laser scans. If we take a bird's eye view for this scanning process and project the vertical features on to a virtual horizontal plane, it is exactly the same as a robot with a horizontal LIDAR moving on a 2D surface. From a mathematical point of view, the vertical surface constrains how laser points at different height should match with each other. With the above intuition, the idea of synthetic LIDAR is proposed. A synthetic LIDAR is a planar 2D LIDAR on the projected virtual plane, where the end points of its laser beams are the projected points from vertical surface in the 3D environment. Figure 3 shows an example of the synthetic LIDAR.

Before looking at the technical details, let us understand the key idea of the synthetic LIDAR first. In the upper left of Fig. 3, 2D LIDAR readings are accumulated over 3D space. The accumulated 3D points are processed to extract interest points. Finally, the post-processed points are mapped into an occupancy grid map for localization and planning.

The idea of synthetic LIDAR helps to solve the 3D localization problem on a 2D plane. Although a vehicle is moving in the 3D world with 6-DOF, generally

¹The push-broom configuration is exactly "tilted-down LIDAR" configuration. The LIDAR is the broom. As the vehicle moves, the LIDAR works in a push-broom way.

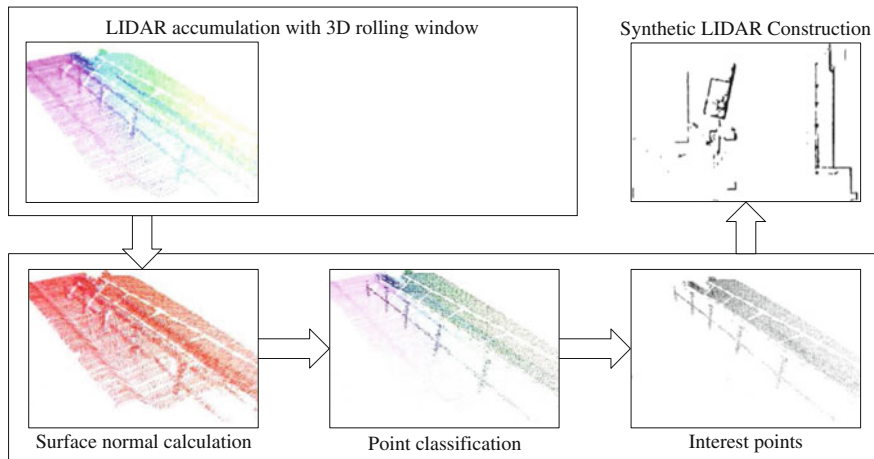


Fig. 3 Construction of synthetic LIDAR

speaking, ground based vehicle is mostly interested in its 2D pose vector (x, y, yaw) . By projecting the 3D vertical features onto a virtual plane, 2D occupancy grid map can be used by marking those vertical features. This way, an a-priori map can be obtained using SLAM with the idea of synthetic LIDAR. It should be clarified that our algorithm only applies to an environment with only one vertical traversable level. For cases with more traversable levels, some other 2.5D or full 3D algorithms may be used, for example [22].

The localization system mainly consists of two parts, 3D perception to extract key feature points, and 2D localization to solve the localization on the horizontal plane. The synthetic LIDAR serves as a bridge to connect the 3D world and the 2D virtual plane, as shown in Fig. 4a. In this work, we use the map-aided localization method where the origin of the coordinate is set to the left-bottom corner of the map.

The system uses an IMU and a wheel encoder to provide 6-DOF odometry information, a 2D tilted-down LIDAR to provide laser scans, and an occupancy grid map serving as a prior for localization. A simple dead reckoning is used to obtain the odometry information. Assuming the distance measured by a wheel encoder at n -th time step is r_n , and the rotation is given by a pitch θ and a yaw Ψ , the change in position of the vehicle is given by:

$$\begin{pmatrix} \Delta x_n \\ \Delta y_n \\ \Delta z_n \end{pmatrix} = \begin{pmatrix} \cos(\theta_n) \cos(\Psi_n) \\ \cos(\theta_n) \sin(\Psi_n) \\ -\sin(\theta_n) \end{pmatrix} \cdot (r_n - r_{n-1}). \tag{1}$$

The 3D perception assumes that odometry system is accurate enough in a short time period, and accumulates the laser scans for 3D range data. A classification procedure is then applied to extract interest points from the accumulated data. The extracted laser points are then projected onto a virtual horizontal plane (by ignoring their z

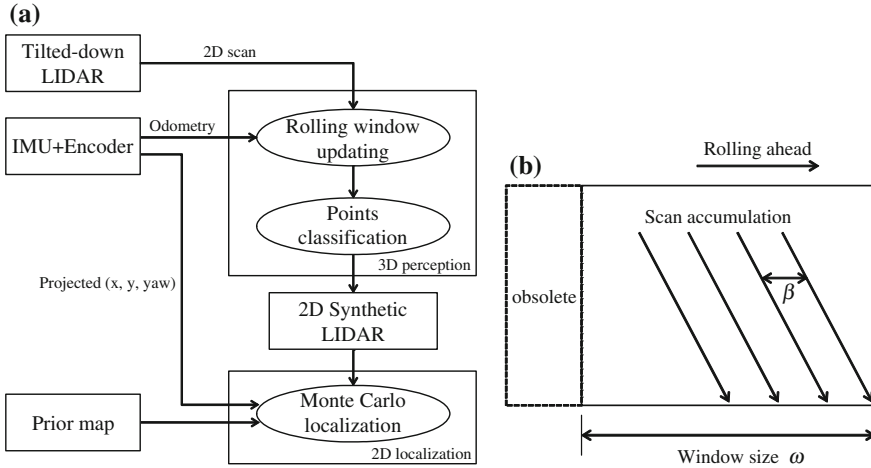


Fig. 4 Localization system overview. **a** Localization flow chart. **b** 3D rolling window

values), and a synthetic 2D LIDAR is constructed. The 2D localization fuses odometry information from odometry and measurements from the synthetic 2D LIDAR in a Monte Carlo Localization scheme. With a prior map of vertical features generated beforehand, localization on the 2D horizontal plane is achieved.

3.2 3D Perception

One of the requirements of the synthetic LIDAR is excellent adaptability that fits with different types of environment in urban scenario. We achieve this by properly extracting interest points from a reconstructed environment model, before the synthetic LIDAR is built.

To be able to recognize features that are perpendicular to the ground, an accurate model of the world is necessary. There are numerous ways that allow building an accurate environmental model, which includes nodding LIDAR and Velodyne [23]. As introduced in Sect. 3.1, a fixed, tilted down single planar LIDAR enables the reconstruction of the environment accurately by sweeping across the ground surface. This is an attractive solution since it is low cost and only requires rigid mounting of the sensor. It also allows optimization to be done that allows real-time computation of feature extraction that is unique to this configuration.

3.2.1 3D Rolling Window

The reconstruction uses rolling window sampling to maintain high probability of reflecting more recent samples by the ranging sensor. As such, a 3D rolling window is used to accumulate different scans recorded in a short distance. The size of the

window is flexible and the rolling window forms a local map of the 3D environment, i.e., it rolls together with the vehicle, where new incoming scans will be added into the window, and the old samples get discarded.

More specifically, given the window size w , the points p in n -th scan, P_n is accumulated according to

$$P_n = \bigcup_{k=n-w} \{p_k, \dots, p_n\} \quad n > w. \quad (2)$$

As shown in Fig. 4b, w is used to control the number of accumulated scans such that the size of the window would not grow unbounded. Also, a new scan is only inserted when a sufficient distance, β is achieved. This has two effects, a small β will have denser points but the overall window size will become shorter and vice versa. In our case, it was set to $\beta = 0.02$ m and $w = 100$ achieve the right compromise. The rolling window works in the odometry frame of the system, where each scan from a physical LIDAR is projected based on the odometry information derived from IMU and wheel encoder.

3.2.2 Point Classification

To extract features that are perpendicular to the ground, surface normal needs to be estimated. While many method exists [24], we used normal estimation proposed by [25]. It is based on first order 3D plane fitting, where the normal of each point in the space is approximated by performing least-square plane fitting to a points local neighborhood P^K [26]. The plane is represented by a point x , its normal vector \mathbf{n} and distance d_i from a point $p_i \in P^K$, where d_i is defined as

$$d_i = (p_i - x) \cdot \mathbf{n}. \quad (3)$$

By taking $x = \bar{p} = \frac{1}{k} \sum_{i=1}^k p_i$ as the centroid of p_k , the values of \mathbf{n} can be computed in a least-square sense such that $d_i = 0$. The solution for \mathbf{n} is given by computing the eigenvalue and eigenvector of the following covariance matrix $C \in \mathbb{R}^{3 \times 3}$ of P^K [27]:

$$C = \frac{1}{k} \sum_{i=1}^k \cdot (p_i - \bar{p}) \cdot (p_i - \bar{p})^T, C \cdot \mathbf{v}_j = \lambda_j \cdot \mathbf{v}_j, j \in \{0, 1, 2\}, \quad (4)$$

where k is the number of points in the local neighborhood, \bar{p} as the centroid of the neighbors, λ_j is the j th eigenvalue with \mathbf{v}_j as the j th eigenvector.

The principal components of P^K corresponds to the eigenvectors \mathbf{v}_j . Hence, the approximation of \mathbf{n} can be found from the smallest eigenvalue λ_0 . Once the normal vector \mathbf{n} is found, the vertical points can then be obtained by simply taking the threshold of \mathbf{n} along the z axis, e.g. 0.5. This can vary depending on how noisy the sensor data is.

To find the local neighborhood points efficiently, KD-tree [28] is built from all the points obtained from the rolling window and perform a fixed radius search at each point. Although the surface normal can be calculated as a whole, performing normal calculation at each point in the rolling window can be very expensive. To further reduce the computation complexity, two successive rolling windows are maintained, where

$$P_{n+1}^{\phi} = P_n^{\phi} \cup \Phi(P_{n+1} \setminus P_n), \quad (5)$$

where Φ can be any points classification function, P^{ϕ} consists of the processed points and P contains the raw points. This way, surface normal calculation is only required for the much smaller rolling window $P_{n+1} \setminus P_n$. In other words, this ensures that classification will only perform on the newly accumulated point cloud and the processed points from the previous instance can be reused.

3.2.3 Synthetic LIDAR Construction

The result from the classified points consists of a collection of interest points in 3D. For the construction of synthetic LIDAR, the interest points in 3D is projected into virtual horizontal plane ($z = 0$). It can be seen that this synthetic LIDAR has a very special feature: the ability to see-through the obstacles. This is possible due to the way that its laser scans are synthesized. During the 3D data accumulation, the collection of laser points is not performed at a single time stamp at the horizontal plane, but in a short time interval while the tilted-down LIDAR sweep the local environment. The objects at a distance (but within the sensor range) will not be blocked by the objects that is nearby but lower than the mounting height of the tilted LIDAR. Laser points cast on the faraway vertical surfaces will be also kept in the synthesis of the laser scan, contributing to the “see-through” characteristics.

This synthetic LIDAR is comprehensive in the sense that it is as if having multiple short ranged LIDARs arranged at different height at a forward facing configuration, with the height adapted to wherever there exist a vertical surface in the environment. The construction of synthetic LIDAR is completed by placing the virtual sensor at the base of the vehicle and performs transformation of all the interest points from odometry to the vehicle’s base. In many applications where a standard LIDAR is desired (equally spaced angle increment), the synthetic LIDAR can be further reconstructed to fulfill this constraint. This would involve performing ray tracing at each fixed angle increment to obtain minimum range value from the possible end points. The overall 3D perception can be summarized in Fig. 3. The 3D perception is done with the Point Cloud library [29] which provides many of the operations described in this section.

3.3 Online Localization

3.3.1 MCL Localization

This paper adopts the Monte Carlo Localization (MCL) scheme in [30] to estimate the vehicle pose. MCL is a probabilistic localization method based on Bayes' Theorem and Monte Carlo idea [14]. The core of MCL is a particle filter, where the belief of vehicle position is maintained by a set of particles. MCL mainly consists of three steps, prediction, correction, and resampling. For the motion model which is required for the prediction step, Pseudo-3D odometry motion model from our previous work [11] is used. The choice of measurement model is discussed in the following.

3.3.2 Virtual LIDAR Measurement Model

To incorporate the measurement into localization, a measurement model is needed for the synthetic LIDAR. The likelihood model is adopted for the synthetic LIDAR. Since the end points of virtual beams are the projection of interest points from vertical surfaces, it is possible that different points from different vertical surfaces may have the same angle. In other works, there exist two laser beams with the same angle while having two different range values. For this reason, synthetic 2D LIDAR is a peculiar LIDAR that only detects vertical surfaces, and can also see-through these surfaces. In light of this, the likelihood model which only requires the end points of laser beams is well suited for the synthetic LIDAR.

3.4 Experimental Results

In this experiment, SICK LMS-151 LIDAR is mounted in the upper-front and tilted-down for localization. A 4-layer LIDAR, SICK LD-MRS400001 is mounted at the waist level for obstacle detection. Both rear wheels of the golf cart are mounted with encoders that provide an estimate of the distance traveled. An IMU MicroStrain 3DM-GX3-25 is mounted at the center of the real axle to provide orientation information of the vehicle. The localization algorithm is tested in the Engineering Campus of National University of Singapore, where the road is up-and-down and many high buildings exist off the road.

A prior map is first generated with graph SLAM techniques by using the synthetic LIDAR as the input. To perform pose optimization, [31] is used as front end to detect loop closure. Then, the fully optimized pose is recovered using optimization library from [32]. To evaluate the quality of the recovered map built from synthetic LIDAR, the map is projected onto a satellite map, as shown in Fig. 5. The map shows consistency with good correlation with the satellite map, with an area of about 550 m × 487 m. Although there are discrepancies towards the left side of the map due to uniform longitudinal features along the road, the overall topology is maintained. This shows that the map can be used for accurate localization.



Fig. 5 Mapping of the NUS engineering area

The synthetic LIDAR using SICK LMS-151 is able to perform at a rate of 50 Hz output on a laptop with Core i7 processor, showing that the synthetic LIDAR can be used to perform a real-time localization. The localization results are shown in Fig. 6. Judging from the prior map, the localization result from our algorithm always aligns with our driving path where a parallel line with the road boundary is clearly shown in the long stretch of road. Since our algorithm does not rely on GPS, our estimation still performs well near areas crowded by tall buildings. Note that in the experiment, a rough initial position is given and hence localization is mostly concerned with pose tracking. However, the system is able to cope with small kidnapping problems, e.g. brief data error from the LIDAR since the odometry system is still able to provide information. Should a large kidnapping occur, e.g. the vehicle was moved in between placed without turning on the localization module, a rough initial position may be provided to speed up the convergence rate.

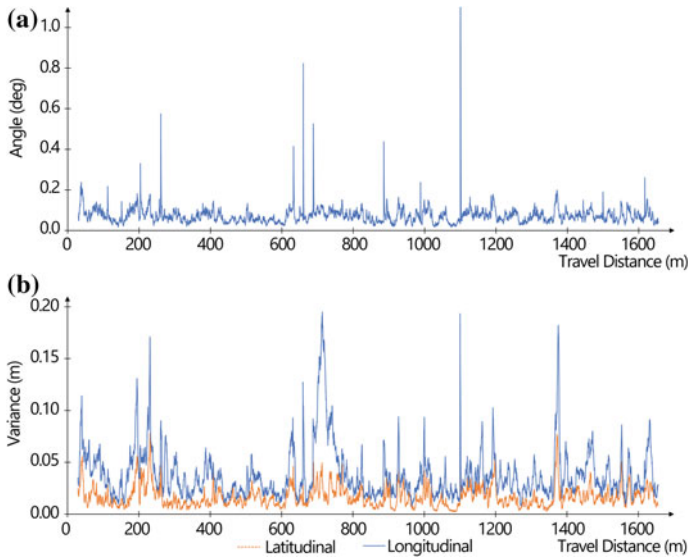


Fig. 7 Localization variance results in terms of angle and position. **a** Angle variance. **b** Position variance

4 Cooperative Perception for Situational Awareness and Vehicle Control

In MoD systems, multiple autonomous vehicles are operating at the same time and equipped with radio devices to handle remote requests from users or operators. In the scenario where multiple vehicles can communicate with other vehicles or infrastructure, perception range can be far extended up to the boundary of connected vehicles by exchanging local perception information. The extended perception range can reach even beyond line-of-sight or field-of-view according to the network connectivity. This augmented perception capability can contribute to safety improvement and traffic flow efficiency [6, 33].

4.1 Cooperative Perception

Cooperative perception is defined as sharing local sensing information with others via wireless communications. The preceding vehicles highly affect the driving decision of the ego driver. In this sense, a leader is defined as a preceding vehicle (1) connected via cooperative perception and (2) observable by the ego vehicle² through its local

²Ego vehicle is a reference point of sensor fusion and a target of planning and control.

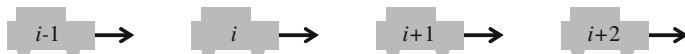


Fig. 8 Concept of leader string where i is an ego vehicle. $i + 1$ and $i + 2$ are the first and second leader of i , respectively. $i - 1$ is a following vehicle behind ego vehicle i

sensors or remote information via cooperative perception. Let $\mathcal{V}_i = \{i + 1, \dots\}$ be a *leader string* of a vehicle i , where, $j, j + 1 \in \mathcal{V}_i$, j and $j + 1$ are connected via cooperative perception and $j + 1$ is observable by local sensors of j . The concept of leader string is depicted in Fig. 8.

In a broad sense, following vehicles of ego vehicle i can be included in the leader string \mathcal{V}_i , although a vehicle $i - 1$ is not a leader literally. The information of a vehicle $i - 1$ can be beneficial for blind spot detection or lane changing assistance, because the following vehicle $i - 1$ can watch the ego vehicle in the third-person view from behind the ego vehicle.

In vehicle driving scenarios, sensing information is typically dealt with as a map. Let $\mathcal{M} = \{\dots, m, \dots\}$ be a map for navigation of vehicles, which consists of the set of points obtained and filtered from sensors, where $m \in \mathbb{R}^n, n = 2, \text{ or } 3$. Given a position $p \in \mathcal{M}$ in a map, $\mathcal{M}[p] \rightarrow \mathbb{R}$ can be defined in several ways such as the height of obstacle in case of $p \in \mathbb{R}^2$, or the belief that the position p is obstacle-free. \mathcal{M}_i denotes a map of a vehicle i .

Now, we can formulate cooperative perception as follows:

$$\mathcal{M}_i = \mathcal{M}_i \bigcup_{j \in \mathcal{V}_i} \mathcal{M}_j, \tag{6}$$

where the operation \bigcup is called *map merging*. The map merging operation is merely a set union operation, if \mathcal{M}_i and $\mathcal{M}_{j \in \mathcal{V}_i}$ are mapped into a global coordinate frame such as GPS coordinates. However, the observation from sensors is typically mapped into a local coordinate frame. Also, there is no guarantee that the initial poses of vehicles are known. In this case, the relative pose between vehicles is necessary to merge different spatial information. The relative pose can be defined as $q = (\tau, \theta)$, where τ and θ correspond to translation and rotation, respectively. We define a transformation operator as $p \otimes q = R(\theta)p + \tau$, where $R(\theta)$ is a rotation matrix. Finally, (6) can be rewritten in a more general form as follows:

$$\mathcal{M}_i = \mathcal{M}_i \bigcup_{j \in \mathcal{V}_i} \mathcal{M}_j \otimes q_{i,j}, \tag{7}$$

where $q_{i,j}$ is a relative pose between a vehicle i and j . One of the key challenges to solve (7) is to obtain an accurate $q_{i,j}$.

4.2 Map Merging Problem

The primary problem of map merging can be formulated as follows:

$$q_{i,j}^* = \arg \max_{q_{i,j}} S(\mathcal{M}_i, \mathcal{M}_j, q_{i,j}), \quad (8)$$

where the similarity measure S is defined as

$$\sum_p \mathcal{L}(\mathcal{M}_i[p], (\mathcal{M}_j \otimes q_{i,j})[p]), \quad (9)$$

where $\mathcal{L}(a_i, a_j)$ is a point-to-point similarity measure, which is positive if $a_i = a_j$; 0, otherwise. Equations (8) and (9) attempt to find the relative pose that maximizes the overlapping area between two maps.

Equations (8) and (9) can be extended to more than two vehicles. Let $Q_i = \{q_{i,i+1}, \dots, q_{i,i+N}\}$ be the set of relative poses w.r.t the ego vehicle, where N is the number of leader vehicles. The problem can be rewritten as follows:

$$Q_i^* = \arg \max_{Q_i} S(\mathcal{M}_i, \dots, \mathcal{M}_{i+N}, Q_i), \quad (10)$$

where the similarity measure S is defined as

$$\sum_p \mathcal{L}(\mathcal{M}_i[p], \dots, (\mathcal{M}_j \otimes q_{i,j})[p], \dots), \quad (11)$$

where $\mathcal{L}(a_i, \dots, a_{i+N})$ is positive if $a_i = \dots = a_{i+N}$; 0, otherwise. There are various ways to implement the similarity measure. In this work, we use Iterative Closet Point (ICP) [34, 35] and Correlative Scan Matching (CSM) [36] to quantify and implement the measure.

Algorithm 1 is an essential algorithm to solve Eqs.(8)–(11). The first step `LeaderVehicleDetection` responds for vehicle detection and estimating the leader vehicle's pose using on-board LIDAR and return the initial relative pose $q_{i,j}^0$. The detail leader vehicle detection method is described as follows.

Algorithm 1: On-Road Map Merging Algorithm

input : $\mathcal{M}_i, \mathcal{M}_j$
output: Merged map
begin
 $q_{i,j}^0 \leftarrow \text{LeaderVehicleDetection}(\mathcal{M}_i)$
 $q_{i,j}^* \leftarrow \text{ScanMatching}(\mathcal{M}_i, \mathcal{M}_j, q_{i,j}^0)$
 return $\mathcal{M}_i \cup (\mathcal{M}_j \otimes q_{i,j}^*)$
end

One of the distinct characteristics of scan data on the road is that features are not obvious to match due to bushes, trees, or pedestrians. Nonetheless, scan data of a leader has relatively clear features such as the backside or side of vehicles. The initial pose of a leader can be guessed from the features. More specifically, the proposed system keeps checking whether there are consecutive points lying in the desired path ahead. For this task, let \mathcal{D}_i be the set of positions that a vehicle i will/can move to, typically the set of the center positions of the current lane. To find the leader position, the proposed system keeps watching straight or slightly-curved lines in the following point set:

$$\{p | r_{th} > \|p - p_l\|, p_l \in \mathcal{D}_i\}, \quad (12)$$

where r_{th} is the maximum boundary that a vehicle can go away from the desired path, typically half of the lane width. Equation (12) can significantly reduce the search space and false positives of leader vehicle detection.

If (1) the detected points compose a straight or slightly curved line, and (2) the line is almost orthogonal to the desired path, the center position of the line and the orthogonal angle to the line are used as t_0 and θ_0 , respectively, which is used as the initial transformation $q_{i,j}^0 = (t_0, \theta_0)$.

Based on the estimated relative pose $q_{i,j}^0$, ICP and CSM are employed for `ScanMatching` to match the scans within the overlap area to maximize $S(M_i, M_j, q_{i,j})$ and refined the relative pose accordingly. Finally, the map is merged according to (7). More detail information is available in [6].

4.3 Sensor Multi-Modality

Given the relative poses between different vehicles, it is easy to perform map merging for sensory information such as a laser scanner whose readings are a set of physical quantities recorded with their spatial coordinates. However, for the sensory modality of vision whose reading is an image, map merging is not a straightforward task, because the vision image is the result of perspective projection unlike laser scan data.

To deal with the problem, we use the Inverse Perspective Mapping (IPM) [37] method that can map these projected pixels to their spatial coordinates on the road surface. For a fixed-mount on-board vision camera, its pose in the vehicle coordinate and the pose relative to the road surface are usually known. Together with its intrinsic parameters from a calibration process, the perspective transformation matrix from road surface to vision camera image can be calculated. The inverse operation of this perspective matrix will restore the original road surface from the camera image. Each pixel of the image generates one color point on the ground. By applying IPM operation, camera sensing information is transformed from the projected image pixels into the spatial coordinate, with which map merging can be performed.

In the merged map of ego vehicle, three types of information represented: (1) vehicle poses, including poses of ego vehicle and other vehicles; (2) laser scan points; and (3) color points of road surface from vision. By supporting multiple sensory modalities, the cooperation perception system helps ego vehicle to perceive not only occluded vehicles and obstacles, but also road surface that may be out of its sensing ability.

Figure 10a shows an example of the map merging result fused with information from range sensor and vision sensor on the road.

4.4 Experimental Results

In the experiments, we used four vehicles consisting of one Mitsubishi iMiEV and three Yamaha golf cars, in which one vehicle participated as a moving obstacle. Among four vehicles, three vehicle were equipped with the proposed cooperative perception system. The setup of three vehicles is represented in Fig. 8, where i , $i + 1$ and $i + 2$ are corresponding to the ego vehicle, the first leader and the second leader, respectively. In Fig. 8, the second leader transmits its sensing information to the first leader via wireless communications, while moving forward. With the support of our system, the first leader merges the remote information with its local sensing information, and then transmits the combined information to the ego vehicle via wireless communications, while moving forward as well. We conducted experiments using the vehicles on a campus road at NUS on sunny or cloudy days between 12 and 5 PM. The rule of the road is to drive on the left and the speed limit is 40 km/h.

Table 1 shows the performance of map merging algorithm according to the scan matching methods and leader detection method. The LDR is obtained from LIDAR-based LeaderVehicleDetection. It is used as the initial condition for both the ICP and CSM scan matching algorithm. In this work, the ICP algorithm from [38] is used. The comparison is done using a ground truth generated by particle-filter-based localization using LIDAR. In the ICP, $d_{th} = 2.5$ m is used. For CSM, two levels of 0.5 and 0.1 m search grids are used. It is also set to search within a window of $10 \text{ m} \times 10 \text{ m} \times 60^\circ$, and all computations are done with a desktop computer equipped with Core i7 CPU and a Nvidia GTX770 graphics card. Note that among the obtained

Table 1 Performance of map merging in terms of translation and rotation error

Average		LDR	ICP	CSM
Translation error (m)	First leader	0.73	0.54	0.44
	Second leader	1.81	1.76	1.11
Orientation error (degree)	First leader	14.44	12.14	2.77
	Second leader	17.65	13.15	3.51
Computation time (ms)		6	6	108

$q = (\tau, \theta)$, the translation τ can be measured more accurately by range sensors. This accuracy asymmetry between position and orientation results in lever-arm effects. In Table 1, it was found that there is significant improvement of the merged map from the second leader. While CSM performed slower than ICP, it is able to robustly estimate the relative pose since CSM is robust to initialization error. We conclude that the results using the both scan matching approaches perform sufficient for our requirements.

Figures 9 and 10 show the snapshots of cooperative perception visualization. In Fig. 9a–c show the camera views of the ego vehicle, the first leader and the second leader, respectively. (e) and (f) shows see-through views of the first and second leader, respectively. (d) shows the third person view of the ego vehicle, where driver’s seat is lifted at 30 m from the ground. Figure 9 shows the satellite view of the ego vehicle at the same moment of Fig. 9. Note that the ego vehicle cannot see oncoming traffic

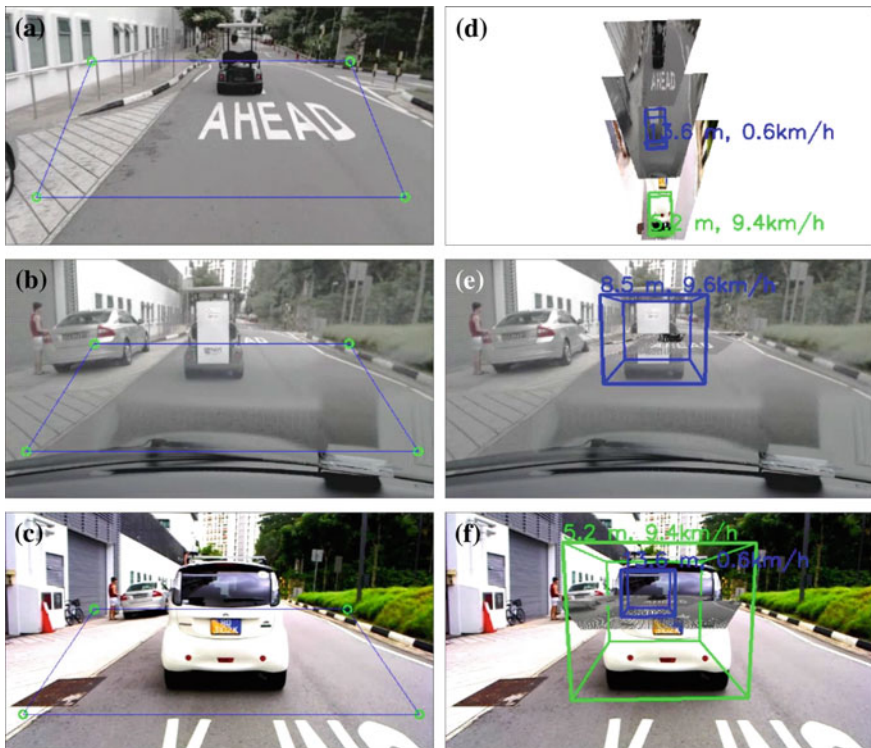


Fig. 9 Map merging results fused with information from range sensor and vision sensor, where the cuboids represent a simplified vehicle model. **a** Second leader’s camera view. **b** First leader’s camera view. **c** Ego vehicle’s camera view. **d** Ego vehicle’s lifted seat view (30 m). **e** First leader’s see-through view. **f** Ego vehicle’s see-through view

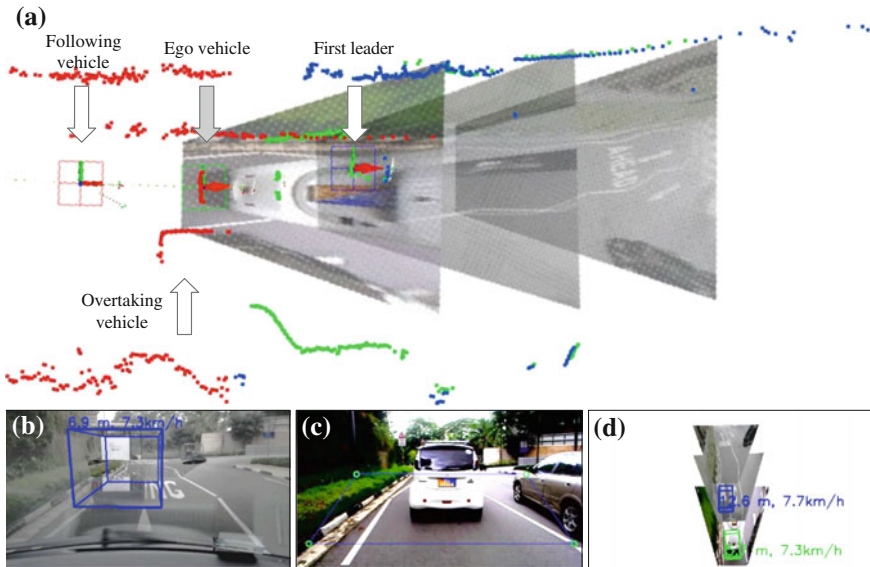


Fig. 10 All-around view supported by cooperative perception. **a** Ego vehicle’s satellite view. **b** Ego vehicle’s see-through view. **c** Following vehicle’s camera view. **d** Ego vehicle’s third person view

situation ahead due to the first leader and the limitation of line-of-sight. However, the ego vehicle equipped with the cooperative perception system can see the second leader and the preceding vehicle of the second leader, and traffic sign on road surface, e.g., “AHEAD”.

Figure 10a–d show how all-around view is supported by a following vehicle connected via cooperative perception. Figure 10a is the satellite view of the ego vehicle, where the grey arrow indicates the ego vehicle. Note that there is another vehicle behind the ego vehicle. The ego vehicle is moving between the first leader and the following. In Fig. 10a, the ego vehicle can detect a vehicle approaching behind, i.e., the overtaking vehicle, because the following vehicle connected via cooperative perception can tell the ego vehicle about what is going on at blind spot of the ego vehicle, which can greatly improve situational awareness.

Figure 10b is a see-through view of the ego vehicle. Figure 10c is a view of the following vehicle $i - 1$, which can be seen by the ego vehicle. (d) is the third person view of the ego vehicle, where driver’s seat is lifted at 30m from the ground. All-around view characteristics of Fig. 10 can be applied to the lane changing and overtaking assistance. In addition, a normal vision camera can sense the adjacent left and right lane, as we can see Fig. 10a and d. A fisheye lens can be considered for multiple lane detection. In this work, we used a processed image and laser scan as a message profile, whose average size is 6.5 K. The average delay is less than 100 ms on IEEE 802.11g, IEE 802.11n, 4G LTE.

5 Automated Early Collision Avoidance Using Cooperative Perception

If the system can let a driver know the moment when the driver has to drive carefully or watch the display through proper visual, audio, or tactile feedback, a driver can focus on driving itself without having to keep watching a driving assistance display. The forward collision warning can trigger path-replanning earlier for early hidden obstacle avoidance. In this context, one of the most desired functions is forward collision warning.

5.1 See-Through Forward Collision Warning

Many forward collision warning algorithms have been proposed along with risk assessment method, which includes time-based, and distance-based approaches. Firstly, the time-based methods usually use Time-To-Collision (TTC), which can be formulated as

$$TTC_{i,j} = \frac{g_{i,j}}{v_i - v_j}, \quad j \in \mathcal{V}_i, \quad (13)$$

where $TTC_{i,j}$ is corresponding to TTC, $g_{i,j}$ is the distance between a vehicle i and j , and v_i is the speed of a vehicle i . A forward collision warning is activated if $TTC_{i,j}$ is less than a certain threshold time, typically 2–3 s according to safety requirements [39].

In distance-based methods, a forward collision warning is activated if, given two vehicles i and j , the recommended safety gap $r_{i,j}$ is larger than the distance between two vehicles $g_{i,j}$, which can be restated as follows: A forward collision warning is activated if

$$g_{i,j} < r_{i,j}, \quad j \in \mathcal{V}_i, \quad (14)$$

where $r_{i,j}$ is a minimum distance to avoid collision to a preceding vehicle j , which can be formulated as follows [40]:

$$r_{i,j} = (v_i - v_j)/2\gamma^2 + (v_i - v_j) * T_{rs}, \quad j \in \mathcal{V}_i, \quad (15)$$

where γ is the deceleration of the ego vehicle, e.g., $-0.2g$ (≈ -2 m/s). T_{rs} is the response time of a driver, e.g., 0.5–1.5 s.

Note that one of the great advantages of our system is that j is not limited to only $i + 1$. In our system, thanks to the capability of a see-through view, j can be beyond the first leader according to the connectivity via cooperative perception, as shown in Fig. 11. This see-through characteristic enables the driver to avoid hidden obstacles earlier than without cooperative perception.

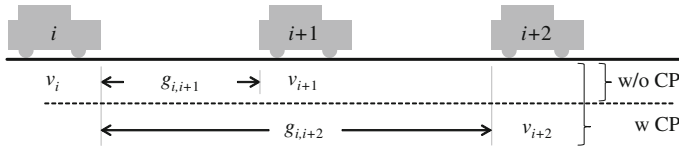


Fig. 11 Comparison of collision warning with and without cooperative perception

5.2 All-Around View Using Cooperative Perception

Not surprisingly, cooperative perception enables all-around view without having to install sensors covering all sides of a vehicle. In particular, a following vehicle can see the ego vehicle in a third-person view, including its blind spots. Figure 10a shows one snapshot of the all-around view of the ego vehicle on the road. Furthermore, our system can detect a vehicle approaching toward the ego vehicle using a spatio-temporal moving obstacle detection and tracking method. From the following, we investigate the benefit of all-around view in terms of overtaking assistance and lane changing assistance.

5.3 Overtaking and Lane Changing Assistance

When a driver should drive slowly or stop due to a slow-moving truck or obstacle ahead, the human or autonomous driver should decide whether to wait longer in the current lane or change lanes, which is an important issue for overtaking on a single-lane road.

Many overtaking assistance methods have been proposed according to the requirements, sensing capability and sensor configurations. In principle, the overtaking decision should be determined with the consideration of (1) the number of lanes, (2) the speed of a preceding vehicle, (3) cut-in space availability, (4) distance from an oncoming vehicle, and (5) the existence of another overtaking vehicle from behind as in Fig. 10d. Our system can inform (3), (4), (5) that are difficult to be provided without cooperative perception. Equations (14) and (15) can be applied to read-end collision warning at lane changing, which are corresponding to $TTC_{j,i}$ and $r_{j,i}$, respectively, where the vehicle j is approaching toward the ego vehicle behind in an adjacent lane, and $j < i$.

5.4 Feedback to a Driver

Once a forward collision is expected, or overtaking is not possible, or lane changing is not possible, it should be notified to a driver through visual, audio, or tactile feedback. In Fig. 12, some examples of visual warnings are illustrated for a forward

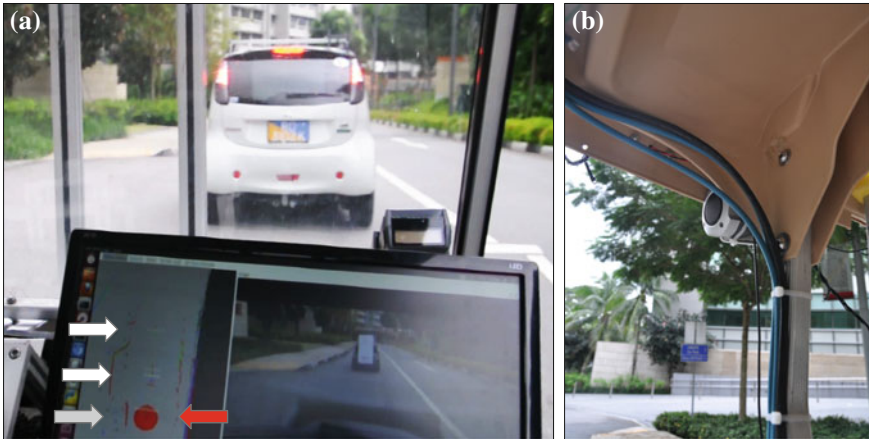


Fig. 12 See through forward collision warning is provided both as a visual warning (seen in **a**) as well as audio warning on speakers (seen in **b**)

collision warning, overtaking possibility, and lane-changing possibility, respectively. In the next subsection, we evaluate how our system can contribute to the see-through forward collision warning, overtaking assistance, and lane changing assistance are, through real experiments on the road using vehicles equipped with our system. Furthermore, we will address in the next experimental results how the feedback can be used for autonomous driver from the perspective of planning and control.

5.5 Experimental Results

We performed (1) see-through forward collision warning tests with real human drivers using real vehicles equipped with the driving assistance system, and (2) automated early lane change experiments using our autonomous vehicles.

5.5.1 See-Through Forward Collision Warning

The test scenario is as follows. Three vehicles move forward on single-lane road like Fig. 1, where a test driver drives the ego vehicle. Then, the second leader suddenly stops at a certain position. To avoid collision, the first leader and the ego vehicle stop accordingly. Real tests were conducted with three human drivers on the road eight times.

In this experiment, the second leader was moving forward at 2–5 m/s before sudden braking. Accordingly, TTC was set to 15 s. These are somewhat conservative parameters primarily for safety concerns. Note that the brake lights of the second leader were turned off. It makes the first leader difficult to notice whether the second

leader decelerates or not. The warning sound includes two loud beeps that last for 2 s per one warning activation.

We used IEEE 802.11n as a radio interface of vehicle-to-vehicle communications. Laser scan and compressed vision image were used as a message profile. The average communication delay between the first leader and the ego vehicle is 16 ms by measurement. IEEE 802.11n performed well at more than 30 m distance on the road.

Figure 12a shows one camera shots captured at the front passenger seat. The left-bottom screen of Fig. 12a shows the satellite view, where the three left-most arrows indicate the ego vehicle, the first leader, and the second leader as like Fig. 9a. The red circle indicated by vertical red arrow represents a high collision probability, because the vehicle the first leader is suddenly stopped in front of the first leader.

In principle, the test driver cannot see the second leader due to the limitation of line-of-sight. However, the test driver can see the second leader through the cooperative perception system. The right bottom screen of Fig. 12a is the see-through view. The color of the red dot becomes green when no collision is expected. The collision warning sign turned red along with a loud sound alarm using a speaker in Fig. 12b. The first leader also stopped to avoid collision.

Average forward collision warning activating time is 2.1 s according to the preset TTC and the distance from the second leader. The test drivers pushed the brake pedal averagely 0.82 s later after a forward collision warning sound is activating, which is a time from processing the sound signal via making a decision in a brain to actuating foot muscles. The delay is widely ranged from 0.7 s to 1 s according to the drivers. In this experiment, the test drivers detected the sudden braking of the second leader earlier than the first leader and accordingly stopped earlier as much as 0.33 s. With the support of cooperative perception, the test driver has more time to cope with the traffic situation.

5.5.2 Early Automated Lane Changing

The see-through collision warning can be used for triggering path-replanning for collision avoidance. Figure 13 shows simulations of the scenario. These simulations utilized a RRT* path planner [41].

Once a collision warning is triggered, the motion planner is called to search a feasible trajectory for obstacle avoidance. Thanks to cooperative perception, the planning space can be extended to as far as the neighboring vehicles connected via wireless communications can perceive; this allows motion planning to be conducted in a long-term perspective. To alleviate the time-constraint issue of long-term planning, the anytime RRT* [42] is used, where a feasible trajectory can be quickly found and the solution path will then be asymptotically optimized.

Additionally, while cooperative perception could provide an enlarged planning space, the inherent sensing uncertainty and transmission delay need to be carefully considered. A cost map based approach is proposed in our previous work in [13] to handle this issue. Specifically, the sensing uncertainty and transmission delay are factored into a cooperative-perception-based cost map, then an anytime RRT* algorithm is employed to search for an optimal trajectory subject to this cost map.

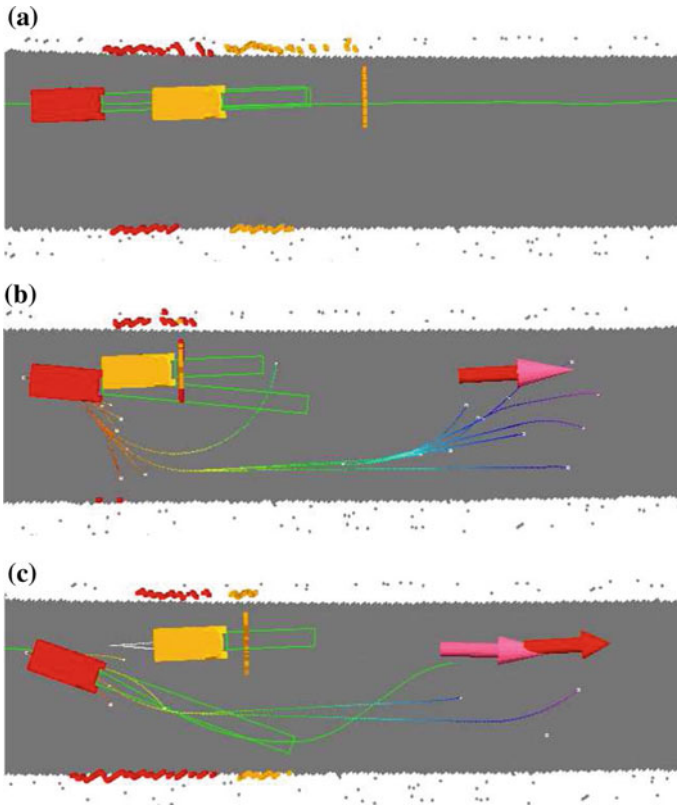


Fig. 13 Early lane changing with the support of a see-through collision warning using cooperative perception. The leader is represented by an *orange rectangle*, and the ego vehicle is represented by a *red rectangle*. Both vehicles are traveling with a lane from *left to right*, with the lane blocked by an obstruction shown as a *vertical line*. Feasible vehicle trajectories are shown by *multicolored curved lines*. **a** shows the point of sudden obstacle appearance, **b** shows path plan of ego vehicle without cooperative perception, and **c** shows planning with the see-through forward collision warning

In this manner, the solution trajectory will be located in the regions with lowest collision risk.

In Fig. 13a shows a situation without cooperative perception, during which a sudden obstacle appears in front of the leader; only the leader detects the obstacle whereas the following ego vehicle has a limited line-of-sight range only up to the rear of the leader. In Fig. 13b, the both of the leader and the ego vehicle suddenly stop, then the ego vehicle performs re-path planning to overtake the stopped leader. However, it is hard to find the feasible path to overtake the leader in the case of Fig. 13b, because the ego vehicle and the leader are too close after the sudden stop. In Fig. 13c, early lane changing is triggered with the support of the see-through collision warning. Since sufficient space is occupied to overtake the leader, the path planner can find the overtaking path promptly.

6 Results and Impact

Since the end of 2011, our main testing grounds have been within the campus grounds of the National University of Singapore. There are two main testing sites: One on a road involving vehicular and pedestrian movement, and another on a non-road area involving heavy pedestrian movement.

Our work in 2013 mainly focused on the second site involving heavy pedestrian movement. Pedestrian interaction is usually with students who are busy rushing off to their next lectures. Students' movements are tied to their lecture schedules. It is common to see groups of students appear around the same time, rushing off to the bus-stop to catch the approaching bus for their next lecture. This presents a nice testing grounds for our vehicle to not only prove the reliability of the vehicle's implemented dynamic safety zone but as well as to further improve the vehicle's mobility within a densely crowded pedestrian environment.

6.1 *An Invitation to a Live Showcase*

The Singapore Science Centre held its first ever Street Fair from 8–11 November 2013. This was in conjunction with the celebrations of its 35th Anniversary in the ongoing endeavor to promote science and technology to the public. The SMART Autonomous Vehicles Group was invited to exhibit the driverless buggy at this Street Fair. We were also given the opportunity to showcase the vehicle in operation amongst the pedestrian crowd. And the most fun part of this was giving out free rides. Figure. 14, captures one such ride.

This was an exciting milestone to our research work. This was the first time the vehicle was operating outside of university campus. This was a very important validation to our work as public interaction with our vehicle was involved—not just with the vehicle motion but as passengers as well.

While roboticists in the community of autonomous vehicles are more comfortable with how such driverless vehicles work, the same cannot be said with the general public. One of the purposes for this live showcase was to raise public awareness of the maturity of driverless technology and that driverless vehicles can be very safe. This was achieved by allowing the public to freely interact with the vehicle. This provided a good opportunity for everyone to get comfortable with a moving vehicle without any human driver. People would learn from first-hand experience that the vehicle will slow down when someone comes near to the vehicle and that the vehicle would reliably come to a halt when anything comes too near it. Free rides were given as well. This allowed the public to have a first-hand experience in sitting in a self-driving vehicle that would bring them to their destination reliably, safely and in a comfortable manner.



Fig. 14 Live showcase at the Singapore Science Center Street Fair

6.2 Humans Are Unpredictable

Most path planners make a basic assumption, i.e., pedestrian movement will largely remain unchanged if there is no sudden perturbation to the overall state of the environment. This is a very safe assumption as people generally tend to have a certain destination they want to walk to and this destination largely remains unchanged. This predicated behavior, however, was not entirely observed at the Science Centre. People behaved differently with the vehicle upon learning that it was driving all by itself. On the one hand, there were those who were unsure if the vehicle would knock them down as there was no human controller. And on the other hand, there were those who were jumping directly in front of the vehicle, from all possible angles, trying to find a scenario where the autonomous vehicle would fail. Thankfully, the implemented dynamic safety zone in the vehicle worked as intended.

It was an interesting phenomenon that people behaved very differently upon knowing that the vehicle is self-driven. Most path planners would assume that pedestrians would take a more conservative approach in their movement in the presence of a vehicle. However, what we observed was a number of pedestrians acting in a more “aggressive” manner—their motion would abruptly change and intersect with the vehicle’s path. It was also observed that this unpredictability in human behavior usually happened to people who have never seen a driverless vehicle before.

6.3 Safety Proven

Allowing the public to interact with the vehicle both as a passenger and as a pedestrian helped to reinforce the reliability of the system. Using laser sensors with a range of

50 m, the vehicle is able to have an early detection of obstacles. In a dynamic safety zone, the speed of the vehicle is monitored in relation to the presence of near-by pedestrians. This allows the vehicle to stop in a gradual manner in the presence of an approaching obstacle. The dynamic safety zone also allows the vehicle to automatically adapt to the speed of the moving crowd, allowing the vehicle to still move as opposed to being completely stationary. The dynamic safety zone also allows passengers in the vehicle to not experience “vehicle jerkiness” in a crowded environment.

6.4 The Next Step

The main thrust of this research work is to use driverless vehicles for Mobility on Demand. One of the key elements to make this a reality is to bridge the gap of between public acceptance of driverless vehicles and the state-of-the-art enabling technologies for driverless vehicles. The experience at the Singapore Science Centre definitely aided to not only raise public awareness of driverless vehicles, but also public acceptance of it through experiential interaction with the driverless buggy. It is important to continually engage the public with regards to the development of autonomous vehicles. After all, the public will eventually be the end-users of such a system.

Ongoing research work aims at enabling the driverless vehicle with greater mobility within a dense pedestrian environment while maintaining an overall satisfactory user experience.

7 Conclusion

In this chapter, we introduced vehicle autonomy using cooperative perception to enable Mobility-on-Demand system. Our MoD system currently operates in a restricted section of the National University of Singapore campus, where autonomous personal transporters have been autonomously servicing numerous requests from visitors and dignitaries during the course of multiple demonstrations. The rich sensing and actuating capabilities of autonomous vehicles can improve traffic flow and safety of MoD services. We proposed and demonstrated driving assistance system and autonomous driving system using cooperative perception, which include a see-through/lifted-seat/satellite view, a see-through forward collision warning, and overtaking/lane changing assistance. All these functions and systems are in operation with a number of autonomous vehicles in the NUS campus road. We have extended the test road to the outside of the campus.

References

1. Mitchell WJ, Borroni-Bird CE, Burns LD (2010) *Reinventing the automobile: personal urban mobility for the 21st century*. MIT Press, Cambridge
2. Car2Go. [Online]. Available: <http://www.car2go.com/>
3. SMOVE. [Online]. Available: <http://smove.sg/>
4. Shaheen SA, Guzman S, Zhang H (2010) Bikesharing in Europe, the Americas, and Asia. *Transp Res Res J Transp Res Board* 2143(1):159–167
5. Bouraoui L, Boussard C, Charlot F, Holguin C, Nashashibi F, Parent M, Resende P (2011) An on-demand personal automated transport system: the citymobil demonstration in La Rochelle. In: *IEEE Intelligent vehicles symposium (IV)*, pp 1086–1091
6. Kim S-W, Qin B, Chong ZJ, Shen X, Liu W, Ang MH Jr, Frazzoli E, Rus D (2014) Multivehicle cooperative driving using cooperative perception: design and experimental validation. In: *IEEE Transaction on Intelligent Transportation Systems*
7. Quigley M, Conley K, Gerkey B, Faust J, Foote T, Leibs J, Wheeler R, Ng AY (2009) Ros: an open-source robot operating system. In: *ICRA workshop on open source software*, vol 3. no 2
8. Chong ZJ, Qin B, Bandyopadhyay T, Wongpiromsarn T, Rebsamen B, Dai P, Kim S-W, Ang MH Jr, Hsu D, Rus D, Frazzoli E (2012) Autonomy for mobility on demand. In: *IEEE/RSJ international conference Intelligent Robots and systems*, Oct 2012
9. Kim S-W, Chong ZJ, Qin B, Shen X, Cheng Z, Liu W, Ang MH Jr (2013) Cooperative perception for autonomous vehicle control on the road: motivation and experimental results. In: *IEEE/RSJ international conference intelligent Robots and systems*, Nov 2013
10. Bandyopadhyay T, Jie CZ, Hsu D, Ang MH Jr, Rus D, Frazzoli E (2012) Intention-aware pedestrian avoidance. In: *International symposium on experimental robotics*, June 2012
11. Qin B, Chong Z, Bandyopadhyay T, Ang MH, Frazzoli E, Rus D (2012) Curb-intersection feature based Monte Carlo localization on urban roads. In: *Robotics and automation (ICRA), 2012 IEEE international conference on*, pp 2640–2646
12. Qin B, Lie W, Shen X, Chong ZJ, Bandyopadhyay T, Ang MH Jr, Frazzoli E, Rus D (2013) A general framework for road marking detection and analysis. In: *16th international IEEE conference on intelligent transportation systems (ITSC)*
13. Liu W, Kim S-W, Chong ZJ, Shen X, Ang MH Jr (2013) Motion planning using cooperative perception on urban road. In: *IEEE international conferences on Cybernetics and intelligent systems, and robotics, automation and mechantronics*
14. Thrun S, Burgard W, Fox D (2005) *Probabilistic robotics*. MIT Press, Cambridge
15. Thrun S (2000) A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In: *Proceedings of the robotics and automation, 2000, ICRA'00, IEEE international conference on*, vol 1. pp 321–328
16. Baldwin I, Newman P (2012) Road vehicle localization with 2d push-broom lidar and 3d priors. In: *Robotics and automation (ICRA), 2012 IEEE international conference on*, pp 2611–2617
17. Nüchter A, Lingemann K, Hertzberg J, Surmann H (2007) 6D SLAM—3D mapping outdoor environments. *J Field Robot* 24(8–9):699–722
18. Harrison A, Newman P (2008) High quality 3d laser ranging under general vehicle motion. In: *Robotics and automation, (2008) ICRA 2008, IEEE international conference on*, pp 7–12
19. Kohlbrecher S, von Stryk O, Meyer J, Klingauf U (2011) A flexible and scalable slam system with full 3d motion estimation. In: *IEEE international symposium of safety, security, and rescue robotics (SSRR)*, (2011) on, pp 155–160
20. Weingarten J, Gruener G, Siegwart R (2003) A fast and robust 3d feature extraction algorithm for structured environment reconstruction. In: *International conference on advanced robotics*
21. Rezaei S, Guivant J, Nieto J, Nebot EM (2004) Simultaneous information and global motion analysis (SIGMA) for car-like robots. In: *Proceedings of IEEE international conference of the robotics and automation, 2004, ICRA'04. 2004*, vol 2. pp 1939–1944
22. Kümmerle R, Triebel R, Pfaff P, Burgard W (2008) Monte Carlo localization in outdoor terrains using multilevel surface maps. *J Field Robot* 25(6–7):346–359

23. Velodyne Lidar. [Online]. Available: <http://velodynelidar.com/>
24. Klasing K, Althoff D, Wollherr D, Buss M (2009) Comparison of surface normal estimation methods for range sensing applications. In: IEEE international conference on Robotics and automation, (2009) ICRA'09, pp 3206–3211
25. Berkmann J, Caelli T (1994) Computation of surface geometry and segmentation using covariance techniques. *Pattern Anal Mach Intell IEEE Trans* 16(11):1114–1116
26. Shakarji CM et al (1998) Least-squares fitting algorithms of the nist algorithm testing system. *J Res-Natl Inst Stand Technol* 103:633–641
27. Rusu RB (2010) Semantic 3d object maps for everyday manipulation in human living environments. *KI-Künstliche Intell* 24(4):345–348
28. Muja M, Lowe DG (2009) Fast approximate nearest neighbors with automatic algorithm configuration. In: *VISAPP* (1), pp 331–340
29. Rusu RB, Cousins S (2011) 3d is here: point cloud library (pcl). In: IEEE international conference on Robotics and automation (ICRA), 2011, pp 1–4
30. ROS AMCL. [Online]. Available: <http://www.ros.org/wiki/amcl>
31. Tipaldi GD, Arras KO (2010) Flirt-interest regions for 2d range data. In: IEEE international conference on Robotics and automation (ICRA), 2010, pp 3616–3622
32. Kaess M, Ranganathan A, Dellaert F (2008) Isam: incremental smoothing and mapping. *Robot IEEE Trans* 24(6):1365–1378
33. Kim S-W, Gwon G-P, Choi S-T, Kang S-N, Shin M-O, Yoo I-S, Lee E-D, Seo S-W (2012) Multiple vehicle driving control for traffic flow efficiency. In: IEEE intelligent vehicles symposium, pp 462–468, June 2012
34. Besl PJ, McKay ND (1992) Method for registration of 3-d shapes. In: *Robotics-DL tentative, international society for optics and photonics*, pp 586–606
35. Yang C, Medioni G (1992) Object modelling by registration of multiple range images. *Image vis comput* 10(3):145–155
36. Olson EB (2009) Real-time correlative scan matching. In: IEEE international conference on Robotics and automation, ICRA'09, pp 4387–4393
37. Mallot HA, Blthoff HH, Little J, Bohrer S (1991) Inverse perspective mapping simplifies optical flow computation and obstacle detection. *Biol cybern* 64(3):177–185
38. Claraco JLB (2008) Development of scientific applications with the mobile robot programming toolkit., Machine Perception and Intelligent Robotics Laboratory, University of Málaga. MRPT reference book, Málaga
39. Dagan E, Mano O, Stein GP, Shashua A (2004) Forward collision warning with a single camera. In: *Intelligent vehicles symposium, 2004 IEEE*, pp 37–42
40. Parasuraman R, Hancock P, Olofinboba O (1997) Alarm effectiveness in driver-centred collision-warning systems. *Ergonomics* 40(3):390–399
41. Karaman S, Walter MR, Perez A, Frazzoli E, Teller S (2011) Anytime motion planning using the RRT*. In: IEEE international conference on Robotics and automation, May 2011
42. Karaman S (2011) Anytime motion planning using the RRT*. In: IEEE international conference on Robotics and automation (ICRA), pp 1478–1483