Remote Sens. **2013**, *5*, 3662-3680; doi:10.3390/rs5083662

OPEN ACCESS

Remote Sensing

ISSN 2072-4292

www.mdpi.com/journal/remotesensing

Article

Hybrid Map-Based Navigation Method for Unmanned Ground Vehicle in Urban Scenario

Yuwen Hu [1], Jianwei Gong [1,2,*], Yan Jiang [1,2], Lu Liu [1], Guangming Xiong [1,2] and Huiyan Chen [1]

[1] Intelligent Vehicle Research Center, Beijing Institute of Technology, 5 South Zhongguancun Street, Beijing 10081, China; E-Mails: huyuwen1984@gmail.com (Y.H.); jiangyan@bit.edu.cn (Y.J.); liulu6@sina.com (L.L.); xiongguangming@bit.edu.cn (G.X.); chen_h_y@263.net (H.C.)

[2] Key Laboratory of Biomimetic Robots and Systems, Ministry of Education, 5 South Zhongguancun Street, Beijing 10081, China

* Author to whom correspondence should be addressed; E-Mail: gongjianwei@bit.edu.cn; Tel.: +86-10-6891-2329.

Received: 31 May 2013; in revised form: 17 July 2013 / Accepted: 17 July 2013 / Published: 25 July 2013

Abstract: To reduce the data size of metric map and map matching computational cost in unmanned ground vehicle self-driving navigation in urban scenarios, a metric-topological hybrid map navigation system is proposed in this paper. According to the different positioning accuracy requirements, urban areas are divided into strong constraint (SC) areas, such as roads with lanes, and loose constraint (LC) areas, such as intersections and open areas. As direction of the self-driving vehicle is provided by traffic lanes and global waypoints in the road network, a simple topological map is fit for the navigation in the SC areas. While in the LC areas, the navigation of the self-driving vehicle mainly relies on the positioning information. Simultaneous localization and mapping technology is used to provide a detailed metric map in the LC areas, and a window constraint Markov localization algorithm is introduced to achieve accurate position using laser scanner. Furthermore, the real-time performance of the Markov algorithm is enhanced by using a constraint window to restrict the size of the state space. By registering the metric maps into the road network, a hybrid map of the urban scenario can be constructed. Real unmanned vehicle mapping and navigation tests demonstrated the capabilities of the proposed method.

Keywords: hybrid map; unmanned ground vehicle; topological map; metric map; simultaneous localization and mapping; Markov localization; laser scanner

1. Introduction

Autonomous-driving technologies have attracted considerable academic and industrial interests in recent years. Navigating an unmanned ground vehicle (UGV) driving through urban areas is a difficult task. The UGV must know exactly where it is in dynamic traffic scenarios. The accuracy of the location information directly impacts the safety and control stability of the UGV.

The localization methods of autonomous driving can be divided into satellite signal-based solutions and map matching solutions. In satellite signal-based solutions, GPS is the most common positioning way. However, because of the multipath effect and the occlusion of satellite signals caused by buildings, trees or clouds, the ordinary GPS equipment with normal accuracy cannot meet the requirement of the positioning system. Therefore, the differential GPS (DGPS) or GPS-inertial navigation system (INS) positioning system comes into use. In the 2007 Defense Advanced Research Projects Agency (DARPA) Urban Challenge, most of the unmanned vehicles were equipped with costly DGPS-based navigation systems [1,2]. The expensive equipment will increase the cost of the autonomous driving vehicle. When in a big city, DGPS positioning signals cannot cover everywhere. Moreover, when there are high buildings, big trees, bridges or tunnels, it is impossible for the positioning system to obtain even ordinary GPS satellite signals.

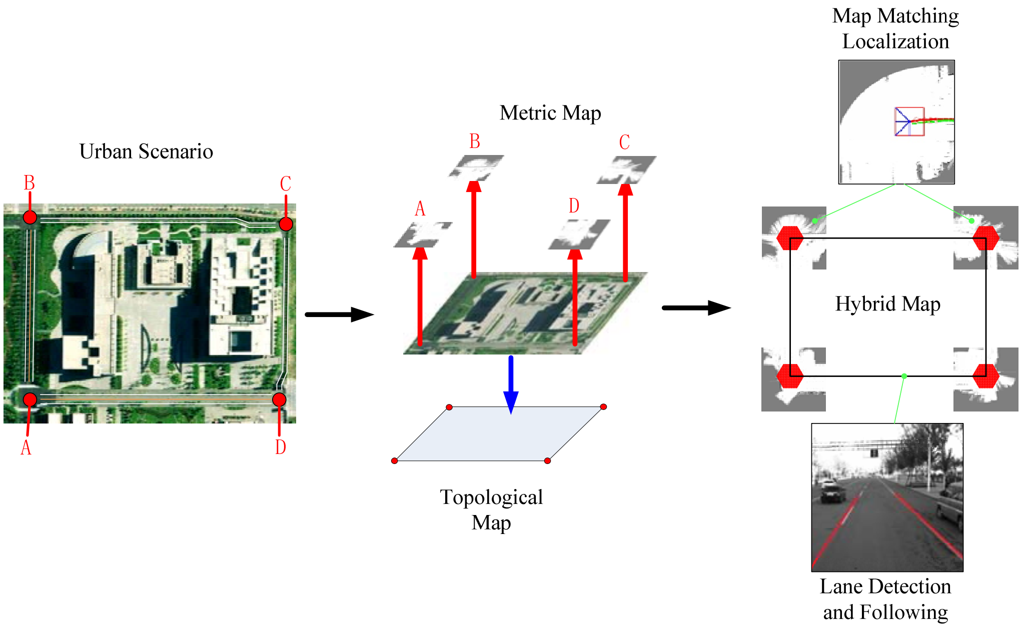
For the map matching-based method, an unmanned vehicle gets its position by comparing the environment map with real-time perception data from mounted sensors, such as laser scanners or cameras [3–7]. If the map is large and accurate enough, map matching-based approaches show better stability than the GPS-based method in the urban scenario. Such an environment map can be obtained by various approaches [8–10]. However, the tremendous size and the huge cost of a detailed map of a whole city is a big problem. Storage and computational complexity, due to a massive amount of map data, will affect the efficiency of positioning system. In early 2013, the Google self-driving car was reported to be commercially available to customers in five to seven years [11]. However, how to obtain the accurate position of the self-driving car is still a challenge. In this kind of map matching approach, the determination of the autonomous vehicle relies on very detailed maps of the roads and terrain. Before sending the self-driving car on a road test, a human driver must drive along the route to gather data about the environment. Then, the autonomous vehicle can compare the data acquired from the perception sensors to the previously recorded map data [12]. Therefore, a very detailed, accurate and high-cost digital map along the driving routes is essential. The high cost refers to not only the map building, but the calculation load of the matching process during autonomous driving.

From the experiences of participation in the Intelligent Vehicle Future Challenge competitions of China since 2009 [13], we found that a UGV could take advantage of a variety of positioning methods, rather than a particular one in the entire self-driving journey in urban areas. In most cases, the UGV drives on the structured roads with lanes and does not need precise position data. Lanes, road curbs and global path points ahead provide the direction for the UGV that drives itself like water flows in a pipe.

Other environment perception technologies, such as traffic sign detection, can offer enough information to constrain the vehicle's movement. However, when the vehicle enters unstructured areas, such as intersections and other areas without guide lanes, accurate localization information becomes essential to its self-driving. If the detailed map of these unstructured areas is known, the vehicle could locate itself using the map matching method of positioning to follow a pre-planned path. Therefore, for urban self-driving, we only need to build environment maps for unstructured areas and regard structured roads with lanes as channels between the unstructured maps. This approach makes the urban map become a hybrid map integrated with the local environment map and topological map. Compared with an ordinary digital map, the hybrid map is able to provide enough information without substantially increasing data quantity. It will make the urban driving task for a UGV not rely on costly positioning equipment for navigation.

Based on the above analysis, our aim is to build a low-cost hybrid map-based navigation system for a UGV self-driving in urban scenarios, as shown in Figure 1. The research is focused on the hybrid map building process and localization in the local map. In the following sections, we will describe the process of using OpenStreetMap [14] for extracting the topological structure of a city map and using a laser scanner to construct a local grid map in detail. Next, an improved Markov localization method is proposed to increase the real-time positioning. Then, we introduce the Beijing Institute of Technology (BIT) unmanned ground vehicle used in this work. Furthermore, the hybrid map mapping and map matching localization in the local map are proposed. At the end, we will discuss the advantages of the hybrid map-based navigation method and future work.

Figure 1. Topological-metric hybrid map for navigation.

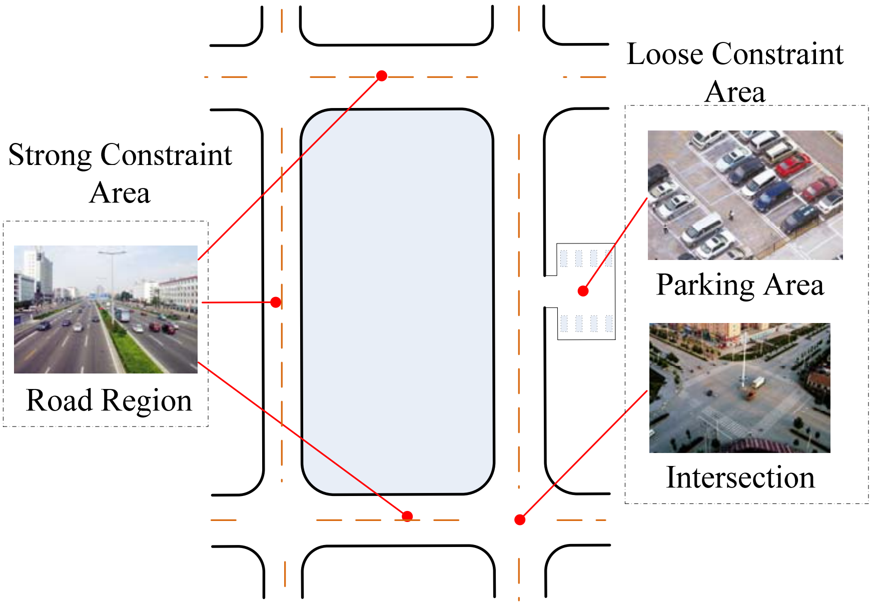


2. Construction of the Hybrid Map

In order to reduce the dependence on a high-precision positioning system and the amount of map data, we divide the whole urban space into two different areas. Because the traffic lanes can provide forward direction for a UGV, the district with visible lanes is defined as a strong constraint (SC) area. On the other hand, the district without lanes is defined as a loose constraint (LC) area. As shown in

Figure 2, the SC areas mainly consist of road regions, and the LC areas may include intersections, parking lots, and so on. Traffic lanes are the primary difference to distinguish an SC area from an LC area. Dividing urban scenarios into these two categories allows us to use a low accurate positioning method in SC areas and precise localization mapping in LC areas.

Figure 2. Definitions of strong constraint area and loose constraint area.



2.1. Topological Map of Urban Scenario

The global path for a UGV self-driving navigation in either SC areas or LC areas is described in the topological map. Basic elements of the topological map include nodes and edges. Nodes represent significant places in the map, and edges show the relationship between two nodes. In this work, the nodes are defined as GPS positions of LC areas, and the edges represent certain roads between two adjacent nodes.

The common way to get the GPS positions is to drive the vehicle with a GPS receiver and record key waypoints on the routes. In a small area, the GPS data collection work can be done manually. However, when building a whole city map of urban size, the work may be too tedious and time-consuming. Therefore, we extract the urban topological structure from OpenStreetMap (OSM) [14]. The OSM is a well-known project that provides user-generated street maps. The GPS data downloaded from the OSM website is in XML form, organized in a treelike structure.

In the XML file structure shown in Figure 3, the root element OSM illustrates that this is a file downloaded from the OpenStreetMap website. There are four elements under this root, including bound, node, way and relation. Node elements are shown in orange. Attributes from a node provide latitude, longitude and the ID of the node. Some nodes may have sub-elements, which include more knowledge about this node, such as whether it is an intersection or whether it belongs to a building, bus stop, restaurant, *etc.* If a node does not have a sub-element, that means it belongs to a road or highway instead of an intersection. The attribute of way element indicates its identification, and the sub-elements of way include its category, traffic rules and member nodes information.

Figure 3. Treelike structure of OpenStreetMap data.

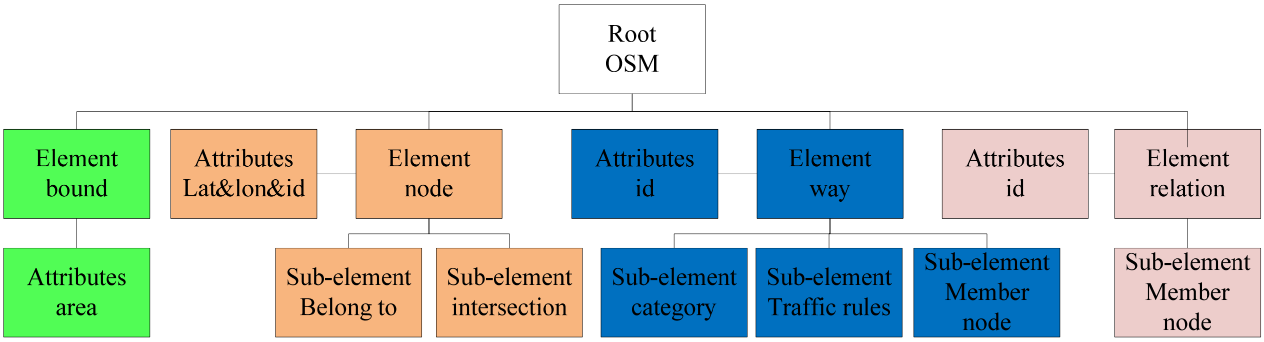
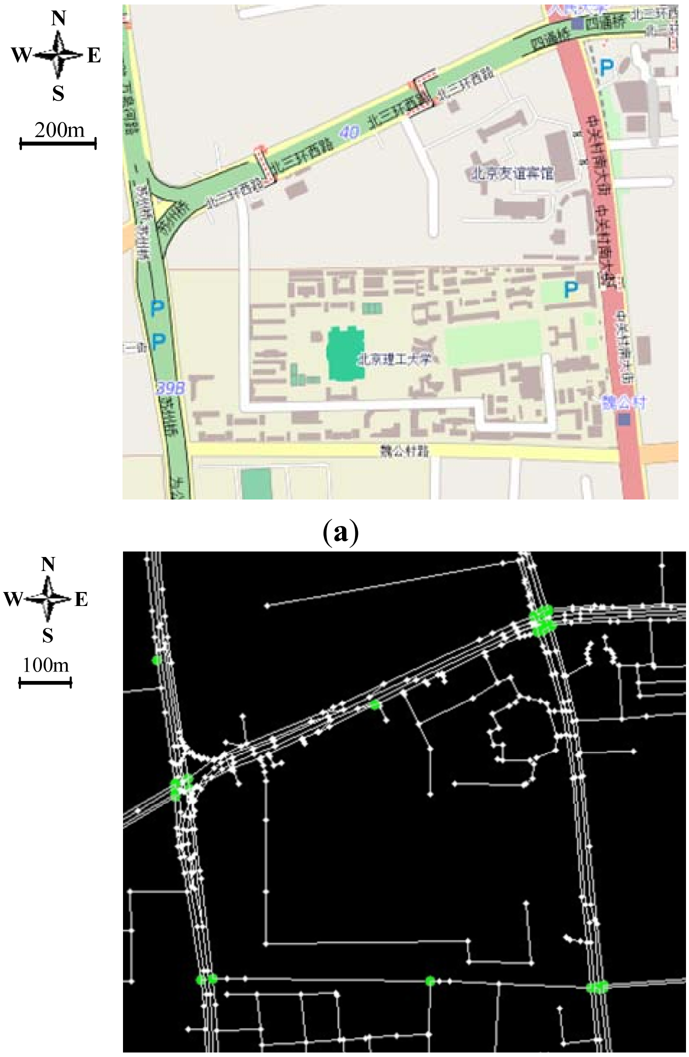


Figure 4. Illustration of building topological map by OpenStreetMap (OSM). (**a**) Urban scenario. (**b**) Corresponding topological map extracted from the OSM. Nodes can provide GPS data, and highlighted green notes are intersections. White lines show roads between connected nodes.



(**a**)

(**b**)

Figure 4a shows urban scenario while Figure 4b shows the topological map extracted from OSM in the scale of Figure 4a. We only extract the points from the highway. These points are the nodes of the topological map. Furthermore, based on the sub-element information, the points with the description of "traffic light" are defined as the locations of intersections, as the green points in Figure 4b. Additionally, these points indicate the position of LC areas where the local metric map should be built for self-driving navigation.

2.2. Metric Map of Loose Constraint Area

For LC areas, there are no lanes to guide the UGV. When the vehicle enters into these areas, either the sparse GPS waypoints in the topological map or the positioning error could cause unstable driving behavior. Therefore, the primary task is to add more environmental constraints to restrict the vehicle movement. The constraints are added in the form of metric map. Additionally, the local metric map can be used for localization, which will be discussed in Section 3.

For environment mapping, there have been many impressive results shown in recent years [15–19]. In this work, the occupancy grid mapping method is applied in local metric map building. To build an occupancy grid map, the environment is discretized into grids, and the value of each cell is allocated in accordance with the probability that the cell is occupied by obstacles. The grid map is built according to measurements collected by a laser scanner mounted on the vehicle. Additionally, the simultaneous localization and mapping algorithm is employed in real-time dynamic outdoor environment mapping [20]. The mapping algorithm includes three basic components: map update, pose estimation and moving object elimination.

2.2.1. Map Update

Given that the observation, z_t, is the measurement of the laser scanner at time *t*, and x_t is the vehicle's corresponding pose, the posterior probability of the occupancy grid map is governed by:

$$P(m \mid x_{1:t}, z_{1:t}) = \frac{P(z_t \mid x_{1:t}, z_{1:t-1}, m) P(m \mid x_{1:t-1}, z_{1:t-1})}{P(z_t \mid x_{1:t}, z_{1:t-1})} \tag{1}$$

Assume the environment is static. When the map, *m*, is known, the measurement, z_t, is independent from the vehicle's historical state, $x_{1:t-1}$, and measurements, $z_{1:t-1}$, Equation (1) can be rewritten as:

$$P(m \mid x_{1:t}, z_{1:t}) = \frac{P(z_t \mid x_t, m) P(m \mid x_{1:t-1}, z_{1:t-1})}{P(z_t \mid x_{1:t}, z_{1:t-1})} \tag{2}$$

Based on reference [21], Equation (2) can be further simplified into the form as:

$$P(m \mid x_{1:t}, z_{1:t}) = \frac{S}{1+S} \tag{3}$$

where:

$$S = \frac{P(m \mid z_t, x_t)}{1 - P(m \mid z_t, x_t)} \cdot \frac{P(m \mid x_{1:t-1}, z_{1:t-1})}{1 - P(m \mid x_{1:t-1}, z_{1:t-1})} \tag{4}$$

The inverse sensor model, $P(m|z_t, x_t)$, is determined based on the laser measurements as:

$$P(m_i \mid z_t, x_t) = \begin{cases} 0.8 & \text{occupied} \\ 0.4 & \text{free} \\ 0.5 & \text{unkown} \end{cases} \tag{5}$$

Equation (5) determines the measured occupied probability of each grid. The cell is occupied if it is hit by a laser point. The cell is free if it lies between the endpoint and the vehicle on the laser beam. If the cell is on a laser beam and behind the endpoint, its state is unknown. As the final occupancy probability of a cell is determined by a sequence of measurements, the value of a free cell in the sensor model has a remarkable impact on the generated grid map. Large value for a free cell can make the probability of a previously occupied cell decrease at a lower ratio when it becomes free. This is necessary if we need to preserve small obstacles, such as traffic guardrails, in the generated grid map. Such obstacles are likely to be invisible in sequential frames of detections, as a result of the vehicle motion and the angle resolution of the laser scanner.

2.2.2. Pose Estimation

Calculating a vehicle's pose is a maximum likelihood estimation problem in the local mapping method, which is finding the most likely pose of the vehicle in the configuration space. Given the posterior likelihood of the vehicle's pose at the last time step, $\hat{x}_{t-1}$, the *a priori* likelihood of the pose is predicted based on the vehicle's control inputs using $P(x_t|\hat{x}_{t-1}, u_t)$. After the vehicle makes a new measurement of the environment, its pose can be determined by maximizing the posterior likelihood:

$$\hat{x}_t = \arg\max_{x_t} \{P(z_t \mid x_t, m_{t-1}) \cdot p(x_t \mid \hat{x}_{t-1}, u_t)\} \tag{6}$$

where *m* is the grid map and *u* is the control inputs of the vehicle.

The pose estimation is resolved using a particle filter. Initially, particles are uniformly distributed in the area without any prior knowledge of the vehicle's position. A simplified velocity model is employed to predict the particle's motion:

$$\begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix} = \begin{bmatrix} \hat{x}_{t-1} \\ \hat{y}_{t-1} \\ \hat{\theta}_{t-1} \end{bmatrix} + \begin{bmatrix} \cos\hat{\theta}_{t-1} & 0 \\ \sin\hat{\theta}_{t-1} & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_t \\ w_t \end{bmatrix} \tag{7}$$

where the motion of the vehicle between two measurements is assumed to be straight and θ is the heading angle of the vehicle. The measurement model used in the update procedure of the particle filter is [20]:

$$P(z_t \mid x_t, m_{t-1}) \propto \sum_{k=1}^{\mathrm{N}} P(m_{t-1}^k) \tag{8}$$

where m_{t-1}^k is the cell hit by the end-point of laser beam, *k*. N is the total number of laser scanner's data.

The local map method simplifies the map generation by decoupling pose estimation from a map update. This may introduce errors in the generated map, but the errors are acceptable.

2.2.3. Moving Obstacle Elimination

In the map update step, cells occupied with moving obstacles have to be removed in Equation (4), otherwise the dynamic elements can make the Markov assumption invalid. By comparing the posterior likelihood, $p(m_i|z_t^k)$, of an occupied cell, i, at the direction of a laser beam, k, with the likelihood of occupied cells in the sensor model, the cell could be classified into static or dynamic [19]. If the $p(m_i|z_t^k)$ is smaller than the occupied threshold, the cell will be treated as dynamic and not counted in Equation (3) for the map update. As only static cells are included in the map update, Equation (4) can be written as:

$$S = \frac{P(m \mid z_{t,\text{static}}, x_t)}{1 - P(m \mid z_{t,\text{static}}, x_t)} \cdot \frac{P(m \mid x_{1:t-1}, z_{1:t-1,\text{static}})}{1 - P(m \mid x_{1:t-1}, z_{1:t-1,\text{static}})} \tag{9}$$

2.3. Metric Map Registration

To integrate the local metric map with the urban topological map, we collect some GPS positioning data in the area during a local metric map mapping process. After the local map is established, only one GPS location in each local map is selected as a registration point, which can be used to register the local metric map to a nearest node in the topological map. The registration point's position is also stored in the corresponding node.

3. Hybrid Map-Based Navigation

In Section 2, we have constructed the topological map of the urban scenario and the local metric maps of the LC areas. In the SC area, our previous works can get enough information, including detections of lane markings [22], traffic lights [23] and signs [24], to provide the direction and the speed limit for the UGV. Because the GPS signal only provides global path information, the GPS receiver used in our UGV can be just an ordinary one with low positioning accuracy.

There are lots of map-matching-based localization methods. In reference [25], light, humidity and other data are used for radio frequency (RF)-based localization. Furthermore, Markov localization [26] is widely used. Based on the discretization of a vehicle's state space, the position is recursively determined by calculating the posterior probability. The original Markov localization method needs to maintain a likelihood matrix of the state space with the same size of the entire metric map. Although the probabilities of most elements in the matrix are negligible, they have to be updated every time when a new measurement is made. In this research, assuming that the vehicle's true position is close to its GPS indications, we can adopt a small likelihood matrix instead of the whole state space to reduce the computational cost. The likelihood matrix is displayed as a constraint window with its center fixed on the vehicle's GPS position. Therefore, the window will scroll forward with the vehicle's movement through the map.

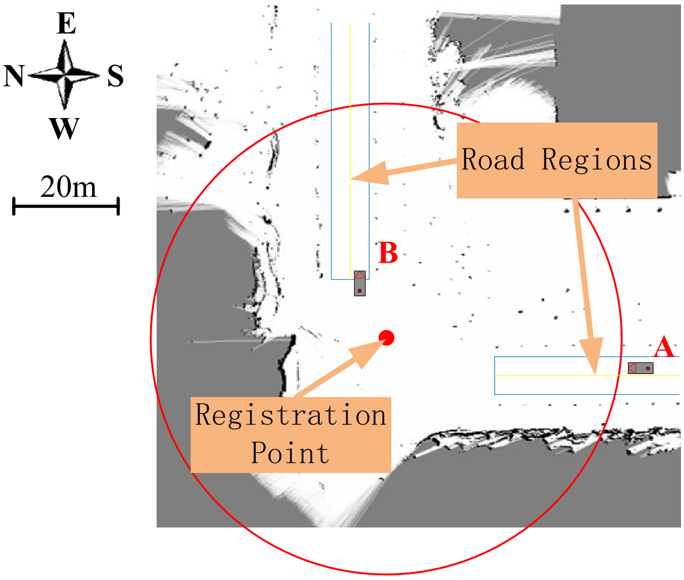
3.1. Navigation in the Hybrid Map

3.1.1. Switch from Strong Constraint (SC) Area to Loose Constraint (LC) Area

The corresponding metric map will be loaded into memory when the vehicle is approaching a node in the digital map. Map matching-based localization will be launched when the vehicle approaches the scope of the metric map and registers the vehicle with the local environment, as location A in Figure 5.

Since the localization in the metric map is independent with the global position of the vehicle and the metric map, the position of the metric map in the global topological map is not required to be very accurate.

Figure 5. Navigation methods switching.



3.1.2. Switch from Loose Constraint (LC) Area to Strong Constraint (SC) Area

Following the global path in the topological map, the autonomous vehicle will drive into the expected edge path of the digital map when it leaves the local metric map. The local metric map contains not only the static structure of an area, which is used for localization and obstacle avoidance, but also the connections with edges in the digital map. This information is recorded when the map is generated. The connection with the edge on the global path can be used as the target state in the vehicle's configuration space for local path planning. Once strong constrained elements are detected, when the vehicle is driving along the local path, the driving mode will be switched to strong constrained mode, for example, as shown in Figure 5; when the vehicle arrives at location B, the vehicle will detect the road lanes.

3.2. Localization in Loose Constraint (LC) Area

The original Markov localization method needs to maintain a likelihood matrix with the same size of the entire metric map. Except for elements closed to the true position, the probabilities of most elements in the matrix are negligible, but they have to be updated every time after a new measurement is made. This updated probability can be expressed as $Bel(L_t = l|s_1,\dots,s_t,a_1,\dots,a_{t-1})$, where L_t denotes the vehicle state, l denotes one of the discrete states and t denotes the time. In this work, s and a indicate the data from the odometer and laser scanner, respectively. Based on the Bayesian theorem, this expression can be transformed into:

$$Bel(L_t = l \mid s_1,\dots,s_t,a_1,\dots,a_{t-1}) = \frac{P(s_t \mid L_t = l, s_1,\dots,s_{t-1},a_1,\dots,a_{t-1})P(L_t = l \mid s_1,\dots,s_{t-1},a_1,\dots,a_{t-1})}{P(s_t \mid s_1,\dots,s_{t-1},a_1,\dots,a_{t-1})} \quad (10)$$

For the simplification of real-time computation, with the Markov assumption of the vehicle's movement and the independence of environment perception on the vehicle's previous position, the expression can be further simplified as:

$$Bel(L_t = l \mid s_1,...,s_t,a_1,...,a_{t-1}) = \eta P(s_t \mid L_t = l)P(L_t = l \mid L_{t-1},a_{t-1}) \tag{11}$$

where $P(L_t = l|L_{t-1},a_{t-1})$ and $P(s_t|L_t = l)$ represent the influence of movement model and perception model on the posterior probability of the discrete state, respectively. η is a normalized coefficient.

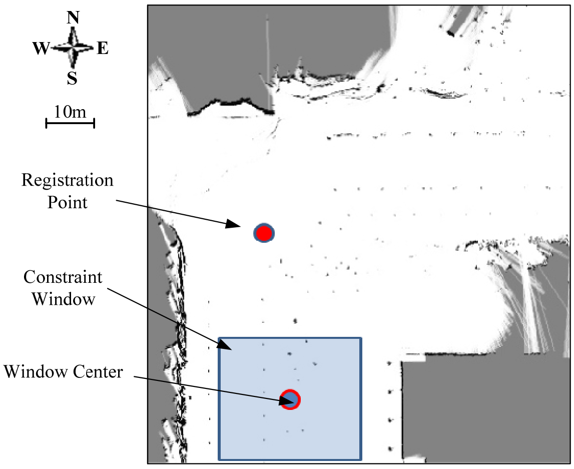
The Markov localization method is generally used for indoor robotics. When it is applied in a large space, the main problem is low positioning frequency caused by increased state space for computation. With the assumption that the vehicle's true position is close to its GPS indications, we propose a window constraint Markov localization (WCML) method, using a small likelihood matrix to reduce the computational cost in the Markov localization process.

3.2.1. Constraint Window

As shown in Figure 6, the size of the constraint window (the blue square) is 15 m × 15 m, and its center is fixed on the vehicle's GPS position. The scope of the constraint window is set to be much larger than normal GPS positioning error. As a result, the true state of the UGV can be covered by the window. Once the corresponding metric map is loaded, the vehicle pose will be transformed into the local coordinate of the map. The relative position of the UGV, which is also the center of the constraint window, can be expressed in the local coordinate in the metric map as:

$$\begin{bmatrix} x_{\mathrm{w}} \\ y_{\mathrm{w}} \end{bmatrix}_{t=0} = \frac{1}{r} \cdot \begin{bmatrix} x_{\mathrm{gv}} - x_{\mathrm{gl}} \\ y_{\mathrm{gv}} - y_{\mathrm{gl}} \end{bmatrix} \tag{12}$$

Figure 6. Constraint window in the grid map of the loose constraint (LC) area.



3.2.2. Window Constraint Markov Localization

In the window constraint Markov localization process, the states needed to be updated in the state space are limited inside the window. Therefore, the movement model of Markov localization can be rewritten as:

$$P(L_t = l \mid L_{t-1}, a_{t-1}) = \sum_{l' \in W_{t-1}} P(L_t = l, l \in W_t \mid L_{t-1} = l', a_{t-1}) Bel(L_{t-1} = l') \tag{13}$$

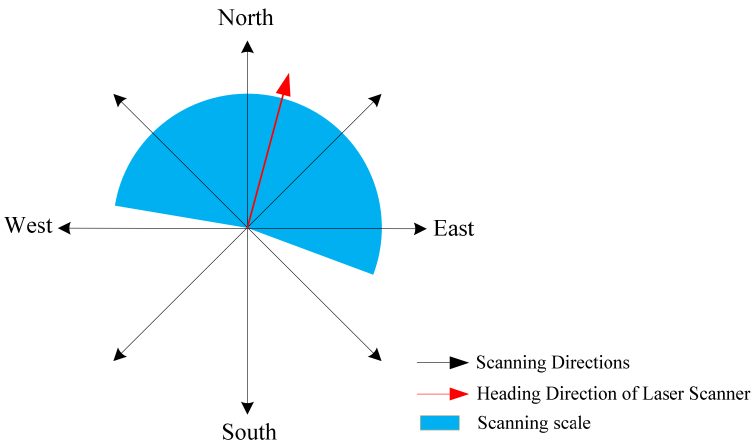
where W_t denotes the state space in the window scale at time t. We use $Bel(L_{t-1} = l')$ instead of $Bel(L_{t-1} = l' | s_1, \ldots, s_{t-1}, a_1, \ldots, a_{t-2})$ for the expression. Equation (13) describes the state transferring probability matrix. In this work, the matrix is designed as a Gaussian distribution, and its parameters are determined by the error of the odometer mounted on the UGV.

Another part of Equation (11) is the perception model, $P(s_t|L_t = l)$, which determines the similarity between the laser data and the occupancy grid map (OGM) at state l. It is defined as:

$$P(S_t \mid L_t = l) = \prod_{i=1}^{K} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(S_t^i{}' - S_t^i)^2}{2\sigma^2}} \tag{14}$$

where S_t is the distances from the laser scanner and S_t' is the desired value measured from the map with the assumption $L_t = l$ using the ray tracing method. K is the number of scanning directions of the laser scanner. For computational consideration, the scanning directions are previously chosen, and all the values used in the nonlinear computation, such as sine and cosine, are stored for real-time calculation. As shown in Figure 7, there are eight total directions in the local coordinates of the metric map. Only four or five directions of them are selected to be used in each localization process.

Figure 7. Selection of spare directions for ray tracing. The directions in the scanning scale of the laser sensor are chosen for the ray tracing operation.

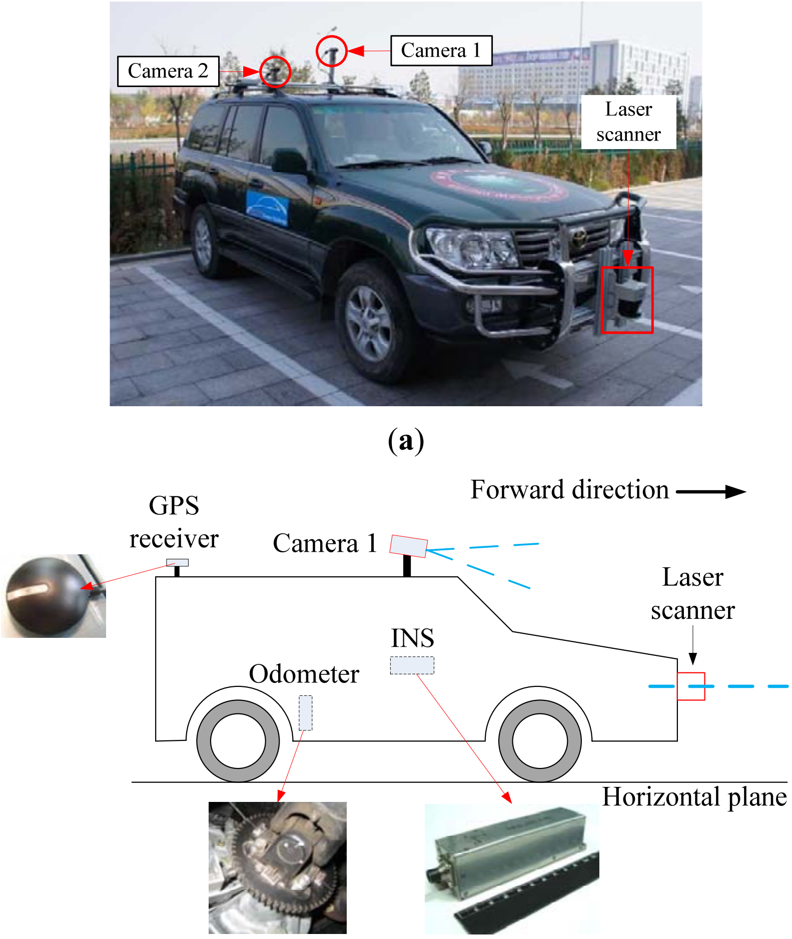


4. Experimental Platform and Results

4.1. The Beijing Institute of Technology Unmanned Ground Vehicle

The Beijing Institute of Technology unmanned ground vehicle (BITUGV) as shown in Figure 8a is set up based on the concept of low-cost design for urban driving [13,27]. The sensors equipped for environment detection include one horizontal laser scanner and one camera. In this research, the laser scanner is used for static and moving obstacle range detection in the map mapping and localization process. The laser scanner used in the experiment is LMS291. It has a detection range of 80 m in 0°–180°. Camera 1 is used for lane detection, and camera 2 is only for data recording.

Figure 8. The Beijing Institute of Technology Unmanned Ground Vehicle. (**a**) BITUGV and its environment sensors, including one laser sensor and two cameras. (**b**) Side view of the BITUGV and its localization equipment, including a global positioning system (GPS) receiver, an odometer and an inertial navigation system (INS).



(**a**)

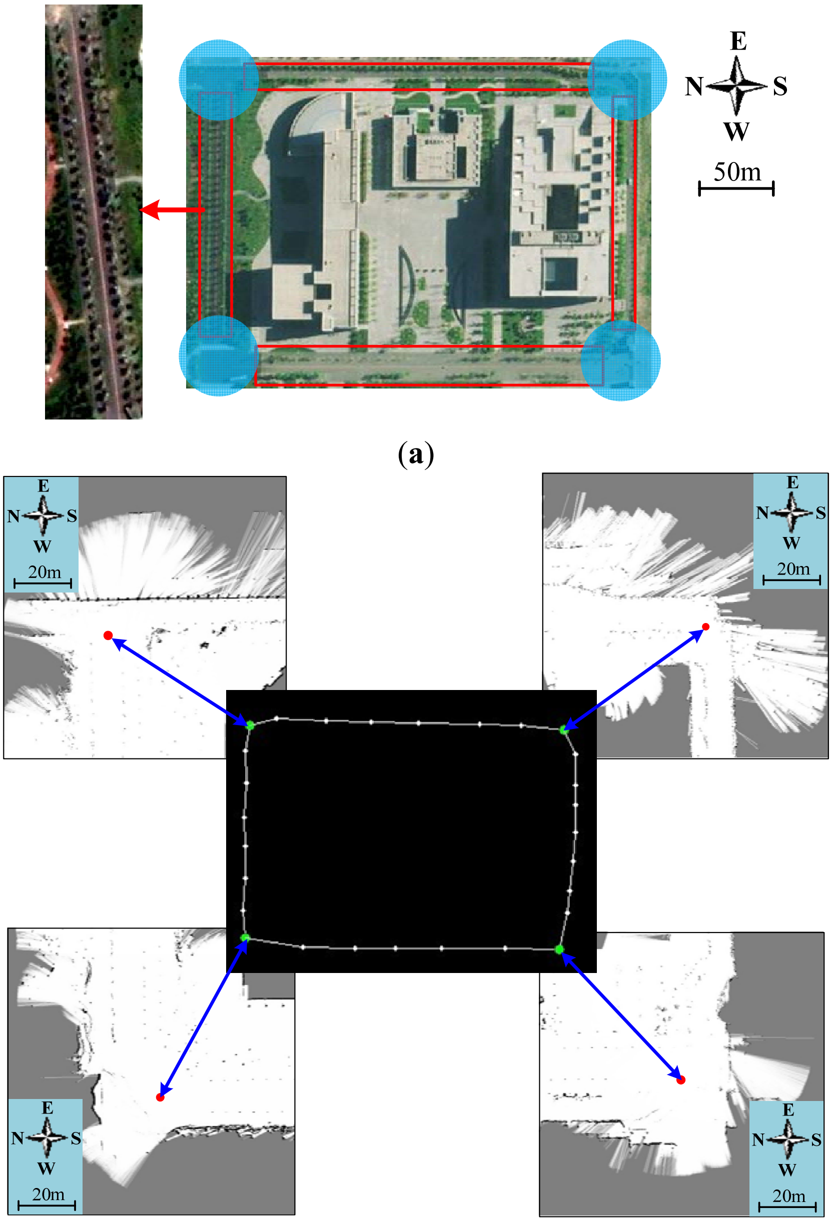
(**b**)

Other equipment for UGV positioning consists of one GPS receiver, one inertial navigation system (INS) and one odometer. The GPS receiver provides the position to locate the UGV in the road network. The INS mainly provides the heading direction of the UGV. The odometer records the distance the UGV traveled. It should be noted that the GPS receiver's accuracy we used is relatively low. The positioning error is about 3 m at least.

4.2. Results of Hybrid Map Construction

Figure 9a shows our experiment environment. The red rectangles in it describe the SC regions, and the blue circles indicate the 4 LC areas' positions. Figure 9b demonstrates the hybrid map of the experiment environment. The picture in the middle of Figure 9b is the topological structure of this scenario, and other pictures display the metric maps of 4 intersections in the environment. The red point in each metric map shows the registration point of each local map. In this 350 m × 240 m scenario, there could be 2,100,000 grids in a traditional metric map, with the grid size of 0.2 m × 0.2 m. However, using our hybrid map, only 4 local metric maps with 1,000,000 total grids are needed.

Figure 9. Metric-topological-based hybrid map of experiment environment. (**a**) Experiment scenario. The loose constraint areas include 4 intersections in blue circles, and the strong constraint areas include the roads in the red rectangles. (**b**) Hybrid map for unmanned ground vehicle navigation. The topological map constructed by GPS data is shown in the middle, with green points for intersections and white points on the road.



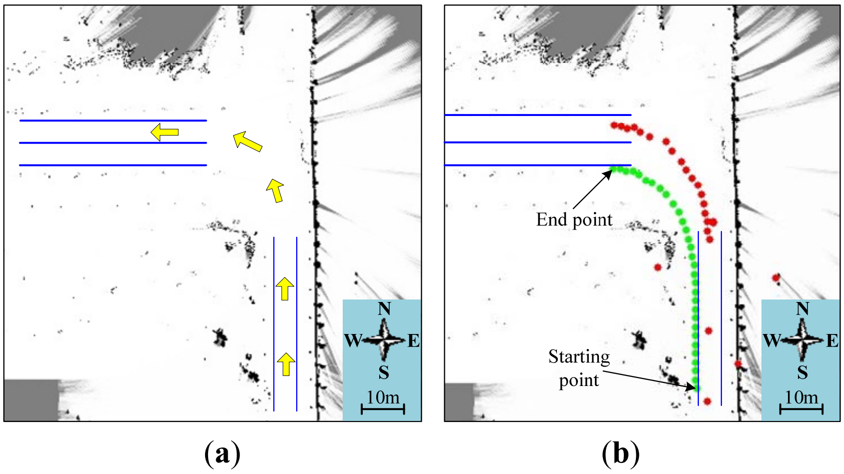
(**a**)

(**b**)

4.3. Results of Window Constraint Markov Localization (WCML) in the Loose Constraint Areas

Figure 10 is an experiment of the window constraint Markov localization (WCML) process in an intersection. During this process, the odometer provided the rough positioning data every 0.1 s and the GPS receiver provided the localization at 10 Hz frequency. The WCML method was launched if the vehicle moved more than 1 m. Figure 10a shows the metric map of this LC area, in which blue lines stand for lane markings and yellow arrows present the rough route of the vehicle. Figure 10b compares localization results from the odometer with the WCML method. Green points are localization information provided by odometer. The start point is decided by the relative position between the GPS position at that time and the registration point of the map. GPS positioning noise will cause a deviation from the actual vehicle position on the start point. Although the cumulative error of the odometer is small in such a short distance, it may result in dangerous consequences. Red dots represent the positioning results from the WCML method.

Figure 10. (**a**) Original scenario. Road regions are between each pair of adjacent traffic lanes shown as blue lines, and the vehicle rough route is shown as yellow arrows. (**b**) Localization results of odometer (green points) and the window constraint Markov localization (WCML) method (red points).



The quantitative analysis of the WCML process is summarized in Figure 11. The time consumption of each WCML localization process is lower than 0.1s, as shown in Figure 11a. Figure 11b shows the slowdown process of UGV velocity while driving through the intersection. Figure 11c,d illustrates the deviations of localization results between odometer and WCML method along the east (*dx*) and north (*dy*) direction separately. The differences gradually become constant, which means the WCML method provides a stable localization result after a period of time.

Figure 11. (**a**) Time consumptions of window constraint Markov localization process. (**b**) Vehicle velocities during experiment. (**c**) Localization deviations (*dx*) between the odometer and WCML along the east direction. (**d**) Localization deviations (*dy*) between the odometer and WCML along the north direction.

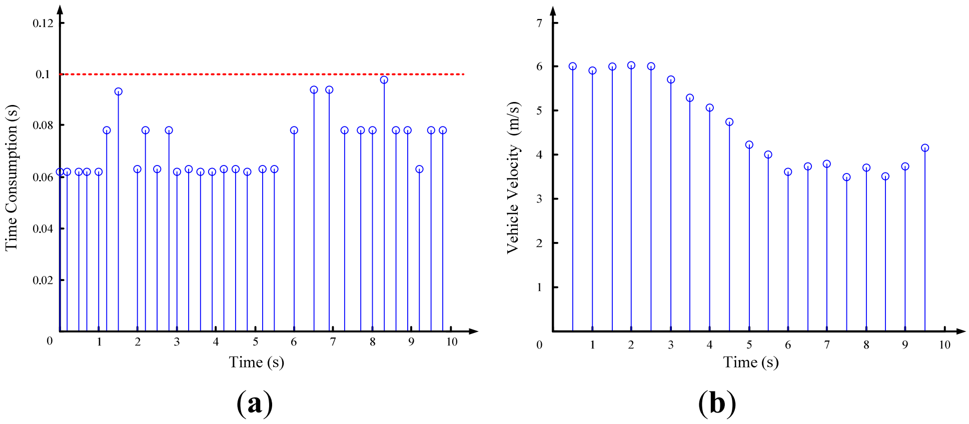
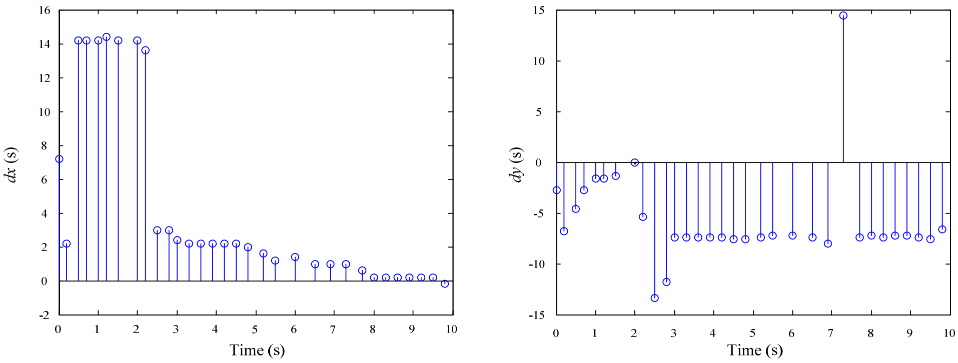


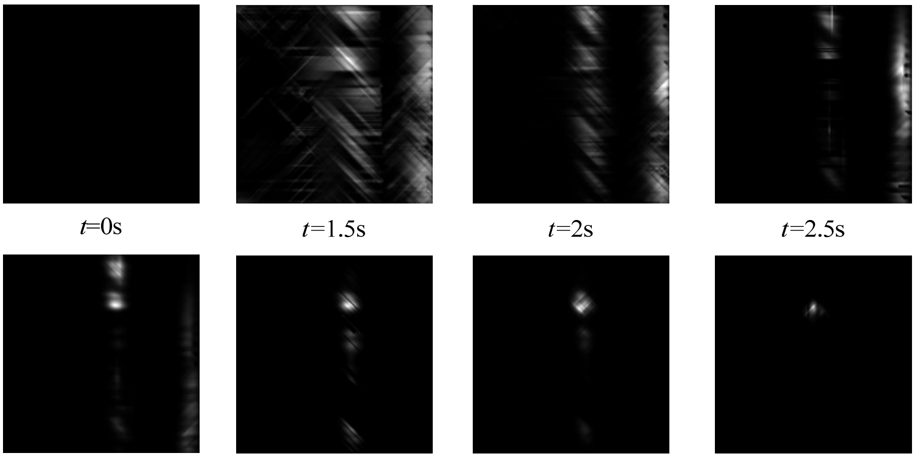
Figure 11. *Cont.*



(c) **(d)**

Figure 12 describes the changing process of probability distribution in the constraint window during the localization process. Probability values in the same window are normalized. Brighter color in the figure shows higher probability. Parameter *t* denotes the time. With UGV moving and scanning, these figures show the recursive update process of probability distribution in the constraint window. The probability distribution has changed from a relatively decentralized state to a centralized state. This process indicates that the vehicle can gradually find its relative position in a metric map.

Figure 12. Probability distribution in constraint window scale during the WCML process. Brighter color shows higher probability. The distribution is changed from a decentralized state to a centralized state. The centralized state in t = 9 s shows that the vehicle finally finds its position.

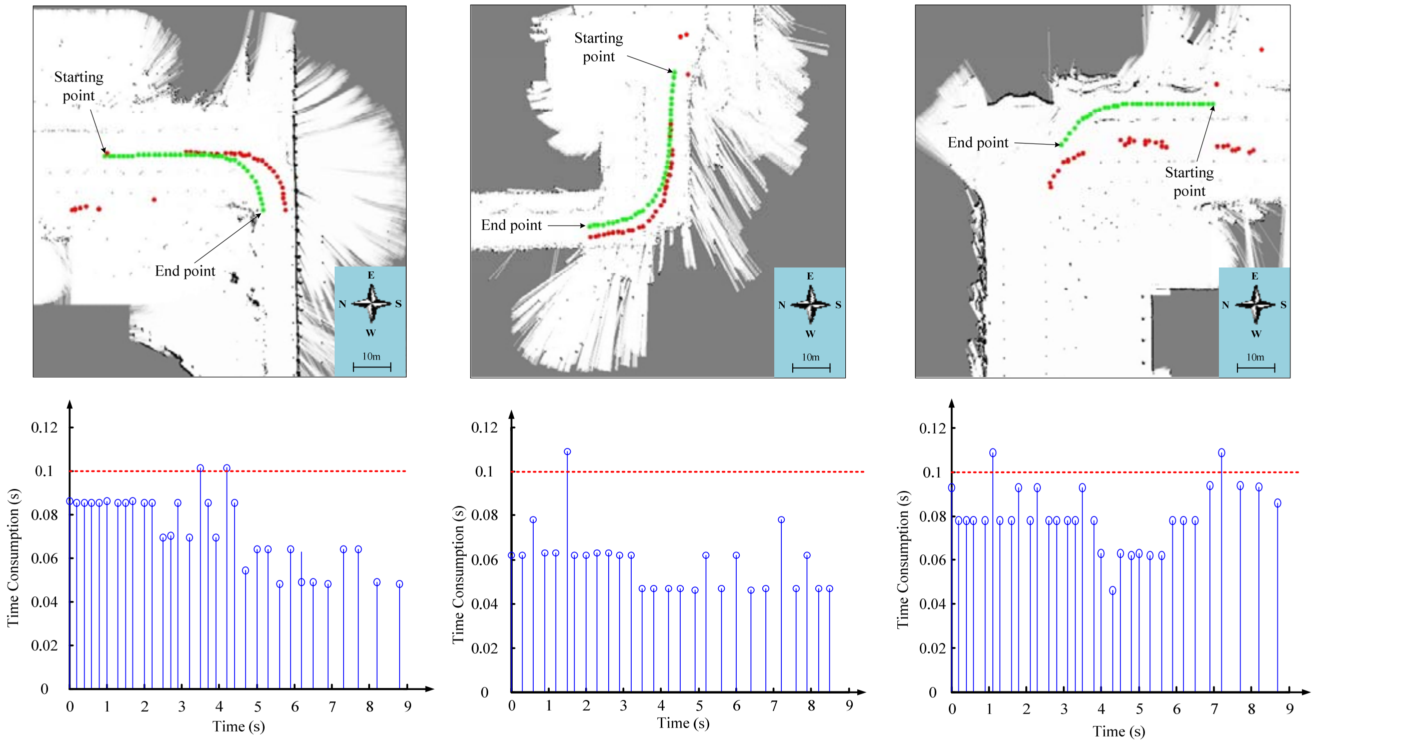


t=0s *t*=1.5s *t*=2s *t*=2.5s

t=4s *t*=6s *t*=7s *t*=9s

The experiment results and analysis show that the WCML method has the ability for UGV to localize in the loose constraint area. When entering the LC area, the positioning system adapts itself and corrects the initial positioning error before getting into the intersection area. It also can provide about a 10 Hz localization frequency for UGV, which is enough for relatively low speed driving in intersections. Results in Figure 13 illustrate other WCML processes with different initial positioning error. The time consumptions prove the real-time performance.

Figure 13. Window constraint Markov localization in loose constraint areas.



5. Conclusion

This paper proposes a hybrid map-based navigation method for unmanned ground vehicles in the urban scenario, without using accurate positioning equipment. We divide the urban scenario into strong constraint areas, where the traffic lanes exist, and loose constraint areas, where no lanes can guide the vehicle. In the loose constraint area, the Markov localization method is involved to achieve high positioning accuracy. By combing the local metric maps for loose constraint areas and the global topological map, a hybrid map is established. In our experiment, we have successfully reduced 50% of the map data. If the scale of the scenario is enlarged, the reduction effect could be more obvious.

In order to improve the real-time performance of grid-based Markov localization, we introduce a window constraint Markov localization method for accurate positioning. A constraint window moving with the vehicle can restrict the size of the state space and, thus, reduce the computational cost. According to our experiments, most positioning results can be achieved within 0.1 s. The update frequency and positioning accuracy can meet the navigation requirements for relatively low speed driving in loose constraint areas.

For future research, we will try to involve vision features to improve the performance of the Markov localization process in the environment map building. Furthermore, the vehicle-to-infrastructure and vehicle-to-vehicle [28,29] communications should be adopted for the driving safety of unmanned ground vehicles.

Acknowledgment

This study is supported by the National Natural Science Foundation of China (No. 91120010 and No. 51275041), the Doctoral Fund of Ministry of Education of China (No. 20121101120015) and the Fundamental Research Funds from Beijing Institute of Technology (No. 20120342011).

Conflict of Interest

The authors declare no conflict of interest.

References

1. Urmson, C.; Anhalt, J.; Bagnell, D.; Baker, C.; Bittner, R.; Clark, M.N.; Dolan, J.; Duggins, D.; Galatali, T.; Geyer, C.; *et al.* Autonomous driving in urban environments: Boss and the urban challenge. *J. Field Robot.* **2008**, *25*, 425–466.
2. Montemerlo, M.; Becker, J.; Bhat, S.; Dahlkamp, H.; Dolgov, D.; Ettinger, S.; Haehnel, D.; Hilden, T.; Hoffmann, G.; Huhnke, B.; *et al.* Junior: The Stanford entry in the urban challenge. *J. Field Robot.* **2008**, *25*, 569–597.
3. Fox, D.; Burgard, W.; Dellaert, F.; Thrun, S. Monte Carlo Localization: Efficient Position Estimation for Mobile Robots. In Proceedings of the 1999 16th National Conference on Artificial Intelligence, Orlando, FL, USA, 18–22 July 1999; pp. 343–349.
4. Dellaert, F.; Fox, D.; Burgard, W.; Thrun, S. Monte Carlo Localization for Mobile Robots. In Proceedings of 1999 International Conference on Robotics and Automation, Detroit, MI, USA, 10–15 May 1999; Volume 2, pp. 1322–1328.
5. Burgard, W.; Derr, A.; Fox, D.; Cremers, A.B. Integrating Global Position Estimation and Position Tracking for Mobile Robots: The Dynamic Markov Localization Approach. In Proceedings of 1998 International Conference on Intelligent Robots and Systems, Victoria, BC, Canada, 13–17 October 1998; Volume 2, pp. 730–735.
6. Burgard, W.; Fox, D.; Hennig, D.; Schmidt, T. Estimating the Absolute Position of a Mobile Robot Using Position Probability Grids. In Proceedings of National Conference on Artificial Intelligence, Portland, OR, USA, 4–8 August 1996; Volume 2, pp. 896–901.
7. Payá, L.; Fernández, L.; Gil, A.; Reinoso, O. Map building and monte carlo localization using global appearance of omnidirectional images. *Sensors* **2010**, *10*, 11468–11497.
8. Junichi, S. Adaptive slope filtering of airborne LiDAR data in urban areas for digital terrain model (DTM) generation. *Remote Sens*. **2012**, *4*, 1804–1819.
9. Kyle, A.; Katheryn, I.L.; Willem, J.D. Fusion of high resolution aerial multispectral and LiDAR data: land cover in the context of urban mosquito habitat. *Remote Sens*. **2011**, *3*, 2364–2383.
10. Lehtomäki, M.; Jaakkola, A.; Hyyppä, J.; Kukko, A.; Kaartinen, H. Detection of vertical pole-like objects in a road environment using vehicle-based laser scanning data. *Remote Sens*. **2010**, *2*, 641–664.
11. Howard, B. Google: Self-Driving Cars in 3–5 Years. Feds: Not So Fast. Available online: http://www.extremetech.com/extreme/147940-google-self-driving-cars-in-3–5-years-feds-not-so-fast (accessed on 8 April 2013).

12. Guizzo, E. How Google's Self-Driving Car Works. Available online: http://spectrum.ieee.org/automaton/robotics/artificial-intelligence/how-google-self-driving-car-works (accessed on 18 October 2011).
13. Xiong, G.; Zhou, P.; Zhou, S.; Zhao, X.; Zhang, H.; Gong, J.; Chen, H. Autonomous Driving of Intelligent Vehicle BIT in 2009 Future Challenge of China. In Proceedings of 2010 IEEE Intelligent Vehicles Symposium, San Diego, NJ, USA, 21–24 June 2010; pp. 1049–1053.
14. Haklay, M.; Weber, P. Openstreetmap: user-generated street maps. *IEEE Perv. Comput.* **2008**, *7*, 12–18.
15. Gil, A.; Reinoso, Ó.; Ballesta, M.; Juliá, M.; Payá, L. Estimation of visual maps with a robot network equipped with vision sensors. *Sensors* **2010**, *10*, 5209–5232.
16. He, B.; Zhang, S.; Yan, T.; Zhang, T.; Liang, Y.; Zhang, H. A novel combined SLAM based on RBPF-SLAM and EIF-SLAM for mobile system sensing in a large scale environment. *Sensors* **2011**, *11*, 10197–10219.
17. Zhang, X.; Rad, A.B.; Wong, Y. Sensor fusion of monocular cameras and laser rangefinders for line-based Simultaneous Localization and Mapping (SLAM) tasks in autonomous mobile robots. *Sensors* **2012**, *12*, 429–452.
18. Shi, Y.; Ji, S.; Shi, Z.; Duan, Y.; Shibasaki, R. GPS-supported visual SLAM with a rigorous sensor model for a panoramic camera in outdoor environments. *Sensors* **2012**, *13*, 119–136.
19. Zhao, H.; Chiba, M.; Shibasaki, R.; Shao, X.; Cui, J.; Zha, H. SLAM in a Dynamic Large Outdoor Environment Using a Laser Scanner. In Proceedings of 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 1455–1462.
20. Trung-Dung, V.; Burlet, J.; Aycard, O. Grid-based localization and local mapping with moving object detection and tracking. *Inform. Fusion* **2011**, *12*, 58–69.
21. Weiss, T.; Schiele, B.; Dietmayer, K. Robust Driving Path Detection in Urban and Highway Scenarios Using a Laser Scanner and Online Occupancy Grids. In Proceedings of 2007 IEEE Intelligent Vehicles Symposium, Istanbul, Turkey, 13–15 June 2007; pp. 184–189.
22. Zhou, S.; Jiang, Y.; Gong, J.; Xiong, G.; Chen, H. A Novel Lane Detection Based on Geometrical Model and Gabor Filter. In Proceedings of 2010 IEEE Intelligent Vehicles Symposium, San Diego, CA, USA, 21–24 June 2010; pp. 59–64.
23. Gong, J.; Jiang, Y.; Xiong, G.; Guan, C.; Tao, G.; Chen, H. The Recognition and Tracking of Traffic Lights Based on Color Segmentation and CAMSHIFT for Intelligent Vehicles. In Proceedings of 2010 IEEE Intelligent Vehicles Symposium, San Diego, CA, USA, 21–24 June 2010; pp. 431–435.
24. Jiang, Y.; Zhou, S.; Jiang, Y.; Gong, J.; Xiong, G.; Chen, H. Traffic Sign Recognition Using Ridge Regression and OTSU Method. In Proceedings of 2011 IEEE Intelligent Vehicles Symposium, Baden-Baden, Germany, 5–9 June 2011; pp. 613–618.
25. Rowe, A.; Starr, Z.; Rajkumar, R. Using Micro-Climate Sensing to Enhance RF Localization in Assisted Living Environments. In Proceedings of IEEE International Conference on Systems, Man and Cybernetics, Montreal, QC, Canada, 7–10 October 2007; pp. 3368–3375.
26. Fox, D. Markov Localization: A Probabilistic Frame for Mobile Robot Localization and Navigation. Ph.D. Thesis, University of Bonn, Bonn, Germany, 1998.

27. Jiang, Y.; Zhao, S.; Gong, J.; Xiong, G.; Chen, H. System design of self-driving in simplified urban environments (in Chinese). *Chin. J. Mech. Eng.* **2012**, *20*, 103–112.
28. Azimi, S.R.; Bhatia, G.; Rajkumar, R.; Mudalige, P. Vehicular networks for collision avoidance at intersections. *SAE Inter. J. Passenger Car.-Mech. Syst.* **2011**, *4*, 406–416.
29. Belanovic, P.; Valerio, D.; Paier, A.; Zemen, T.; Ricciato, F.; Mecklenbrauker, C.F. On wireless links for vehicle-to-infrastructure communications. *IEEE Trans. Veh. Technol.* **2010**, *59*, 269–282.