Cloud Based Localization for Mobile Robot in Outdoors

Xiaorui Zhu, *Member, IEEE*, Chunxin Qiu, Yulong Tao and Qi Jin

***Abstract*— Cloud Robotics is the application of the cloud computing concept to the robot. It utilizes modern cloud computing infrastructure to distribute computing resources and datasets. A cloud based localization technique is proposed in this paper to allow the robot to identify its location relative to a road network map in the cloud. The update of the road network map and the extraction of the robot-terrain inclination model (RTI model) are running in the cloud. A particle filter localization is achieved on the mobile robot based on the local RTI model sent from the cloud. Experiments were carried out for validation of the proposed cloud based localization technique. Preliminary results show that this method could be potentially applicable to long-term autonomous.**

***Index Terms*—Cloud robotics, localization, Mobile robot.**

I. Introduction

Localization is an important problem for autonomous mobile robot. Traditional robotic technologies, however, have been limited by the inherent physical constraints especially for large-scale explorations since all the computations have to be conducted in the onboard computers/microchips of the robot that have limited computing capabilities. Hence this paper proposes a cloud-based architecture to achieve long-term autonomous localization of mobile robots in outdoor environments taking advantage of the powerful computation, storage and other shared resources of the cloud.

In the last decade, many researchers have started to focus on the robot localization in outdoor environments. GPS is a popular tool for localization in outdoor environments[1-2]. However, in some urban circumstances, the signals would be blocked resulting in degrading of the position estimates. Laser range scanner is an option for autonomous outdoors navigation where the point clouds were generated for representation of the surroundings[3-9]. However, the con of the point clouds representation was higher computational load [3]. The combination of point reduction and kd-tree was proposed to reduce the computational load of point clouds representation[4]. Some researchers used the occupancy grid map to divide the whole point clouds into a grid of cells with the occupancy evidence inferred from sensors[5]. Since the representation of the entire space must be stored in memory, even 2D evidence grids are large and expensive to copy. In order to reduce the processing and storage requirements, an octree data structure was further developed to finish a underwater tunnel exploration project using a simplified occupancy grid map[6]. On the other hand, several researchers proposed to sort the raw point clouds data in the way of the standard elevation map (DEM)[7-8] and the multi-level surface map (MLS)[9]. The application of the vision systems for outdoor localization has also received increasing attention[10]. Different robust core algorithms such as the Scale Invariant Feature Transform (SIFT), the Position Invariant Robust Feature (PIRF) and etc. have been developed to adapt to the complex outdoor environments[11-12]. But unpredictable long computation periods still made the above techniques fail for many real-time applications[13]. Outsourcing map based localization is another kind of the methods to estimate the robot position. Mandel et al. proposed a novel approach to take advantage of the road network's structure and its height profile for position estimation when GPS was lost [14]. Our research group also proposed a new localization method with less occupied memory where the topographical map was utilized as the prior available terrain map for localization[15]. However, in the case of the large-scale exploration and long-term autonomous, all the above efforts are not enough.

Recently few researchers have tried to face the challenges on the long-term autonomous robot in outdoor environments where the mobile robot is expected to run autonomously over a long period of time and adapt to the real dynamic scenarios. Zhao et al. proposed a simultaneous localization and mapping (SLAM) method to simultaneously detect and track the moving objects using a laser scanner in a dynamic environment[16]. The authors found out that the method was very time-consuming to track many static or moving objects. Badino et al. and Neubert et al. described a novel concept of appearance change prediction to learn how the environment changes over time beforehand, and then take advantage of the learned knowledge to predict its appearance under different environmental conditions[17-18]. The key limitation of this method was the requirement of a large storage space to store different environmental conditions as many as possible and the map information of the navigation area. On the other hand, if the real environment changed, it would be hard to update the map. Cloud robotics provides a very promising solution to overcome

This research is currently supported by the NSF of China under NSFC Grant No. 60905052.

Xiaorui Zhu is the corresponding author with Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, Guangdong 518055, China (e-mail: xiaoruizhu@hitsz.edu.cn).

C. Qiu, Y. Tao and Q. Jin are with the Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, Guangdong 518055, China (qiu_chun_xin@163.com).

such problems[19]. Cloud robotics is applying the cloud computing concept to robots in order to augment the robots capabilities by off-loading computation and shares huge data or new skills via the internet.

The related works on cloud robotics are still rare so far. Arumugam et al. built a cloud computing infrastructure "DAvinCi" to improve the implementation speed of simultaneous localization and mapping (SLAM) [20]. Kehoe et al. developed an architecture for a cloud robotics system to recognize and grasp the common household objects[21]. Wang et al. introduced a generic infrastructure of cloud robotic system to enable several poor-equipped robots to retrieve location data from a dynamically updated map which was built by a well-equipped robot[22].

In this paper, we proposed a cloud based localization technique using outsourcing road network maps to achieve long-term/large-scale autonomous navigation of the mobile robot in outdoor environments. The proposed technique in this paper is aimed to solve two problems. One is that GPS would receive insufficient satellite signals among the buildings and other constructions during the long-term outdoor autonomous navigation. The other problem is that the long-term/large-scale autonomous navigation would require greatly increasing computational payloads. Hence the road network maps are first extracted by Google Earth, OpenStreetMap or other commercially available resources such that the new roads could be added into the map database in the cloud. The Terrain Inclination Aided Localization algorithms recently proposed by our research group will then be applied to achieve online localization only taking advantage of the road network maps stored in the cloud [15]. In this paper, the cloud can not only provide storage space to store the large amount of map data, but also provide a new way to obtain the latest map information.

The structure of this paper follows. Background and Related work are discussed in Section 1. The cloud robotics architecture and the detailed algorithms are proposed in Section2. Experimental results and discussion are presented in Section 3. Conclusions are described in Section4.

II. Cloud Robotics Architecture and Algorithms

The proposed cloud robotics architecture has two phases: offline and online, Figure 1.

A. *Offline phases*

The offline phase is to extract the new road networks and add them to the road network map stored in the cloud before each task is executed, Figure 1. In order to obtain the new road networks, a set of points were labeled along the new roads, Figure.2. Then the geodetic coordinate (latitude, longitude, altitude) of these points located on the road networks together with the new road names can be extracted from the Google Earth that could provide the

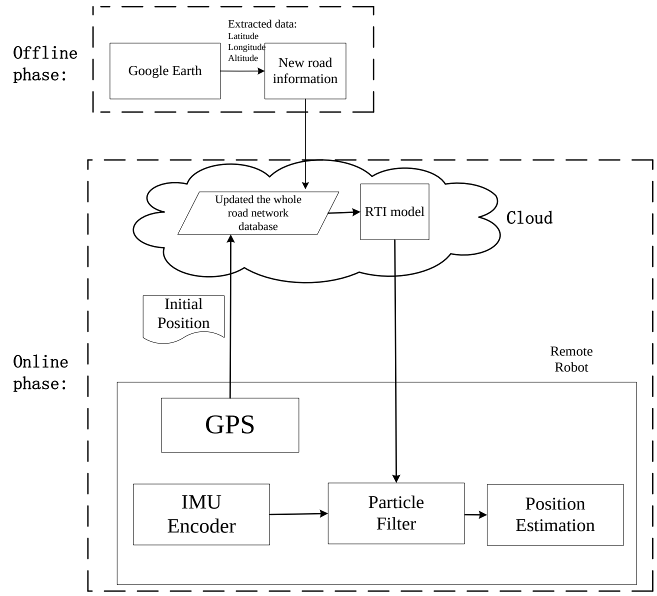


Figure 1.Cloud-based localization Architecture.

latest information of the roads. Therefore the road network map in the cloud is updated and ready for online task execution.

B. *Online phases*

The online phase is to achieve the localization task based on the latest road network map. According to Figure 1, the cloud based architecture consists of two sections: the mobile robot and the cloud. The cloud part includes the updated road network map and the RTI model proposed by our research group [23]. RTI model is used to describe the relationships between the robot attitude and the robot position and can be extracted from the road network map.

When the robot moves on a road, the initial position estimated by the GPS will be sent to the cloud. If GPS signal is not on the road, the estimated initial position will be pulled to a nearest road point by searching the road network map. This road point would be treated as the new estimation of the initial position. All the road segments within the neighborhood of the initial estimation with the radius of δ are searched. The robot-terrain inclination (RTI) model for the corresponding road segments is then

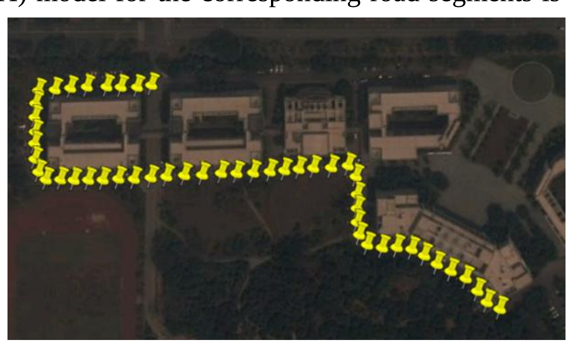


Figure.2. Google Earth and the points sets on the pre-planned path.

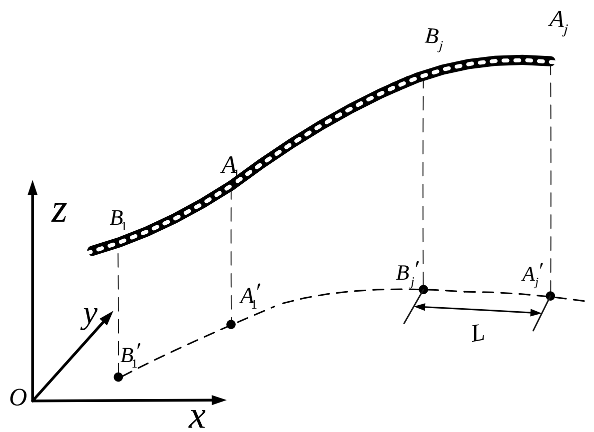


Figure 3.Robot path is segmented into a series of line segments.

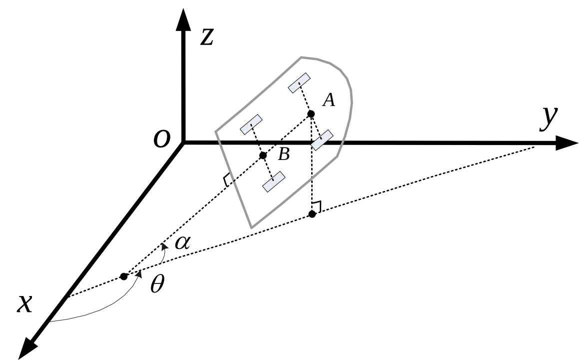


Figure 4.Geometric extraction of RTI model.

computed in the cloud and sent back to the robot.

On the robot part, a 3-D inertial sensor is installed to measure the attitude and velocities of the robot. Then a particle filter algorithm is used to incorporate the inertial sensor data to determine the 3-D position of the robot based on the RTI model sent from the cloud.

RTI Model in the Cloud:

Suppose there is one portion of the road networks $B_1A_1B_jA_j$, Figure 3. The positions of the point set on the road are first transformed from the geodetic coordinate (latitude, longitude, altitude) to the Cartesian coordinate (*x*, *y*, *z*). It is then projected onto the *x*-*O*-*y* plane as $B_1'A_1'B_j'A_j'$, Figure 3. The projected road is segmented into a series of line segments with a fixed interval L/k where L is the length of the robot. The $B_j'A_j'$ represents the j^{th} line segment. The points B_j' and A_j' are the projections of the road points B_j and A_j, respectively. The *z* value of these road points can be obtained by a weighted average interpolation method from the road network map [24]. When the robot moves above the j^{th} line segment along the pre-planned path, the points B_j and A_j represent the midpoint between two ground contact points of the front wheels and the left-rear wheel, respectively.

The $\overrightarrow{B_jA_j}$ represents the direction of the robot motion, Figure 4. The heading angle, θ_j ,is defined as the angle between the $\overrightarrow{B_j'A_j'}$ and the *x* axis. The heading angle is exclusively determined by the path. The angle α_j is defined as the one between the robot direction $\overrightarrow{B_jA_j}$ and the *x*-*O*-*y* plane. So the angles α_j can be obtained from the following equations,

$$\alpha_j = \sin^{-1}\left(\frac{(z_A - z_B)}{\left|\overrightarrow{B_jA_j}\right|}\right) \tag{1}$$

$$\left|\overrightarrow{B_jA_j}\right| = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2} \tag{2}$$

where the coordinate information of the points includes $B_j = (x_B, y_B, z_B)$ and $A_j = (x_A, y_A, z_A)$ .Therefore, a number of angles (θ_j , α_j) can be extracted from the serial line segments $B_j'A_j'$. Then the robot position (x_j, y_j, z_j) at each line segments corresponds with each group (θ_j , α_j).By linear interpolation of the above discrete relationship, $[\theta_k \quad \alpha_k]^T = RTI\_Model(x_k, y_k, z_k)$ can be obtained, $k = 1, 2, \ldots, N$. The number *N* can be adjusted for the accuracy requirement.

Communication between the Cloud and the Robot:

Assumption in this paper is that the robot and the cloud share with the same network. The cloud service creates the listener socket that is waiting for remote clients to connect. The client issues the connect() socket function to start the TCP handshake. This socket contains many parameters of the client, such as IP address, port number and so on. If these parameters are the same as those in the listener socket, then the cloud server issues the accept() socket function to accept the connection request. Thus the communication between the cloud and the robots can be established.

Particle Filter Algorithm on the Robot:

See the detailed algorithm in Table 1.The system state, X_t , represents the three-dimensional position of the robot in the inertial frame (*x*, *y*, *z*) at the time *t*. The superscript [*m*] denotes the particle *m*, *T* is the sampling period, and v_t is the linear velocity in the direction of robot movement.

Table 1 Algorithm: Particle Filter based localization

1:	(X_{t-1} , v_t , z_t)
2:	$X_{t-1}=\langle \chi_{t-1}^{[1]},\chi_{t-1}^{[2]},...,\chi_{t-1}^{[M]}\rangle$, $z_t=\langle \theta_t,\alpha_t,dist_t\rangle$, Q_t , $\bar{X}_t=X_t=\phi$
3:	*for m=1 to M do*
4:	$\chi_t^{[m]}=\begin{bmatrix} x_t^{[m]} \\ y_t^{[m]} \\ z_t^{[m]} \end{bmatrix}=\begin{bmatrix} x_{t-1}^{[m]}+\cos\theta_t\cdot\cos\alpha_t\cdot v_t\cdot T \\ y_{t-1}^{[m]}+\sin\theta_t\cdot\cos\alpha_t\cdot v_t\cdot T \\ z_{t-1}^{[m]}+\sin\alpha_t\cdot v_t\cdot T \end{bmatrix}$ *//motion model*
5:	$\hat{z}_t^{[m]}=\begin{bmatrix} \hat{\theta}_t^{[m]} \\ \hat{\alpha}_t^{[m]} \\ \hat{dist}_t^{[m]} \end{bmatrix}=\begin{bmatrix} \hat{\theta}_t^{[m]}=RTI\_Model_\theta(x_t^{[m]},y_t^{[m]},z_t^{[m]}) \\ \hat{\alpha}_t^{[m]}=RTI\_Model_\alpha(x_t^{[m]},y_t^{[m]},z_t^{[m]}) \\ \hat{dist}_t^{[m]}=dist(p(x,y,z)-p(x_t^{[m]},y_t^{[m]},z_t^{[m]})) \end{bmatrix}$ *// measurement model*
6:	$w_t^{[m]}=\lvert 2\pi Q_t\rvert^{-\frac{1}{2}}\exp\left\{-\frac{1}{2}\left(z_t-\hat{z}_t^{[m]}\right)^T\right\}Q_t^{-1}\left(z_t-\hat{z}_t^{[m]}\right)$ *// weight calculation*
7:	*Add* $\chi_t^{[m]}$ *and* $w_t^{[m]}$ *to* $\bar{X}_t$
8:	*endfor*
9:	*for m=1 to M do*
10:	*Draw i with probability* $\propto w_t^{[i]}$
11:	*Add* $\chi_t^{[i]}$ *to* X_t
12:	*endfor*
13:	*return* $X_t=\langle \chi_t^{[1]},\chi_t^{[2]},...,\chi_t^{[M]}\rangle$

RTI_Model_θ and RTI_Model_α are the RTI model downloaded from the cloud that is treated as the measurement model. $p(x_t^{[m]},y_t^{[m]},z_t^{[m]})$ represents the position of the particle *m* and $p(x,y,z)$ represents the road point on the robot path closest to this particle that is gained from the map. $\hat{dist}_t^{[m]}$ is the distance between the particle *m* and the point $p(x,y,z)$ [14]. $w_t^{[m]}$ is the weighting factor of the particle *m* for resampling of the particle filter.

III. Experiments and Discussion

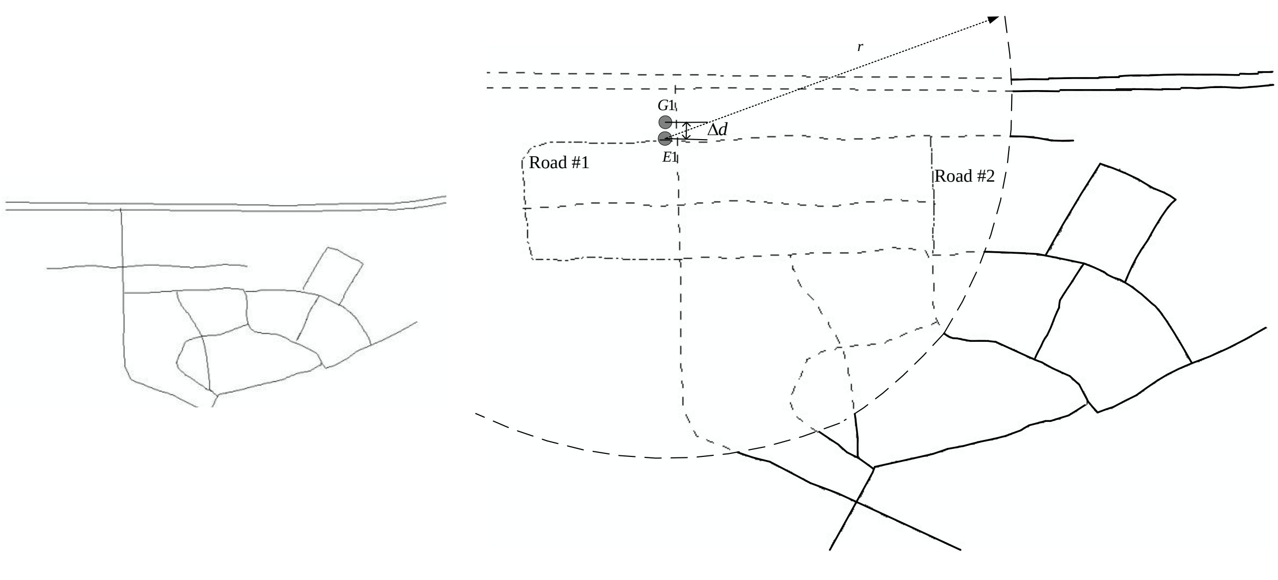
A. Methods and Procedures

The experiments were conducted on the platform, a Summit XL mobile robot. The NAV440 from Crossbow Technology® was used as the inertial measurement unit (IMU) that mounted on the top surface of the robot in order to measure the roll, pitch, yaw angles. The measurement accuracy was 0.5 degree in the roll and pitch directions while 1 degree in the yaw direction. The line velocity of the robot was provided by the encoders. An outdoor environment with the area of 200m x 400m in the Shenzhen University Town, Figure.2, was used for performance evaluation. The sampling period for all experiments is 0.1s.

B. Results and Discussion

The road networks of the selected area were extracted from the Google Earth and sent to the cloud service before the localization task started. Comparing with the road network database saved previously (Figure 5 (a)-(b)), it was found out that Road #1 and #2 on the current road networks were new, Figure 5 (b). Therefore,the latest road network information was updated in the cloud.

When the Summit robot started to move on the road, the initial position estimation by GPS, $G1$, was transmitted to the cloud service. So the short-distance road point, $E1$, was searched in the cloud because this estimated position was not on any of the road. The roads within the neighborhood of the point $E1$ with the radius 200m were obtained, Figure 5 (b). Then a local RTI model starting from this new initial point along the pre-planned path was computed in the cloud service and sent back to the mobile



(a) Road networks previously existing

(b)Updated road networks from the Google Earth

Figure 5.Road networks of the experimental area.

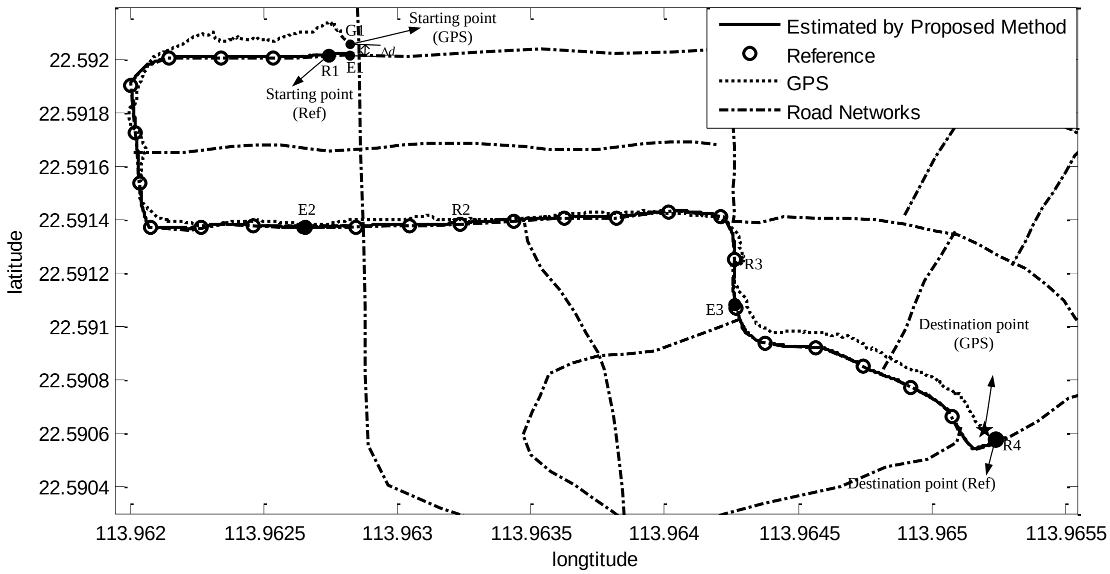


Figure 6. The estimation of the robot position by the proposed technique.

robot. Finally, the localization can be achieved on the robot by applying the particle filter algorithm in Section 2. The estimated position at the end of each 200m travelling distance of the robot (E2 and E3 in Figure 6) was sent to the cloud again, and the above procedures were repeated.

Figure 6 depicts the position estimation of the robot by the proposed technique (solid line) compared with the GPS alone (dashed line) and the reference positions (circle signs) with the robot speed 1.0m/s. According to Figure 6, the position estimation by the proposed method was much closer to the ground truth values inside the Area 1 (from the reference point $R1 \rightarrow R2$) and Area 3 ($R3 \rightarrow R4$). On the Area 2(from the reference point $R2 \rightarrow R3$), the performance of the proposed method was quite similar to the one estimated by GPS alone. It was found out the Area 2 was a wide playground and GPS signal was already effective while Area 1 was surrounded by two buildings and GPS signals were worse. The same phenomena were also observed in Figure 7. Figure 7 shows the comparison of the position estimation errors using the proposed technique (circle sign) and the GPS alone (circle sign). The estimation error at the travelling distance of 380m was up to (9m, 9m) using GPS alone according to Figure 7 because GPS was partially blocked. This result has apparently pointed to somewhere off-road. At the same time, the estimation error using the proposed method has reached to the value (0.3m, 3m). This estimated position was still on the road, which was coincident with the actual fact. Hence it is concluded the proposed cloud based technique can achieve online localization for large-scale road networks. In the near future, more long-term experiments will be carried out, and more complex scenarios will be considered.

IV. CONCLUSIONS

This paper introduces a cloud-based outsourcing localization technique for a mobile robot on outdoor road networks. Preliminary experimental results validate the proposed technique and illustrate that the proposed technique has capability to achieve online localization taking advantage of outsourcing road network maps and the relative algorithms in the cloud. This method will be applied to more large-scale/long-term circumstances.

REFERENCES

[1] P. Bonnifait, P. Bouron, P. Crubille, and D. Meizel, "Data fusion of four ABS sensors and GPS for an enhanced localization of car-like vehicles," presented at *IEEE International Conference on Robotics and Automation(ICRA)*, 21-26 May 2001, Seoul, South Korea, 2001.

[2] D. Bouvet and G. Garcia, "Civil-engineering articulated vehicle localization: solutions to deal with GPS masking phases," presented at

IEEE International Conference on Robotics and Automation(ICRA), 24-28 April 2000, San Francisco, CA, USA, 2000.

[3] C. Brenneke, O. Wulf, and B. Wagner, "Using 3D Laser Range Data for SLAM in Outdoor Environments," presented at 2003 IEEE/RSJ *International Conference on Intelligent Robots and Systems*, Oct 27-31 2003, Las Vegas, NV, United States, 2003.

[4] A. Nüchter, H. Surmann, K. Lingemann, J. Hertzbergand S. Thrun."6D SLAM with an Application in Autonomous Mine Mapping," presented at *International Conference on Robotics and Automation(ICRA)*, April 26-May 1, New Orleans, USA, 2004.

[5] M. C. Martin and H. Moravec," Robot evidence grids. Technical Report CMU-RI-TR-96-06," Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.

[6] N. Fairfield, G. Kantor and D. Wettergreen, " Real-Time SLAM with Octree Evidence Grids for Exploration in Underwater Tunnels," Journal of Field Robotics, vol. 23, pp. 3-21.

[7] E. Hygounenc, I.-K. Jung, P. Soueres, and S. Lacroix. " The autonomous blimp project of laas-cnrs: Achievements in flight control and terrain mapping". Int'l Journal of Robotics Research, Vol.23, No.4-5, pp.473–511, 2004.

[8] P. Pfaff, R. Triebel, and W. Burgard. " An efficient extension to elevation maps for outdoor terrain mapping and loop closing ". Int'l Journal of Robotics Research, Vol. 26, No. 2, pp. 217–230, 2007.

[9] R. Triebel, P. Pfaff, W. Burgard."Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing. " presented at *International Conference on Intelligent Robots and Systems (IROS)*,9-15 Oct. 2006, Beijing, China ,2006.

[10] P. Blaer and P. Allen, "Topological Mobile Robot Localization Using Fast Vision Techniques," presented at *International Conference on Robotics and Automa*tion(IC*RA)*, 11-15 May 2002, Washington DC, USA, 2002.

[11] T. D. Barfoot, "Online visual motion estimation using fastslam with sift features," presented at *International Conference on Intelligent Robots and Systems (IROS)*, 2-6 Aug, Edmonton, Canada, 2005.

[12] A. Kawewong, N.Tongprasit, S.Tangruamsub, O.Hasegawa,"Online and Incremental Appearance-based SLAM in Highly Dynamic Environments," Int'l Journal of Robotics Research, Vol.30, No.1, pp.33–33, 2010.

[13] N. Tongprasit, A. Kawewong, O. Hasegawa," PIRF-Nav 2: Speeded-up Online and Incremental Appearance-based SLAM in An Indoor Environment," presented at *IEEE Workshop on Applications of Computer Vision*, Washington, DC, USA.

[14] C. Mandel, T. Laue, " Particle filter-based position estimation in road networks using digital elevation models," presented at IEEE/RSJ *International Conference on Intelligent Robots and Systems*, Taipei, Taiwan.2010

[15] X. Zhu, C.Qiu, M. A. Minor, "Terrain Inclination Aided Three-Dimensional Localization and Mapping for an Outdoor Mobile Robot," International Journal of advanced Robotic Systems, vol.10,76.

[16] H. Zhao et al., "SLAM in a Dynamic Large Outdoor Environment using a Laser Scanner," presented at*International Conference on Robotics and Automation (ICRA)*, 19-23 May. 2008, Pasadena,USA, 2008.

[17] H. Badino, D. Huber, T. Kanade, "Real-Time Topometric Localization," presented at *International Conference on Robotics and Automation (ICRA)*, June. 2013, Berlin, Germany, 2013.

[18] N. Sunderhauf, P. Neubert, P. Protzel, "Predicting the change-A Step Towards Life-Long Operation in Everyday Environments," presented at 2013*Robotics: Science and Systems (RSS) Robotics Challenges and Vision Workshop*,14-18 May. 2012, Minnesota,USA, 2012.

[19] J. Kuffner, "Cloud-EnabledRobots," In IEEE-RAS *International Conference on Humanoid Robots*, 6-8 December, Nashville,USA,2010.

[20] R. Arumugam et al., "DAvinCi: A Cloud Computing Framework for Service Robots," presented at *International Conference on Robotics and Automation (ICRA)*, 3-8 May, Anchorage,USA, 2010.

[21] B. Kehoe et al., "Cloud-Based Robot Grasping with the Google Object Recognition Engine," presented at *International Conference on Robotics and Automation (ICRA)*, 6-10 May, Karlsruhe, Germany, 2013.

[22] L.Wang, M. Liu and M. Meng, "Towards Cloud Robotic System: A Case Study of Online Co-localization for Fair resource Competence," presented at *International Conference on Robotics and Biomimetics*, 11-14 December, Guangzhou, China, 2012.

[23] C. Qiu, X. Zhu and L. Liu, "Robot Terrain Inclination Model Extracted from Laser Scanner Data for Outdoor Environment."presented at*International Conference on Mechatronics and Automation*, 5-8 August,Chengdu, China, 2012.

[24] H. Chen, X. Li and W. Ding, "Twelve Kinds of Gridding Methods of Surfer 8.0 in Isoline Drawing," Chinese journal of engineering geophysics, vol. 4, pp. 52-57, 2007.

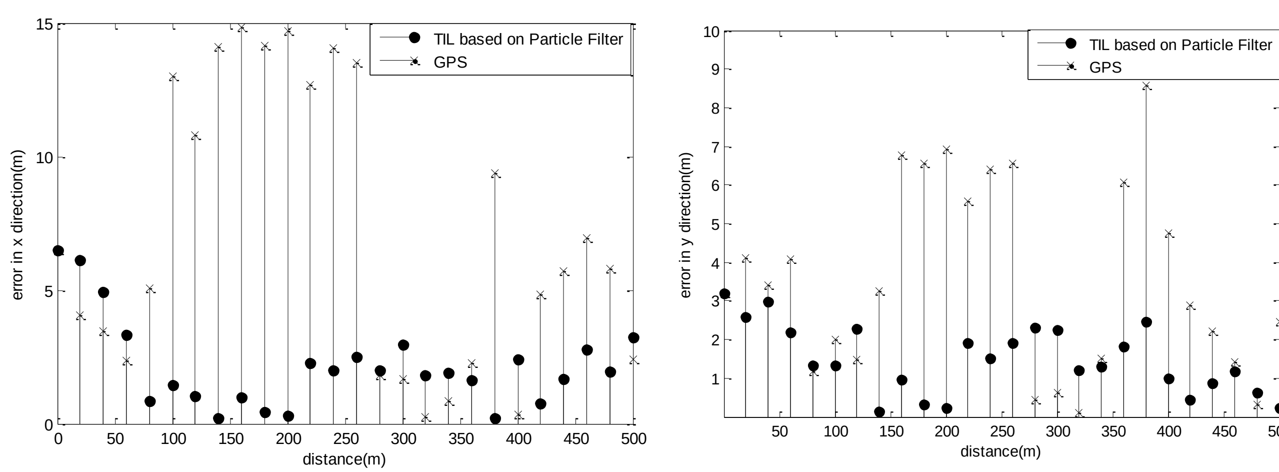


Figure 7.The position estimation errors of the robot using the proposed technique.