

## Performance Improvement of ICP-based Outdoor SLAM Using Terrain Classification

Yong-Ju Lee<sup>\*1</sup>, Yong-Hoon Ji<sup>\*2</sup>, Jae-Bok Song<sup>\*1</sup>, and Sang-Hyun Joo<sup>\*3</sup>

\*1 School of Mechanical Engineering, Korea University  
5, Anam-dong, Seongbuk-gu, Seoul, 136-713, Korea

\*2 Department of Mechatronics, Korea University  
5, Anam-dong, Seongbuk-gu, Seoul, 136-713, Korea

\*3 Agency for Defense Development, Daejeon, Korea

### Abstract:

**This research deals with outdoor SLAM based on local elevation map matching and proposes a new terrain classification method for efficient map matching. The main map used for outdoor SLAM is an elevation map which consists of 2-D grids with elevation information on each cell. 3-D environmental data acquired by tilting a 2-D laser scanner are used to build a local elevation map and classify the local terrain. The experimental results show that the proposed method can enhance performance of outdoor SLAM.**

### 1. INTRODUCTION

When a robot navigates in an unknown environment, both accurate pose estimation of the robot and mapping of the environment are important. Therefore, SLAM (Simultaneous Localization And Mapping) has been one of the most fundamental and challenging issues in the field of mobile robotics. A 2-D grid map is efficient and sufficient when the robot navigates in the indoor environment with a range sensor. In this case, the motion of a robot can be expressed by 3 degrees of freedom ( $x$ ,  $y$ ,  $\theta$ ) in 2-D space. However, outdoor navigation usually requires the estimation of 6 DOF motion ( $x$ ,  $y$ ,  $z$ , roll  $\psi$ , pitch  $\theta$ , yaw  $\phi$ ) in 3-D space, and therefore both the robot and the environment should be represented in 3-D space.

An elevation map is the most popular map to represent 3-D outdoor environment. In this map, the environment is regularly divided into small cells (for example, 0.1m\*0.1m) and each cell has height information. This type of map is suitable for large outdoor environments and used as a main map for special applications such as military robots. A 3-D point cloud, which can also model an outdoor environment, is inefficient and unsuitable for outdoor navigation because it requires a huge amount of memory.

Building of an elevation map for campus or urban environments was conducted in [1]. 3-D range data were obtained by tilting a 2-D laser scanner. Whenever a robot moves a constant distance, local elevation maps are built using 3-D range data and a global elevation map is constructed by integrating these local elevation maps. When two different local elevation maps are matched, the rotation and translation between them are obtained by the

ICP (Iterative Closest Points) algorithm [2]. To increase the accuracy and efficiency of ICP, cells of elevation maps are classified into 4 groups (traversable, vertical object, gap, and edge). The robot pose was accurately estimated and environments were represented well even for the case of the robot's movement more than 100 m. However, the ICP algorithm for the flat paved area failed to provide accurate estimation of the robot pose unlike the ICP for vertical structures.

Modeling urban environments was also conducted in [3]. Two 2-D laser scanners were used for acquiring 3-D data from the environment. One laser scanner was mounted horizontally and the other vertically. The horizontal laser scanner estimated the 3 DOF motion of the robot in 2-D space and the vertical one obtained vertical scan data for 3-D modeling. SLAM for an urban environment using features such as vertical lines was conducted in [4]. It could significantly reduce the computational load for outdoor SLAM. However, the vertical lines were not likely to be extracted well in a general outdoor environment. Among various outdoor SLAM techniques, ICP seems to be the most adequate because it has been applied and the results were demonstrated for various 2-D or 3-D environments.

This research is concerned with elevation map building for unknown outdoor environments and the main issue of the research is how to improve the performance of ICP-based outdoor SLAM. A new terrain classification is proposed to improve ICP performance in outdoor environments. As mentioned above, ICP based on the flat area often fails. However, we present a new method to use the flat area to estimate the robot pose and evaluate how the ICP performance can be improved when the proposed method is used instead of the conventional ICP.

### 2. ELEVATION MAP

Figure 1 shows our experimental setup, which consists of a Pioneer 3AT mobile robot and a SICK laser scanner. This laser scanner senses the environment within 32 m and can be tilted by a DC servomotor. The absolute roll and pitch angles of a robot are sensed by the inertial measurement unit (IMU), and the yaw angle and small motion increments are sensed by both the wheel encoder and the IMU. Combining these data allows the estimation

of six DOF motion in the global coordinate frame.

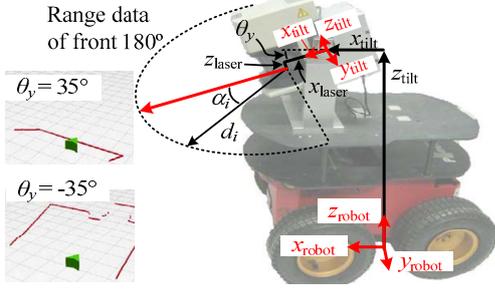


Fig. 1. Experimental setup and local coordinate frames.

To collect the 3-D range data, a robot stops and then a 2-D laser scanner is tilted by a DC motor from 35° (downward) to -35° (upward) continuously in this research. An obstacle is sensed by a tilted laser scanner, and its elevation can be calculated based on a 6 DOF robot pose. The elevation of each cell is updated when it is sensed as follows:

$$e_t(i, j) = \begin{cases} z_t(i, j) + z_{t, \text{robot}} & , \text{ if } e_{t-1}(i, j) < z_t(i, j) + z_{t, \text{robot}} \\ e_{t-1}(i, j) & , \text{ otherwise.} \end{cases} \quad (1)$$

where  $e_t(i, j)$  is the elevation of cell  $(i, j)$  at time  $t$ ,  $z_{t, \text{robot}}$  is the  $z$  position of a robot, and  $z_t(i, j)$  is the sensed elevation of cell  $(i, j)$  at time  $t$ . If the newly-sensed elevation of cell  $(i, j)$  is higher than the previous elevation, then the elevation of cell  $(i, j)$  is replaced by this new elevation.

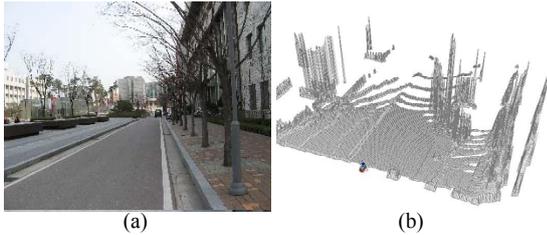


Fig. 2. Elevation map building: (a) real environment and (b) elevation map.

### 3. TERRAIN CLASSIFICATION

A robot builds local elevation maps using sensor data and then constructs a global elevation map by integrating local elevation maps. This integration requires the accurate alignment of two local elevation maps, which is performed by the ICP algorithm in this study. In most cases, ICP presents correct matching results. However, the ICP scheme is computationally intensive because pairs of corresponding points must be found by searching all the

3-D points. Moreover, it is known that the ICP algorithm often fails to calculate the transform when a robot is located on the flat or featureless surface such as a corridor [2].

If all the cells of an elevation map are classified into several groups, and if the corresponding points are found in the same group, the correspondence can be found more accurately and efficiently. In this paper, each cell of an elevation map is classified as one of 5 groups according to the statistical measure of 3-D range data pertaining to the cell. A robot collects all the height values for each cell  $(i, j)$  and extracts their standard deviation and the highest elevation, as shown in Fig. 3. Classification methods are described in detail below and summarized in Table 1.

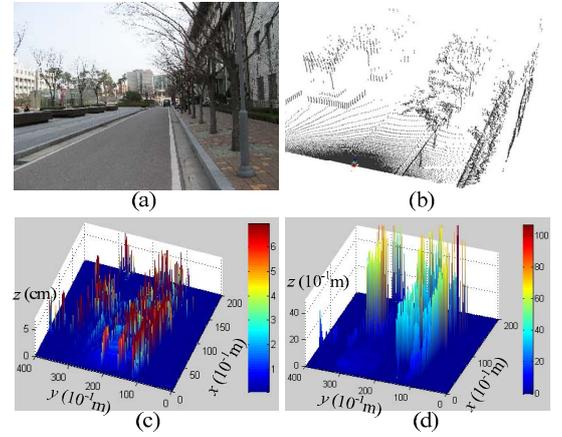


Fig. 3. Environmental data: (a) real environment, (b) 3-D point cloud, (c) standard deviation of elevations of scan data, and (d) highest elevation.

1. Ground: A cell is classified as ‘ground’ when the highest elevation pertaining to the cell is low (for example, lower than 20 cm), and the standard deviation of the elevations pertaining to the cell is also small (for example, smaller than 5 cm). The examples are flat surfaces or roads.
2. Obstacle: A cell is classified as an ‘obstacle’ when the highest elevation pertaining to the cell is similar to the height of the robot and the standard deviation of the elevations pertaining to the cell is small.
3. Overhanging object: A cell is classified as an ‘overhanging object’ when the highest elevation pertaining to the cell is higher than 1.5m and there exists a gap larger than 1.0 m among the elevation information pertaining to the cell. A cell is also classified as an ‘overhanging object’ when the highest elevation pertaining to the cell is higher than the height of the robot and the standard deviation of the elevations pertaining to the cell is small. The typical examples are the eaves of a roof and branches of a tree.

4. Vertical object: A cell is classified as a ‘vertical object’ when the highest elevation pertaining to the cell is higher than the height of the robot and the standard deviation of elevations pertaining to the cell is large. Contrary to an ‘overhanging object’, elevations are uniformly distributed. The typical examples are walls and poles.
5. Horizontal edge: A cell is classified as a ‘horizontal edge’ when there are other types of terrains around the ‘ground’. The example is the curb which is the boundary of the ground.

The terrain classification of 3-D point clouds in Fig. 3(b) is shown in Fig. 4(b)-(f). The proposed terrain classification process takes about 200 ms for the elevation map of 40 m \* 20 m. (1.7GHz CPU, the cell size of the elevation map is 10 cm \* 10 cm.)

Table 1. Terrain classification according to distribution of elevation.

		Highest elevation among elevations pertaining to one cell		
		Low	Middle	High
Standard deviation of elevations pertaining to one cell	Small	Horizontal edge or Ground	Obstacle	Overhanging object
	Large	Vertical object (if the cell contains a gap) Overhanging object (if the cell does not contain a gap)		

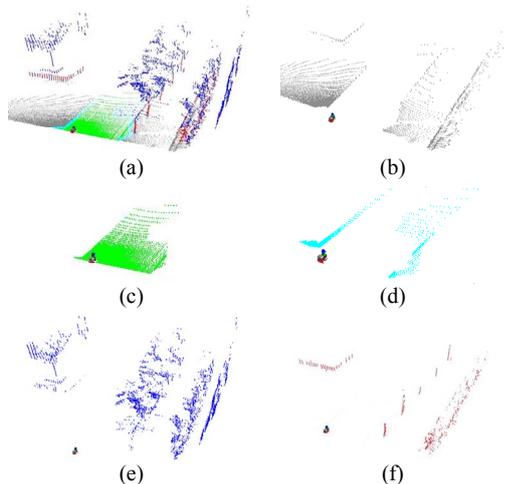


Fig. 4. Example of terrain classification; (a) classified 3-D point cloud, (b) obstacle, (c) ground, (d) horizontal edge, (e) overhanging object, and (f) vertical object.

#### 4. CLASSIFIED TERRAIN-BASED ELEVATION MAP MATCHING

The result of the proposed terrain classification is used

to align two different local elevation maps. For example, a robot builds an elevation map at time  $t-1$ , moves a constant distance (for example, 3 m), and builds another elevation map at time  $t$ . Let us denote the elevation map at time  $t-1$  and  $t$  as  $X$  and  $Y$ , respectively. ICP assumes the closest points as identical points and forces them to be located at the same position. More details on ICP can be referred to [2].

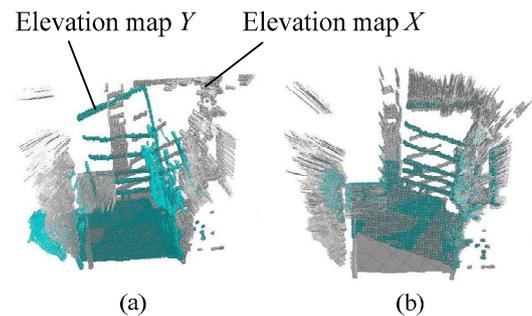


Fig. 5. Matching of elevation maps  $X$  and  $Y$ : (a) initial state before matching, and (b) final state after matching.

Figure 5 illustrates the alignment of elevation maps  $X$  and  $Y$ . Cells of the same type are matched for computing the alignment of two local elevation maps. However, the cells belongs to the ‘ground’ are not used for ICP because their convergence is not very good in most cases. The matching error of ICP between the two local elevation maps is shown in Fig. 6. The matching error continuously decreases. If the matching error converges below the pre-specified threshold, the iteration for ICP is terminated.

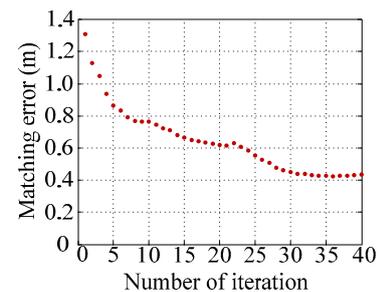


Fig. 6. Change of matching error between two elevation maps during ICP scan matching.

Figure 7 illustrates the integration of local elevation maps and Fig. 8 shows a global elevation map. The robot pose relative to the global frame was obtained by accumulating rotations and translations computed by ICP. In this experiment, the ICP algorithm was executed after the robot moved about every 3 m. ICP started with two elevation maps and an initial guess for their relative transform. The initial guess was generated using odometry and IMU. Therefore, the ICP with terrain classification

could estimate the robot pose more accurately than the ICP using 3-D point cloud alone.

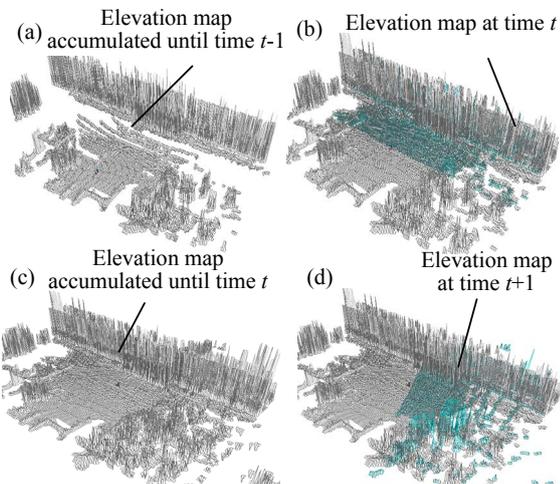


Fig. 7. Elevation map building of outdoor environment using ICP scan matching.

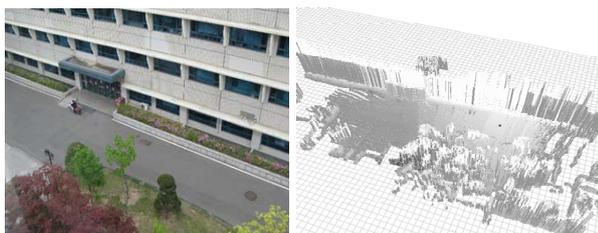


Fig. 8. Global elevation map for outdoor environments.

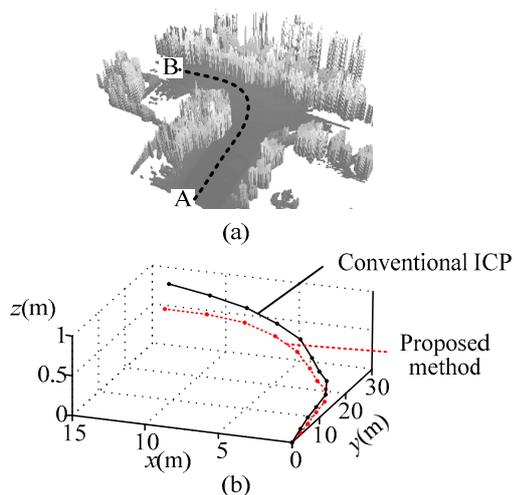


Fig. 9. Trajectory of robot during navigation on flat ground.

Figure 9 shows the comparison between the proposed ICP and the conventional ICP. The robot moved from region A to region B, where both regions were on a flat road (i.e., nearly horizontal surface). A robot could detect

a few trees and some parts of long walls that were used to estimate the robot pose. In case of executing the conventional ICP, the elevation of the robot increased continuously due to both the uncertainty of vertical structures and low convergence of flat surfaces. The elevation of region B was estimated to be 80 cm higher than that of region A. However, the difference of elevations between region A and B was estimated 40 cm from the proposed ICP algorithm. Therefore, the ICP using classified terrain estimated the robot pose more accurately than the ICP using 3-D point cloud only.

## 5. CONCLUSION

This paper describes a SLAM scheme with ICP in outdoor environments. A robot builds local elevation maps using range sensor data and integrates them into a global elevation map. For efficient map matching, a new terrain classification method was proposed based on the statistical measure of the points stored in each cell of the elevation map. The proposed terrain classification scheme can classify all the terrain data into 5 groups. The proposed method exploits the boundaries of the flat area (i.e., ‘horizontal edge’) instead of using the whole flat area and the  $z$ -direction of the robot pose was able to be accurately estimated by using the proposed terrain classification scheme.

## ACKNOWLEDGEMENT

This work was supported in part by the Agency for Defense Development and Unmanned Technology Research Center at KAIST and by the Center for Autonomous Intelligent Manipulation under Human Resources Development Program for Robot Specialists (Ministry of Knowledge Economy).

## REFERENCE

- [1] P. Pfaff and W. Burgard, “An efficient extension of elevation map for outdoor terrain mapping,” *Proc. of International Conference on Field and Service Robotics*, pp. 165-176, 2005.
- [2] S. RusinKiewicz and M. Levoy, “Efficient Variants of the ICP Algorithm,” *Proc. of International Conference on 3D Digital Imaging and Modeling*, pp. 145-152, 2001.
- [3] C. Frueh, and A. Zakhor, “3-D Model Generation for Cities Using Aerial Photographs and Ground Level Laser Scans,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 31-38, 2001.
- [4] R. Triebel and W. Burgard, “Improving Simultaneous Mapping and Localization in 3-D Using Global Constraints,” *Proc. of the National Conference on Artificial Intelligence*, pp. 1330-1335, 2005.