A New Feature Commonly Observed from Air and Ground for Outdoor Localization with Elevation Map Built by Aerial Mapping System

Tae-Bum Kwon
Cognitive Robotics Center, Korea Institute of Science and Technology, Seoul, Korea
e-mail: artistkwon@gmail.com
Jae-Bok Song
School of Mechanical Engineering, Korea University, Seoul, Korea
e-mail: jbsong@korea.ac.kr

Received 10 March 2010; accepted 17 August 2010

Monte Carlo localization (MCL) uses a reference map to estimate a pose of a ground robot in outdoor environments. However, MCL shows low performance when it uses an elevation map built by an aerial mapping system because three-dimensional (3D) environments are observed differently from the air and the ground and such an elevation map cannot represent outdoor environments in detail. Although other types of maps have been proposed to improve localization performance, an elevation map is still used as the main reference map in some applications. Therefore, we propose a new feature to improve localization performance with an elevation map. This feature is extracted from 3D range data and represents the part of an object that can be commonly observed from both the air and the ground. Therefore, this feature is likely to be accurately matched with an elevation map, and the average error of this feature is much smaller than that of unclassified sensing data. Experimental results in real environments show that the success rate of global localization increased and the error of local tracking decreased. Thus, the proposed feature can be very useful for localization of an outdoor ground robot when an elevation map is used as a reference map.

1. INTRODUCTION

Localization is one of the most important techniques for mobile robot navigation in indoor and outdoor environments. Various types of maps, including grid maps, elevation maps, and topological maps, can be exploited for localization (Choset & Nagatani, 2001; Kummerle, Triebel, Pfaff, & Burgard, 2008; Kwon & Song, 2008; Thrun, 1998). When a robot navigates with a range sensor in an indoor environment, the ground is flat. Therefore, only the distance to the object at a fixed height is measured by a sensor, and a two-dimensional (2D) grid map is sufficient for localization. In this case, the motion of a robot can be expressed by three-degree-of-freedom (DOF) motion (x, y, θ) in 2D space. In contrast, outdoor navigation usually requires the estimation of six-DOF motion (x, y, z, roll ψ, pitch θ, yaw φ) in three-dimensional (3D) space, so the environment should be represented by a 3D map for localization.

An elevation map is the most popular map to represent the 3D outdoor environment (Nashashibi, Fillatreau, Dacre-Wright, & Simeon, 1994; Parra, Murrieta, Devy, & Briot, 1999). In this map, the environment is regularly divided into small cells, for example 0.1×0.1 m, and each cell has its own elevation information. A common method of building an elevation map is to use an aerial mapping system equipped with both a global positioning system (GPS) and an inertial navigation system (INS) for localization and a light detection and ranging (LIDAR) sensor for range data acquisition. This type of map is suitable for a large outdoor environment and is used as the main map for special applications such as military vehicles (Frederick, Kania, Rose, Ward, Benz, et al., 2005). However, the elevation map cannot accurately describe the 3D environment, and it can cause an important problem in some applications. In this research, an elevation map built by an aerial mapping system was given in advance and used as the main reference map for localization.

If GPS/DGPS (differential GPS) data are available, the localization problem in the outdoor environment can be solved in a practical way to some extent. Therefore, little research has been done on outdoor localization with respect to a pregiven reference map, whereas much research in other fields such as environmental modeling based on the simultaneous localization and mapping (SLAM) technique (Cole & Newman, 2006; Durrant-Whyte & Bailey, 2006; Fairfield, Wettergreen, & Kantor, 2010; Katz, Melkumyan, Guivant, Bailey, Nieto, et al., 2006; Liu & Thrun, 2003) and obstacle negotiation (Ye & Borenstein, 2004) has been done thus far. However, GPS/DGPS-based localization schemes cannot be successfully used for certain applications

(e.g., military robots) that may be exposed to anti-GPS signals. Furthermore, GPS signals are often blocked by nearby tall buildings or trees, which severely degrade the accuracy of GPS-based localization. In this case, the given elevation map is the only reference data, and the range data obtained from sensors mounted on the robot should be matched with this reference map for localization.

Among many localization techniques such as Kalman filter–based localization (Lee, Kwon, & Song, 2007; Leonard & Durrant-Whyte, 1991), Markov localization (Fox, Burgard, & Thrun, 1998), and Monte Carlo localization (MCL) (Dellaert, Fox, & Thrun, 1999; Fox, Burgard, Dellaert, & Thrun, 1999), MCL has been the most widely used due to several advantages (Gutmann, Burgard, Fox, & Konolige, 1998; Thrun, Burgard, & Fox, 2005). Its good performance in indoor environments has been widely demonstrated in much literature, and in recent years, MCL has also been applied to outdoor localization problems. However, its performance in terms of the success rate of global localization and local tracking error was generally worse than that of indoor localization. Moreover, the computational cost of MCL in the outdoor environment is much higher than that in the indoor environment because of factors including the size of the environment, the complexity of map representation, and the large uncertainty of a map. In this research, MCL was also used for outdoor localization with an elevation map.

City modeling using a digital surface map (DSM), which is identical to the elevation map in this research, has been conducted by Frueh and Zakhor (2004). A DSM is used as a reference elevation map, and a vehicle pose with respect to the DSM can be estimated through the MCL method. The range data measured by a laser scanner are compared with data predicted from the DSM to estimate a vehicle pose. The path estimated by MCL is very accurate without error accumulation, even after more than 10 km, because it estimates a pose with respect to the reference map. However, it takes a very long time to generate the path because of high computational complexity, and thus the vehicle position cannot be estimated online. A similar study was conducted by Kummerle et al. (2008). This research also estimates a robot pose through the MCL method with a newly developed multilevel surface (MLS) map. The proposed map was very useful for outdoor localization because it could accurately describe the 3D environment. However, it was generally more difficult to build such a map compared to an elevation map. For some applications, an elevation map built by an aerial mapping system is the only available map for localization. In this case, other types of map such as the MLS map cannot be used to improve localization performance, and a new method that uses an elevation map as the main reference map may be more useful.

In this study, an elevation map built by an aerial mapping system was used as the main reference map to estimate the pose of a ground robot in large outdoor environments. Because some parts of 3D environments can be seen differently from the air and the ground, the sensing data of a mobile robot traveling on the ground may not be accurately matched with the elevation map built from the air. In this study, we propose a novel feature that can consider such a discrepancy between the 3D environments and its 2.5-D elevation map to improve localization performance. This feature can be used to match the real sensing data with the predicted data in MCL. A comparison of the proposed feature with other features for MCL shows its improved performance. The main contribution of this research is the improvement in the localization performance of an outdoor mobile robot using the proposed feature in outdoor environments when an elevation map built by an aerial mapping system is used as the main reference map.

When the proposed feature is used, a huge amount of 3D data can be handled efficiently and meaningfully. The proposed algorithm selects a subset of the 3D data, such as a 3D point cloud, to reduce the size of the data set for matching. This smartly selected data set is much smaller and easier to handle than the raw 3D data, and it can be matched accurately with the elevation map, avoiding errors due to the viewpoint difference. So our algorithm is very useful for a system that handles the high data volume of a 3D scan sensor such as a Velodyne, and experiments with mobile robot systems with a tilting laser scanner and a Velodyne will show the performance.

We also state that this study is related to our previous one (Kwon, Song, & Joo, 2010). In that research, elevation moment of inertia (EMOI) was proposed for MCL with an elevation map in an outdoor environment. It represents the distribution of elevation around a robot. The EMOI was also designed to improve the localization performance, especially the time for global localization. Both the EMOI and the new feature proposed in this paper deal with outdoor localization but are totally different from each other. After we investigated the results from the previous research, some limitations were found and they were clearly stated in the previous paper. To overcome these limitations and improve the localization performance with an elevation map, the new feature proposed in this paper was developed.

This paper is organized as follows. Section 2 describes the research background. Section 3 explains the concept of the proposed feature and extraction method. The usefulness of this feature is shown through experimental results described in Section 3, and the improvement of localization performance is analyzed in Section 4. Finally, Section 5 presents our conclusions.

2. VIEWPOINT-DEPENDENT CHARACTERISTICS OF ELEVATION MAP

2.1. Elevation Map Built by Aerial Mapping System

As mentioned in the Introduction, an elevation map is one of the most popular maps to represent outdoor environments, and this type of map is used as the main reference

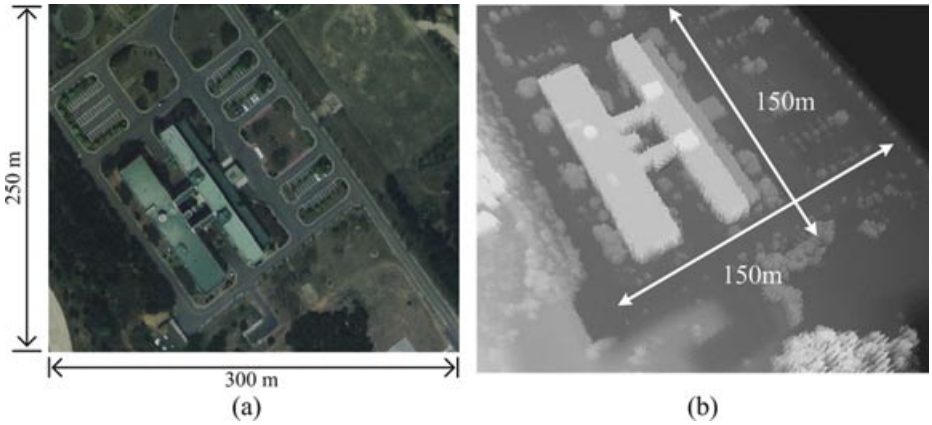

Figure 1. (a) Experimental environment and (b) its elevation map built by an aerial mapping system.

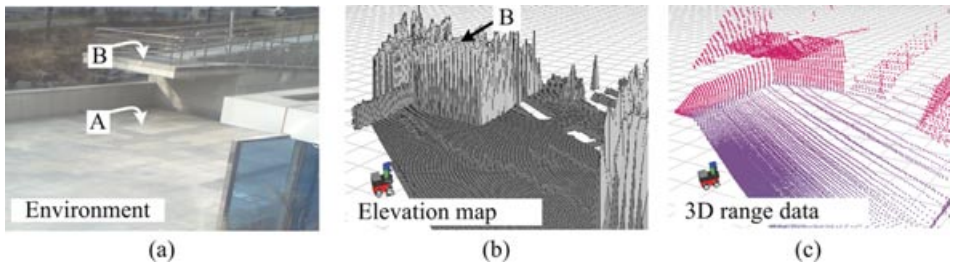

Figure 2. Examples of discrepancy between elevation map and real range sensor data.

map in this study. This 2.5-D map cannot accurately describe the 3D environment because each cell of an elevation map has only the highest elevation of the environment. However, it is suitable for large outdoor environments because of its use of a relatively small amount of memory in comparison with a real 3D environmental map. Moreover, this map can be relatively easily built by an aerial mapping system in a very large and complex environment. Figure 1 shows the experimental environment and its elevation map generated by an aerial mapping system. This environment includes various types of buildings, roads, trees, steeply inclined ground, and overhanging structures. The cell size of the map is 0.25×0.25 m.

2.2. Discrepancy between Environment and Elevation Map

Each cell of an elevation map has the highest elevation if more than two objects with different height values are located vertically in the same cell. For example, if part A on the lower floor and part B on the upper floor are located in the same cell of the elevation map in Figure 2(a), then this cell contains the elevation of part B, and the information on part A cannot be restored from the elevation map as shown in Figure 2(b). However, all parts of the environment shown in Figure 2(c) can be sensed using a 2D or 3D range sensor. A large discrepancy, therefore, might occur between the elevation map and the range sensor data in some parts of the environment. This discrepancy is attributed to the difference in viewpoint. That is, the environment was sensed by a mobile robot on the ground, whereas the identical environment was observed from the air when the elevation map was built. This is an important feature of an elevation map constructed by an aerial mapping system.

Many environments with different shapes can be modeled into one elevation map. For example, the environments in Figures 3(b)–3(d) were modeled into the

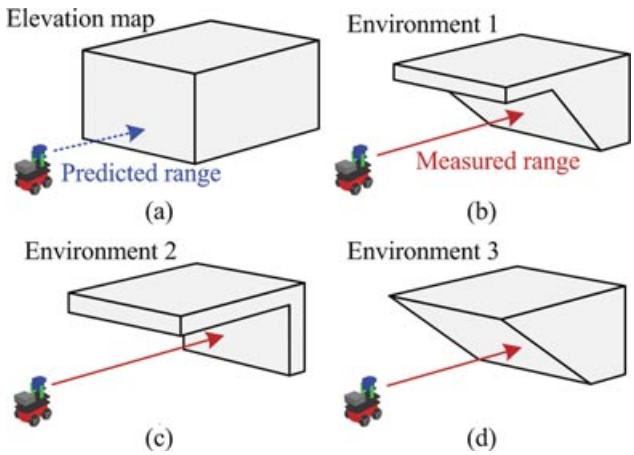

Figure 3. Examples of environments to be modeled as an identical elevation map.

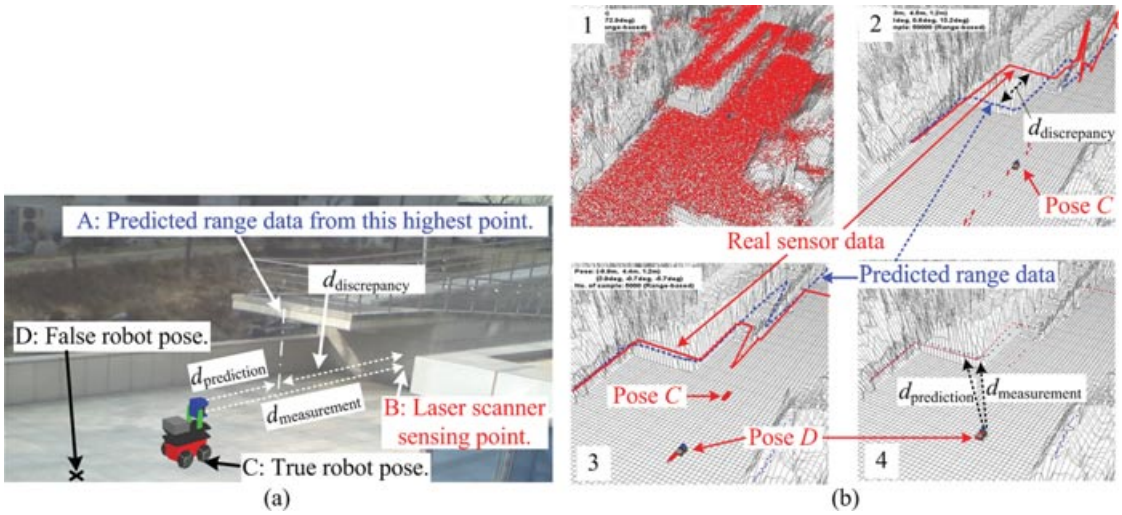

Figure 4. Incorrect convergence of samples during global localization because of discrepancy between elevation map and real range sensor data: (a) experimental environment and (b) convergence of samples.

identical elevation map shown in Figure 3(a) because of their identical greatest heights. A robot computed the range data from the elevation map shown in Figure 3(a) by a ray-casting algorithm to perform MCL for localization. This discrepancy deteriorates the accuracy of localization because MCL compares the range data measured from the real environments, such as those shown in Figures 3(b)–3(d), with those predicted from the map. The 3D ray-casting algorithm used in our experiments was implemented based on a 2D ray-casting algorithm included in CARMEN, which is the most famous open-source navigation program (CARMEN, 2008).

Figure 4 shows an example of global localization with an elevation map based on the conventional MCL approach in which a laser scanner is installed at a fixed tilt angle. The environment shown in Figure 4(a) and its elevation map are similar to environment 2 and its elevation map as shown in Figures 3(c) and 3(a), respectively. Suppose that the robot's true pose was pose C at the instant under consideration. As shown in Figure 4(b-2), the samples near pose C predicted the ranges to A of the elevation map ($d_{\text{prediction}}$), and the 2D laser scanner mounted on a robot measured the real range to B ($d_{\text{measurement}}$). These two ranges differed by $d_{\text{discrepancy}}$ as shown in Figure 4(a). However, some samples near pose D (i.e., incorrect robot pose) could predict $d_{\text{prediction}}$ similar to $d_{\text{measurement}}$, and samples tended to converge to pose D as shown in Figure 4(b-3) because samples whose predicted data are similar to the real sensing data have a high probability in MCL. As a result, the incorrect robot pose would likely be estimated as shown in Figure 4(b-4), which means the failure of MCL-based global localization. Similar problems can easily occur when there are any overhanging structures (such as a tree), and it is a major cause of poor localization performance when an elevation map is used as a reference map for outdoor localization.

3. A NEW FEATURE COMMONLY OBSERVED FROM AIR AND GROUND

In this research, we propose a novel feature that represents part of an object commonly observed from air and ground (COAG). Therefore, this feature can be accurately mapped into the elevation map built by an aerial mapping system and also sensed by a range sensor mounted on a mobile robot. A method to extract the COAG features from a 3D point cloud was also investigated, and the matching accuracy with the elevation map was analyzed.

3.1. Experimental Setup

Figure 5 shows a mobile robot system equipped with a SICK LMS291 laser scanner that was tilted by a dc motor with the tilt angle measured by a high-resolution encoder. The dc motor has sufficient power to keep the tilt angle at a desired angle during navigation. The absolute roll and pitch angles of the robot were sensed by an

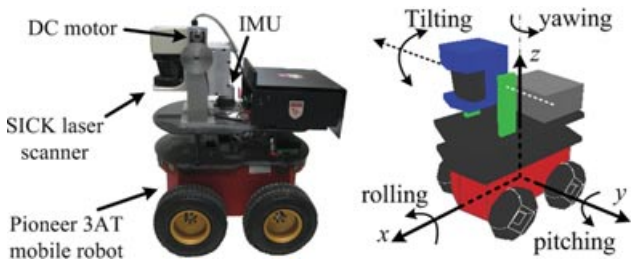

Figure 5. Robot used in the experiments.

inertial measurement unit (IMU), MI-GA3350M (MicroInfinity Corp.), and the yaw angle and motion increments were sensed by both the wheel encoder and the IMU. All sensors were carefully installed and calibrated with respect to the robot coordinate frame shown on the right in Figure 5. Combining these data allowed the estimation of six-DOF motion in the global coordinate frame (Lacroix, Mallet, Bonnafous, Bauzil, Fleury, et al., 2002). Because the absolute roll and pitch angles with respect to the global coordinate frame were measured by the IMU, the remaining four states (translations in x, y, and z axes, yawing) of the robot pose were estimated.

3.2. COAG Feature and Its Characteristics

COAG point features were classified into four groups as shown in Figure 6: commonly observed (CO), potentially commonly observed (PCO), not commonly observed (NCO), and unknown (UN). CO points are commonly observed from both the air and the ground, and NCO points are points that can be observed from the ground but not from the air. The points that could be observed from the air but not from the ground were not considered because such points are not useful for localization. On the other hand, PCO points are usually on the vertical surface, and the parts higher than PCO points were not sensed because of limited sensing ability as shown in Figure 6. They are, strictly speaking, not commonly observed from both the air and the ground. If higher parts can be sensed, the PCO points are likely to be classified into CO points because no objects block their observation from the air; thus these points are called PCO points.

When the elevation map of the environment is given and the range from a robot to an object is predicted from this map, the characteristics of the four types of points are as follows:

1. CO (commonly observed) point
   - It is highly probable that the elevation of the cell on which the CO point is located is correctly sensed.
   - The range to an object along the direction to the CO point is predicted from the elevation map, and it can be assumed that this range is very close to the range measured by a laser scanner.
   - It is the most useful information for localization.
2. PCO (potentially commonly observed) point
   - The parts higher than the PCO point are not fully sensed, mainly because the range of the tilt angle of a laser scanner is limited. Therefore, we cannot be assured that the elevation of the cell on which the PCO point is located is correctly sensed.
   - It is probable that the predicted range to an object on the map is close to the range measured by a laser scanner because the PCO point is most likely to be on the vertical surface.
   - It is relatively useful information for localization.
3. NCO (not commonly observed) point
   - It is highly probable that the elevation of the cell on which the NCO point is located is not correctly sensed.
   - It is assumed that the range to an object along the direction to the NCO point predicted from the elevation map is different from the range measured by a laser scanner.
   - It is harmful information for localization.
4. UN (unknown) point
   - The similarity of the sensing range to the range predicted from the elevation map cannot be estimated.
   - It is useless information for localization.

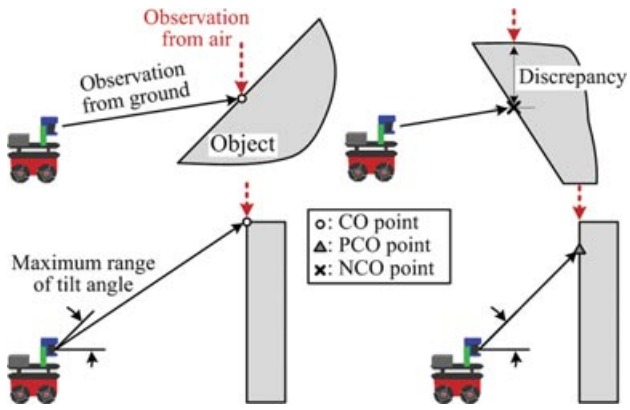

Figure 6. Examples of COAG features.

3.3. Extraction Scheme of COAG Features

A laser scanner was continuously tilted from 0 deg (forward) to −45 deg (upward) to collect 3D sensing data from which COAG features were extracted. First, all sensing points should be described in the local polar coordinate system as shown in Figure 7(a). In this system, each sensing point is described by (r_i, φ_i, z_i). The environment is divided at regular intervals to generate many partitions as shown in Figure 7(b). The appropriate angle of a partition, α, can be determined by the user ($\alpha = 2$ deg in this study). The environment is sensed in 3D space, and all sensing points belong to one of the partitions. Several sensing points with different elevations are usually included in each partition as shown in Figure 7(c). We defined the closest one to the origin in (r, φ) space as the *nearest point* regardless of its height z_i. This means that the nearest point has the minimum radius r in that partition. If the r of a new sensing point is smaller than the r of the nearest point while the laser scanner is tilted, the nearest point will be replaced by this new sensing point.

Algorithm 1 is the algorithm to extract the COAG feature in one partition, where k is the partition number, ${}^k\varphi_{\min}$ and ${}^k\varphi_{\max}$ are the boundary angles of partition k in (r, φ) space, $\theta_{\text{tilt,step}}$ is the tilt resolution of a laser scanner ($\theta_{\text{tilt,step}} = 1$ deg in this study), $<r_{\text{np}}, \varphi_{\text{np}}, z_{\text{np}}>$ is the

Algorithm 1. COAG feature extraction

1: **Algorithm: COAG feature extraction in partition k**
2: $n = 0$
3: for $\theta_{\text{tilt}} = 0^\circ$ to $\theta_{\text{tilt,min}}$ with -$\theta_{\text{tilt,step}}$
4: calculate $<r_s, \varphi_s, z_s>$ using sensing data and tilt angle
5: if $(\varphi_s >= {}^k\varphi_{\min})$ & $(\varphi_s < {}^k\varphi_{\max})$
6: $n = n + 1$
7: $<r_n, \varphi_n, z_n> = <r_s, \varphi_s, z_s>$
8: endif
9: endfor
10: $<r_{np}, \varphi_{np}, z_{np}> = <r_1, \varphi_1, z_1>$
11: $C_{np} = UN$
12: for $i = 1$ to n
13: if $(r_i > r_{np} + \varepsilon)$
14: $C_{np} = CO$
15: elseif $(|r_i - r_{np}| < \varepsilon)$
16: $C_{np} = PCO$
17: $<r_{np}, \varphi_{np}, z_{np}> = <r_i, \varphi_i, z_i>$
18: elseif $(r_i < r_{np} - \varepsilon)$
19: $C_{np} = NCO$
20: $<r_{np}, \varphi_{np}, z_{np}> = <r_i, \varphi_i, z_i>$
21: endif
22: endfor
23: return $<r_{np}, \varphi_{np}, z_{np}>$ & C_{np}

location of the nearest point, C_{np} is the class of the nearest point, and ε is a small margin for sensing error and noise. When a laser scanner is tilted from 0 to -45 deg ($\theta_{\text{tilt,min}}$) in this experiment, the positions of many sensing points in 3D space are continuously computed (lines 3–9) and some of them will be stored for COAG feature extraction if they are in partition k (lines 5–8). The position and class of the nearest point are initialized (lines 10–11), and the nearest point in partition k is classified (lines 12–22). Figure 8 also shows

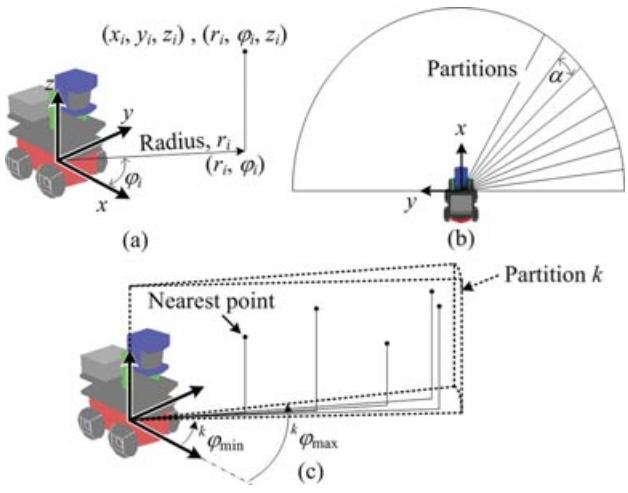

Figure 7. (a) Coordinate systems, (b) partitions divided at regular intervals, and (c) example of one partition including several sensing points.

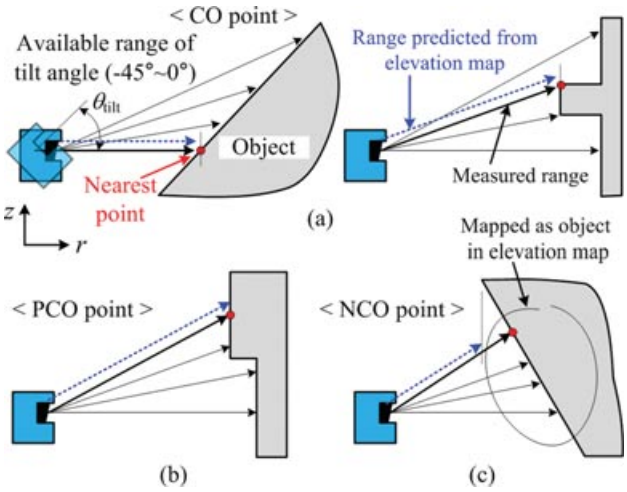

Figure 8. Classification of nearest point using sensing points included in one partition: (a) CO, (b) PCO, and (c) NCO.

the classification scheme. In Figure 8(a), the nearest point is classified as a CO point because the points higher than the nearest point are farther than the nearest point in (r, z) space (lines 13–14). In Figure 8(b), the nearest point is classified as a PCO point because there are some points whose radii are similar to the nearest point in (r, z) space (lines 15–17). Unlike a CO point, there should be no point that is both farther from and higher than the nearest point. In Figure 8(c), the radius (r_{np}) and height (z_{np}) of the nearest point become shorter and higher, respectively, as the scanner is tilted from 0 to -45 deg (lines 18–20), and this type of nearest point is classified as a NCO point. Unlike a CO point, there is no point that is both farther from and higher than the nearest point in (r, z) space. In addition, unlike a PCO point, there is no point whose radius is close to the nearest point. If classification is impossible because there is just one available sensing point in partition k, that sensing point will remain as a UN point. Finally, the position and class of the nearest point of partition k are returned (line 23). The nearest point of each partition in Figure 7(b) is classified using the proposed algorithm. Note that each partition has only one feature point, if any. Algorithm 1 shows only the feature extraction in one partition to easily explain the algorithm, but this task can be executed in parallel for each partition. So all COAG features of all partitions are extracted in a single 3D scan, and the whole algorithm complexity is $O(n)$, where n is the number of points in the point cloud obtained by tilting a laser scanner.

The algorithm for COAG feature extraction was explained without considering the roll and pitch angles of the robot in Algorithm 1 and Figures 7 and 8 for an easy and simple explanation. However, the angles should be considered in an outdoor environment. The x–y plane of the local coordinate system in Figure 7 is parallel to the

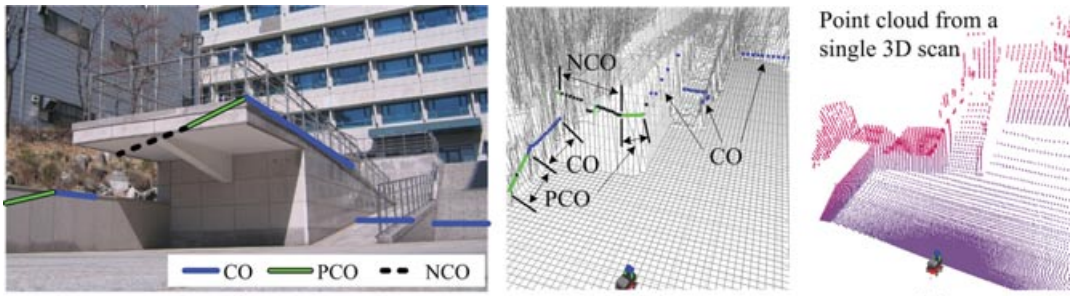

Figure 9. Experimental results of COAG feature extraction: (a) environment, (b) extraction result superimposed on elevation map, and (c) 3D point cloud.

x–y plane of the global coordinate system, and so the roll and pitch angles can be easily considered in line 4 of Algorithm 1. The laser scanner should also be tilted from positive angle (downward) to −45 deg with respect to the robot coordinate system in Figure 5 if necessary, and line 3 of Algorithm 1 should be modified. All experiments in this paper were performed considering the roll and pitch angles of the robot.

3.4. Extraction Results and Error Comparison

Figure 9 shows an example of extraction of the COAG feature in a real environment. The sensing points corresponding to the top of the structure were classified as CO points, and the range to these points predicted from the elevation map might be close to the real sensor data measuring the range to the points classified as CO points in the environment. Some sensing points for the vertical part of the structure were classified as PCO points. The predicted range to these points might also be close to the range measured by a sensor, but PCO points can be less accurate than CO points. The sensing points for the lower part of the overhanging structure were grouped as NCO points, and the range to these points predicted from the elevation map might be very different from the real measurements because this part was not correctly mapped in the elevation map. The experimental results show that the CO points can be matched well with an elevation map but the NCO points cannot be matched.

The errors between the real sensor data and the data predicted from the map are very important in MCL because all samples converge to a small region in which the errors of samples are minimized. MCL will not be able to accurately estimate a robot pose without the proper model of the error between a true robot pose and a hypothesis (sample). In the environment shown in Figure 9(a), a robot obtains the 3D range data shown in Figure 9(c). There is an overhanging structure, and only the shape of the structure is mapped into the elevation map. The space under the structure is empty, and thus the shape detected by a laser scanner as shown on the left-hand side of Figure 10(a) is very different from the elevation map if the tilt angle of a laser scanner is fixed. To install a laser scanner horizontally or in parallel with the local coordinate system with a fixed tilt angle is a common approach to sense the environment in conventional MCL. This approach causes a large error between the range sensor data and the predicted range data from the elevation map using a ray-casting algorithm. On the left-hand side of Figure 10(c), a robot extracts CO and PCO points that correspond to the top of the trees and overhanging structures, and they do not penetrate the cells whose elevations are higher than the beams of a laser scanner. The range data to objects along the directions to the CO and PCO points are predicted from the elevation map shown on the right-hand side of Figure 10(c), and they are very similar to the CO and PCO points extracted from the real sensor data. Figure 10(b) shows the error of each partition for the experiment in Figure 10(a). The dark (red) and light (green) points denote sensing and predicted points, respectively, and the lines show the errors between them. The average error was 3.1 m when the unclassified sensing points with a fixed tilt angle were used, but this error was reduced to 0.6 m when the proposed CO and PCO points were used. That is, the error for the unclassified sensing points was much larger than the error for the CO and PCO points because of the viewpoint-dependent characteristics of an elevation map. When there are many trees or overhanging structures in the environment, the difference between the two errors will become large. This is always true if the elevation map is properly built and the sensor data are accurate.

Figures 11(b) and 11(c) show other examples, and Figure 11(a) shows the part of the environment where the results in Figure 11(b) were obtained. The maximum range of laser scanner data was 30 m in this study. The magnitude of the error was dependent on several characteristics of the environment, including the amount of range data longer than the maximum available range of a laser scanner, the number of trees, and the number of overhanging structures. Therefore, a theoretical analysis that is adaptable to every

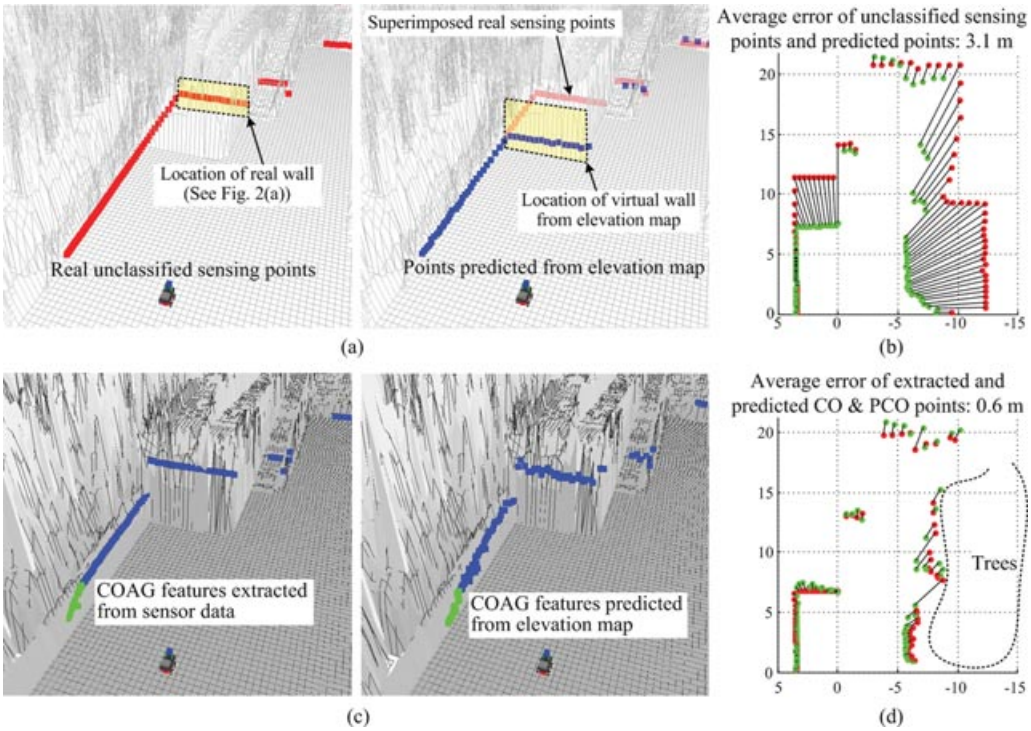

Figure 10. Comparison of real sensor data and predicted data: (a) unclassified range data and predicted range data, (b) average error of unclassified sensing points, (c) extracted COAG feature and COAG feature predicted from elevation map, and (d) average error of COAG features.

environment is impossible, and only a statistical analysis for each environment is possible.

In the environment shown in Figure 1, errors between the real sensor data and the data predicted from correct robot poses were computed from more than 1,000 random places using an actual robot. The results are shown in Table I. When all points (CO, PCO, NCO, and UN) were used, the average error was 2.61 m. This error was reduced using only the CO and PCO points that could be correctly matched with the elevation map. When the unclassified sensing points were used, the average error was 5.75 m, which was much larger than the error for COAG features. This verifies that the classified sensing points, especially the CO and PCO points, were more useful than the unclassified sensing points for outdoor localization with an elevation map. The absolute values shown in Table I are meaningful for the environment shown in Figure 1, but the errors for the CO and PCO points are always smaller than the errors for unclassified sensing points in every environment. The errors seem to be relatively higher than those in other literature, and this will be analyzed in Section 4.3.

It might seem that this comparison is unfair because COAG features are the points properly selected from a much larger set of information, 3D point clouds. However, it is an improvement of the proposed features compared with the previous features, unclassified sensing points, and the results in Table I experimentally support the idea that COAG feature-based matching is more advantageous than conventional matching. The proposed features, of course, have drawbacks, although their accuracy is improved as explained in this section, and this will be mentioned in detail in Section 5.

Table I. Statistical analysis of errors between sensor data and predicted data.

Average errors of COAG features (m)		Average error of unclassified sensing points (m)
CO and PCO points	1.51	5.75
CO, PCO, NCO, and UN points	2.61	5.75
NCO and UN points	8.2	5.75

4. MCL-BASED OUTDOOR LOCALIZATION USING COAG FEATURES

MCL was used as the main paradigm for localization in this study. It is one of the popular Bayesian filters that can estimate the distribution of probability using a set of random

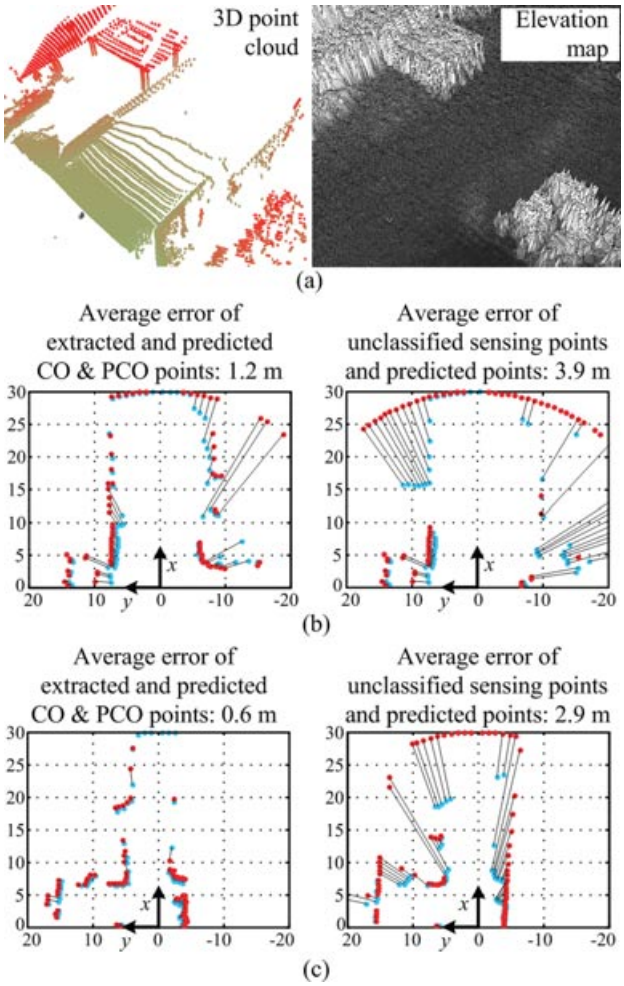

Figure 11. Two comparisons of real sensor data and predicted data.

samples. At each time step, the probabilities of samples were updated using a motion model and a sensor model, and then the samples were resampled. The state (robot pose in this case) was represented by the weighted sum of all samples. More details on MCL can be found in Doucet, Godsill, and Andrieu (2000), Frueh and Zakhor (2004), and Thrun et al. (2005).

4.1. Computation of Sample Weighting Using COAG Features

Each sample has the range data predicted from the map, which are compared to the real sensor data in the MCL. The difference between the two sets of data is transformed into the weighting of that sample. The weighting is calculated using a sensor model $p(z|x)$, where x and z are a state and an observation, respectively. The most popular sensor model has been used for a laser scanner by many researchers (Dellaert et al., 1999; Fox et al., 1998, 1999; Frueh & Zakhor, 2004; Kummerle et al., 2008; Lacroix et al., 2002; Thrun et al., 2005):

$$p(z|x) = \prod_{k=1}^{K} p({}^k z|x), \tag{1}$$

where K is the number of sensing points, or beams, of one laser scan z. The weightings for all beams are multiplied because it is assumed that each beam was independent. The sensor model $p({}^k z|x)$, which is a sum of three different distributions, is given by

$$p({}^k z|x) = \alpha_{\text{obst}} \cdot p_{\text{obst}}({}^k z|x) + \alpha_{\text{rand}} \cdot p_{\text{rand}}({}^k z|x) + \alpha_{\text{max}} \cdot p_{\text{max}}({}^k z|x), \tag{2}$$

where $p_{\text{obst}}({}^k z|x)$ calculates a weighting based on the difference between the predicted data and the sensor data and is a Gaussian distribution with variance σ^2. $p_{\text{rand}}({}^k z|x)$ models random measurements and is a uniform distribution from zero to the maximum sensing range. $p_{\text{max}}({}^k z|x)$ models the maximum range measurements using a point mass distribution. The three coefficients α_{obst}, α_{rand}, and α_{max} can be empirically determined.

Figure 12(a) shows the sensor model used in this study. The values of α_{obst}, α_{rand}, and α_{max} were set to 0.05, 0.001, and 0.005, respectively. The variance σ^2 was chosen as 1.0 m^2, which was empirically estimated with respect to the

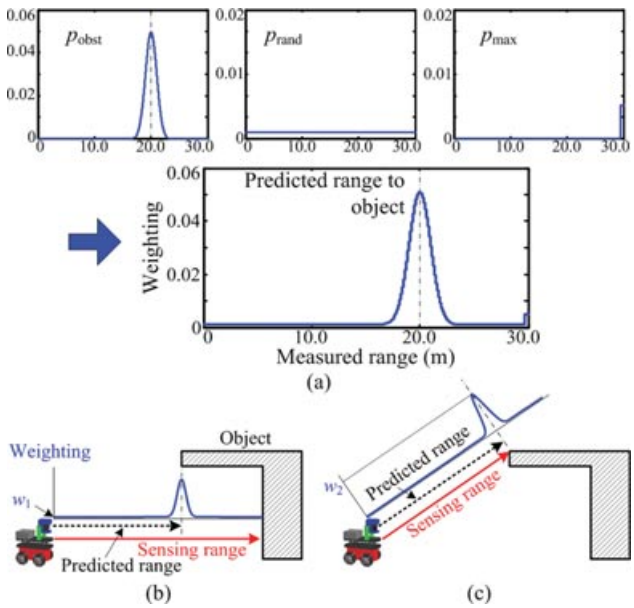

Figure 12. Weighting computation by sensor model: (a) sensor model, (b) unsuitable weighting computation for matching with unclassified sensing points, and (c) suitable weighting computation for matching with classified sensing point.

uncertainty of a sensor, the size of a cell, the uncertainty of a map, and so on. This variance was rather large compared to the σ^2 of other studies, but it was a suitable value because the cell size was large and the accuracy of an elevation map built by an aerial mapping system is not very high. In Figure 12(b), the range predicted from the elevation map differed from the range sensor data even though the robot pose was correct because of the viewpoint-dependent characteristics of an elevation map. Therefore, the weighting w_1 of a sample was assigned a very low value, which mainly depended on $p_{\text{rand}}({}^k z|x)$ in Eq. (2). It could cause samples to converge to an incorrect pose during global localization because the samples close to the correct robot pose might be easily eliminated in resampling steps of the MCL. This problem can often occur in outdoor environments with an elevation map, which is a major cause of poor localization performance as mentioned in Section 2. The classified points, however, were used as shown in Figure 12(c), and a robot could predict a range that was very close to the sensing range on the correct robot pose. Therefore, the weighting w_2 of a sample became very high because the CO and PCO points were more likely to be matched accurately with an elevation map than the unclassified sensing data.

4.2. Results of Global Localization

To remedy poor localization performance due to the viewpoint-dependent characteristics of an elevation map, we proposed the use of COAG features. The use of these features enables the viewpoint-dependent characteristic of an elevation map to be taken into consideration in localization, which results in significantly improved localization performance.

First, global localization was performed in the real environment in Figure 1. Figure 13 shows the several stages of global localization by MCL using COAG features (MCL/COAG). The initial number of samples was 100,000, and the CO and PCO points were used for matching. Figure 14(a) depicts the convergence of samples to the correct pose as a function of the resampling step. The convergence rate R is defined as

$$R = \frac{\text{Number of samples closer than 1.0 m to correct position}}{\text{Number of all samples}}. \tag{3}$$

Using the CO and PCO points, it took fewer resampling steps for samples to converge to the correct pose because they were more accurate than the unclassified sensing points obtained with a fixed tilt angle. Figure 14(b) shows the success rate of global localization, and we assumed that global localization was successful when all samples were closer than 1.0 m to the correct location of a robot, or $R = 1.0$, in fewer than 15 resampling steps. A similar criterion was used in other studies in defining the success rate of

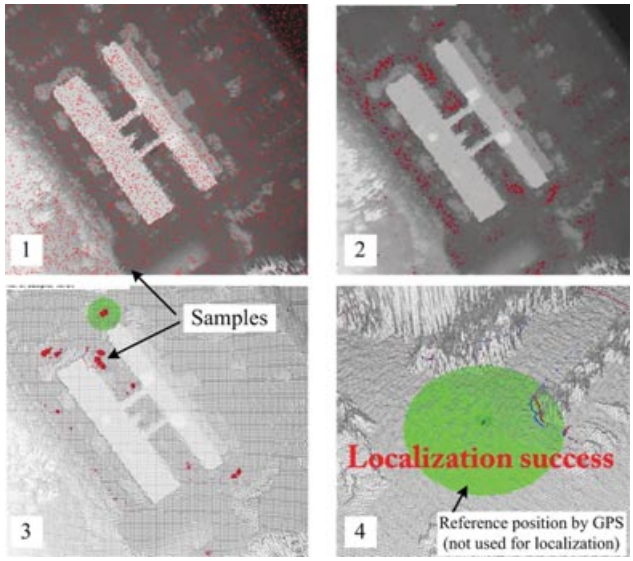

Figure 13. Experimental results from global localization.

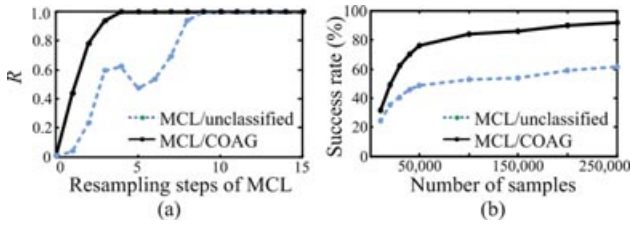

Figure 14. Characteristics of sample convergence: (a) sample convergence to correct pose versus resampling steps (initial number of samples: 100,000) and (b) success rate of global localization versus number of samples.

global localization (Kummerle et al., 2008). A robot started from arbitrary locations in the experimental environment with different numbers of samples that were globally distributed as shown in Figure 13-1. It follows that the success rate of global localization by MCL/COAG was higher than the success rate by MCL using unclassified sensing points (MCL/unclassified).

4.3. Results of Local Tracking

In the local tracking experiments, a robot was controlled manually to travel more than 600 m. In Figure 15, the dark (black) solid line, light (white) solid line, and light (yellow) dotted line show the path estimated by MCL/COAG and by MCL/unclassified and the odometric path, respectively. One thousand samples were used, and all processes were executed online on a 1.7-GHz notebook computer in which about 100 ms was required to perform one cycle of MCL, which includes all processes of MCL such as motion prediction, measurement update, weighting computation,

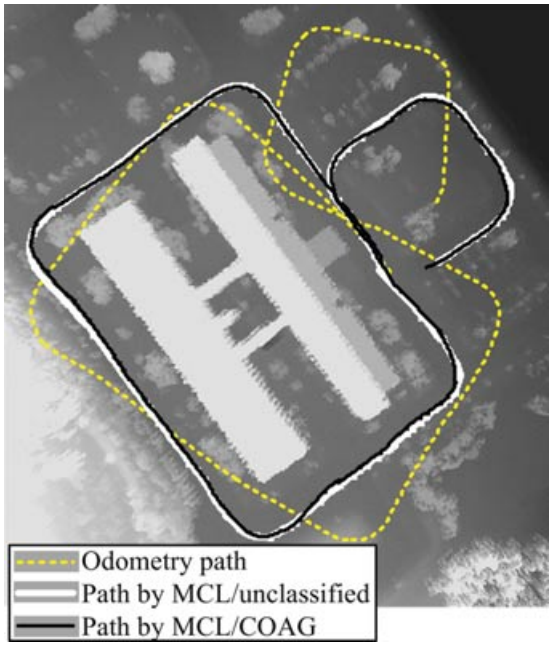

Figure 15. Paths estimated by odometry and MCL/COAG.

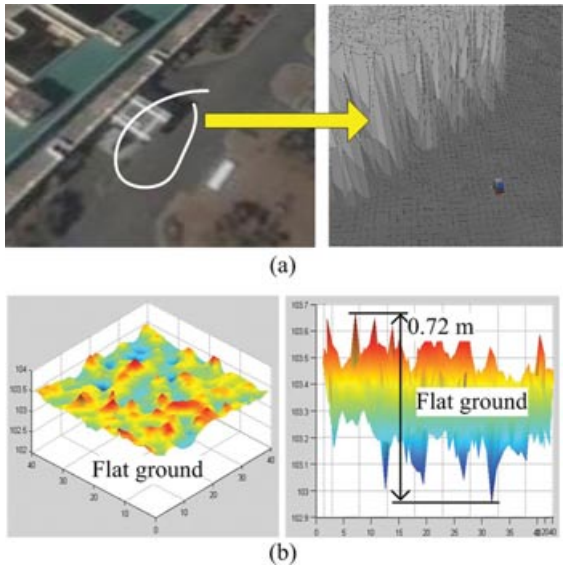

Figure 17. Examples of uncertainty of elevation map built by aerial mapping system: (a) roof and wall and (b) flat ground.

normalization, resampling, and so on. Figure 16 shows the translational errors of local tracking. The average errors of MCL/unclassified and MCL/COAG are 2.0 and 1.2 m, respectively. Their average rotation errors (yaw axis) are about 5 and 2 deg, respectively. The matching of COAG feature points showed more accurate results than the matching of unclassified sensing points. Moreover, MCL/COAG did not fail to track a robot pose along the trajectory in Figure 15 when more than 100 experiments were conducted, but local tracking based on MCL/unclassified easily failed after the robot traveled about 250 m. The path by MCL/unclassified in Figure 15 was estimated when MCL/unclassified did not fail. Note that the robot started from the same arbitrary locations on the trajectory for both cases. Therefore, the accuracy and reliability of local tracking by MCL/COAG were much better than the accuracy and reliability by MCL/unclassified.

The average errors of local tracking (2.0 and 1.2 m) seem to be very large compared to the results from other studies. However, the cell size (0.25 m) in this study is somewhat large, and there are many discrepancies between the environment and the elevation map because of the viewpoint-dependent characteristics of the elevation map. Moreover, the accuracy of the elevation map is relatively low, as shown in Figure 17, because an aerial mapping system measured the distance to the terrain using a LIDAR sensor at an altitude of up to 2 km. The height difference between the highest and the lowest cells for a flat ground may be very large, as shown in Figure 17(b). Therefore, we think that the errors of local tracking are within a reasonable range.

An aerial mapping system can relatively easily build an elevation map in a very large environment, but its accuracy is limited. We are investigating methods to measure the accuracy and error of an elevation map built by an aerial mapping system compared to the real environment, and this will be a subject for additional research.

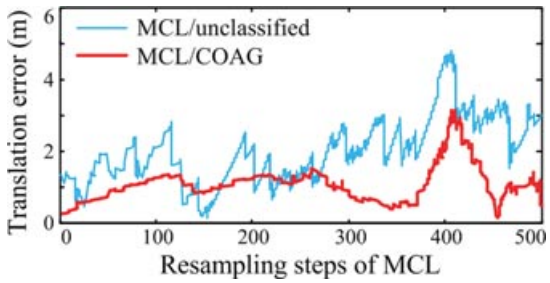

Figure 16. Translation errors between estimated positions and true positions.

5. FURTHER CONSIDERATIONS

Though the MCL-based localization performance in outdoor environments can be improved by the proposed COAG features, localization with COAG features also has limitations. First, the proposed approach cannot work well with overhanging structures higher than the maximum height that a robot can sense. This is because of the limited sensing ability of a system, and the proposed COAG features were designed to minimize the effects of this limitation by sensing point classification. In Figure 9, for example, our robot did not sense the whole parts of the overhanging

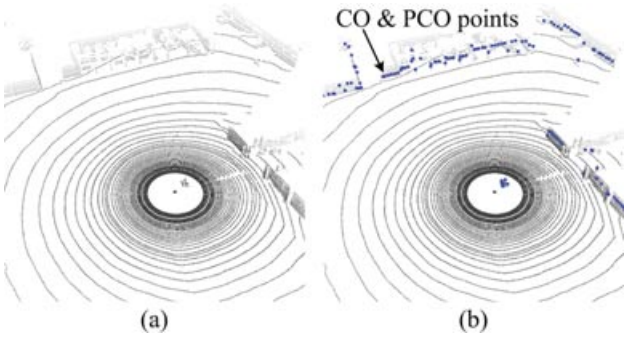

Figure 18. Feature extraction from 3D range data: (a) 3D range data from a Velodyne and (b) COAG feature points.

structure and all points in a certain partition were obtained only from the bottom of the overhanging structure because the tilting angle of the laser scanner was limited from 0 to −45 deg.

However, all points were properly classified into CO, PCO, and NCO points, and more reliable points could be used for matching. Undoubtedly, the classification result, however, may be incorrect in some situations. For example, if there is a vertical wall that is taller than the maximum height a robot can sense and an overhanging structure such as an eave is on the top of the wall, only the wall is sensed and the sensing points on the wall will be classified into PCO points though they should be classified into NCO points because the elevation map represents just the overhanging structure and not the wall. And if the robot is maneuvering 1 m away from a wall, then most points on that wall will be classified as PCO points regardless of the actual structure. A too-short distance to obstacles may deteriorate the classification performance. These may cause inaccurate matching and localization failure. They are limitations of COAG features, and research on solving them is currently underway.

Another important limitation is the need to stop the robot while each 3D scan data set is collected, tilting the

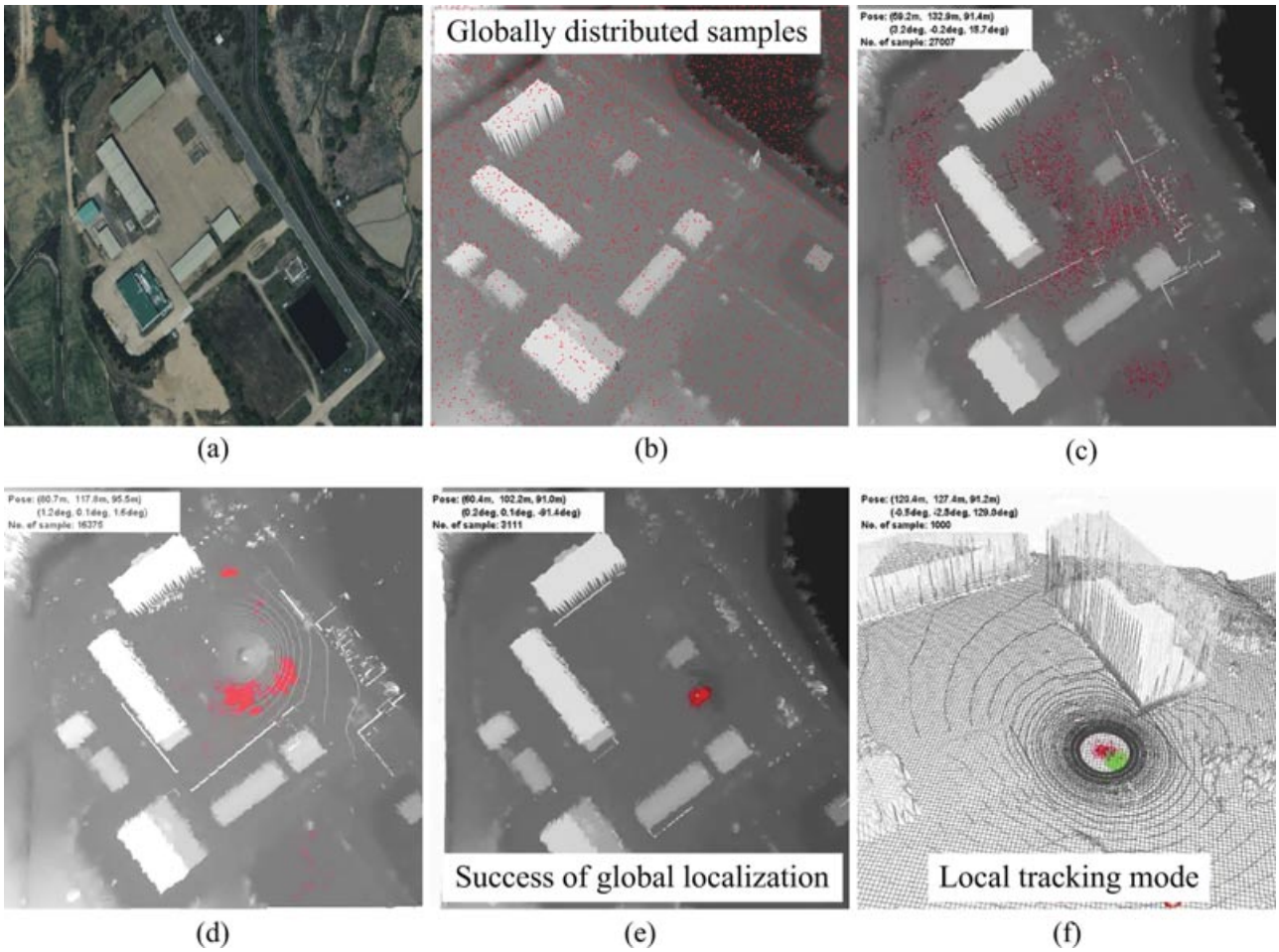

Figure 19. Experimental results from global localization using a Velodyne: (a) experimental environment and (b)–(f) sample convergence during global localization.

laser scanner. In our system, it takes about 1.5 s to tilt the laser scanner from 0 deg (forward) to −45 deg (upward). This approach (so-called stop-scan-go fashion), however, is common in mapping and localization research with a laser scanner tilted by a motor because tilting a laser scanner in motion can cause large errors due to vibrations, measurement errors of tilt angles, range errors, and so on (Kummerle et al., 2008; Nuchter, Lingemann, Hertzberg, & Surmann, 2007). A commercial 3D LIDAR such as a Velodyne can be used to get reliable 3D range data in motion, and recently it has been widely used in much research. Our algorithm was also tested using a Velodyne, and Figures 18(a) and 18(b) show the 3D range data of a single scan of a Velodyne sensor and the extracted COAG feature points, respectively.

The proposed approach can be used as a very good criterion for selecting points that are suitable for matching with a map when a Velodyne is used. Choosing suitable sensing points for matching is essential because a Velodyne gives a huge amount of sensing data and it is impossible to match all of them with a map. Figure 19 shows the result of global localization using COAG features with a Velodyne, and this result verifies that the proposed COAG features are also useful for a system equipped with a Velodyne. In this experiment, the robot did not stop to get 3D scan data and the COAG feature extraction algorithm worked within the 120-m range, which was the maximum available range of a Velodyne, although we used the laser scanner data only within the 30-m range, which was the maximum available range of a SICK LMS laser scanner with millimeter resolution.

6. CONCLUSIONS

MCL-based outdoor localization with an elevation map is difficult. Its performance is worse than that of indoor localization because an elevation map built by an aerial mapping system cannot accurately describe the 3D environment, and there may be many discrepancies between them. To overcome the effects of these viewpoint-dependent characteristics of an elevation map and improve localization performance with an elevation map, we proposed a new method to classify 3D sensing points into four groups (CO, PCO, NCO, and UN) according to the possibilities of being matched with an elevation map. These classified points are more suitable for matching with an elevation map than the unclassified sensing points, and localization performance can be improved using these COAG feature points.

Experiments on the extraction of COAG feature points were carried out using real sensor data. The CO and PCO points showed much smaller average errors than the unclassified sensing points in matching with an elevation map. This result improves the localization performance by allowing the more correct weighting computation of a sample in the case of the existence of a discrepancy between an environment and its elevation map. The success rate of global localization increased, and a robot could find its pose with fewer resampling steps when COAG feature points were used for matching. Furthermore, the error of local tracking was reduced, and MCL/COAG did not fail to track a robot pose, whereas MCL/unclassified easily failed. Therefore, the proposed COAG feature is considered to be very useful for outdoor localization based on the MCL scheme when an elevation map is used as the main reference map.

ACKNOWLEDGMENTS

This work was supported in part by the Agency for Defense Development and the Unmanned Technology Research Center at KAIST and in part by the Intelligent Robotics Development Program (Frontier R&D Program) funded by the Ministry of Knowledge Economy.

REFERENCES

CARMEN. (2008). Robot Navigation Toolkit 0.7.4_beta. Retrieved October 23, 2008, from http://carmen.sourceforge.net/.

Choset, H., & Nagatani, K. (2001). Topological simultaneous localization and mapping (SLAM): Toward exact localization without explicit localization. IEEE Transactions on Robotics and Automation, 17(2), 125–137.

Cole, D. M., & Newman, P. M. (2006, May). Using laser range data for 3D SLAM in outdoor environments. Paper presented at the IEEE International Conference on Robotics and Automation, Orlando, FL.

Dellaert, F., Fox, D., & Thrun, S. (1999, May). Monte Carlo localization for mobile robots. Paper presented at the IEEE International Conference on Robotics and Automation, Detroit, MI.

Doucet, A., Godsill, S., & Andrieu, C. (2000). On sequential Monte Carlo sampling methods for Bayesian filtering. Statics and Computing, 10(3), 197–208.

Durrant-Whyte, H., & Bailey, T. (2006). Simultaneous localization and mapping: Part I, The essential algorithms. IEEE Transactions on Robotics and Automation Magazine, 13(2), 99–110.

Fairfield, N., Wettergreen, D., & Kantor, G. (2010). Segmented SLAM in three-dimensional environments. Journal of Field Robotics, 27(1), 85–103.

Fox, D., Burgard, W., Dellaert, F., & Thrun, S. (1999, July). Monte Carlo localization: Efficient position estimation for mobile robots. Paper presented at the 6th National Conference on Artificial Intelligence, Orlando, FL.

Fox, D., Burgard, W., & Thrun, S. (1998). Active Markov localization for mobile robots. Robotics and Autonomous Systems, 25(3–4), 195–207.

Frederick, P., Kania, R., Rose, M. D., Ward, D., Benz, U., Baylot, A., Willis, M. J., & Yamauchi, H. (2005, October). Spaceborne path planning for unmanned ground vehicles (UGVs). Paper presented at the IEEE Conference on Military Communications (MILCOM), Atlantic City, NJ.

Frueh, C., & Zakhor, A. (2004). An automated method for large-scale ground-based city model acquisition. International Journal of Computer Vision, 60(1), 5–24.

Gutmann, J. S., Burgard, W., Fox, D., & Konolige, K. (1998, October). An experimental comparison of localization methods. Paper presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, Victoria, Canada.

Katz, R., Melkumyan, N., Guivant, J., Bailey, T., Nieto, J., & Nebot, E. (2006, October). Integrated sensing framework for 3D mapping in outdoor navigation. Paper presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China.

Kummerle, R., Triebel, R., Pfaff, P., & Burgard, W. (2008). Monte Carlo localization in outdoor terrains using multilevel surface maps. Journal of Field Robotics, 25(6–7), 346–359.

Kwon, T. B., & Song, J. B. (2008). Thinning-based topological exploration using position possibility of topological nodes. Advanced Robotics, 22(2–3), 210–220.

Kwon, T. B., Song, J. B., & Joo, S. H. (2010). Elevation moment of inertia: A new feature for Monte Carlo localization in outdoor environment with elevation map. Journal of Field Robotics, 27(3), 371–386.

Lacroix, S., Mallet, A., Bonnafous, D., Bauzil, G., Fleury, S., & Herrb, M. (2002). Autonomous rover navigation on unknown terrains: Functions and integration. International Journal of Robotics Research, 21(10–11), 917–942.

Lee, Y. J., Kwon, T. B., & Song, J. B. (2007). SLAM of a mobile robot using thinning-based topological information. International Journal of Control, Automation and Systems, 5(5), 577–583.

Leonard, J., & Durrrant-Whyte, H. (1991). Mobile robot localization by tracking geometric beacons. IEEE Transactions on Robotics and Automation, 7(3), 376–382.

Liu, L., & Thrun, S. (2003, September). Results for outdoor-SLAM using sparse extended information filters. Paper presented at the IEEE International Conference on Robotics and Automation, Taipei, Taiwan.

Nashashibi, F., Fillatreau, P., Dacre-Wright, B., & Simeon, T. (1994, May). 3-D autonomous navigation in a natural environment. Paper presented at the IEEE International Conference on Robotics and Automation, San Diego, CA.

Nuchter, A., Lingemann, K., Hertzberg, J., & Surmann, H. (2007). 6D SLAM—3D mapping outdoor environments. Journal of Field Robotics, 24(8–9), 699–722.

Parra, C., Murrieta, R., Devy, D., & Briot, M. (1999). 3-D modelling and robot localization from visual and range data in natural scenes. Lecture Notes in Computer Science, 1542, 450–468.

Thrun, S. (1998). Learning metric-topological maps for indoor mobile robot navigation. Artificial Intelligence, 99(1), 21–71.

Thrun, S., Burgard, W., & Fox, D. (2005). Probabilistic robotics. Cambridge, MA: MIT Press.

Ye, C., & Borenstein, J. (2004). A novel filter for terrain mapping with laser rangefinders. IEEE Transactions on Robotics, 20(5), 913–921.