The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Mobile Robot Localization Using Stereo Vision in Outdoor Environments under Various Illumination Conditions

Kiyoshi Irie, Tomoaki Yoshida, and Masahiro Tomono

Abstract— **This paper proposes a new localization method for outdoor navigation using a stereo camera only. Vision-based navigation in outdoor environments is still challenging because of large illumination changes. To cope with various illumination conditions, we use 2D occupancy grid maps generated from 3D point clouds obtained by a stereo camera. Furthermore, we incorporate salient line segments extracted from the ground into the grid maps. This grid map building is not much affected by illumination conditions. On the grid maps, the robot poses are estimated using a particle filter that combines visual odometry and map-matching. Experimental results showed the effectiveness and robustness of the proposed method under various weather and illumination conditions.**

I. Introduction

Autonomous navigation in outdoor environments is an important issue in mobile robotics. One of the key issues in outdoor navigation is localization under various conditions. Localization for outdoor navigation has been studied extensively and there have been proposed many methods combining inertial sensors (such as odometry), and external sensors such as GPS, laser scanners and cameras [1][2][3].

Stereo vision-based methods are a promising approach to mobile robot localization. Stereo cameras can obtain 3D range data at high frame rate and also capture colors and textures, which cannot be detected sufficiently by laser scanners. Recently, image features with distinctive local descriptors, such as SIFT, have been employed for mobile robot localization [4]. These image features are useful to identify landmarks in indoor environments without large illumination changes. However, drastic illumination changes, which often occur in outdoor environments, make it difficult to obtain stable image features.

Fig. 1 shows examples of image feature extraction in outdoor environments. The middle column of the figure shows SIFT key points detected from two images of the same place which were captured at different times. As can be seen, the results of the SIFT key point detector vary depending on illumination conditions. Additionally, none of the detected key points was matched between two images using the SIFT key point matcher. Similar results were obtained for edge points as shown in the right column of the figure. Thus, currently, it would be difficult to implement robust outdoor localization using image features only, and we employ a different approach.

This paper proposes a localization method using a stereo camera for outdoor navigation to cope with the problems mentioned above. The proposed method estimates the robot motion using visual odometry and corrects its accumulated errors using a map-matching algorithm, which is based on the shapes of 3D point clouds obtained by the stereo camera. The map-matching is actually performed using 2D grid maps generated by projecting 3D point clouds onto the ground. The projected 2D grid maps are stable under various illumination conditions and also are less computationally expensive than 3D maps. For environments without 3D features, such as wide roads and open spaces, salient line segments are extracted from the ground surface and incorporated into the grid map as additional landmarks. In this paper, we call them *road landmarks*. A particle filter is used to fuse visual odometry and map-matching. While a stereo camera is the only required sensor in our method, other sensors, such as wheel odometry and gyroscope, can be used to improve localization accuracy.

Fig. 1. Two images of the same place captured at different times and result of image feature detection. Left: original image. Center: Key points detected by SIFT. Right: Detected edge points. Lowe's SIFT Keypoint Detector [5] was used to detect the SIFT key points.

Our method combines 3D range data and image features in an effective manner to enhance robustness to illumination changes. The shapes of the 2D grid maps generated from 3D point clouds are not much affected by illumination conditions. Salient line segments on the ground can be extracted stably under various illumination conditions. In urban environments, there are plenty of 3D features including walls, curbs, bushes, and trees. Also, a number of line segments, such as road boundaries and traffic signs, can be found on the ground. Thus, our method is applicable to many man-made outdoor environments. We found that our method was successfully performed in experiments on paved roads and tiled pedestrian areas under various weather conditions.

The reminder of this paper is organized as follows. After presenting related work in section II, we present our method in sections III and IV. Experiments under various illumination conditions are presented in section V.

All authors are with Future Robotics Technology Center, Chiba Institute of Technology, 2-17-1 Tsudanuma, Narashino-shi, Chiba , Japan {irie, yoshida, tomono}@furo.org

II. Related Works

Vision-based outdoor navigation has been studied for decades [6]. Many methods of ego-motion estimation, such as visual odometry, have been proposed [7][8] but visual odometry is not sufficient because errors accumulate over time. To correct the accumulated errors, landmark-based localization is necessary. Many features and objects have been used as landmarks for outdoor navigation; road boundary detection for autonomous driving [9], buildings [10] and Braille blocks [11].

Some navigation methods do not use explicit landmarks [12][13]. In these methods, the robot navigates along a prelearned path given as an image sequence, but precise robot positions on the map cannot be obtained.

Recently, appearance-based localization methods which are robust to changes in lighting have been proposed [14][15]. These methods do not provide precise localization on a metric map since they provide only topological mapping and localization.

In contrast to the above approaches, our method uses a map containing both 3D shapes and image features, and can be used in structured environments as well as in less-structured environments such as passage without apparent road boundaries and open spaces without 3D features.

III. Map Building

A 2D occupancy grid map is used in our method. The robot pose is denoted by $\mathbf{x} = (x, y, \theta)$, assuming the robot moves on a 2D ground plane. Each grid cell is labeled as *occupied*, *free*, *road landmark* or *unknown*.

First, we collect a stereo image sequence from the target environment using a manually operated robot that is equipped with a stereo camera. The collected stereo images are then processed off-line to build a map in the following steps:

1) Create a 3D point cloud from stereo images and simultaneously estimate the trajectory of the robot.
2) Project the 3D point cloud onto a 2D grid map and label each cell as occupied or free according to the height of the points in the cell.
3) Extract salient line features on the ground from the images and label the cells that contain the line features as road landmarks.
4) Close the loop based on 2D graph SLAM.

A. 3D SLAM based on Edge-Point ICP

A 3D point cloud map is built based on the method proposed by Tomono [16]. The method uses image edge points which are detected from not only long segments but also fine textures. The number of edge points detected is usually much larger than other local features (typically thousands per one QVGA (320×240) image), and it is favorable for the purpose of building occupancy maps.

We refer to a pair of left and right images as *stereo frame* (*frame*, for short). The 3D edge point $P_c = (X, Y, Z)^T$ is calculated from point $(x_l, y_l)^T$ on the left image and point $(x_r, y_r)^T$ on the right image based on the parallel stereo formula.

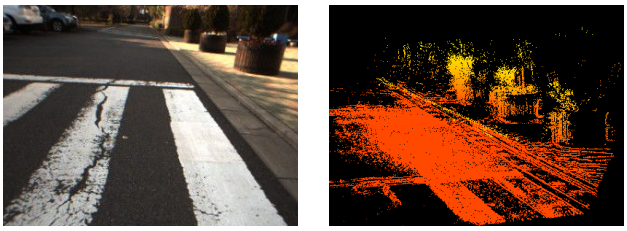

Fig. 2. Example of stereo SLAM based on edge points. Left: One of input images. Right: point cloud map built from the input images. The yellow points are higher and the red points are lower in height.

The camera motion from time $t-1$ to t is estimated by matching the 3D points reconstructed from frame I_{t-1} with the 2D points detected in frame I_t. The registration is performed using a variant of the ICP algorithm on the image plane. Let $\mathbf{r}_t$ be the camera pose at t, P^i_{t-1} be the i-th 3D edge point reconstructed at $t-1$, and p^i_{t-1} be the projected point of P^i_{t-1} onto image I_t. Let q^i_t be the image edge point at t, which corresponds to p^i_{t-1}. A cost function F is defined as follows:

$$F(\mathbf{r}_t) \quad = \quad \frac{1}{N}\sum_{i=1}^{N} d(q^i_t, p^i_{t-1}) \qquad (1)$$

Here, $d(q^i_t, p^i_{t-1})$ is the perpendicular distance between p^i_{t-1} and the edge segment on which q^i_t lies.

Camera motion $\mathbf{r}_t$ and edge point correspondences are searched by minimizing $F(\mathbf{r}_t)$ using the ICP algorithm. The initial value of $\mathbf{r}_t$ is set to $\mathbf{r}_{t-1}$, and the initial correspondence q^i_t of p^i_{t-1} is set to the edge point that is the closest to p^i_{t-1} in terms of Euclidean distance. By repeating the minimization of $F(\mathbf{r}_t)$ and edge point matching, the optimal $\mathbf{r}_t$ and edge point correspondences are obtained.

Based on the obtained camera pose $\mathbf{r}_t$, a 3D map is built by transforming the intra-frame 3D points from the camera coordinate system to the world coordinate system. Only the 3D points tracked for more than n_1 frames (typically $n_1 = 4$) are added to the 3D map. Also, 3D points with large variance are removed. This filter is useful to eliminate blurred edges and moving objects.

The procedure described in this section is also used in visual odometry, as described in section IV-A. An example of a point cloud map built by this method is shown in Fig. 2.

B. Generating 2D Occupancy Grid Maps

A 2D occupancy grid map is generated by projecting the 3D point cloud map onto the ground. The ground plane is divided into square grid cells and the 3D points in the point cloud are projected onto them. To remove 3D points on the ground and also to reduce the noise caused by errors in stereo matching, the grid cells are labeled as occupied or free according to the number of the contained 3D points that are higher than th_1. In our implementation, the size of the cells was $10cm$ and $th_1 = 15cm$.

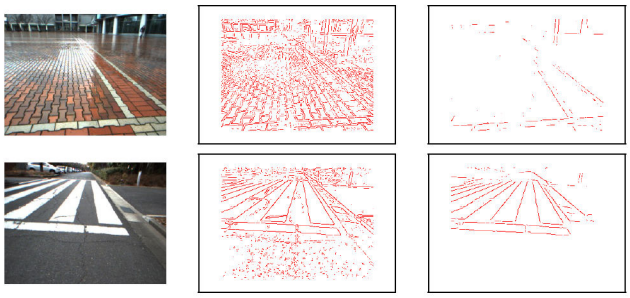

Fig. 3. Detection of road landmarks. Left: original image. Center: detected edge points. Right: detected salient lines for landmarks.

The 6-DOF camera trajectory estimated by 3D SLAM has accumulated errors. When the camera moves long distance, accumulated errors can be large in the height direction, which increase spuriously-labeled cells in the 2D grid map. To address this problem, we make the camera height constant on the assumption that the robot moves on a flat ground. 3D points are rearranged on the ground plane based on the robot's 3-DOF poses and the camera pose relative to the robot.

C. Detecting Road Landmarks

In contrast to indoor environments, which usually have rich 3D features such as walls and furniture, some outdoor environments have very few 3D features. Even laser scanners can be affected by this problem, and it is even worse for stereo vision which usually has a small field of view and a limited range of stereo reconstruction. For stable localization in such areas, other landmarks than 3D features are needed.

To cope with this problem, we use salient line segments on the ground surface. In urban environments, various line segments can be found on the ground, such as road boundaries, patterns in tiled floors, and traffic signs. These salient features are detected stably under various illumination conditions partly because the distance between the camera and the ground is small.

Road landmarks are detected by the following procedure. First, edge points are detected from the input image by the Canny detector [17]. Second, lines (continuous edge points) are extracted using the Hough transform. Through these two steps, not only salient long segments but also short segments, which are useless for localization, are extracted from fine textures on paved roads, boundaries between tiles of the same color, etc. Since most of these short segments are roof or valley edges, we remove them using a filter that only extracts step edges. This procedure removes most of the useless tiny textures and the remaining features are useful for road landmarks. Fig. 3 shows examples of detection of road landmarks.

The extracted edge points for road landmarks are projected onto a 2D grid map. To filter out noises, only the cells that contain edge points more than a threshold are labeled as road landmarks.

D. Loop Closure

To reduce accumulated errors from stereo SLAM, loop closure is performed to correct the robot trajectory based on 2D graph SLAM. We use a graph based SLAM formulation [18] with optimization, as presented in [19].

The graph is constructed as follows. When the robot moves for a certain distance, a node representing the robot pose is automatically added to the graph, and also an arc is added to represent geometric constraints between the new node and the previous node. In our current implementation, an arc to close the loop by connecting the nodes of the same place is created manually.

IV. Monte Carlo Localization

Our localization method uses a particle filter based on Monte Carlo Localization [20]. In the prediction step of the particle filter, we draw a set of particles based on the robot motion estimated by visual odometry. In the update step, the particles are weighted by the map-matching score and re-sampled according to the weights. The following subsections describe the detailed procedure.

A. Visual Odometry

3-DOF robot motion is calculated by firstly estimating 6-DOF camera motion using visual odometry and then projecting the motion onto the ground plane. The 6-DOF camera motion estimation is done basically in the same manner with the stereo SLAM described in section III-A. The difference is that no global maps are generated by the visual odometry, to reduce memory consumption. The visual odometry uses local point cloud maps to estimate camera motion. A local map is created by integrating 3D points from multiple frames since 3D points reconstructed from one stereo frame can have large errors. The local maps created in this procedure are re-used in grid map matching described in section IV-C.

B. Prediction Step

In the prediction step, the particle filter uses 3-DOF robot motion $\mathbf{u}_t = (\Delta x_t, \Delta y_t, \Delta\theta_t)^T$, which is calculated by projecting the 6-DOF camera motion according to Eq.(3). Here, T_{camera}^{robot} is the transformation from the camera coordinate system to the robot coordinate system. $\mathbf{x'_t} = (x, y, z, \phi, \theta, \psi)^T$ is the 6-DOF robot pose (ϕ: roll, θ: pitch, ψ: yaw), and $\mathbf{r_t}$ is the 6-DOF camera pose. $\Delta\mathbf{x'_t}$ is the relative pose of $\mathbf{x'_t}$ with respect to $\mathbf{x'_{t-1}}$.

$$\mathbf{x'_t} = T_{camera}^{robot}\mathbf{r_t} \tag{2}$$

$$\mathbf{u_t} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \Delta\mathbf{x'_t} \tag{3}$$

Approximating the error by a normal distribution, the robot pose represented by i-th particle $\mathbf{x}_t^i = (x_t^i, y_t^i, \theta_t^i)^T$ is calculated by using Eq.(4).

$$\mathbf{x}_t^i = \mathbf{x}_{t-1}^i + R(\theta_{t-1}^i)(\mathbf{u}_t + \mathbf{w}_t^i) \quad (4)$$

$$R(\theta) = \begin{pmatrix} cos(\theta) & sin(\theta) & 0 \\ -sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (5)$$

$$\mathbf{w}_t^i \sim N(0, \Sigma_t) \quad (6)$$

The covariance matrix Σ_t is determined experimentally.

C. Update Step

The update step of the particle filter is based on a map-matching between a global 2D grid map M_{global} and a local 2D grid map M_{local}. The global 2D grid map is built as described in section III. The local 2D grid map is built by the same procedure, using a point cloud generated through the camera motion estimation (as described in section IV-A).

In the update step, particles are re-sampled according to the weight w^i proportional to the posterior probability in Eq. (7).

$$w^i \propto p(M_{local}|\mathbf{x}_t^i, M_{global}) \quad (7)$$

In our implementation, $p(M_{local}|\mathbf{x}_t^i, M_{global})$ is approximated by cosine correlation between the local grid map and the global grid map. Let o_k^l be the occupancy value of the k-th cell in the local grid map, and r_k^l be the road landmark value of the cell (Eq. (8) and (9)). Local map vector $\mathbf{m}_{local}$ is defined as Eq. (10).

$$o_k^l = \begin{cases} 1 & \text{(occupied)} \\ 0 & \text{(not occupied)} \end{cases} \quad (8)$$

$$r_k^l = \begin{cases} 1 & \text{(road landmark)} \\ 0 & \text{(not road ladnamrk)} \end{cases} \quad (9)$$

$$\mathbf{m}_{local} = (o_1^l, r_1^l, o_2^l, r_2^l, ..., o_N^l, r_N^l) \quad (10)$$

Let $o_{k,\mathbf{x}}^g$ be the occupancy value of a cell in the global grid map corresponding to the k-th cell in the local grid map when the robot is at $\mathbf{x}$ (and $r_{k,\mathbf{x}}^g$ is defined similarly). Global map vector $\mathbf{m}_{global,\mathbf{x}}$ is defined as Eq. (11).

$$\mathbf{m}_{global,\mathbf{x}} = (o_{1,\mathbf{x}}^g, r_{1,\mathbf{x}}^g, o_{2,\mathbf{x}}^g, r_{2,\mathbf{x}}^g, ..., o_{N,\mathbf{x}}^g, r_{N,\mathbf{x}}^g) \quad (11)$$

The cosine correlation between the local grid map and global grid map for the i-th particle is calculated as Eq. (12). The weight w^i is calculated by normalizing the correlation ρ^i as Eq. (13).

$$\rho^i = \frac{\mathbf{m}_{local} \cdot \mathbf{m}_{global,\mathbf{x}_t^i}}{\|\mathbf{m}_{local}\| \left\|\mathbf{m}_{global,\mathbf{x}_t^i}\right\|} \quad (12)$$

$$w^i = \rho_i / \sum_j \rho_j \quad (13)$$

Fig. 4. Robot used in experiments.

D. Recovery from Localization Failure

Although our visual odometry works well under various illumination conditions, it can fail under extremely poor conditions. For example, direct sunlight can saturate a large part of the captured image to black or white due to the limited dynamic range of the camera. In such a case, sufficient edge points cannot be detected, which causes large errors in motion estimation.

We found that this problem is similar to slip of the wheels in the case of wheel odometry, and considered it as a kind of kidnapped robot problem. Several methods have been proposed for the kidnapped robot problem [21][22]. Our solution is similar to Expanding Reset method described in [23], which is suitable when the distance of kidnap is relatively small.

V. Experiments

We implemented the proposed method on a wheeled mobile robot, which is equipped with a Bumblebee2 stereo camera (Point Gray Research, Inc.). The camera was mounted at a height of 86cm from the ground, tilted at a pitch angle of -21°. The image size used was QVGA.

A. Map Building under Various Illumination Conditions

Before localization experiments, we evaluated how our maps are affected by illumination conditions. For comparison, we built 2D grid maps of four areas under sunny and rainy weather conditions, respectively. Fig. 5 shows the images of the four areas and the maps built from them. A lens flare seen in (a)-sunny did not affect the map. The shadow of the building in this image was not detected as a road landmark since the shadow boundary was blurred. The shadows of several people in (b)-sunny were mostly filtered out through stereo SLAM and map generation mentioned in Section III-A. The white lines in (c) and (d) are detected as road landmarks under sunny and rainy conditions despite light reflection by water. As can be seen, our map building method generates similar maps even under different illumination conditions, and this validates our map-matching approach to outdoor localization.

B. Localization under Various Illumination Conditions

We conducted experiments on robot localization in our campus. The first experiment was conducted on a path of

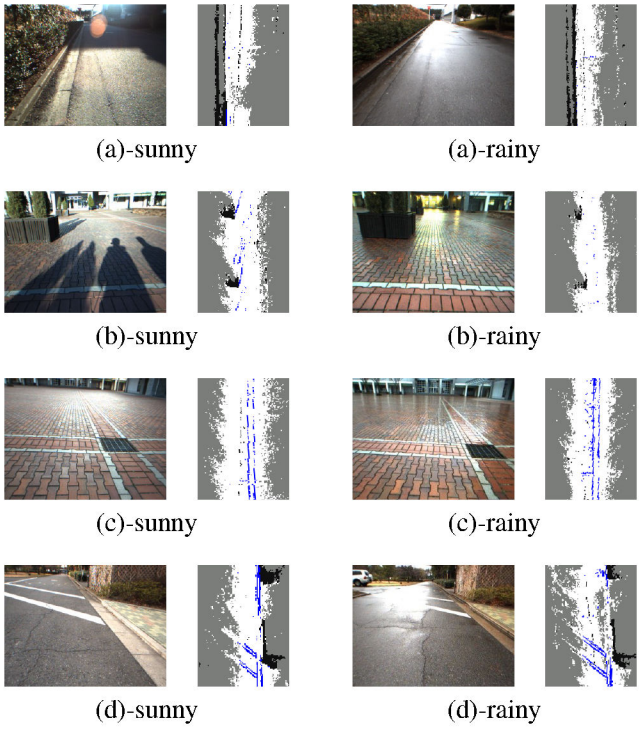

Fig. 5. Images and grid maps under various illumination conditions. Each row shows two sets of an image and a map of the same location obtained under sunny and rainy weather conditions. Colors in the maps indicate labels for cells: occupied (black), free (white), road landmarks (blue), unknown (gray).

400m, which is relatively rich in 3D features. A 2D grid map was built from an image sequence captured on a cloudy day (Fig. 6 (a)), and localization was performed on the map off-line with an image sequence captured on a sunny day (Fig. 6 (b)).

The robot was manually operated to run along the path at a maximum speed of 75cm/sec, taking stereo images at 20fps, 16,530 pairs of images in total. To evaluate localization accuracy, we operated the robot to pass through seven reference points, which we had determined in advance.

We compared two methods using the same data set, one using only occupancy information without road landmarks and the other using both occupancy information and road landmarks. In the first method (without road landmarks), the average pose error at the seven reference points was 60cm, 3.5°; the maximum pose error was 202cm, 12.3°. In the second method (with road landmarks), the average pose error was 46cm, 3.2°; the maximum pose error was 75cm, 10.9°.

Fig. 7 shows experimental results by our method (with road landmarks). While the trajectory of the visual odometry is distorted by accumulated errors, our localization method successfully tracked the robot position throughout the path.

We performed the same experiment with two different data sets (rainy / rainy and dark). In all experiments, our method successfully kept track of the robot position. Table I shows the localization results by the proposed method. Several images captured by the robot are shown in Fig. 6.

In these experiments, we used 1,000 particles. The initial

Input Data Set	Use Road Landmark	Average Error	Maximum Error
sunny	yes	59cm, 2.3°	156cm, 7.6°
sunny	no	60cm, 3.5°	202cm, 12.3°
rainy	yes	37cm, 2.4°	159cm, 6.7°
rainy	no	98cm, 3.0°	277cm, 5.4°
rainy and dark	yes	47cm, 2.1°	156cm, 4.4°
rainy and dark	no	62cm, 2.6°	152cm, 8.9°

TABLE I

LOCALIZATION RESULTS BY PROPOSED METHOD

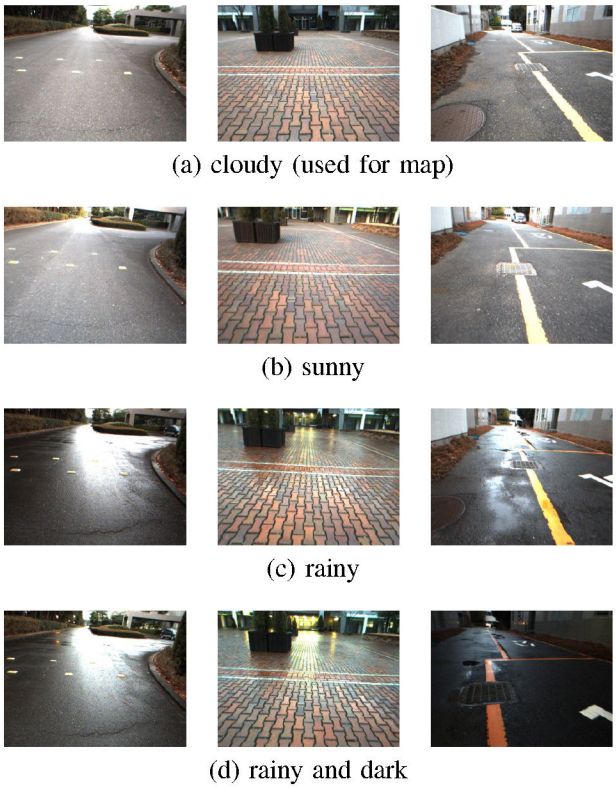

Fig. 6. Images used in the first experiment.

pose of the robot was given as a normal distribution with a standard deviation of 10cm. The prediction step of the particle filter was carried out for each frame of stereo images, and the update step was executed at every 60 frames. The processing time on a laptop with 2.4GHz Dual-Core CPU was 60 to 120ms for each prediction step, depending on the number of the edge points in the images, and approximately 300ms for each update step.

C. Localization in Environment with Wide Open Space

The second experiment was conducted on a path of 800m, which includes an open space of approximately 50m × 50m (Top right of Fig. 8) with few 3D features. In the open space, the 2D grid map did not have any valid occupied cells due to lack of 3D features, and only road landmarks coming from white tiles on the floor could be used for localization.

We again compared the two methods (with and without road landmarks). A 2D grid map was built from an image sequence captured on a rainy day, and localization was performed on the map off-line with an image sequence

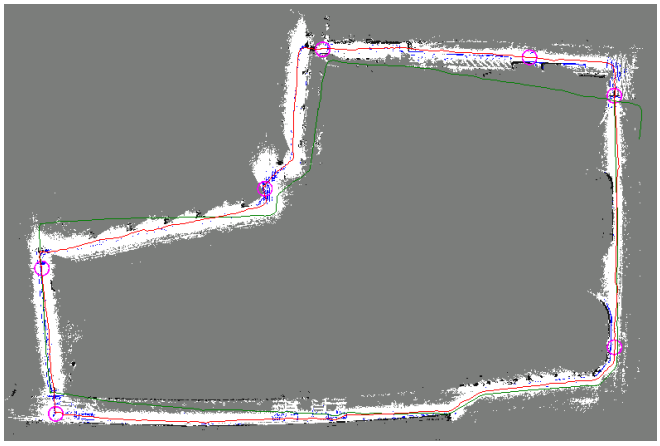

Fig. 7. Localization result by proposed method (red line) and estimation by visual odometry (green line). Magenta circles show reference points used to measure the accuracy of the localization.

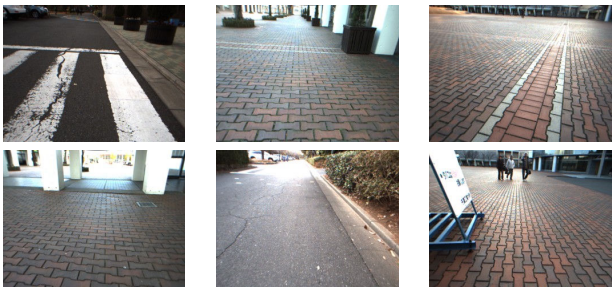

Fig. 8. Images captured by the robot during the second experiment.

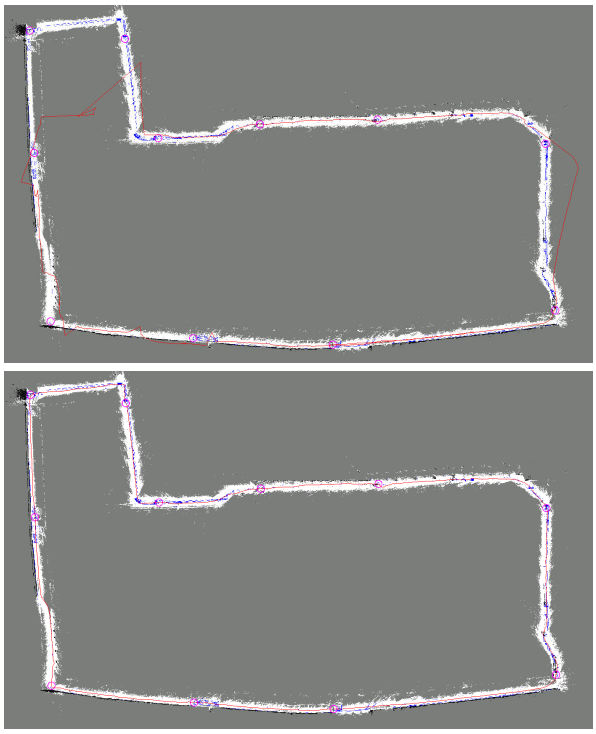

Fig. 9. Comparison of two methods. Top: trajectory obtained without road landmarks. Bottom: trajectory obtained using road landmarks. Magenta circles show reference points used to measure the accuracy of the localization.

captured on a sunny day. The accuracy of localization was measured at 11 reference points. The method without road landmarks had significant errors in the open space and at other areas with few 3D features (see Fig. 9), resulting in an average pose error of 313cm and 20.2°; the maximum error was 670cm and 86.3°. With the road landmarks, on the other hand, the average pose error was 41cm and 2.3°; the maximum error was 161cm and 5.3°.

Fig. 10 shows a zoomed comparison of localization in the open space. The error ellipses were calculated approximately from particles. As can be seen in the figure, the method with road landmarks provided better estimation.

D. Recovery from Localization Failure

Finally, we show an example of recovery from localization failure. In an experiment on a sunny day, we found an extremely poor condition shown in Fig. 11, in which a large part of the images was blacked out because of sunlight and shadow. As mentioned in section IV-D, visual odometry cannot estimate the motion of the robot correctly in such conditions.

We carried out an experiment with this image sequence and a map built from a rainy data set. The result is shown in Fig. 12. Visual odometry incorrectly estimated the motion of the robot for approximately 70 frames immediately before the robot turned right. After the robot finished turning to the right, localization failure is detected and expansion reset occurred, and eventually the robot was re-localized. The

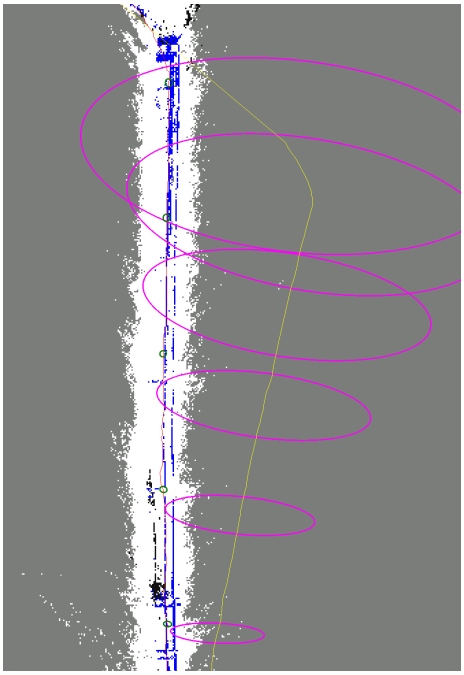

Fig. 10. Localization results of two methods in open space. The red line and the green ellipses show trajectory and error ellipses obtained with road landmarks. The yellow line and the pink ellipses show trajectory and error ellipses obtained without road landmarks.

Fig. 11. Images obtained during experiment. Left: visual odometry not functional because of few edge points. Center: pole in the image used for visual odometry while turning to right. Right: camera exposure was adjusted to the shadow after turning.

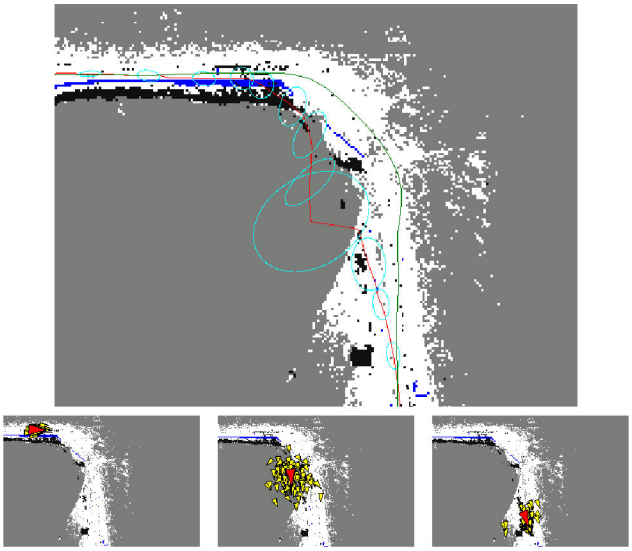

Fig. 12. Recovery from localization failure. Top: estimated position (red) and error ellipses (cyan). The trajectory obtained by gyro-assisted wheel odometry (green) is shown for comparison. The robot navigated from top-left to bottom- right in this figure. Bottom: distribution of particles. The yellow triangles show particles. The red triangles show estimated position. (a) Before kidnapping. (b) Kidnapping is detected. (c) Robot re-localized.

recovery from localization failure enables the robot to resume localization even if it encounters extremely poor illumination conditions as long as they are transient.

VI. Conclusion

In this paper, we have proposed a new localization method for outdoor navigation using a stereo camera only. The proposed method works robustly under various illumination conditions due to map-matching using 2D grid maps generated from 3D point clouds obtained by a stereo camera. We incorporated salient line segments extracted from the ground into the grid maps, making it possible to localize in environments without 3D features. Experimental results showed the effectiveness and robustness of the proposed method under various weather and illumination conditions.

Future work includes map building and localization in larger environments using a camera with a wider field of view. Also, we plan to see if employing high dynamic range images improves the robustness of this method.

References

[1] M. Adams, S. Zhang, and L. Xie, "Particle filter based outdoor robot localization using natural features extracted from laser scanners," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2004, pp. 1493–1498.

[2] K. Ohno, T. Tsubouchi, B. Shigematsu, S. Maeyama, and S. Yuta, "Outdoor navigation of a mobile robot between buildings based on dgps and odometry data fusion," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2003, pp. 1978–1984.

[3] J. R. Schoenberg, M. Campbell, and I. Miller, "Localization with multi-modal vision measurements in limited gps environments using gaussian sum filters," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2009, pp. 1423–1428.

[4] S. Se, D. G. Lowe, and J. J. Little, "Vision-based global localization and mapping for mobile robots," *IEEE Trans. on Robotics*, vol. 21, pp. 364–375, 2005.

[5] D. Lowe. (2005) Demo software: Sift keypoint detector. [Online]. Available: http://www.cs.ubc.ca/ lowe/keypoints/

[6] C. Thorpe, M. Hebert, T. Kanade, and S. Shafer, "Vision and navigation for the carnegie-mellon navlab," *IEEE Trans. on Pattern Analysis & Machine Intelligence*, vol. 10, no. 3, pp. 362 –373, 1988.

[7] A. Chilian and H. Hirschmuller, "Stereo camera based navigation of mobile robots on rough terrain," in *Proc. of the IEEE Int. Conf. on Intelligent Robots & Systems (IROS)*, 2009, pp. 4571–4576.

[8] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proc. of IEEE Computer Society Conference on Vision & Pattern Recognition*, vol. 1, 2004, pp. 652 – 659.

[9] C. Guo, S. Mita, and D. McAllester, "Stereovision-based road boundary detection for intelligent vehicles in challenging scenarios," in *Proc. of the IEEE Int. Conf. on Intelligent Robots & Systems (IROS)*, 2009, pp. 1723 –1728.

[10] A. Georgiev and P. Allen, "Vision for mobile robot localization in urban environments," in *Proc. of the IEEE Int. Conf. on Intelligent Robots & Systems (IROS)*, vol. 1, 2002, pp. 472 – 477.

[11] T. Yoshida, A. Ohya, and S. Yuta, "Braille block detection for autonomous mobile robot navigation," in *Proc. of the IEEE Int. Conf. on Intelligent Robots & Systems (IROS)*, 2000, pp. 633 – 638.

[12] H. Katsura, J. Miura, M. Hild, and Y. Shirai, "A view-based outdoor navigation using object recognition robust to changes of weather and seasons," in *Proc. of the IEEE Int. Conf. on Intelligent Robots & Systems (IROS)*, vol. 3, oct. 2003, pp. 2974 – 2979.

[13] Y. Yamagi, J. Ido, K. Takemura, Y. Matsumoto, J. Takamatsu, and T. Ogasawara, "View-sequece based indoor/outdoor navigation robust to illumination changes," in *Proc. of the IEEE Int. Conf. on Intelligent Robots & Systems (IROS)*, 2009, pp. 1229–1234.

[14] M. Cummins and P. Newman, "Fab-map: Probabilistic localization and mapping in the space of appearance," *The International Journal of Robotics Research*, vol. 6, no. 27, pp. 647–665, 2008.

[15] C. Valgren and A. J. Lilienthal, "Sift, surf & seasons: Appearance-based long-term localization in outdoor environments," *Robotics and Autonomous Systems*, vol. 58, no. 2, pp. 149–156, 2010.

[16] M. Tomono, "Robust 3d slam with a stereo camera based on an edge-point icp algorithm," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2009, pp. 4306–4311.

[17] J. Canny, "A computational approach to edge detection," *IEEE Trans. on Pattern Analysis & Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[18] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol. 4, pp. 333–349, 1997.

[19] E. Takeuchi and T. Tsubouchi, "Multi sensor map building based on sparse linear equations solver," in *Proc. of the IEEE Int. Conf. on Intelligent Robots & Systems (IROS)*, 2008, pp. 2511–2518.

[20] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, vol. 2, 1999, pp. 1322 – 1328.

[21] D. Fox, W. Burgard, and S. Thrun, "Markov localization for mobile robots in dynamic environments," *Journal of Artificial Intelligence Research*, vol. 11, pp. 391–427, 1999.

[22] S. Lenser and M. Veloso, "Sensor resetting localization for poorly modelled mobile robots," in *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, vol. 2, 2000, pp. 1225 –1232.

[23] R. Ueda, T. Arai, K. Sakamoto, T. Kikuchi, and S. Kamiya, "Expansion resetting for recovery from fatal error in monte carlo localization - comparison with sensor resetting methods," in *Proc. of the IEEE Int. Conf. on Intelligent Robots & Systems (IROS)*, vol. 3, 2004, pp. 2481 – 2486.