Multi-robot SLAM Using Ceiling Vision

Hee Seok Lee and Kyoung Mu Lee

Abstract— In this paper we present a new vision-based SLAM approach for multi-robot formulation. For a cooperative map reconstruction, the robots have to know each other's relative poses, but estimating these at the start of operation puts a limit on real applications. In our study, the robots start the single SLAM with their own global coordinate, and merge their maps during the operation by detecting the overlapped region of their maps. The robots automatically recognize the occurrence of map overlapping by matching their current frame with the maps built by other robots. With the robust data association technique from the ceiling-vision based SLAM, the proposed algorithm robustly detects the overlapping regions and estimates the accurate transformations for map alignment. In our experiment, we have verified that our algorithm successfully enables the multi-robot SLAM without any initial correspondence or encounter of robots.

I. INTRODUCTION

In real applications of mobile robots, robots sometimes have to organize themselves into a team and cooperate with one another to perform their tasks. Here, the robots need to know the position of the others. This can be achieved by measuring the relative position of each robot, or by having a common map for all robots and exchanging the information of their global position. In the case of a general sensor such as laser and vision, it is difficult to measure the position of robots that are located at a distance due to the limitation of the sensor range or obstacles between the robots. Therefore, having a common map is more efficient than estimating the relative position of robots every time. This cooperative map-building problem is known as the multi-robot SLAM, or the cooperative SLAM [1], [2]. Recently, the multi-robot SLAM has also received considerable attention in robotics.

There are two important issues in the multi-robot SLAM problem. The first issue is how to simultaneously estimate the position of the multiple robots and their map. The position estimation in the multi-robot SLAM is more complex than a single-robot's estimation. We have to localize each robot simultaneously and, in map building the sensor measurements from all the available robots, have to be consistent with the estimated result. However, the multi-robot operation has the ability to give a more accurate estimation as compensation for this complexity. In [3], [4] it has been demonstrated that the error of a landmark position decreases in the case of the multi-robot SLAM when compared to that of the single-robot SLAM. Generally, the estimation of a landmark position with

H. Lee is with the Department of Electrical Engineering and Computer Science, Seoul National University, 599 Gwanangno, Gwanak-gu, Seoul, Korea. `ultra21@snu.ac.kr`
K. Lee is with Faculty of the Department of Electrical Engineering and Computer Science, Seoul National University, 599 Gwanangno, Gwanak-gu, Seoul, Korea. `kyoungmu@snu.ac.kr`

multiple observations can be integrated by EKF, and our system also uses this approach.

The second issue in the multi-robot SLAM is how to put the robots together in a common global map. It is necessary to have a global coordinate for every robot in order to perform the cooperative SLAM operation in the common map. This global map can be obtained at the start of operation or during the operation. To start SLAM operation with the common global coordinate, the robots have to estimate one another's relative positions before starting the SLAM operation. To estimate the relative position, the robots have to start at a location where their sensors can observe one another.

Without this initial correspondence between the robots, another possible way to obtain a common map is to find a relationship between the maps during the operation and merge the maps that have been independently built by each robot. In [5], the encounter of the two robots provides the transform which performs the map merging operation by moving one map to the others. When the two robots meet each other, using the image containing the appearance of the other robot, the robots estimate the relative pose between them and calculate the transform for map merging using this relative pose. However, this encounter does not occur frequently, leading to the low probability of map merging.

In contrast, our proposed algorithm does not require the encounter of the robots. Our algorithm finds the relation between the maps not by using a direct observation of each robot but by using a matching between the map itself and current observations from the robot. Each robot automatically detects the overlapping of its map with others and, if there is map overlapping, the relation between the maps is estimated and is followed by map merging. We utilize the robust data association method from the ceiling vision-based SLAM (CV-SLAM) [6], which uses ceiling scenes as observations. This enables very stable result of the map overlap detection and an accurate estimation of the relation between the maps.

Ceiling scenes from a planar robot motion have little scale change and affine deformation, so the scene matching is easily done just by using local feature matching based on the correlation window such as NCC (normalized cross correlation) [7] or SAD (sum of absolute difference). Therefore, we do not have to calculate a complicated feature descriptor like SIFT [8] and a handling rotational changes of scene is sufficient to match the features.Thus, fast feature matching is possible. Ceiling scenes, however, can suffer when there is a similar structure on the ceiling; this can cause an incorrect scene matching. To overcome this problem, we tested a sequence of the scene matching results of each step and

confirmed the consistency of the sequence. The matching with one ceiling scene is error prone, but the series of tests gives robust detection of map overlapping even though there are some similar patterns on the ceiling.

As our algorithm does not require an encounter of the robots, it is possible to obtain the global map early and with more chances. Moreover, map merging strategy based on the relation of the map, not on the robots' relation, enables the reuse of maps that were previously built by other robots. Suppose a robot enters the new region to perform its task. If the map of the new environment is available, then the robot can merge its map and obtained map after a short SLAM operation in the new area. This leads to fast adjustment in a new region.

II. FORMULATION OF MULTI-ROBOT SLAM

A. Modelling of robots and maps

The observation models of our algorithm for a single robot follow those of ceiling vision-based SLAM. Each robot is equipped with a single camera pointing in an upward direction to view the ceiling. With a wide field of view, the camera can capture scenes not only on the ceiling but also on the wall near the robot. From these scenes, the robot detects feature points on the ceiling and on the wall. If the robot moves in a plane, then the features on the ceiling and on the wall have the characteristics of scale invariance because the distance from the feature to the robot remains almost the same. If the robot leaves the position under the landmark, then the landmark disappears in the observed image and we do not need to consider it. This results in only sheer deformations on the feature appearance by the motion of the robot. With this property, we merely have to handle the rotation changes of the feature appearance to match them.

Although we use this scale invariant property of ceiling vision, it does not restrict the ceiling's height a constant. We extract the features on the wall, use them as landmarks and estimate the height of the landmark position. However, the robot should remain in a plane motion to estimate the height of the landmarks. The pose of the robot at time index t can be represented as $\mathbf{r}_t = [r_x, r_y, r_\theta]_t^T$, and the landmarks in the 3D map can be represented as $\mathbf{L} = [L_x, L_y, L_z]^T$.

The observation of landmarks is performed using a single 2D image captured at the current position. The observation $\mathbf{z}_t$ for one landmark is composed of a distance z_r and a direction z_θ from the image center, so $\mathbf{z}_t = [z_r, z_\theta]^T$. To predict the projected position $\hat{\mathbf{z}}_t$ for the landmark in 2D image, we use the relation between the robot pose $\mathbf{r}_t$ and landmark position $\mathbf{L}$ given by the following equation:

$$\begin{bmatrix} \hat{z}_r \\ \hat{z}_\theta \end{bmatrix} = \begin{bmatrix} \sqrt{(L_x - r_x)^2 + (L_y - r_y)^2} \times \frac{f}{L_z} \\ \tan^{-1} \frac{L_y - r_y}{L_x - r_x} - (r_\theta - 90) \end{bmatrix}. \quad (1)$$

Where f is the focal of the camera. Using the difference between the observed position on the image and the landmark's projected position, we can update the estimated 3D position of the landmark.

B. Particle filtering for Multi-robot SLAM

We employ the Rao-Blackwellized Particle Filter (RBPF) [9], which is widely used in the SLAM algorithm, to simultaneously estimate the posterior probability of poses of the multiple robots and a map of the environment. Each robot estimates the posterior of its pose by sampling from the motion information $\mathbf{u}_t$, and for each sample the position of the landmarks is estimated by EKF. When the two robots share the common map $\mathbf{m}$ to perform the cooperative SLAM, we have to estimate the posterior of both robots $\mathbf{r}_t^1$, and $\mathbf{r}_t^2$, and the one common map $\mathbf{m}$. This entire posterior can be factorized as follows [10]:

$$\begin{aligned} p(\mathbf{r}_t^1, \mathbf{r}_t^2, \mathbf{m} | \mathbf{z}_t^1, \mathbf{u}_{t-1}^1, \mathbf{r}_{t-1}^1, \mathbf{z}_t^2, \mathbf{u}_{t-1}^2, \mathbf{r}_{t-1}^2) = \\ p(\mathbf{m} | \mathbf{r}_t^1, \mathbf{z}_t^1, \mathbf{r}_t^2, \mathbf{z}_t^2) \\ p(\mathbf{r}_t^1 | \mathbf{z}_t^1, \mathbf{u}_{t-1}^1, \mathbf{r}_{t-1}^1) p(\mathbf{r}_t^2 | \mathbf{z}_t^2, \mathbf{u}_{t-1}^2, \mathbf{r}_{t-1}^2) \end{aligned} \quad (2)$$

The first term, the distribution over the map, is estimated using EKF, and the second term is estimated by sampling the robot poses. More accurate motion information is available than from a monocular SLAM [11] or visual odometry system [12] such that the camera moves freely in 3D space because our robots always move on a plane. Therefore, in the last two terms, the robot pose $\mathbf{r}_t$ is independent from the map $\mathbf{m}$ because we do not use observations to estimate the robot pose. After we get samples of the robots' pose, we marginalize the distribution of the map by applying EKF for each landmark.

In an EKF update for a landmark, the state vector is $\mathbf{X}_t = \mathbf{L}_t = [L_x, L_y, L_z]_t^T$ and the sensor input is only the observation images which give the projected 2D position information of landmarks. In our case, the number of observations vary according to the number of robots which can view the landmark at current position. The dimension of the measurement residual vector $\tilde{\mathbf{v}}_t$ is also changed with the number of observations. As our measurement vector is composed of z_r and z_θ, the dimension of $\tilde{\mathbf{v}}$ should be $2k$, where k is the number of robots which can view the landmark. From this measurement residual $\tilde{\mathbf{v}}_t$ and error covariance matrix $\mathbf{R}_t$, we can calculate the innovation covariance $\mathbf{S}_t = \mathbf{J}_{h_t} \mathbf{P}_t \mathbf{J}_{h_t} + \mathbf{R}_t$ using the Jacobian $\mathbf{J}_{h_t}$ given by:

$$\mathbf{J}_{h_t} = \begin{bmatrix} \frac{\partial \hat{z}^1}{\partial L} \\ \frac{\partial \hat{z}^2}{\partial L} \\ \cdots \\ \frac{\partial \hat{z}^k}{\partial L} \end{bmatrix} = \begin{bmatrix} \frac{\partial z_r^1}{\partial L_x} & \frac{\partial z_r^1}{\partial L_y} & \frac{\partial z_r^1}{\partial L_z} \\ \frac{\partial z_\theta^1}{\partial L_x} & \frac{\partial z_\theta^1}{\partial L_y} & \frac{\partial z_\theta^1}{\partial L_z} \\ \frac{\partial z_r^2}{\partial L_x} & \frac{\partial z_r^2}{\partial L_y} & \frac{\partial z_r^2}{\partial L_z} \\ \frac{\partial z_\theta^2}{\partial L_x} & \frac{\partial z_\theta^2}{\partial L_y} & \frac{\partial z_\theta^2}{\partial L_z} \\ \cdots \cdots \cdots \\ \frac{\partial z_r^k}{\partial L_x} & \frac{\partial z_r^k}{\partial L_y} & \frac{\partial z_r^k}{\partial L_z} \\ \frac{\partial z_\theta^k}{\partial L_x} & \frac{\partial z_\theta^k}{\partial L_y} & \frac{\partial z_\theta^k}{\partial L_z} \end{bmatrix}, \quad (3)$$

where $h(\mathbf{X}_t, \mathbf{r}_t^1, \mathbf{r}_t^2, \cdots, \mathbf{r}_t^k) = [\hat{z}^1, \hat{z}^2, \cdots, \hat{z}^k]$ is the project function which calculates the 2D projected position of the 3D landmark. We can obtain a more accurate estimate of the landmark's position when we have larger number of k.
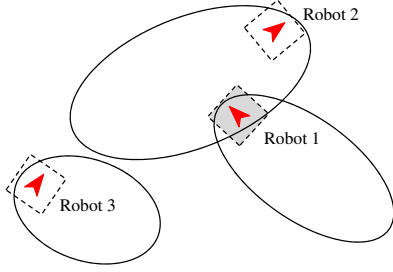
Fig. 1. When a robot enters the region of which map is already built by other robots, the robot recognizes the map overlapping and performs the map alignment. Shaded rectangle represents the observed image by the entering robot.
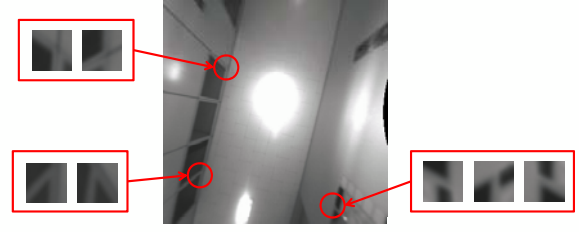

Fig. 2. Examples of cropped image patch from landmark. Due to the variation of patch orientation by view point changes, each landmarks is represented by several difference appearances.

III. MULTI-ROBOT SLAM WITH UNKNOWN INITIAL POSES

To perform the multi-robot SLAM, our method does not require the known initial relative poses and gradually estimates them. Our scenario is as follows. Each robot starts the single-robot SLAM operation independently with unknown initial relative poses. When map overlapping occurs, the robot in the overlap region automatically recognizes that a map overlap has occurred, and the transformation between two maps is estimated to align the two maps. In this process, the critical problem is how to detect the overlapped region of the maps and find the transformation for map alignment. In this section, we discuss this problem in detail.

A. Finding overlap regions with a single frame

Initially, each robot knows only the number of other robots. Each robot starts the SLAM operation with its own coordinate system as it has no information about its relative pose at all. For convenience, we set the initial pose of all robots as $\mathbf{r}_0 = [r_x, r_y, r_\theta]_0^T = [0, 0, 0]^T$. For each SLAM iteration the robot obtains an observation image, updates its map, and searches the overlap region using this observation image. If one robot is in a region where the other robot has reconstructed the map, the ceiling scenes of the robot which has entered the overlapped region and the already reconstructed map by the other robot would thus have some common landmarks. Figure 1 illustrates this situation. The landmarks in the shaded region would have been seen by robot2, and now it is observed by robot1. To search these common landmarks, we employ robust data association technique of the CV-SLAM.

In CV-SLAM, as the data association of landmarks with ceiling scenes is performed under the condition of no scale and no view point change, robust matching can be performed very fast. We modify the global localization algorithm of the original CV-SLAM in order to apply it to the map alignment process in multi-robot SLAM operation.

When each robot performs single robot SLAM, it collects image patches around observed landmarks as a descriptor for data association. We store a new patch of the landmark when the appearance changes significantly because the landmark's appearance changes every time that the robot moves. For every patch, we estimate its orientation and normalize the patch using this orientation. This process removes duplicate patches induced by the rotation of a patch. In our experiment, the required number of patches does not exceed 20. These appearance patches are used in SLAM operations and overlap region detection of multiple robots. Figure 2 shows the example of landmarks patches.

Our multi-robot formulation can be applied to a system composed of an arbitrary number of robots. We may assume that we have two robots, R^1 and R^2. For each observed image frame, the robots perform the single-SLAM operation first and make a decision whether the current location is already mapped by other robots or not. This can be achieved by comparing image patches of landmark in the current frame with patches in the data base (i.e., the map built by other robots). For the given image frame I_t^1, the robot extracts the set of landmarks $Ł^1 = \{\mathbf{L}_1^1, \mathbf{L}_2^1, \cdots, \mathbf{L}_N^1\}$ and attempts to search its matching landmarks from the map built by other robots, where N is the number of observed landmarks in I_t^1. We use height values of landmarks to reduce the search range.

We use the normalized cross correlation as a similarity measure of two image patches of landmarks. From the landmarks in the map of R^2, one landmark L_m^2, which has the highest similarity patch with the patches of $\mathbf{L}_n^1 \in Ł^1$, is selected as a matching landmark of L_n^1 unless the similarity is lesser than the threshold $T_{similarity}$. For every landmark in I_t^1, we find the matching landmark from the map of R^2. In the case of a small number of matching landmarks, we terminate the overlap detection process for the current frame and continue on to the next SLAM iteration. If there are sufficient matches, then next we need to confirm the map overlapping more strictly.

We cannot determine the pose of robot because it is difficult to determine the direction of robot with one correspondence of a landmark. Therefore, we use multiple correspondences of the landmark rather than use the bearing information of the landmarks. Similar to the global localization algorithm of [6], we estimate the pose of R^1 based on voting of multiple landmarks using Hough clustering. The observation of landmark using ceiling image provides a distance measurement z_r between the robot and landmarks.
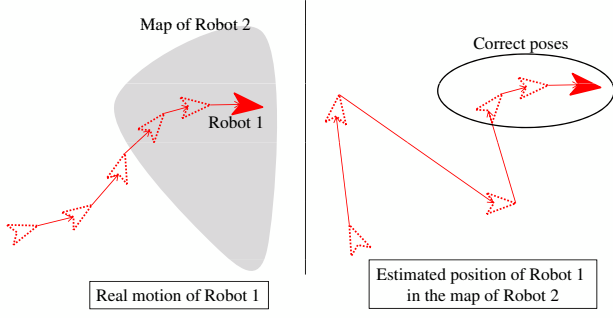
Fig. 3. Before the occurrence of map overlapping, the real motion of Robot 1 and the estimated trajectory in the map of Robot 2 are not matched. After the map overlapping and map alignment, they become consistent.

We can draw a circle with a radius z_r around the landmark which represents possible robot position. For each landmark in the current frame I_t^1, we draw these circles. As more circles intersect at a certain point, the probability of R^1 existing at that point goes higher. If the most probable robot position does not have enough intersecting circles, then we reject the position and the overlap detection process fails.

B. Robust overlap detection using sequential frames

As discussed in the previous section, by using one observed image frame we can determine the occurrence of map overlapping with some confidence. In this process, however, setting a sufficient number of the intersecting circles can be a problem. Let the threshold of the number of intersecting circles which determines the occurrence of map overlap as $T_{intersect}$. If $T_{intersect}$ is too small such that it strictly rejects the intersecting point, we may fail to detect the map overlapping. On the other hand, if $T_{intersect}$ is too large, then a wrong detection result may occur. One way to solve this problem is to use several consecutive frames for map overlap detection.

Let the estimated position of R^1 in $\mathbf{m}^2$ from the result of Hough clustering be $\hat{\mathbf{r}}^* = [x, y]^T$. When R^1 enters the region where R^2 has already reconstructed the map, the estimated motion of R^1 must be consistent with the trajectory of the estimated position $\hat{\mathbf{r}}^*$ on $\mathbf{m}^2$, as illustrated in Figure 3. When R^1 is outside of $\mathbf{m}^2$, the real motion estimated by a particle filter differs from the trajectory of $\hat{\mathbf{r}}^*$, while they are consistent when R^1 goes through m^2.

To measure the consistency between the motion of R^1 estimated by SLAM operation and the trajectory of $\hat{\mathbf{r}}^*$, we calculate the difference of incremental position for a sequential SLAM iteration only when Hough clustering is successful. The incremental position is calculated by $d\hat{\mathbf{r}}_t^1 = \hat{\mathbf{r}}_t^1 - \hat{\mathbf{r}}_{t-1}^1$ for the real motion and $d\hat{\mathbf{r}}_t^* = \hat{\mathbf{r}}_t^* - \hat{\mathbf{r}}_{t-1}^*$ for trajectory of the estimated pose in $\mathbf{m}^2$, where $\hat{\mathbf{r}}_t^1$ represents the x, y component of $\mathbf{r}_t^1$. The difference between $d\hat{\mathbf{r}}_t^1$ and $d\hat{\mathbf{r}}_t^*$ would be quite large in the case of wrong pose estimation of $\mathbf{r}_t^*$. By calculating the incremental positions for a few consecutive frames, we can robustly determine the occurrence of map overlapping. Testing for more frames provides a more robust determination result, and the minimum number of frames

required is two (i.e., one incremental position). According to this procedure, we define the function that estimates the confidence of a map overlapping as

$$confidence = \exp\left[\frac{1}{M}\sum_{i=0}^{M}\frac{\|d\hat{\mathbf{r}}_{t-M}^1 - d\hat{\mathbf{r}}_{t-M}^*\|}{\|d\hat{\mathbf{r}}_{t-M}^1\|}\right], \quad (4)$$

where M is the number of previous frames that we used to check the consistency of incremental position. The difference is normalized by the motion of R^1. When this confidence value becomes larger than 0.5, we conclude that map overlap has occurred.

C. Estimate map transform

A map merging process is performed after the overlap region detection. Again, we assume that R^1 is entering the map of R^2, $\mathbf{m}^2$. We merge the maps $\mathbf{m}^1$ and $\mathbf{m}^2$ by transforming $\mathbf{m}^2$ into $\mathbf{m}^1$. Let $\mathbf{T}_{12}$ be the transformation matrix which transforms the coordinate of R^2 into that of R^1, then $\mathbf{T}_{12}$ is given by

$$\mathbf{X}^* = \mathbf{T}_{12} \cdot \mathbf{X}^2, \quad (5)$$

where $\mathbf{x}^2$ is the original coordinate of R^2 and $\mathbf{x}^*$ is the mapped position. $\mathbf{X}^2$ and $\mathbf{X}^*$ are in the form of 3D homogeneous representations. Due to the assumption of plane motion, the transformation is composed only by translation and rotation in an xy plane. Thus $\mathbf{T}_{12}$ is given by

$$\mathbf{T}_{12} = \begin{bmatrix} \mathbf{R} & & 0 & t_x \\ & & 0 & t_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (6)$$

where $\mathbf{R}$ is a 2×2 rotation matrix, and t_x and t_y are the translation factors in x and y coordinate, respectively. As mentioned in the previous section, a possible robot position $\hat{\mathbf{r}}^*$ from Hough clustering is not an optimal solution and does not give rotation information. Thus we cannot estimate $\mathbf{T}_{12}$ using the robot poses. Instead, by using correspondences of landmarks, we can achieve a more accurate solution for $\mathbf{T}_{12}$.

Finding the optimal solution that satisfies all these correspondences is required since we have a number of correspondences of landmarks. We find the map transform $\mathbf{T}_{12}$ by minimizing error function

$$\sum_{i=1}^{N_{obs}} \left\| \bar{\mathbf{L}}_i^1 - \mathbf{T}_{12} \cdot \bar{\mathbf{L}}_i^2 \right\|^2, \quad (7)$$

where $\bar{\mathbf{L}}_i^1$ and $\bar{\mathbf{L}}_i^2$ are the matching landmarks in current frame I_t^1 with homogeneous representation, and N_{obs} is the number of correspondences observed in I_t^1. We can easily find the solution to this equation by solving a linear equation. With this transformation $\mathbf{T}_{12}$, we put the set of landmarks in $\mathbf{m}^2$ into the coordinate of R^1 using relation given by

$$\bar{\mathbf{L}}_n'^2 = \mathbf{T}_{12} \cdot \bar{\mathbf{L}}_n^2. \quad (8)$$

For every landmark $\mathbf{L}_n^2$ in $\mathbf{m}^2$, we apply this transformation and finally get the merged map. Unfortunately, there may
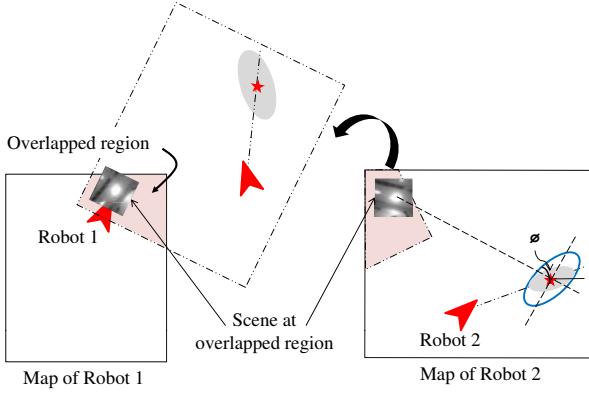
Fig. 4. The uncertainty of each landmark is transferred according to the map transformation.

Fig. 6. Confidence of map overlapping versus frame sequence. The confidence value is high when the map overlapping occurs.

exist errors in $\mathbf{T}_{12}$, especially since even a small error in rotation brings more severe errors in the transformed landmark position when the distance between the overlap region and the landmark increases. To handle this amplified error, we search the duplicate landmark [5] in the map of R^1 and correct this error using newly found correspondences. We introduce an uncertainty range for the transformed landmark to find the duplicate landmark. Originally every landmark has the uncertainty of its position given by the covariance of Kalman filter estimator. When performing the landmark transformation, these covariance matrices are also transformed by the equation given by

$$\mathbf{P}'^2_n = \mathbf{S}(\mathbf{r}^2 - \mathbf{L}^2_n, \phi)\mathbf{R}(\theta)\mathbf{P}^2_n, \qquad (9)$$

where $\mathbf{S}(\mathbf{r}^2 - \mathbf{L}^2_n, \phi)$ stretches the position uncertainty of landmark in a direction of rotational error. In Figure 4, the shaded ellipse in the map of R^2 is the landmark's original uncertainty. Its principal axis is parallel with the line passing through R^2 and the landmark. When this landmark is transformed by $\mathbf{T}_{12}$, the uncertainty range is increased in a tangential direction of the line which passes through the landmark and the axis of the rotation transform. Thus, $\mathbf{S}(\mathbf{r}^2 - \mathbf{L}^2_n, \phi)$ is of the form

$$\mathbf{S}(\mathbf{r}^2 - \mathbf{L}^2_n, \phi) = \begin{bmatrix} c\|\mathbf{r}^2 - \mathbf{L}^2_n\|^2 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{R}(-\phi). \qquad (10)$$

The matrix $\mathbf{R}(\theta)$ rotates the stretched ellipse to adjust the transformed map. Using this covariance, we set the uncertainty range of the position of the transformed landmarks. Within this range, we search a landmark $\mathbf{L}^1_n \in \mathbf{\L}^1$, which is possible to be matched with the transformed one. If the matched landmark is found, then we add this correspondence to Equation 7 and re-estimate the transform $\mathbf{T}_{12}$. We can expect a more accurate $\mathbf{T}_{12}$ in this re-estimation because more distant landmarks are involved.

IV. EXPERIMENTAL RESULTS

We have tested our multi-robot SLAM algorithm in a real indoor environment composed of a long corridor and many
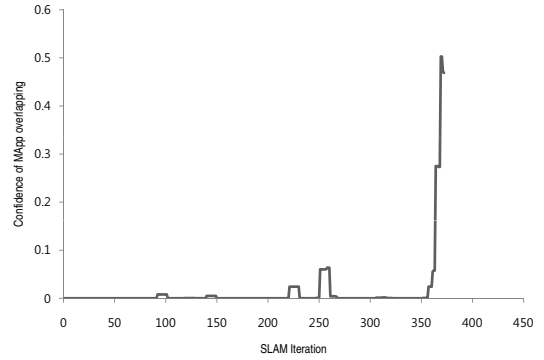
rooms along the corridor. The experiments were performed with the pre-recorded scenes obtained from the human controlled mobile robot. The obtained scenes had 640×480 resolution, and 180 degrees FOV. We corrected the radial distortion of images and resized it to 320×320 size. Harris corner detector [13] was used to obtain landmarks. In the PBRF, we used 200 particles.

In the experiment, we drove the two robots independently with unknown initial correspondences. One robot, specified R^2, moved along with the corridor. The other robot, R^1, started by building the map of the room near the corridor, then exited the room and entered the corridor where R^2 had already built the map. Figure 5 (a) and (b) are the results of the single-robot SLAM operation performed by two robots independently. The orange and violet lines represent the estimated trajectory of each robot, and the ellipses are uncertainty of the landmarks. The red circles in (a) and (b) represent the same region where the map overlapping is going to occur. By applying our proposed map overlap detection and merging procedure, we could align the maps as Figure 5 (c). When R^1 was located at the region indicated by the red circle, it recognized the map overlap between the two robots and performed a map alignment procedure. The landmarks in magenta represent the matching landmarks used in overlap detection.

Figure 6 is the graph that shows the confidence map overlapping explained in Section 3.2. At frame 360, this confidence value is increased rapidly due to the map overlap. This confidence values differ largely according to whether a map overlap occurs or not, which enables accurate decision for map overlap.

After map alignment of the two maps, the two robots continue map building and finally obtain the result as shown in Figure 5 (d).

V. CONCLUSIONS

We have presented a technique for multi-robot SLAM which can operate with unknown initial relative poses between robots. The robust data association algorithm using ceiling vision provides accurate detection of map overlapping
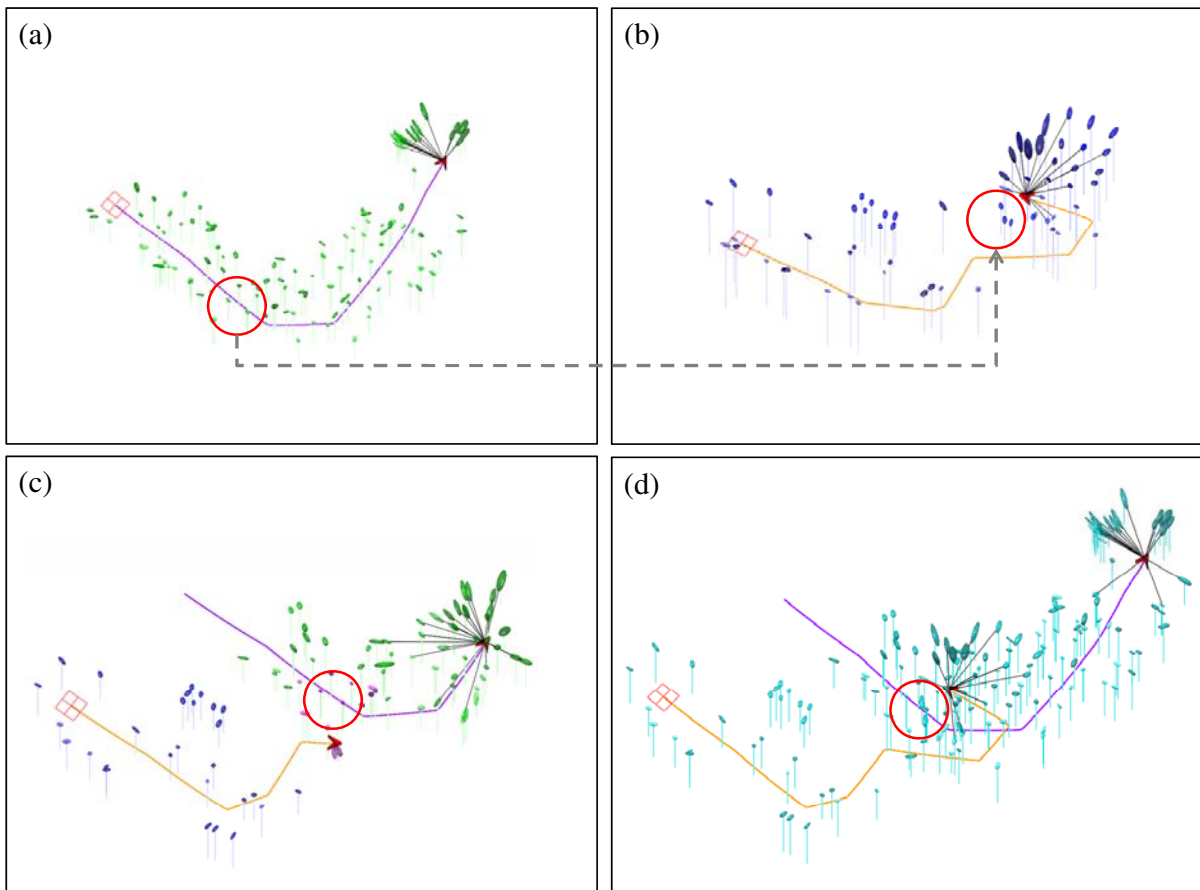
Fig. 5. (a) and (b) are the maps built by two robots independently. (c) shows the result of the map alignment. (d) is the final integrated map by the two robots.

without an encounter of robots, leading to the early and frequent chances for map merging with very low error rate.

In future work, we intend to apply an efficient landmark management algorithm which will efficiently perform the data association of the landmark from different maps. We expect this will improve the computational speed of overlap detection process. In addition, investigating the probabilistic approach for estimating the map alignment transform would be helpful. If we determine the map transform as soon as map overlap occurs, then recovery or correction for wrong estimation of transform will be impossible. The probabilistic approach will lead to a more flexible result of map alignment.

VI. ACKNOWLEDGMENTS

REFERENCES

[1] S. Thrun, "A probabilistic online mapping algorithm for teams of mobile robots," 2001, international Journal of Robotics Research.
[2] Z. Wei, G. Huang, and P. Wang, "The research on multi-robot simultaneous localization mapping algorithm," 2007, international Conference Automation and Logistics.
[3] J. Fenwick, P. Newman, and J. Leonard, "Cooperative concurrent mapping and localization," 2002, international Conference on Robotics and Automation.
[4] A. Mourikis and S. Roumeliotis, "Performance bounds for cooperative simultaneous localization and mapping (c-slam)," 2004, robotics: Science and Systems Conference.
[5] X. Zhou and S. Roumeliotis, "Multi-robot slam with unknown initial correspondence: The robot rendesvous case," 2006, international Conference on Intelligent Robots and Systems.
[6] W. Jeong and K. Lee, "Cv-slam: A new ceiling vision-based slam technique," 2005, international Conference on Intelligent Robots and Systems.
[7] W. Forstner, "A feature-based correspondence algorithm for image matching," international Archives of Photogrammetry and Remote Sensing, vol.26, No.3, pp.150-166, 1986.
[8] D. Lowe, "Distinctive image features from scale-invariant keypoints," international Journal of Computer Vision, Vol. 60, No. 2, pp.91-110, 2004.
[9] K. Murphy, "Bayesian map learning in dynamic environments," 1999, neurla Information Processing Systems.
[10] A. Howard, "Multi-robot simultaneous localization and mapping using particle filters," 2006, international Journal of Robotics Research.
[11] A. Davison, "Real-time simultaneous localization and mapping with a single camera," 2003, international Conference on Computer Vision.
[12] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," 2004, conference on Computer Vision and Pattern Recognition.
[13] C. Harris and M. Stephens, "A combined corner and edge detector," 1988, alvey Vision Conference.