A Cooperative Localization Algorithm for Mobile Sensor Networks

Junghun Suh, Seungil You, and Songhwai Oh

Abstract— **In this paper, we propose a cooperative localization algorithm for mobile sensor networks with camera sensors to operate under GPS denied areas or indoor environments. Mobile robots are partitioned into two groups. One group moves within the field of views of remaining stationary robots. The moving robots are tracked by stationary robots and their trajectories are used as spatio-temporal features. From these spatio-temporal features, relative poses of robots are computed using multi-view geometry. In order to provide the poses of robots with respect to the reference coordinate system, robots take a turn and at least one robot is stationed as robots move across the field of interest. By taking the advantage of the multi-agent system, we can reliably localize robots over time as they perform a group task. In experiment, we demonstrate that the proposed method consistently achieves less than $1\,cm$ of localization error for trajectories of length less than $100\,cm$ and less than 0.34% of localization error for longer trajectories of length between $725\,cm$ and $769\,cm$ using an inexpensive off-the-shelf robotic platform.**

I. INTRODUCTION

A wireless sensor network has been successfully applied to many applications for monitoring, event detection, and control of our environment, including environment monitoring, building comfort control, traffic control, manufacturing and plant automation, and military surveillance applications (see [1] and references therein). However, faced with the uncertain nature of the environment, stationary sensor networks are sometimes inadequate and a mobile sensing technology shows superior performance in terms of its adaptability and high-resolution sampling capability [2].

A mobile sensor network can efficiently acquire information by increasing sensing coverage both in space and time, thereby resulting in robust sensing under the dynamic and uncertain environments. While a mobile sensor networks shares the same limitations of wireless sensor networks in terms of its short communication range, limited memory, and limited computational power, it can perform complex tasks, e.g., exploration, surveillance, and environmental monitoring, distributed coordination, by cooperating with other agents as a group. There is a growing interest in mobile sensor networks and it has received significant attention recently [3]–[6].

In order to perform sensing or coordination using mobile sensor networks, localization of all the nodes is of paramount importance. A number of localization algorithms have been proposed for stationary sensor networks, e.g., [7], [8]. But they are applied in the outdoor environment and precise indoor localization is still a challenging problem [9]. (For more information about various localization methods for wireless sensor networks, see references in [7]–[9].) One promising approach to indoor localization is based on the ultra-wideband (UWB) radio technology [10]. But as stated in [10], the minimum achievable positioning error can be in the order of $10\,cm$'s and it is not accurate enough to control and coordinate a group of robots. In addition, the method requires highly accurate time synchronization.

Localization using camera sensors have been widely studied in the computer vision community. Taylor et al. [11] used controllable light sources to localize sensor nodes in a stationary camera network. A distributed version of camera localization is proposed by Funiak et al. [12], in which relative positions of cameras are recovered by tracking a moving object. Meingast et al. [13] proposed a multi-target tracking based camera network localization algorithm. The critical concept applied in [13] is the use of spatio-temporal features, an approach taken in this paper. These spatio-temporal features were used for finding matching features over a pair of cameras instead of spatial features. Then the relative position and orientation between cameras were computed using multi-view geometry. Since an incorrect matching between spatio-temporal features is extremely rare compared to spatial features, the method provides an outstanding performance under wide baseline and varying lighting conditions.

But the aforementioned methods are designed for stationary camera networks and not optimal for dynamic mobile sensor networks. In fact, in mobile sensor networks, we can take the advantage of mobility to improve the efficiency of localization. For instance, Zhang et al. [14] proposed a method to control the formation of robots for better localization. They estimated the quality of team localization depending on the sensing graph and the shape of formation. Tully et al. [15] used a "leap-frog" method for a team of three robots performing cooperative localization during navigation. Two robots, which are stationary, localize the moving third robot from bearing measurements with an extended Kalman filter. After completing the move, the role of each robot is switched and the process is repeated. They have shown a good localization result while moving in a $20\,m \times 30\,m$ GPS-denied area. However, the experiment was conducted using an expensive hardware platform (three on-board computers and four stereo cameras are used) and it is unclear if the approach is suitable for an inexpensive off-the-shelf robotic platform considered in this paper.

We propose a localization algorithm for mobile sensor

This work has been supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2010-0013354).

Junghun Suh and Songhwai Oh are with the School of Electrical Engineering and Computer Science and ASRI, Seoul National University, Seoul 151-744, Korea (emails: {junghun.suh, songhwai.oh}@cpslab.snu.ac.kr). Seungil You is with the Control and Dynamic Systems, California Institute of Technology, Pasadena, CA 91125 USA (email: syou@caltech.edu).

networks under GPS denied areas or indoor environments using inexpensive off-the-shelf robotic platform. We take the advantage of the multi-agent system of mobile sensor networks. In order to localize mobile robots, we first partition robots into two groups: stationary robots and moving robots. We assume each robot carries a camera and two markers. The moving robots move within the field of views (FOVs) of stationary robots. The stationary robots observe the moving robots and record the positions of the markers of moving robots. Based on the trajectories of markers of moving robots (i.e., spatio-temporal features), we localize all the robots using multi-view geometry. Localization requires recovering relative positions, i.e., translation and orientation. A multi-robot coordination algorithm using the proposed localization algorithm is also presented, which maintains positions of robots with respect to a fixed reference coordinate system as a group of robots moves across the field of interest.

We have implemented the proposed algorithm on a mobile robot platform assembled from an iRobot Create [16] and conducted an extensive set of experiments. From experiments, we have discovered a set of configurations of robots, from which good localization is possible. We then applied these configurations in our cooperative multi-robot localization algorithm. Our experimental results show that the proposed method consistently achieves less than $1\,cm$ of localization error for trajectories of length less than $100\,cm$ and less than 0.34% of localization error for longer trajectories of length between $725\,cm$ and $769\,cm$, making it a promising solution for multi-robot localization in GPS denied or unstructured environments.

This paper is structured as follows. Section II provides an overview of the proposed cooperative multi-robot localization method. In Section III, We present the multi-robot localization algorithm based on planar homography used in our experiments. Multi-robot coordination and communication protocols are described in Section IV. In Section V, the results from experiments are presented.

II. An Overview of Cooperative Multi-Robot Localization

This section gives an overview of the method proposed in this paper. Suppose there are N robots and we index each robot from 1 to N. We assume that each robot's state is determined by its position and orientation in the reference coordinate system. Then the goal of the multi-robot localization problem is to estimate positions and orientations of all robots over time.

Let $X_i(k) = (P_i(k), R_i(k))$ be the state of robot i at time k with respect to the reference coordinate system, where $P_i(k) \in \mathbb{R}^3$ and $R_i(k) \in SO(3)$ are the position and rotation of robot i at time k, respectively. Then the configuration of a multi-robot system at time k is

$$X(k) = (X_1(k), X_2(k), \ldots, X_N(k)).$$

The multi-robot localization problem is to estimate $X(k)$ for all k from sensor data.

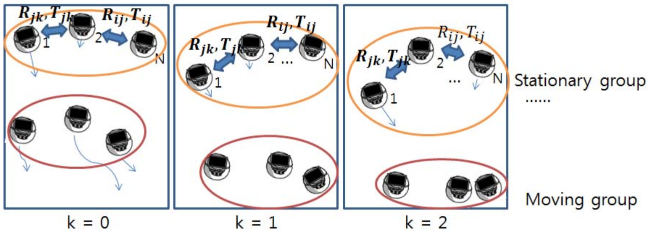


Fig. 1. An overview of the proposed cooperative multi-robot localization algorithm. A group of robots is is moving within the field of views of stationary robots. Stationary robots track markers of moving robots and exchange marker positions with other stationary robots. The translations and orientations of pairs of stationary robots are then computed using multi-view geometry. Finally, all robots are localized with respect to the reference coordinate system based on the position of at least one fixed robot since last update time.

Suppose that we have $X(k-1)$ with respect to the reference coordinate system and computed relative positions, $T_{ij}(k)$, and orientations, $R_{ij}(k)$, for pairs of robots i and j at time k. Then we can easily compute positions and orientations of all robots with respect to a single robot of choice. In order to map new positions of robots in the reference coordinate system, we require that there is at least one robot i such that $X_i(k) = X_i(k-1)$. Then taking positions with respect to this robot, we can recover the positions and orientations of all robots at time k with respect to the reference coordinate system.

Based on this idea, we develop a corporative localization algorithm. At each time instance, we fix robot q and move other robots. Then we compute $T_{ij}(k)$ and $R_{ij}(k)$ for pairs of robots such that the pose of each robot can be computed with respect to robot q. Finally, we compute $X(k)$ based on $X_q(k-1)$. For $k+1$, we fix another robot r and move remaining robots and continue this process. By doing so, we can continuously estimate $X(k)$ for all times.

Now the remaining issue is how to estimate translations $T_{ij}(k)$ and orientations $R_{ij}(k)$ for pairs of robots. For this task, we make the following assumptions:

- Each robot carries a camera and markers.
- The internal parameters of cameras are known (e.g., focal lengths, principal points, distortion coefficients).
- Each robot communicates with other robots via wireless communication.
- The clocks of all robots are synchronized.
- Either the distance between a pair of markers on a robot is known or the height of a single marker is known when a robot is moving on a flat surface.
- At least two robots which capture images are stationary.

Figure 1 illustrates an overview of our method. Robots carrying markers move within the field of views (FOVs) of stationary robots. Each stationary robot performs image processing, detects markers, and localizes positions of markers in its image frame. The marker positions and image capture times are shared with other stationary robots. For a pair of stationary robots i and j, we can compute the relative translation $T_{ij}(k)$ and orientation $R_{ij}(k)$ from pairs of marker trajectories using multi-view geometry.

Since the minimum number of robots required for the proposed cooperative localization algorithm is three, we will discuss our method using a mobile sensor network of three robots for the ease of exposition in this paper. However, the proposed method can be applied to a multi-robot system with a larger number of robots. Furthermore, while a single moving robot is used in our discussion and experiment, the method can be likewise applied to the case with multiple moving robots using a multi-target tracking algorithm as done in [13].

III. Multi-Robot Localization using Multi-View Geometry

In this section, we focus on a single step of the cooperative multi-robot localization algorithm for localizing stationary robots by tracking a moving robot.

When two cameras view the same 3D scene from different viewpoints, we can construct the geometric relation between two views. If we constrain the 3D scene to the planar scene, the relation can be represented using the homography. The homography can be computed from a number of pairs of corresponding points which are the positions of the marker on the robot and they are detected by a color based marker detection algorithm.

In order to robustly estimate the planar homography, we use the random sampling consensus (RANSAC) algorithm [17], which is robust for fitting a model in the presence of outliers. The homography between two views can be expressed as follows:

$$H = K\left(R + \frac{1}{d}TN^T\right)K^{-1} \tag{1}$$

where, $R \in SO(3)$ is the rotation matrix, $T \in \mathbb{R}^3$ is the translation vector, N is the unit normal vector of the plane with respect to the first camera frame and d is the distance from the plane to the optical center of the first camera [18].

We can recover $\left\{R, \frac{1}{d}T, N\right\}$ from H using singular value decomposition. From this derivation, we obtain two possible solutions [18]. Among two solutions, we can find a unique solution since the normal vector of the plane is available in our case. In this paper, corresponding points are located in the plane which is parallel to the ground and the angle each camera views the ground is fixed, so we can compute the normal vector of the plane. As explained in [18], from singular value decomposition of H^TH, we obtain orthogonal matrix $V \in SO(3)$, such that $H^TH = V\sum V^T$, where $V = [v_1, v_2, v_3]$. Let u be the unit-length vector such that $N = v_2 \times u$ and $v_2^T u = 0$ because v_2 is orthogonal to N. Therefore, given v_2 and N, we can solve for u. Once we find u, we can form the new orthonormal basis $\{v_2, u, N\}$ and obtain R and T as follows:

$$R = WU^T \qquad \text{and} \qquad T = d(H-R)N, \tag{2}$$

where $U = [v_2, u, N]$ and $W = [Hv_2, Hu, \widehat{Hv_2}Hu]$.

When we reconstruct the position of markers in the 3D space using data points from the image plane, we can find the exact scale factor using the distance between markers. In order to compute the distance d from camera center to the 2D plane in the 3D space, we can simply measure the height of marker and camera. Considering that the 2D plane of markers is parallel with the plane where the center of camera is located, d is the distance between these two planes.

IV. Multi-Robot Coordination and Communication

The overall cooperative localization algorithm is illustrated in Figure 2. A moving robot (robot C) is not illustrated in this figure. Let the coordinate of robot A in Figure 2 (Step 1) be the reference coordinate. In Step 1, from our proposed localization process, the homography H_{01} and the position of robot B are obtained with respect to the coordinate of robot A. After localization, robot B is stationary and robot A moves forward and rotates to the left. (For more information about movements after localization, see V-B.) Before starting localization, robot A and B check whether robot C is within the FOVs and make it move within the FOVs. Again the homography H_{12}, the relative pose of robot A with respect to the coordinate of robot B, is computed (Step 2). Now we compute the position of robot A with respect to the reference coordinate by using the composition of two homography H_{01} and H_{12}, i.e., $H_{02} = H_{01}H_{12}$. In Step 3, robot A is stationary and robot B moves forward and rotates. And the process repeats as robot A, B, and C moves as a group. Note that this three-robot coordination method can be easily extended to a multi-robot system with a larger number of robots.

An overview of the communication scheme for a three-robot system is as follows. We use a one-server, two-client model. After synchronization, robot A (server) sends a command to robot B (client 1) to capture images while commanding robot C (client 2) to move. Then robot A and B capture images and robot C moves. Since robot C may move beyond the FOV of robot A or robot B, robots A and B check the visibility of robot C, testing whether robot C is within the FOVs. If robot C is not visible by either robot, visibility information is sent to robot C and robot C turns back to be visible by both robots. Once an enough number of marker positions are collected, robot B sends trajectory data and capture times to robot A. Then, robot A estimates homography matrix H using its data and data transmitted by robot B.

V. Experimental Results

In this section, we discuss the results obtained from our indoor experiments. For our experiments, we used a wheeled mobile robot, iRobot Create [16], as a mobile node in our mobile sensor network. Our mobile platform is shown in Figure 3(a). It is equipped with a PS3 Eye camera, HP Netbook Mini 110 which runs Linux OS, and a white LED which works as a marker. Wi-Fi (IEEE 802.11) is used for robot communication. Each camera has a resolution of 320×240 pixels and runs at 100 frames per second (fps).

In order to reduce the computation time, when we detect a marker in an image, we restricted our search over a small patch centered at the position based on the previously detected marker's position and marker's velocity. Suppose that the pixel position of the marker at time k is (x_k, y_k). At time $k+1$, the center of the patch is at $(x_k + (x_k - x_{k-1}), y_k +$

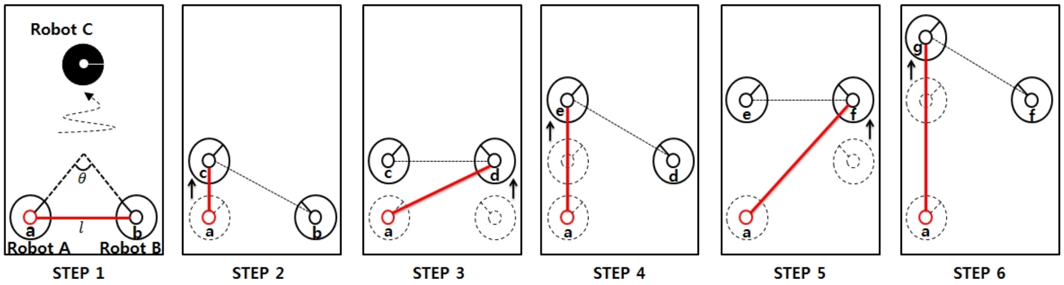


Fig. 2. An illustration of the cooperative multi-robot localization algorithm. The length of each red segment is the distance from the original position of robot A in step 1 to the new position of the robot with motion. Dashed lines show the relative poses that are computed at each step of the algorithm. l is the distance between two stationary robots and θ is the angle between them.

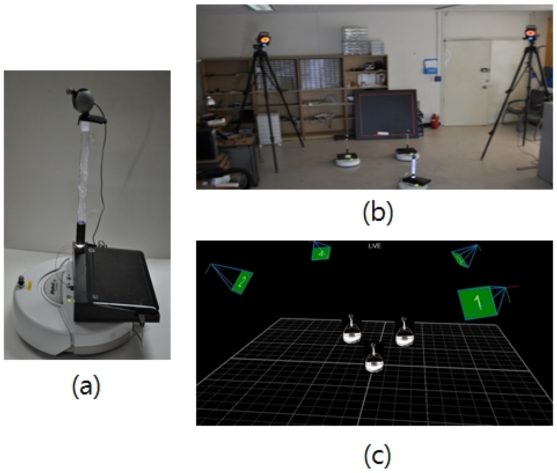


Fig. 3. (a) iRobot Create based mobile platform. (b) A photo from the experiment. The Vicon motion capture system is used for providing the ground truth values. (c) An image obtained from Vicon with robots placed on the reference coordinate system.

$(y_k - y_{k-1}))$. In our experiment, we used a 40×40 patch, which is decided based on the speed of robots used in the experiment.

The time synchronization operation is implemented as follows. Clients wait for a synchronization signal from the server. After receiving a synchronization signal, clients start their respective tasks at the chosen synchronization time. The server robot and client 1 robot capture images at 100 fps and record image capture times.

There are three types of errors that can be introduced. The first type is the environmental error. The planar homography assumes that all correspondences are on the same plane. However, a bumpy movement of a moving robot and debris on the floor can introduce an error. By using RANSAC, we can exclude outliers and reduce this type of error. The second type is the correspondence error. An incorrect correspondence of markers from two images can introduce an estimation error. By comparing timestamps from two image buffers, we find corresponding image pairs when the time difference is less than a threshold. In our experiment, the time synchronization error was at most $5\,ms$. In addition, the center of detected blob may not be the correct position of a marker. The third type is the quantization error from using images with the finite resolution. In experiment, we have found that this error is not fatal for the results and can be reduced by using more marker positions.

A. Cooperative Multi-Robot Localization: Single-Step

We first performed experiments for the single-step of the cooperative multi-robot localization algorithm in order to find a multi-robot configuration which results in good localization.

Figure 3(b)-(c) shows our experiment setup. We used the Vicon motion capture system [19] to collect the ground truth data in order to measure the performance of our algorithm. The Vicon motion capture system operates at over 250Hz and its position error is less than $0.25\,mm$. We conducted our experiments at four different baselines between two stationary robots ($l = 60, 80, 100, 120\,cm$). For each baseline, we tested five different angles between robots ($\theta = 0^\circ, 10^\circ, 20^\circ, 30^\circ, 40^\circ$). See Figure 2 for how l and θ are defined. Hence, there is a total of 20 cases. For each case, we collected about 250 marker positions of a moving robot. We randomly picked 50 data points and performed our algorithm and repeated this process for 500 times for each case. Hence, for each case, we have 500 runs. For each run, we computed the estimation error $\varepsilon_i = |\hat{l}_i - l_{vicon}|$, for $i = 1, 2, \ldots, 500$, where $\hat{l}_i$ is the estimated distance between two robots for the i-th run using our algorithm and l_{vicon} is the ground truth distance obtained from Vicon. Figure 4(a)-4(d) show the results of all 20 cases. The distribution of localization error is shown as a histogram for each case. The bin size of a histogram is $0.5\,cm$ and the color of a bin represents the number of runs with localization errors belonging to the bin. When this number is large, the bin color is red and the bin color is dark blue when this number is low. For instance, when $l = 60$ and $\theta = 0$, more than 200 runs resulted in error between $0.5\,cm$ and $1.0\,cm$. A white circle represents the mean error from 500 runs for each case. For $l = 60$ and $\theta = 0$, the mean error is $0.5\,cm$. As shown in Figure 4(a) and 4(b), when the baseline is $60\,cm$ or $80\,cm$, the mean error is within $1\,cm$, except when $(l = 80, \theta = 0)$ and $(l = 80, \theta = 10)$. On the other hand, as shown in Figure 4(c) and 4(d), the mean errors are relatively high for $l = 100\,cm$ and $l = 120\,cm$, especially at small angles. This is due to the

TABLE I

RESULTS FROM THE MULTI-STEP EXPERIMENT. (SEE FIGURE 2 FOR SEGMENT LABELS)

seg	true	est	seg	true	est
a-b	77.14cm	77.12cm	a-b	77.14cm	77.12cm
b-c	85.76cm	85.72cm	a-c	36.22cm	36.23cm
c-d	78.29cm	78.42cm	a-d	85.73cm	85.21cm
d-e	88.16cm	87.82cm	a-e	74.01cm	73.98cm
e-f	80.47cm	80.05cm	a-f	107.82cm	107.4cm
f-g	89.82cm	89.53cm	a-g	109.73cm	109.36cm

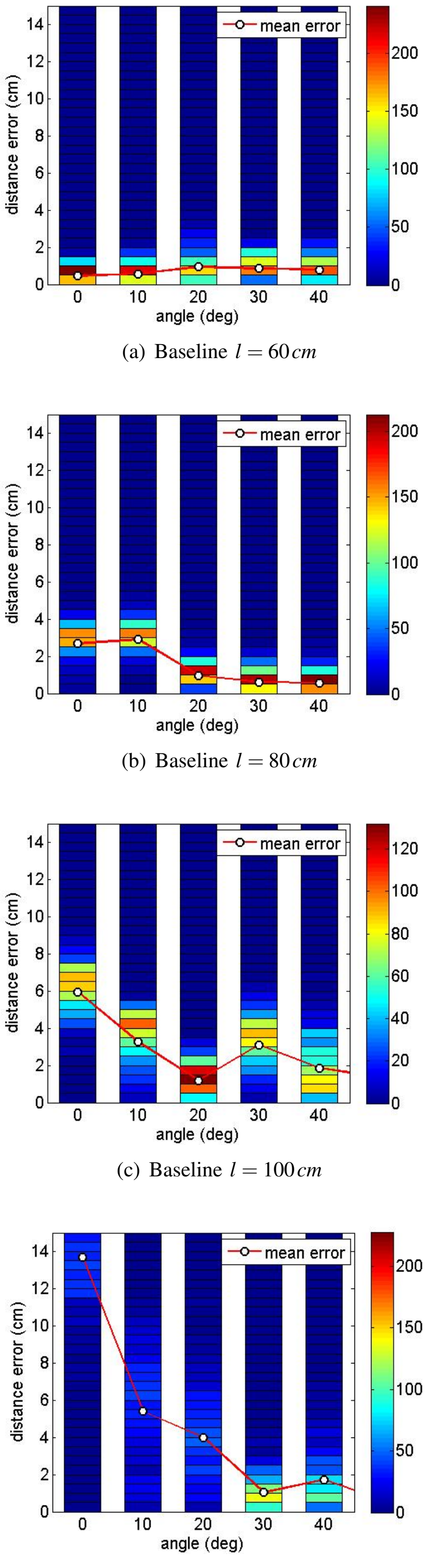
fact that the overlapping area between two cameras is small for those cases. We plotted the localization error as a function of the number of overlapping pixels in Figure 5. Clearly, the size of the overlapping area determines the localization performance and we must account this when designing a multi-robot localization algorithm. Since the baseline distance and angle between robots can be configured in our multi-robot localization algorithm for the best performance, the experimental results show how we should configure robots in our cooperative multi-robot localization algorithm as we do in the next experiment.

B. Cooperative Multi-Robot Localization: Multi-Step

In this experiment, we localize a group of robots as they move from one place to another as described in Section IV. Based on the previous experiment, we found that a baseline of 80cm and an angle between 30° and 40° were ideal and this configuration was used in this multi-step experiment.

Because the space covered by the Vicon motion capture system was limited, we were able to perform six steps of the algorithm as shown in Figure 2. Table I shows localization errors from the experiments. In the table, "seg" represents the line segment shown in Figure 2, "true" is the length of the segment computed by Vicon, and "est" is the length computed by our algorithm. At step 1, the difference between ground truth value and estimation value is 0.02cm. After step 1, robot A goes forward for about 40cm and turns to the left and robot B does not move. Since we know the angle between robot A and B from rotation matrix R computed in step 1, when robot A rotates, we can maintain the pre-defined angle θ. At step 2, the coordinate system with respect to robot B is the reference coordinate system and the localization error for the segment $a-c$ is only 0.01cm. After step 2, robot A is stationed and robot B moves forward for about 40cm. Again, we can maintain the pre-defined distance d by computing the position of robot B in step 2 with respect to the coordinate of robot A in step 1. And this process is repeated as shown in Figure 2. For all steps, the localization error was kept within 1cm and the localization error of the longest segment $a-g$ was only 0.37cm.

We also conducted the experiments in the hallway to demonstrate its performance over a long distance (see Figure 6(a) and 6(b)). Table II shows localization results from the experiments in the hallway. For trajectories with length from 725cm to 769cm, the achieved localization error is



(a) Baseline $l = 60cm$

(b) Baseline $l = 80cm$

(c) Baseline $l = 100cm$

(d) Baseline $l = 120cm$

Fig. 4. Localization error distributions of 20 cases at different baselines ($l = 60, 80, 100, 120cm$) and angles ($\theta = 0°, 10°, 20°, 30°, 40°$) between robots. Each case has 500 runs. An interval of each bin is 0.5cm and the color of each bin represents the number of runs with localization error belonging to the interval. A white circle represents the mean error of 500 runs.

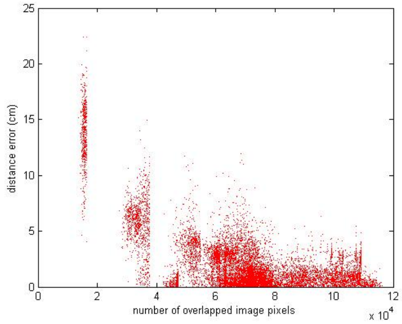
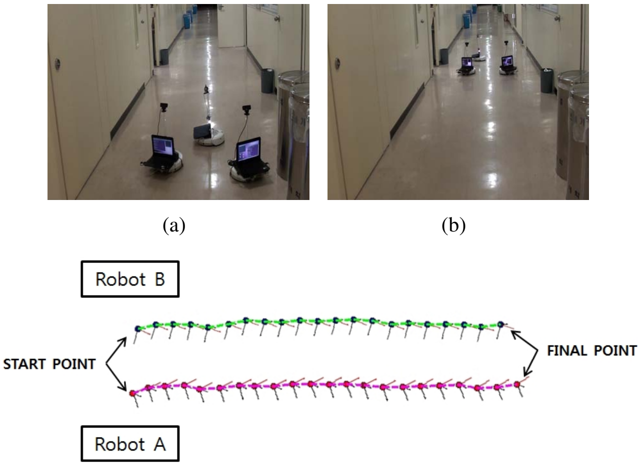


Fig. 5. Scatter plot of the localization error as a function of the number of overlapping pixels between two cameras.

TABLE II

Results from the multi-step experiment in the hallway.

Case	Robot	True	Est.	Error	Err. Rate
1	B	$769\,cm$	$771.6\,cm$	$2.6\,cm$	0.34%
	A	$730\,cm$	$728.1\,cm$	$1.9\,cm$	0.26%
2	B	$725\,cm$	$723.2\,cm$	$1.8\,cm$	0.25%
	A	$732\,cm$	$733.5\,cm$	$1.5\,cm$	0.20%



(a)

(b)

(c)

Fig. 6. (a) and (b) Photos from the hallway experiments. (c) Estimated trajectories of robots (robot A and B).

between $1.5\,cm$ and $2.6\,cm$, making the error less than 0.34% of the length of the trajectory of the robot. See Figure 6 for photos from the experiments and robot labels.

Localization under the GPS denied or unstructured indoor environment is a challenging problem. But the experimental results show that our algorithm can provide a promising solution to this challenging localization problem.

VI. Conclusion

In this paper, we have presented a cooperative localization algorithm for mobile sensor networks. The algorithm is designed to solve the challenging localization problem under the GPS denied or unstructured indoor environment by taking the advantage of the multi-agent system and mobility in mobile sensor networks. Our experiment shows that there exists a configuration of robots for good localization and this configuration is applied to develop a highly accurate cooperative multi-robot localization algorithm. In experiments, the proposed method achieves less than $1\,cm$ of localization error for trajectories of length less than $100\,cm$ and less than 0.34% of localization error for trajectories of length between $725\,cm$ and $769\,cm$. The experimental results show that the localization error increases as a robot travels a longer distance. This propagation of error can be reduced by placing landmarks in the environment and this is our future research topic.

References

[1] S. Oh, L. Schenato, P. Chen, and S. Sastry, "Tracking and coordination of multiple agents using sensor networks: System design, algorithms and experiments," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 234–254, January 2007.

[2] A. Singh, M. Batalin, M. Stealey, V. Chen, Y. Lam, M. Hansen, T. Harmon, G. S. Sukhatme, and W. Kaiser, "Mobile robot sensing for environmental applications," in *Proceedings of the International Conference on Field and Service Robotics*, 2007.

[3] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[4] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Stability of flocking motion," University of Pennsylvania," Technical Report, 2003.

[5] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[6] W. Ren and R. W. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, May 2005.

[7] K. Whitehouse, C. Karlof, A. Woo, F. Jiang, and D. Culler, "The effects of ranging noise on multihop localization: an empirical study," in *Proc. of the ACM/IEEE International Conference on Information Processing in Sensor Networks*, April 2005.

[8] M. Maróti, B. Kusý, G. Balogh, P. Völgyesi, A. Nádas, K. Molnár, S. Dóra, and A. Lédeczi, "Radio interferometric geolocation," in *Proc. of ACM SenSys*, Nov. 2005.

[9] A. Ledeczi, P. Volgyesi, J. Sallai, B. Kusy, X. Koutsoukos, and M. Maroti, "Towards precise indoor rf localization," in *Proc. of the Workshop on Hot Topics in Embedded Networked Sensors (HOTEMNETS)*, June 2008.

[10] S. Gecizi, Z. Tian, G. B. Giannakis, Z. Sahinoglu, H. Kobayashi, A. F. Molisch, and H. V. Poor, "Localization via ultra-wideband radios," *IEEE Signal Process. Mag.,*, vol. 22, no. 4, pp. 70–84, 2005.

[11] C. Taylor and B. Shirmohammadi, "Self localizing smart camera networks and their applications to 3d modeling," in *International Conference on Distributed Smart Cameras (ICDSC-06)*, 2006.

[12] S. Funiak, C. Guestrin, M. Paskin, and R. Sukthankar, "Ad hoc networks for localization and control," in *Proc. of the 5th Int. Conference on Information Processing in Sensor Networks IPSN 06. ACM*, 2006.

[13] M. Meingast, S. Oh, and S. Sastry, "Automatic camera network localization using object image tracks," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV) Workshop Visual Represent. Model. Large-Scale Environ*, Octover 2007.

[14] F. Zhang, B. Grocholsky, R. Kumar, and M. Mintz, "Cooperative control for localization of mobile sensor networks," in *GRASP Laboratory, University of Pennsylvania, internal paper*, 2003.

[15] S. Tully, G. Kantor, and H. Choset, "Leap-frog path design for multi-robot cooperative localization," in *International Conference of Field and Service Robots, Cambridge, MA, USA*, 2009.

[16] "iRobot." [Online]. Available: http://www.irobot.com/

[17] M. Fischler and R. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[18] Y. Ma, J. Kosecka, S. Soatto, and S. Sastry, Eds., *An Invitation to 3-D Vision*. New York: Springer Verlag, 2003.

[19] "Vicon MX motion capture system." [Online]. Available: http://www.vicon.com/